LÜĞƏT VƏ ENSİKLOPEDİYALAR



UŞAQLAR ÜÇÜN

ENSİKLOPEDİYA



İNFORMATİKA



"Şərq-Qərb"
Bakı
2008

-4700-

Azərbaycan Respublikası Prezidentinin
İşlər İdarəsi
PREZİDENT KİTABXANASI

ICES
INTERNATIONAL CENTRE OF EDUCATIONAL SYSTEMS (ICES)
МЕЖДУНАРОДНЫЙ ЦЕНТР ОБУЧАЮЩИХ СИСТЕМ (МЦОС)
CENTRE INTERNATIONAL DES SYSTEMES D'EDUCATION (CISE)
INTERNATIONALES ZENTRUM FUR AUSBILDUNGSSYSTEME (IZAS)
CENTRE
UNDP Reg.№ 95073
UNESCO Agr.of 12.11.95
UNIDO Reg.№ 092353
UNEP Reg.of 24.08.99
UNESCO/ICES BEYNƏLXALQ KAFEDRA-ŞƏBƏKƏSİ
"TEXNİKİ VƏ PEŞƏKAR TƏHSİL VƏ KADRLARIN HAZIRLANMASI"

"Uşaqlar üçün ensiklopediya"lar Beynəlxalq təhsilverici sistemlər mərkəzi (BTSM) və YUNESKO/BTSM Beynəlxalq kafedra-şəbəkəsi tərəfindən hamı üçün fasiləsiz təhsil (continuing longlife education for all) sistemlərində dərs vəsaiti kimi tövsiyə edilmişdir.

Azərbaycan nəşri üzərində işləyənlər:

Əli Abbasov
akademik

Rasim Əliquliyev
AMEA-nın müxbir üzvü,
texnika elmləri doktoru, professor

Ələkbər Əliyev
texnika elmləri doktoru, professor

ISBN 978-9952-34-197-3

004.03-dc22
Uşaq ensiklopediyası – İnformatika

Uşaqlar üçün ensiklopediya. İnformatika.
Bakı, "Şərq-Qərb", 2008, 616 səh.

© "Şərq-Qərb", 2008

OXUCUYA

İnformatika elmini, eləcə də onun xüsusiyyət və qanunlarının yayılmasını öyrənən fənn – informasiyadır. Ona görə də, informatika haqda nə isə başa düşməkdən ötrü gərək "informasiya" anlayışını götür-qoy edəsən.

Necə sadə və vərdiş etdiyimiz sözdür bu – "informasiya". İndi hətta uşaqlar belə ondan asanlıqla istifadə edir, böyüklərin isə o çoxdandır fəal lüğətlərindəndir. Doğrudan da, əgər bu anlayış bizim həyatımıza sıx, özü də gözgörəsi uzun müddətə daxil olmuşdursa burada nəyi götür-qoy etməliyik? Əgər bizdən bu sözün mahiyyəti barədə soruşanda fikirləşmədən deyirik ki, informasiya – bu saxlanılması, ötürülməsi və emal edilməsi mümkün olan istənilən məlumatdır. Bununla belə çox adam informasiyanın bəşəriyyətdən əhəmiyyətli dərəcədə öncə meydana gəldiyini fikirləşibmi heç? Təbiət bitkilərin və canlı orqanizmlərin genetik informasiyalarını qoruyub, ötürüb və emal edəndə hələ insanın bizim planetə zühuru haqqında heç bir xəbər yox idi. Və insan peyda olub, demək olar ki, ilk addımlarını Yerdə atdığı andan informasiyanın ötürülməsi və qorunması üçün vasitə axtarmağa başladı. Qədim Misir papirusları, qədim insanların mağaralardakı qayaüstü rəsmləri və hindi tayfalarının məlumatlar ötürən tonqal ənənələri bunları təsdiqləməyə xidmət edir.

Son illər insanın tələbat duyduğu informasiyanın miqdarı o qədər sürətlə genişlənir ki, biz bu artımla bəzən ayaqlaşa bilmirik, özümüzü rəngarəng məlumatların sahilsiz okeanında azmış və köməksiz hiss edirik. YUNESKO-nun verdiyi xəbərə görə XIX əsrin əvvəllərində bütün dünya üzrə elmi-dövri nəşrlərin sayı 100-ə yaxın idi. 1850-ci ildə dövri nəşrlər 1000-ə qədər artmış, 1900-cü ildə 10000-i keçmiş, indisə bəzi mütəxəssislərin rəyinə görə 100000-ə yaxınlaşır. Bundan əlavə jurnal məqalələri durmadan genişlənir: indi onların dərci il ərzində 3 milyona qədərdir. Son 25 ildə nəşr olunan kitabların sayı son 500 ildəkinə bərabərdir. Bəs bu axının informasiya tufanında müvazinəti necə saxlamalı?

Müasir insan üçün informasiyanın hansı əhəmiyyəti vardır? O bizə nə bağışlayır? İlk növbədə, əlbəttə, qolaylıq. Biz televizora baxır, sorğu bürosuna zəng edir, mətbuatı

oxuyur, suallarımıza internetdə cavab axtarır və bu minvalla həyatımızı sadələşdiririk. Dərindən baxsaq biz görərik ki, əgər fəaliyyəti məlumatların axtarışı və analizə bağlıdırsa informasiya insanı fiziki işdən zehni işə qoşmaqla üstünlük təşkil edir. Bu da ilk öncə rifah kimi qəbul olunur. Amma unutmaq lazım deyil ki, bizim şüurumuz da əzələlərimiz kimi yorula bilir. Əqli, psixoloji yorğunluğun fəsadları fiziki yorğunluğun törətdiyi xəstəliklərdən heç də geri qalmır.

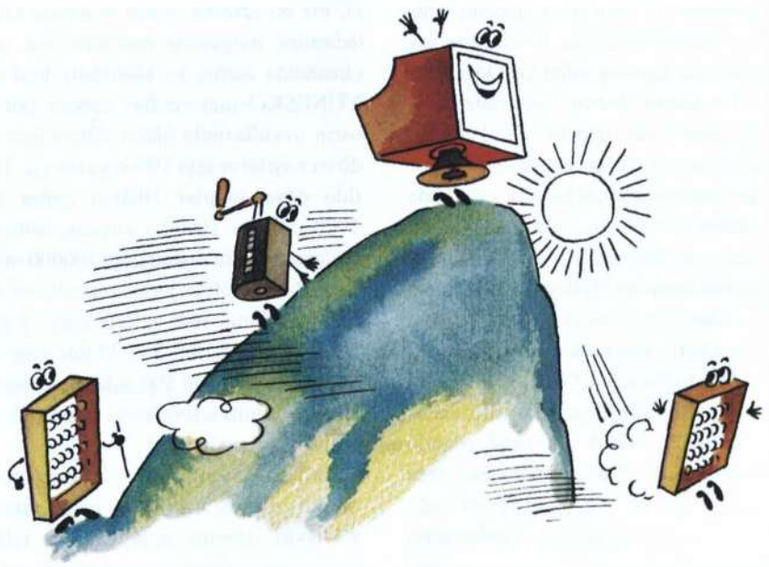
Ona görə də informasiya artımının bəşəriyyət üçün əhəmiyyətindən danışanda, Yer üzərində mövcud olan hər şeyin təzadlı olduğu kimi onun da əhəmiyyətinin təzadlı olduğunu gərək düzünə etiraf edək.

Budur informasiya ilə götür-qoy etdik. İndi davam etməyin əsl vaxtıdır. "İnformatika" termini iki söz birləşməsindən: "informasiya" və "avtomatika"dan əmələ gəlmişdir. Bəs bu elm nə ilə məşğuldur. İngilis alimləri bu suala belə cavab verirlər: "İnformatika son dərəcə professional və akademik fəaliyyəti dəstəkləməyi nəzərdə tutmayan, informasiyanın üstünlüyünün nəzəri, metodiki və praktiki təqdim olunması ilə məşğul olur". Bu məqsədlə o, informasiyanı yaranmasından istifadə olunmasına qədər, eləcə də, müxtəlif formalarda, müxtəlif kanallarla ötürülməsini öyrənir. İnformasiya bu və digər şəkildə milyardlarla insanın həyatında iştirak edir, o elmi-texniki tərəqqiyə, sosial-iqtisadi həyata və hətta insanın özünüdərkinə təsir göstərən ciddi qüvvəyə çevrilmişdir.

İndiki halda informasiyanı getdikcə daha çox və daha artıq əxz eləmək sadəcə mümkün deyildir: indi ən önəmlisi onu işləmək, analiz etmək, müqayisə etmək, nəticə çıxarmaq, daha doğrusu informasiyanı öz məqsəd və tələbatına tabe etməkdir. Məhz buna görə də informatika gərəkdir.

Bu elm – ən gənc elmlərdən biridir. İlk EHM (elektron-hesablama maşını) ikinci dünya müharibəsi zamanı yaradılmışdı, amma bu yeni elm barədə 40 il bundan öncə ciddi və geniş söhbət açmağa başlanmışdır. Kompüterlərin meydana gəlməsi informasiyanın əmək nümunəsi kimi nəzərəçarpacaq rol oynadığı güclü sənaye əsasları olan cəmiyyətlərin yaranmasına səbəb oldu.

İnformasiyanın təkmilləşdirilməsinin səmərəlilik və sürəti, eləcə də informasiya





axınının genişləndirilməsi indi iqtisadi inkişafın vəziyyətini xeyli şərtləndirir və bu, dəfələrlə sübut olunmuşdur. Hazırda istənilən bir ölkənin istənilən cəmiyyətində elektron vasitələrinin verilənlərilə işləməyi bacaran mütəxəssislər tələb olunur.

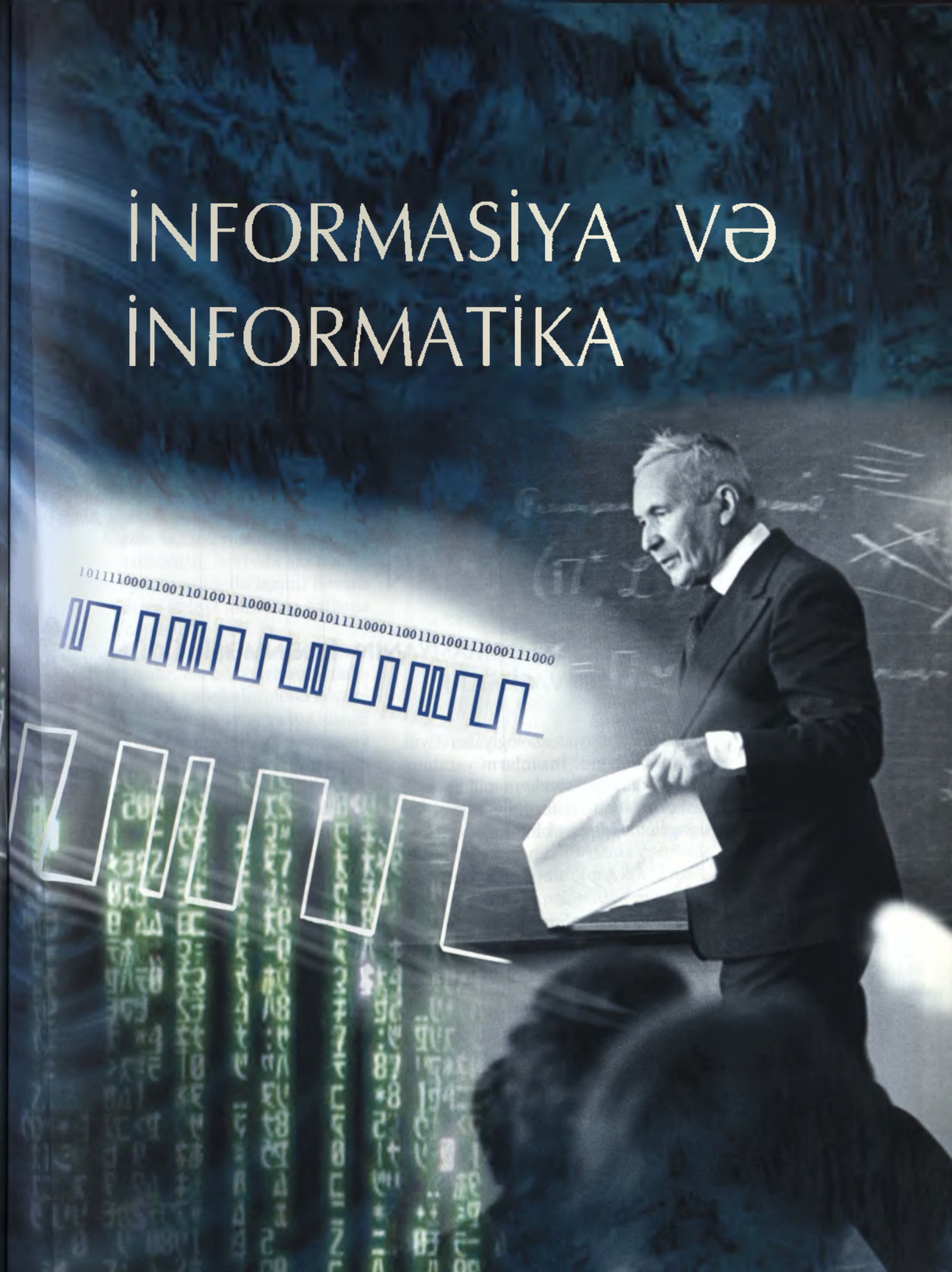
İnformasiya və informatikanın kontekstində bu günkü dünyamızın durumunu iki faktı yada salmaqla xarakterizə etmək olar: bütün bəşəriyyətin və eləcə də ayrı-ayrılıqda hər bir insanın həyatına informasiya texnikasının görünməmiş müdaxiləsi faktını kompüterlərdən bu gün nəinki elmi-tədqiqat institutlarında, istehsalatda, dövlət müəssisələrində, kommersiya sahələrində, eləcə də məktəblərdə, kitabxanalarda, fərdi mənzillərdə də istifadə olunur. Kompüterin, yeni informasiya texnologiyasının köməyilə bu gün işləyir və dincəlirlər, oxuyur və əlaqə saxlayırlar, elmi kəşflər edir və yeni kulinariya reseptləri haqda məlumat alırlar.

İnsanların böyük hissəsi yeni elmdən öz fəaliyyətlərində istifadə etməyə başlamışlar. Dünyanın hər yerində təhsil müəssisələri informasiya və informatika sahəsi üzrə peşəkarlar hazırlayırlar. Statistikaya görə hələ 1975-ci ildə amerikalı işçilərin yarıdan çoxu bu və ya digər cəhətdən verilənlərin emalı və analizilə bağlı idilər.

Demək olar ki, informasiya miqdarına və insan həyatına təsir gücünə görə indi ən qüdrətli bir sivilizasiya yaranmaqdadır.

İNFORMASİYA VƏ İNFORMATİKA



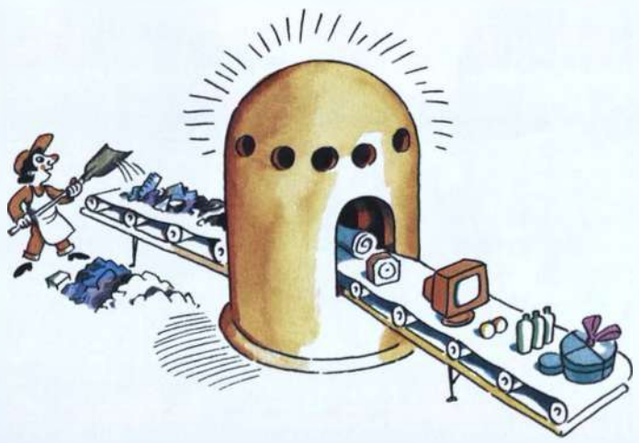




İNFORMASİYA VƏ İNSAN

KOMPÜTER İNFORMASİYANIN İŞLƏNMƏSİ ÜÇÜN UNİVERSAL VASİTƏDİR

Qlobal informasiya texnologiyaları dövrü yenicə başlayıb. İnsanların yaratmış olduqları informasiyaların mühüm hissəsi artıq rəqəmli formaya salınıb və kompüterdə işlənərək informasiya şəbəkələri vasitəsilə dünyanın istənilən nöqtəsinə ötürülə bilər. Ötən əsrlərdə əldə edilmiş informasiyalar (mətnlər, şəkillər, filmlər, audioyazılar və s.) tədricən rəqəmli formaya çevrilir və həmin informasiya şəbəkələrinə daxil edilir. İnformasiyaları bu formada saxlamaq daha asandır və onlar köçürülmə zamanı təhrif olunmur. Fiziki informasiya daşıyıcıları daimi olmasalar da informasiyanın vaxtında köçürülməsi, üzünün çıxarılması, kompüter şəbəkəsinin müxtəlif nöqtələrində saxlanması lokal qəzalardan – kompüterlərin sınmasından, yanğınlardan qorxmamağa əsas verir. İnformasiyanın işlənməsi zamanı fiziki işlərdən tutmuş yaradıcı işlərə qədər istənilən əməliyyatlar kompüter vasitəsilə yerinə yetirilir. Hələ bir neçə il öncəyədək şahmat oyunu son



dərəcə yaradıcı məşğuliyyət hesab edilirdi, ancaq bu gün məlum olur ki, kompüter üçün bu oyun "xaçlar və sıfırlar" (*ing.* tic-tac-toe) oyunundan azca çətindir.

Gərək olduqları yerlərdə kompüterlər artıq var, yaxud yaxın zamanlarda olacaqlar. Müasir kompüter praktiki olaraq öz imkanlarının son həddinə çatıb. Kompüterdə saxlanılan mətnlərin həcmi o qədər böyükdür ki, onun sahibi bu mətnləri bütün ömrü boyu oxuyub qurtara bilməz. Bütün dünyanın riyazi ictimaiyyətinin il ərzində hazırladığı mətnlərin həcmi bir müasir kompüterin yaddaşında saxlanıla bilən mətnlərin həcmindən azdır.

Böyüklər kompüterdən iş görmək üçün, uşaqlar isə dərs öyrənmək üçün istifadə edir. İşgüzar məktublar kompüterdə tərtib edilir, arxivdə saxlanılır, elektron poçt vasitəsilə lazımi ünvana göndərilir və yalnız müstəsna hallarda kağızda çap edilir.

Bu gün kitablar artıq rəqəmli formada yaradılır. Bu cür "elektron kitablar"ı kompüter şəbəkələrindən köçürmək və fərdi kompüterin, yaxud xüsusi cib kompüterinin monitorundan oxumaq olar. Bu gün oxucu kitab üçün mağazaya getməyə, yaxud onu poçt vasitəsilə sifariş etməyə vərdiş edibsə, sabah nəşriyyata müvafiq məbləği (kompüter vasitəsilə) ödəməklə həmin kitabı ekrandan oxumaq və onun bir nüsxəsini öz printerində çap etmək hüququ əldə edəcək, ya da məbləği bilavasitə müəllifə ödəmək öhdəliyi götürməklə kitabı onun kompüterindən köçürə biləcək (əlbəttə, kitab xoşuna gələrsə).

Müasir kompüter həmçinin sonsuz sayda statik görüntüləri – şəkil və fotoqrafiyaları özündə saxlayır. Görüntünü işığa həssas filmdə (plyonkada) qeydə alan ənənəvi optik fotoaparatlar hələ də geniş yayılıb. Amma artıq görüntünü birbaşa rəqəmli şəkildə qeydə alan və onu öz elektron yaddaşına köçürən filmsiz (və ümumiyyətlə, hərəkətedici hissələri olmayan), rəqəmli fotokameralardan aktiv şəkildə istifadə edilməyə başlanıb. Rəqəmli formada çəkilmiş kadrları fərdi kompüterə köçürmək, ekranda baxmaq, printerdə çap etmək, yaxud xüsusi proqramın köməyilə redaktə edərək kompüterdəki ailə albomuna yerləşdirmək olar. Tezliklə (150 il ərzində, demək olar ki, dəyişmədən mövcud olan) ənənəvi fotoaparatlar mexaniki arifmometrlər və lampalı radioqəbuledicilər kimi muzey eksponatına çevriləcək.

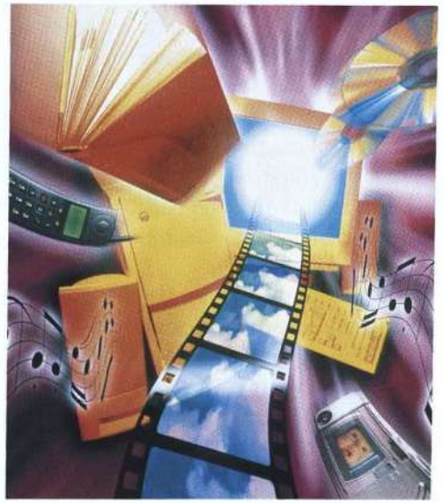
Kino və səsyazma texnologiyalarını da belə bir aqibət gözləyir. Yeni rəqəmli texnologiyalar burada da köhnə analoq texnologiyalarını sıxışdırıb aradan çıxa-

XIX-XX ƏSRLƏRİN BAŞLICA ELMİ VƏ TEXNOLOJİ NAİLİYYƏTLƏRİ

- İnformasiyanın işlənməsi üçün universal qurğunun – kompüterin yaradılması ideyası.
- İstənilən informasiyanın rəqəmli, ikilik kodlaşdırılmasından istifadə etməklə kompüterin elektron sxemlərdə gerçəkləşdirilməsi ideyası.
- İstənilən informasiyanın ikilik formada verilməsi üçün standartların yaradılması.
- İnformasiyanın işlənməsi alqoritmlərinin yazılması üçün proqramlaşdırma dillərinin yaradılması və insan tərəfindən yazılmış alqoritmlərin ikilik proqramlara çevrilməsinin avtomatlaşdırılması.
- Kompüterləri öz aralarında əlaqələndirən ümumdünya şəbəkəsinin yaradılması.
- Mikroelektronika texnologiyası (elektron sxemlərin vahid kristalda reallaşdırılması; bunun nəticəsində milyonlarla və milyardlarla dəyişdirici elementlərdən ibarət elektron sxemləri milyon tirajlarla hazırlamaq mümkündür).
- Diskretləşdirmə texnologiyası (istənilən növ informasiyanın rəqəmli formada təqdimatı). İnformasiyanın analoqlu, fasiləsiz yazılış, ötürülmə və işlənmə metodları rəqəmli metodlarla əvəz olunur, nəticədə informasiyanın saxlanma və ötürülmə zamanı təhrif olunması aradan qalxır, informasiyanı istənilən sayda oxumaq və üzünü köçürmək mümkün olur.
- Ümumdünya şəbəkəsinin yaradılması üçün bütün müasir rabitə növlərinin istifadə etdiyi rəqəmli kommunikasiyalar texnologiyası.



Funksional savadsız insanlar indi də var. Məsələn, harasa getmək istəyən bir nəfər dəmiryol vağzalına gəlir, amma burada qatarların hərəkət cədvəlindən baş çıxara bilmir. Eynilə bunun kimi elektron funksional savadsızlıq da mövcuddur. İnsanın sosial statusu onun müasir informasiya texnologiyalarını necə mənimsəməsilə deyil, yeni texnologiyalara necə uyğunlaşa bilməsilə müəyyən olunur.



İNFORMASİYA NƏDİR

XX yüzilin sonlarından "informasiya" sözü nə qədər geniş işlədilirsə, bir o qədər qeyri-müəyyən görünür. "Fəlsəfi ensiklopedik lüğət"də bu sözün dörd müxtəlif mənası verilir:

• işlərin vəziyyəti haqqında bilgi, xəbər; hər hansı bir şey haqqında insanların verdiyi məlumat;

• məlumatların alınması nəticəsində azalan, aradan qaldırılan qeyri-müəyyənlik;

• idarəetmə ilə qırılmaz bağlı olan məlumat; sintetik, semantik və praqmatik xarakteristikalardan ibarət siqnallar;

• canlı və cansız təbiətin istənilən obyektlərindəki müxtəlifliklərin əksi (inikası).

"İnformasiya – insanlar arasında, insanla avtomat arasında, avtomatla avtomat arasında məlumat mübadiləsini; heyvanlar və bitkilər aləmində siqnalların mübadiləsini; hüceyrədən hüceyrəyə, orqanizmdən orqanizmə əlamətlərin ötürülməsini (genetik informasiyanı) özündə əks etdirən ümumi elmi anlayışdır".

"Böyük ensiklopedik lüğət", 2001.

1948-ci ildə amerikalı alim və mühəndis Klod Şennon rabitənin riyazi nəzəriyyəsinin əsasları – informasiya nəzəriyyəsi haqqında elmi işini dərc etdirəndən sonra, həmin nəzəriyyə informasiyanın ötürülməsinin önəm daşıdığı bütün elm sahələrində tətbiq edilməyə başladı. Təxminən həmin illərdə, sonra-

racaq. Hərəkətedici hissələri olmayan rəqəmli videokameralar və maqnitofonlara keçid üçün cəmi bir neçə il gərəkdir. Lakin kompüterlər hələ ki sonsuz sayda filmləri saxlaya və ötürə bilmir.

Yeni informasiya texnologiyalarına keçid müxtəlif mövqelərin, texnologiyaların, standartların, proqram təminatı protokollarının və avadanlıqların rəqabəti şəraitində baş verir. İnformasiyanın təqdim edilmə formalarının və onlara müraciət metodlarının müxtəlifliyi qaçılmazdır. Sürətlə yeniləşən dünya ilə ayaqlaşmaq üçün informasiya infrastrukturlarına dair baza biliklərini və onlarla praktiki iş vərdişlərini ömür boyu öyrənmək və yeniləmək gərəkdir.

Yeni minilliyin tam hüquqlu üzvü lokal şəbəkələrdən tutmuş dünya miqyaslı şəbəkələrə qədər istənilən səviyyəli kompüter şəbəkələrilə hər gün və səmərəli surətdə təmasda olmalıdır. İnformasiya infrastrukturları ilə işləmək bacarığının səviyyəsi bir çox hallarda şəxsiyyətin sosial statusunu müəyyən edəcək.

İNFORMASİYANIN ATOMLARI

İnformasiya hansı üsulla əldə edilməsindən asılı olmayaraq, həmişə fərqləri və dəyişiklikləri qeydə alır. "Gecdir" ona görə məna kəsb edir ki, "tezdir" mövcuddur; yaz qışdan sonra gəlir. Biz qarşılaşdırmalarla düşünürük, bu da insan təfəkküründən fundamental xüsusiyyətidir. Axı təbiətdə heç bir konkret an mövcud deyil ki, yaz qışı əvəz etsin. Dünyanın dərkolunmasında passiv olmayan, ancaq ondan informasiya götürən insan kəsilməz gerçəkliyi xəyali hədlər, sərhədlər, böhran nöqtələri vasitəsilə elementlərə bölür ki, həmin sərhədlərədək nə isə "bir cür" idi, amma sonra "başqa cür" oldu. İstənilən "hə" yalnız "yox"la qarşılaşdırmada, istənilən "çox" yalnız "az"la qarşılaşdırmada, "müharibə" – "sülh"lə, "saraylar" – "daxmalar"la və s. qarşılaşdırmada məna kəsb edir.

Beləliklə, hər dəfə informasiya yaradarkən, yaxud onu ötürərkən biz tamı diskret edirik, elementləri hissələrə ayırırıq. Bəzi hallarda diskretlik aşkardır: sirena qoşulub – qoşulmayıb, əlini uzatdı – uzatmadı. "Ya – ya da" prinsipinə görə qurulmuş bu cür əks qütblüklər binar adlanır. Bu cür hallarda bir vəziyyət başqasından təbii, aşkar şəkildə ayrılır, onlar arasındakı fərq "özünü gözə soxur" (sirena misalında – "qulağa"). Yazın gəlişilə bağlı misalda isə hər şey daha mürəkkəbdir. Dünyada aşkar hüdudları olmayan çoxlu proseslər var. Özümüz bildiyimiz diskretlik daxil etməklə bu hüdudları biz qoyuruq. "Saçın çox uzanıb, – anası deyir, – qırxdırmaq gərəkdir". – "O qədər də çox deyil, gözümə ki tökülmür, – oğlu cavab verir. – Hələ belə gəzəcəyəm". Burada müxtəlif situasiyalar əlamətlərin sayına, yaxud dərəcəsinə görə bir-birindən ayrılır. Tutaq ki, saçın qalınlıq dərəcəsi belə verilib: daz – sıfırıncı dərəcə, qısa vurulmuş saç – birinci dərəcə, bir qədər uzanmış saçlar – ikinci dərəcə, gözə tökülən saçlar – üçüncü dərəcə, hörülmüş saçlar – dördüncü dərəcə və s. Dərəcəli əks qütblük belə təşkil olunur. Kəsilməzliyin bölünmə üsulları müxtəlif olur. Aydındır ki, dərəcəli əks qütblük istənilən qaydada müəyyən olunarsa, bu informasiyanın ötürülməsinin ən mühüm şərtinin – dəqiqliyin yerinə yetirilməsinə imkan verməyəcək. Fərz edək ki, vahid təqvim mövcud deyil, yazın nə vaxt gəlməsini, bu günün nə vaxt sona çatacağını və sabahkı günün nə vaxt başlanacağını hər kəs öz şəxsi hisslərinə əsaslanaraq müəyyən edir. İnformasiyanın dəqiqliklə ötürülməsini təmin edən və hamı üçün ümumi olan işarə sisteminin yaradılması – insanlar tərəfindən həllinə dəfələrlə cəhd göstərilmiş məsələdir. Kəsilməzliyi kəsilən kimi göstərən bu cür sistemlər olmadan müasir həyatı təsəvvür etmək mümkün deyil, çünki bu, not savadıdır, pul sistemidir, zamanın hesablanması sistemidir, coğrafi koordinatlardır, temperaturlar şkalasıdır və çoxlu başqa şeylərdir.



Ferdinand de Sossür.

İlk dəfə qarşılaşdırmanın fundamental əhəmiyyətini dili işarə sistemi kimi öyrənən, müasir linqvistikanın banisi Ferdinand de Sossür (1857-1913) izah etdi. Məhz o, göstərdi ki, heç bir işarə öz-özlüyündə məna kəsb etmir, o, yalnız və yalnız başqa işarələrlə əkslikdə (qarşılaşdırmada) mövcuddur. Bir fanarlı işıqfor, bir hərfdən ibarət əlifba ola bilməz. Hətta sizə elə gələ bilər ki, cəmi bir işarə var, amma əslində, bu belə deyil. Məsələn, sirena – "diqqət", yaxud "təhlükə"ni bildirən səs işarəsidir. Ancaq dərindən düşünsək, aydın olar ki, burada da qarşılaşdırma var: səsin olması – səsin olmaması. Sirenanın susmasının da əhəmiyyəti var, o, "hər şey həmişə olduğu kimidir", "heç bir xüsusi hadisə baş vermir" kimi mənaları ifadə edir. Xarici forması obyektin olmamasından ibarət olan belə işarələrə sıfırıncı işarələr deyilir. Sıfırıncı işarələrə örnək olaraq sözlər arasındakı boşluğu (pauzanı), cihazdakı yanmayan lampaları, hörmətsizlik əlaməti olaraq uzadılmayan əli göstərmək olar.



dan informasiyanın işlənməsi üçün universal vasitəyə çevrilən ilk kompüterlər meydana çıxdı. Tədricən informasiya haqqında bu təlim kibernetika və informatikadan fizikaya, kimyaya, biologiyaya, fəlsəfəyə, məntiqə və digər elm sahələrinə keçməyə başladı. "İnformasiya" anlayışı artıq elmin fundamental anlayışlarından birinə çevrilib və alimlərin çoxsaylı elmi işlərində bu anlayışa müxtəlif baxışlar var, ona müxtəlif təriflər verilir.

"İnformasiya – ilkin mənbəyi təcrübə olan xüsusi məlumatların məcmusudur".

A.A.Krasovski, Q.S.Pospelov, 1961-ci il.

Bəs bu qarmaqarışıqlıqdan necə baş çıxarmaq olar? Biz bir, yoxsa bir neçə anlayışdan istifadə etməliyik? Bu anlayışlar arasında hansısa ortaq nöqtələr mövcuddurmu? Bütün bunları aydınlaşdırmağa çalışacağıq.

İlkin və daha dar mənada informasiya – şüurlu məxluqatın, insanların atributudur. O, təcrübə, müşahidə, yaxud düşüncə yolu ilə əldə edilən və başqasına, yaxud özünə verilmək üçün maddi formada qeyd olunmuş hansısa xəbərlər, məlumatlar, faktlardır. Bu anlamda istənilən informasiyada iki komponentin – məzmun və maddi hissənin olması labüddür. Yəni, birincisi, o kimlər üçün nəzərdə tutulubsa, onlar üçün aydın olmalıdır (məzmun komponenti), ikincisi, bu və ya digər fiziki daşıyıcıda, məsələn, kağızda təqdim edilməlidir (forma komponenti).

Aydındır ki, eyni bir informasiyanı tamamilə müxtəlif cür (məsələn, türk, yaxud ingilis dilində) və müxtəlif daşıyıcılarda (kitabda, maqnit lentində) təqdim etmək olar.

Ancaq informasiyanın məzmun komponentini bir tərəfə qoysaq və eyni zamanda onu istənilən konkret fiziki daşıyıcısından təcrid etsək, onda hələ də araşdırma predmeti ola biləcək nəsə qalacaq. Bu, formal komponentdir. O yalnız kodlaşdırma ideyasından ibarətdir, yəni kodlaşdırılan informasiyanın mənasından və informasiya daşıyıcısının fiziki xüsusiyyətlərindən asılı deyil. Formal komponent üçün yalnız informasiyanın kəmiyyət, tezlik-ehtimal və struktur xassələrinin əhəmiyyəti var. Məsələn, informasiya ehtimallı və az ehtimallı, mürəkkəb ola bilər, yəni bir deyil, bir neçə asılı olmayan faktdan ibarət ola bilər.

Əslində "0"ın yekun kodlaşdırılmasında "Sizin oğlunuz olub" informasiyası yoxdur, ancaq biz onu əlavə informasiya – "Doğum evindən məlumat" əsasında bərpa edə bilərik.

İnformasiyanın formal xassələrini riyazi metodlarla araşdırmaq olar; informasiya nəzəriyyəsi bu cür yaranıb. Çox yaxşıdır ki, bu tədqiqatların nəticələri istənilən proseslərə və hadisələrə, o cümlədən sözün dar mənasında informasiya prosesləri və hadisələrinə aid olmayan, ancaq yalnız informasiya proseslərinin formal xassələrinə malik olan proseslərə və hadisələrə də tətbiq oluna bilər. Ona görə də bu gün "informasiya" sözünün olduqca geniş mənası olması tamamilə məntiqlidir. Bəzi tədqiqatçılar informasiya anlayışını hətta təbiətdəki və cəmiyyətdəki müxtəliflikləri, ümumilikdə nizamlılıqla əlaqələndirirlər.

İNFORMASİYANIN MİQDARI

Hər hansı kəmiyyəti ölçmək üçün, birincisi, ölçü vahidini və etalonu seçmək, ikincisi isə istənilən kəmiyyəti etalonla ifadə etməyin prosedurunu müəyyənləşdirmək gərəkdir.

Əgər bu qayda ilə informasiyanın ölçüsünü müəyyənləşdirməyə cəhd göstərilsə, bunu müxtəlif üsullarla etməyin mümkünlüyü aydınlaşar.

Sadə bir halı nəzərdən keçirək. Tutaq ki, informasiya – sıfırların və vahidlərin istənilən ardıcıllığıdır. Onda belə ardıcıllıqda bir yeri (mərtəbəni) məntiqi olaraq informasiya vahidi, ardıcıllığın uzunluğunu isə informasiyanın miqdarı hesab etmək olar.

Məsələn, doğum evindən belə bir xəbər gəlir: "Sizin oğlunuz oldu". Kompüterdə qəbul olunduğu kimi hər bir hərfi (o cümlədən, sözlərarası boşluğu da) bir baytla kodlaşdırsaq, göstərilən məlumat 18 bayt, yaxud $18 \cdot 8 = 144$ bit yer tutacaq. Məlumatı "Sizdə oğlandı", yaxud, sadəcə, "Oğlan" şəklində qısaltsaq həmin qiymət uyğun olaraq 104 və 40 bit olacaq. Qabaqcadan razılaşılsa ki, 0 "oğlan", 1 isə "qız" deməkdir, onda bütün informasiya 1 bitə sığışacaq.

İndi bu cür yanaşmanın əsas catışmazlığı aydın olur: informasiyanın miqdarı onun məzmunu ilə deyil, istifadə olunan kodla müəyyənləşir. Kodun

İNFORMASİYANIN TƏQDİM EDİLMƏSİ

İnformasiyanın saxlanması, çevrilməsi və ötürülməsi üçün ciddi tərifdənsə, onun cürbəcür təqdim olunması və onlardan istifadə metodları daha vacibdir.

İnformasiya işarələr, simvollar və kodlar şəklində mütləq şifrələnməlidir. Kodlaşdırma dəfələrlə və çeşidli məqsədlərlə aparıla bilər: bu və ya digər rabitə kanalları ilə ötürməni təmin etmək, onlardan hər hansı birinin yükünü azaltmaq, informasiyanın məxfiliyini saxlamaq və s. Bir dildən başqasına tərcümə də kodlaşdırmadır ki, burada da məqsəd dili bilməyən, yəni informasiyanın ilkin təqdimatı ilə tanış olmayan insanın informasiyanı qavramasıdır.

Son zamanlar IP-telefoniya geniş yayılmağa başlayıb. İstənilən şəxs öz kodu olan kart alıb başqa şəhərə, yaxud başqa ölkəyə zəng edə bilər. Danışıq zamanı belə bir informasiya prosesi baş verir. Öncə informasiya insanın beynində ardıcıl olaraq obrazlar, fikirlər, nitqlər şəklində şifrələnir. Telefon qurğusunun köməyilə rəqslər (danışıq) elektrik impulslarına çevrilir. Kompüter stansiya-kommutatorlarında analoq elektrik impulsları rəqəmli koda çevrilir və o, ötürülməni tezləşdirmək üçün xüsusi alqoritmlər vasitəsilə emal olunur. Sonra alana düzgün çatdırılması üçün kod yenidən kodlaşdırılaraq İnternetlə ötürülür. Qəbuledici stansiyada əks proses – kodaçma baş verir: rəqəmli kod analoq elektrik impulsuna, sonra telefonda səs rəqslərinə çevrilir ki, nəhayət, ikinci həmsöhbət tərəfindən başa düşülür.

Bu cür uzun zəncirdə informasiyanın təqdim edilməsi dəyişilib, ancaq məqsəd (göndərilən informasiyanın məğzi) dəyişməz qalıb.

Dünyada informasiyanın cürbəcür növləri mövcuddur. Onların insan qavrayışına əsaslanan təsnifatına baxaq:

1. Vizual (görmə) informasiya. Belə informasiyanı, əlbəttə, təkcə insan yox, həm də heyvanlar anlayır. Ətraf aləmin başlıca olaraq yalnız əksini mənimsəyən heyvanlardan fərqli olaraq insan əyani informasiyanı öncə qayaüstü rəsmlər, sonra rəssamlıq əsərləri, fotoqrafiya, video şəklində yaza, saxlaya bildi. İnkişaf irəliyə doğru gedir, XXI əsr yazı texnologiyalarının yaradıldığı və geniş şəkildə tətbiq edildiyi müstəvi təsvirlərin deyil, əsl üçölçülü təsvirlərin hazırlandığı dövrdür. Bu təsvirlərə yandan, yuxarıdan baxmaq olar, onun arxasına keçib o biri tərəfdə nə baş verdiyini görmək olar.

Artıq doğmalarınızın, ev heyvanlarınızın holoqrafik, psevdoüçölçülü fotoqrafiyasını hazırlamağı təklif edən fotostudiyalar meydana çıxıb.

2. Akustik informasiya (nitq, səs). Demək olar ki, bütün heyvanlar bu və ya digər dərəcədə səs qavrama qabiliyyətinə malikdir: onlar bir-birlərini təhlükə barəsində xəbərdar edir, cəmiyyətdə öz statuslarını müəyyən edir, hətta, ola bilsin, birgə ov zamanı öz hərəkətlərini uzlaşdırırlar. Ancaq təkcə insan nitq vasitəsilə hisslərini, fikirlərini ifadə edir, müəyyən dərəcədə informasiyanın vizual qavranılmasını əvəz edə bilir. Danışan nəyisə gözəl (bəlağətli), aydın, canlı təsvir edərsə, dinləyici həmin predmeti elə təsəvvür edə bilər ki, sanki onu görür. Kompüterdə nitq istənilən başqa bir səs, yaxud musiqi kimi verilə bilər, ya da ayrı-ayrı səslərə – fonemlərə, yaxud hətta bütöv sözlərə bölünə bilər.

3. Simvollu informasiya (mətn, yazı). Yazının köməyilə insan bir növ ölməzlik qazandı: papirusa, perqamentə, kağıza, müxtəlif rəqəmli daşıyıcılara bir dəfə yazılmış fikirlər, düzgün şəraitdə saxlanılarsa, demək olar ki, həmişəlik qalacaq.

İstənilən bu cür təsnifat tam olmayacaq – həmişə informasiyanın elə təqdimat forması var ki, yuxarıda qurulmuş sxemə uyğun gəlmir. DNK-da olan bioloji informasiyanı hara aid etməli? Bəs, Yer qabığında olan və onun dərinliklərindəki tektonik proseslərə dəlalət edən gərginlik? Bəs, uzaq ulduzların növünü, tərkibini, onlarda gedən prosesləri müəyyənləşdirməyə kömək edən onların şüalanmasını?

Buna görə də informasiyanı istifadəsindən asılı olaraq təsnif etmək gərəkdir. Bu, istər insanın başa düşdüyü informasiya olsun, istər kompüterin emal etdiyi informasiya, istərsə də bəşəriyyətin çoxəsirlik bilikləri – hər yerdə məqsədli təsnifat aparmaq gərəkdir.





"İnformasiya – təsirin məzmunudur, onun kəmiyyətidir, mürəkkəb sistemlərin rabitə vasitəsi kimi istifadə edilən məkan və zamanda dəyişiklikdir".

N.M.Amosov, 1963-cü il.

optimallığından, yaxud artıqlığından asılı olaraq eyni bir məlumatdakı informasiyanın miqdarı az, yaxud çox olacaq.

Başqa yolla gedək. Hər şeydən öncə bu və ya digər informasiyanı konkret təqdimetmə üsulundan təcrid edək. Tutaq ki, informasiya (məlumat) A-dan (göndərəndən) B-yə (alana) ötürülür. Məlumatın həqiqi uzunluğundan və onun kodlaşdırılması üsulundan asılı olmayaraq, hesab edəcəyik ki, alan hər hansı fakt haqqında həmin məlumatdan nəsə yeni bir şey öyrənir (verilmiş faktla bağlı qeyri-müəyyənlik dərəcəsi azalır). Ona görə də tutaq ki, informasiyanın miqdarı həmin informasiyanın alınması nəticəsində onun qeyri-müəyyənlik dərəcəsinin azalmasından başqa bir şey deyil. Bu, hələ informasiyanın miqdarının tərifi deyil. Bundan sonra biz yuxarıda göstərilənləri dəqiqləşdirərək, addım-addım əsl tərifi almağa çalışacağıq.

1-ci addım. *İki eyniqiymətli (eyniehtimallı) variantdan ibarət qeyri-müəyyənlik.* Məsələn, oğlan və ya qız uşağı doğuldu, metal sikkənin ön və ya arxa üzü düşdü. İnformasiyanın alınması nəticəsində qeyri-müəyyənlik aradan qalxırsa (bizə iki variantdan məhz hansının baş verdiyini bildirdilər), hesab edəcəyik ki, bir vahid informasiya alınıb (informasiyanın ölçü vahidi "bit" adlanır).

2-ci addım. *N sayda eyniehtimallı variantdan ibarət qeyri-müəyyənlik.* Məsələn, xüsusi zər oyununu götürək. İntuitiv olaraq aydındır ki, bu halda qeyri-müəyyənlik yüksəkdir. Amma nə qədərdir? Bax, elə məsələ də bundadır! Tutaq ki, $N=8$. Onda cəmi üç suala cavab almaqla ("hə", yaxud "yox") səkkiz variant arasında birmənalı seçim etmək olar. Birinci sualla biz variantların sayını 8-dən 4-ə, ikinci sualla 4-dən 2-yə, üçüncü sualla 2-dən 1-ə azaldırıq. Hər bir cavabdan sonra biz 1 bit informasiya əldə edirik. Deməli, cəmi 3 bit.

Daha ümumi halda, $N=2^k$ olarsa, onda k sayda bit alırıq. Yəni bir neçə eyniehtimallı variantdan ibarət qeyri-müəyyənliyin aradan qaldırılması üçün zəruri olan informasiyanın miqdarı: $k=\log_2 N$.

3-cü addım. *İki müxtəlif ehtimallı variantdan ibarət qeyri-müəyyənlik.* Tutaq ki, səkkiz eyniehtimallı variant verilib. Bizə yalnız onlardan hansının (birincinin, yaxud hansısa başqasının) qəbul olunduğunu bilmək vacibdir. İki variant, özü də müxtəlif ehtimallı variantlar arasında seçim edilir. Tutaq ki, cavab belədir: "Hə, birincidir". Məntiqi olaraq hesab etmək olar ki, 2-ci addımda olduğu kimi, biz 3 bit informasiya əldə etdik, çünki başlanğıcdakı səkkiz variantdan ibarət qeyri-müəyyənlik bir varianta qədər azaldı.

"İnformasiyanın miqdarı anlayışı tamamilə təbii olaraq statistik mexanikanın klassik anlayışı – entropiya ilə əlaqələndirilir. Sistemdə informasiyanın miqdarı sistemin mütəşəkkillik ölçüsü olduğu kimi, entropiya da sistemin qeyri-mütəşəkkillik ölçüsüdür. Biri digərinə əks işarə ilə bərabərdir".

N.Viner, "Kibernetika".

İndi, fərz edək ki, cavab belədir: "Yox, birinci deyil". Qeyri-müəyyənlik tamamilə aradan qaldırılmasa da, hər halda o, 8 variantdan 7 variantadək azaldı. Deməli, artıq biz nə qədərsə, tutaq ki, x bit informasiya əldə etdik.

Qalan yeddi variantdan ibarət qeyri-müəyyənliyi aradan qaldırmaq üçün bizə $\log_2 7$ bit informasiya da lazımdır. Onlar birlikdə 3 bit informasiya verir, çünki qeyri-müəyyənliyi 8 variantdan 1 varianta endirir. Ona görə də $x+\log_2 7=3$, yaxud $x=3-\log_2 7$.

Beləliklə, "Hə, birincidir" məlumatı bizə $3=\log_2 8$ bit informasiya verdi. Variantların sayını 8-dən 7-ə endirən "Yox, birinci deyil" məlumatı isə $3-\log_2 7=\log_2(8/7)$ bit informasiya verdi.

Ümumi halda, N sayda eyniehtimallı variant olduqda və k ($k<N$) variantdan birinin baş verməsi haqda məlumat varsa, ancaq onlardan məhz hansının olması məlum deyilsə, onda biz $\log_2(N/k)$ bit informasiya alırıq.

Ehtimal nəzəriyyəsindən məlumdur ki, k/N – verilmiş hadisənin p ehtimalıdır. Ona görə də yekunda alırıq ki, $\log_2(1/p)=-\log_2 p$.

Məlumatdakı informasiyanın miqdarı müəyyən mənada məlumatın optimal uzunluğudur.

KOMPÜTERLƏR İNFORMASİYA YARADIRMI

Bütün riyazi nəzəriyyələr kimi informasiyanın riyazi nəzəriyyəsi də məntiqi baxımdan mükəmməl və həm də ən müxtəlif sahələrdə əla "işləyən" nəzəriyyə kimi görünür. Ancaq, bununla belə, o, daxili ziddiyyətlərdən də azad deyil.

Belə bir paradoksu nəzərdən keçirək. Tutaq ki, kimsə belə bir məlumat alır: "İki dəfə iki – dördə bərabərdir". Bu, olduqca məzmunlu məlumatdır, vurma cədvəlinin bir hissəsidir. Bununla belə, formal baxımdan bu məlumatda

MƏZMUN İNFORMASİYASI



Y.A.Şreyder.

Rabitə nəzəriyyəsində informasiyanın miqdarı məlumatın ötürülməsinə çəkilən xərclər baxımından qiymətləndirilir. Verici-qəbuledici qurğuların işləməsi üçün birinci dərəcəli məsələ siqnalların mənası deyil, onların statik xüsusiyyətləridir: siqnalların müxtəlifliyi, onların gözlənilməzliyi, artıqlığı və s. İnsan üçün isə məhz ötürülən məlumatın məzmunu və ondan əldə edilən biliklərin, yəni məzmun informasiyasının önəmi var.

Məzmun informasiyasının əsasında gələcək simvollar, yaxud girişdəki obrazlar ilə zəruri fəaliyyət arasında uyğunluq hasil edilir və beyində möhkəmlənir. Ali heyvanlarda ətraf aləmdəki situasiyaların müxtəlif siqnalları (vəhşi heyvanın qoxusu, qənimətin qoxusu və s.) ilə bu situasiyalardakı məqsədyönlü fəaliyyət arasında kifayət qədər uyğunluq ("məzmun lüğəti" kimi bir şey) olmasaydı, bu canlılar mövcud ola bilməzdi. Bundan başqa, sürünənlərdə bu cür uyğunluq əsasən anadangəlmə olursa və onların davranışı fitri instiktlər əsasında formalaşırsa (irsi informasiya ilə müəyyən edilirsə), məməli heyvanlarda bu, bütün ömür boyu təkmilləşir və hissiyat orqanları vasitəsilə alınan informasiyalar daha çox məzmun xüsusiyyəti daşıyır, onların yaddaşındakı uyğunluq ehtiyatının artırılmasına xidmət edir. İnsanlarda məzmun informasiyası təkcə xarici siqnallar və hərəkətlər arasındakı deyil, eləcə də anlayışlar arasındakı uyğunluqların artırılmasına səbəb olur.

Beləliklə, məzmun informasiyasının ümumi cəhəti odur ki, o, informasiyanı qəbul edənin məlumat ehtiyatını, uyğunluq ehtiyatını dəyişir. İlkin (və informasiya daxil olduqca dəyişən) uyğunluq ehtiyatını hər hansı ümumiləşdirilmiş lüğət, yaxud soraqça kimi təsəvvür etmək olar. Y.A.Şreyder bunu "tezaurus" (*yun.* "tezaurus" – "xəzinə" deməkdir; təkcə sözlərin mənasının yox, həm də onlar arasında əlaqələrin verildiyi lüğətlər nəzərdə tutulur) adlandırmağı təklif edib. Məzmun informasiyasının miqdar ölçüsü kimi daxil olan informasiyanın təsiri altında qəbul edənin tezaurusunun dəyişilməsini götürmək təbii olardı.

İnformasiyanın miqdarı onu qəbul edəndən, onun tezaurusundan asılıdır. Qəbul edənin tezaurusu olduqca kasaddırsa, məlumatdakı informasiyanın miqdarı, ümumiyyətlə, sıfra bərabər olur. Məsələn, "Bu, ali məktəbin birinci kursu üçün ali riyaziyyat dərsliyidir" məlumatından üç yaşlı uşaq tamamilə heç nə başa düşməyəcək, yuxarı sinif şagirdi isə artıq nəyisə anlayacaq. Şübhəsiz, buradan maksimal informasiyanı dərsliyin nəzərdə tutulduğu kursun tələbəsi götürəcək. Maraqlıdır ki, qəbul edənin tezaurusu genişləndikcə mənimsənilən informasiyanın miqdarı azalmağa başlayır. Artıq ikinci kurs tələbəsi birinci kursun dərsliyini oxuduqda onun tezaurusu az dəyişilir, ali riyaziyyatı yaxşı bilən insan üçün isə bu dərslik heç bir informasiya daşımır.

İncəsənət, ədəbiyyat, təhsil sahələrində informasiya nəzəriyyəsinin metodları yalnız o halda səmərəli ola bilər ki, informasiyanın (təkcə statistik yox, həm də semantik və məzmun) miqdarının qiymətləndirilməsi yolları, eləcə də incəsənət əsərlərinin təsiri altında, biliklərin müəllimdən şagirdə ötürülməsi prosesində və s. hallarda insanın tezaurusunun (o cümlədən, onun emosional tezaurusunun) dəyişməsinin qiymətləndirilməsi üsulları işlənib hazırlansın.

Azərbaycan Respublikası Prezidentinin İşlər İdarəsi
PREZİDENT KİTABXANASI



İnformasiya (*lat.* informatio – aydınlaşdırma, izah etmə, məlumatlılıq) – elmin ən ümumi anlayışlarından biri olub, müəyyən məlumatı, hər hansı xüsusi biliklərin toplusunu və s. bildirir.

"Kibernetika ensiklopediyası", 1974.

heç bir informasiya yoxdur. Doğrudan da, iki dəfə ikinin beşə, altıya və s. deyil, məhz dördə bərabər olmasının ehtimalı 1-ə bərabərdir: $\log_2 1 = 0$.

Deməli, məlumatda 0 bit informasiya var. Eyni məntiqlə, formal baxımdan istənilən riyazi teoremdə, fizikanın istənilən qanununda, istənilən düzgün mühakimədə, yaxud isbatda olan informasiyanın miqdarı sıfra bərabərdir.

İldən-ilə kompüterlər daha da "ağıllanırlar". Onlar ustalıqla şahmat oynayır, hava haqqında proqnoz verir. Kompüter vasitəsilə teoremin isbat olunması halları da məlumdur (bax: "Riyaziyyat. Uşaq ensiklopediyası" cildliyindəki "Dörd rəng haqqında məsələ" əlavə oçerki). Kompüterlərin informasiya yaratması sanki aydındır. Lakin, məsələn, differensial tənliyin həllində olan informasiyanın hamısı, əslində, onun kompüterə buyruq vasitəsilə verilmiş yazılışında və başlanğıc şərtlərdədir. Kompüter, sadəcə, həmin gizli informasiyanı "aşkara çıxardır", bizim üçün daha münasib olan formaya çevirir.

Analoji olaraq, danışıqda "elektrik stansiyası elektrik enerjisi istehsal edir" ifadəsi təbii kimi görünür. Amma fizika kursundan enerjinin saxlanması qanunu məlumdur və əslində, elektrik stansiyası heç bir enerji istehsal etmir, onu yalnız bir formadan digərinə çevirir.

İNFORMASİYA NƏZƏRİYYƏSİ

Məlumatların ötürülməsi insan fəaliyyətinin ən mühüm növlərindən biridir. Teleqrafin, telefonun və İnternetin ixtira edilməsi bəşəriyyətin özünün varlığını dəyişdirdi. Məlumatların ötürülməsi prosesini təhlil etsək, bütün rəngarəngliklərə (danışıqdan tutmuş kompüter komandalarına qədər) baxmayaraq, iki əsas mərhələni ayırırlar. Öncə məlumat ötürülmək (saxlanmaq) üçün əlverişli olan formaya salınır, sonra ona müəyyən verilənlər artırılır ki, ötürmə zamanı heç bir maneə qəbulun keyfiyyətinə təsir göstərməsin. Birinci mərhələyə məlumat mənbəyinin kodlaşdırılması, ikinci mərhələyə maneələrə dayanıqlı kodlaşdırma deyilir.

Nümunə kimi kompüterdə yığılmış adi mətni götürsək, onda əlifbanın hərflərinə, eləcə də durğu işarələrinə, rəqəmlərə sıfır və birlərin müəyyən ardıcıllığını uyğun qoymaq olar. Bu düzümə iki işarənin – sıfır və birin kombinasiyasından ibarət söz kimi baxmaq daha əlverişlidir (bu cür ikilik sözlərin mərtəbələrinə "bit" deyilir). Azərbaycan dilinin bütün hərflərini, rəqəmləri, durğu işarələrini və s. yazmaq üçün 7 bit bəs edir (7 bit $2^7=128$ müxtəlif məlumatı yazmağa imkan verir), ancaq verilmiş hərfin ötürülməsi (yazılması, saxlanılması) zamanı baş verən bir çox səhvləri (hamısını yox) aşkara çıxartmağa imkan verən əlavə yoxlama bitilə birlikdə 8 bitdən istifadə edilir. Musiqinin CD-yə yazılması nümunəsi daha çox təsir bağışlayır, belə ki, bu zaman hətta cızıqlar da, demək olar ki, səslənmənin keyfiyyətinə təsir göstərmir (bu, maneələrə dayanıqlı kodlaşdırma sayəsində mümkün olmuşdur).

Məlumatın ötürülməsinin riyazi nəzəriyyəsi, yaxud informasiya nəzəriyyəsi məşhur amerikan alimi və mühəndisi Klod Şennonun (1916-2001) işlərində formalaşmışdır.

Şennon ehtimal nəzəriyyəsinin ideya və metodlarından geniş istifadə etmiş və informasiyanın miqdarı kimi çox mühüm entropiya anlayışını vermişdir.

İnformasiya mənbəyinin kodlaşdırılmasına ən populyar örnəklərdən biri teleqrafiyada tətbiq edilən Morze əlifbasıdır (yaxud kodudur). Burada müxtəlif hərflərə müxtəlif ikilik sözlər qarşı qoyulur (tire və nöqtələr sıfır və birlərlə əvəz edilir), çox istifadə edilən hərflərə ("A", "O" kimi) qısa sözlər, az işlənən hərflərə isə ("J" kimi) uzun sözlər uyğun gəlir. Bu, Morzeyə ikilik sözlərin orta uzunluğunun minimal mümkün olan həddə yaxın etməyə imkan vermişdir. Bəs minimal orta uzunluq nə qədərdir? Şennonun yolu ilə gedərək bu məsələnin ehtimalla qoyuluşuna baxmaq olar. Tutaq ki, P_i – əlifbanın i-ci hərfinin sözlərdə rast gəlinmə tezliyidir (ehtimalıdır). Sadə olsun deyə, hesab edək ki, əlifbanın i-ci hərfi sözlərdə əvvəlki hərflərdən asılı olmayaraq görünür (əlbəttə, istənilən təbii dildə, o cümlədən Azərbaycan dilində müəyyən qaydalar mövcuddur). Əlifbanın hərflərinə ikilik sözlər uyğun qoyulur. Məsələn, B hərfinə (0) sözü, U hərfinə (1) sözü, O hərfinə (01) sözü uyğun qoyulduqda belə bir çözülməz vəziyyət alınır: qabaqcadan məlum deyil ki, (01) şəklindəki ikilik yazı ilə nə kodlaşdırılıb – "O" sözü, yoxsa "BU" sözü.

Birqiymətli kodlaşdırmanı təmin etmək üçün ən sadə üsul yalnız prefiks kodlara baxmaqdır ki, bu halda heç bir kod sözü başqa kod sözünün başlanğıcı olmur. Yuxarıda göstərilən misal, şübhəsiz, prefiks kodu deyil. Ən sadə prefiks kodu bərabərölçülü koddur ki, burada bütün sözlərin uzunluğu eynidir. Kod sözlərinin uzunluğu eyni olmayan prefiks koduna bir qədər mürəkkəb misal:

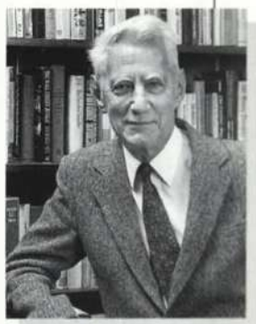
$$A \to 0;\ B \to 10;\ C \to 110;\ D \to 111.$$

Bu kodun sözlərindən ibarət olan ikilik ardıcıllığı dekodlaşdırmaq üçün birinci simvoldan başlamaq və yalnız o kod sözlərini saxlamaq lazımdır ki, o, onlar üçün başlanğıc olsun. Belə sözlər bir neçə olarsa, onda ilk iki simvolu nəzərdən keçirmək və yenə də yalnız verilən simvollarla başlayan sözləri saxlamaqla mümkün sözlər siyahısını ixtisar etmək lazım gələcək. Sonda ilk bitlərin bir neçəsilə (bizim misalda üçdən çox olmayan) üst-üstə düşən yeganə kod sözü alınacaq, bütün qalan kod sözləri isə həmin mövqelərdə qəbul edilmiş bitlərdən fərqlənəcək. İndi həmin bitlərin "üzərindən xətt çəkilir" və prosedur davam etdirilir. Beləliklə, 100111 ikilik ardıcıllığı üçün:

$$100111 = (10)0111 = (B)(0)111 = (B)(A)(111) = (B)(A)(D) = BAD.$$

Müxtəlif (birqiymətli) kodlaşdırmaları səmərəlilik baxımından necə müqayisə etməli, səmərəlilik ölçüsü olaraq nəyi götürməli?

Şennon kodlaşdırmanın orta uzunluğunu (kod sözlərinin orta uzunluğunu) – $E(L)=p_1L_1+\ldots+p_mL_m$, burada L_i – i-ci



Klod Elvud Şennon.





sözün uzunluğudur, nəzərdən keçirməyi və minimal orta uzunluqlu kodlaşdırmanı optimal hesab etməyi təklif etmişdir. Aydındır ki, A, B..., yaxud X, Y... hərflərinin necə adlanması mühüm deyil, mühüm olan yalnız neçə hərfin olması və onların rast gəlinməsinin P_i ehtimalıdır, yəni hər şey ehtimalların paylanması $P=\{p_1, \dots, p_m\}$ ilə müəyyən edilir. Ehtimalların paylanması ilə verilmiş hərflər çoxluğuna (əlifbaya) məlumatların mənbəyi deyilir.

Dörd hərfdən ibarət olan (A, B, C, D) və ehtimalların paylanması ($P(A) = 1/2$, $P(B) = 1/4$, $P(C) = P(D) = 1/8$)-dir. Bu isə belədir: $P(A) + P(B) + P(C) + P(D) = 1/2 + 1/4 + 1/8 + 1/8 = 1$. Kodun orta uzunluğu $E(L) = p_1L_1 + \dots + p_mL_m$ 7/4-ə bərabərdir ki, bu da bərabərölçülü kodlaşdırma üçün lazım olandan 2 bit azdır.

$P = \{p_1, \dots, p_m\}$ ehtimalların paylanmasına informasiya mənbəyinin entropiyası adlanan və qeyri-müəyyənlik ölçüsü rolunu oynayan $H(P)=p_1\log_2(1/p_1)+\dots+p_m\log_2(1/p_m)$ ədədi qarşı qoyulur. Bu düsturda daha çox iki əsaslı loqarifmlərdən istifadə edilir və bu zaman informasiya mənbəyinin qiymətinin bərpası (yəni qeyri-müəyyənliyinin aradan qaldırılması) üçün zəruri olan informasiya miqdarı bitlərlə ölçülür. Bir bit iki bərabər ehtimallı cavabı olan sualın cavabından alınan informasiya miqdarına uyğun gəlir.

Şennonun birinci teoreminə – küysüz kanal üçün kodlaşdırma teoreminə görə istənilən kodlaşdırma (birqiymətli kod-açma ilə) üçün onun $E(L)$ orta uzunluğu $H(P)$ entropiyasından az deyil, digər tərəfdən, $H(P)$-yə istənilən qədər yaxın olan orta uzunluqlu kodlaşdırma mövcuddur. Bununla bu teorem onu ifadə edir ki, mənbədən alınan məlumat, orta uzunluğu mənbənin entropiyasından azca böyük olan ikilik formada kodlaşdırıla bilər və entropiyadan "yığcam" heç bir məlumat mənbəyini təsəvvür etmək olmaz. $r(L)=E(L) - H(P)$ kəmiyyətinə L kodlaşdırmasının artıqlığı deyilir (birhərfli kodlaşdırmalar üçün $0<(L)<1$ olur).

Hərf-hərf kodlaşdırmalar üçün kodlaşdırma artıqlığını istənilən qədər kiçik etməyin mümkünlüyünü iddia etmək olmaz, ancaq kodlaşdırma artıqlığını 1-dən kiçik etməyin mümkünlüyünə zəmanət vermək olar. Kiçik kodlaşdırma



artıqlığı almaq üçün hərf-hərf kodlaşdırmadan blok kodlaşdırmasına keçirlər. Bu zaman əlifbanın n hərfindən ibarət bloklar kodlaşdırılır. Təhlilin sadəliyi üçün, tutaq ki, mənbə hərflərinə bir-birindən asılı olmadan rast gəlinir. Onda $A=(a_1, \dots, a_n)$ blokunun rast gəlinmə ehtimalı $P(A)=p(a_1)\cdot\dots\cdot p(a_n)$-ə, hərflər bloku ansamblının entropiyası isə $H(P^n)=nH(P)$-ə bərabərdir. Deməli, blok kodlaşdırılmaları üçün bir hərfə (mətnə) düşən kodun orta uzunluğu $H(P)$-dən az deyil, ancaq digər tərəfdən o,

$$\frac{1}{n}H((P^n)+1) = H(P) + \frac{1}{n} \text{-dən}$$

kiçikdir.

Beləliklə, kodlaşdırılan blokların uzunluğu artırıldıqda kodlaşdırmanın orta uzunluğu mənbənin entropiyasına çox yaxın ola bilər, kodlaşdırma artıqlığı isə yetərincə az ola bilər. Kodlaşdırma alqoritmlərinə informasiyanın sıxılması alqoritmləri də deyirlər, çünki bu alqoritmlər verilənlərdəki təbii artıqlığı aradan qaldırmaqla onları daha yığcam formada göstərməyə imkan verir.

Bu vaxta qədər informasiya (məlumat) mənbəyinin kodlaşdırılması məsələsinə, yəni informasiyanın yığcam təsvirini əlverişli şəkildə (məsələn, ikilik formada) göstərmək məsələsinə baxılırdı və hesab edilirdi ki, kodlaşdırılmış informasiya hər hansı təhrifə məruz qalmır – küy yoxdur. Bu, həm də Şennonun birinci teoreminin – "küysüz kanal üçün kodlaşdırma teoremi"nin niyə belə adlandırılmasına aydınlıq gətirir. Lakin, əlbəttə ki, real vəziyyətlərin çoxunda küy olur. Bu küy bəzən əhəmiyyət verilməyəcək qədər cüzi olsa da, olduqca tez-tez elə də olur ki, bizim seçdiyimiz təsvir (kodlaşdırma) formasının dəyişməsinə gətirib çıxarır. Onda artıq kod açmanın birqiymətliliyinə zəmanət vermək olmaz. Yaxud "kod açma" birqiymətli olacaq, ancaq yanlış məlumata (göndərilənə deyil) gətirib çıxaracaq.



Beləliklə, BAD sözü üçün birinci bitin dəyişdirilməsi onun ikili təqdim ediləni (yəni, vahid səhvi) olan 100111 aşağıdakı 000111 = (0)(0)(0)(111) = = AAAD-yə gətirib çıxaracaqdır.

Səhvlərin düzəldilməsi məsələsinin formal təsviri üçün Şennon informasiyanın ötürülmə kanalı anlayışını daxil etmişdir. Kanalın girişi vardır və ora X əlifbasının simvolları daxil olur (başqa simvolları kanal, sadəcə, qəbul etmir). Kanala "qara qutu" kimi baxılır, yəni kanalın daxilində nə baş verdiyinin əhəmiyyəti yoxdur (yaxud məlum deyil), yalnız nəticə – y çıxışı (y müəyyən bir Y əlifbasına aiddir) haqqında məlumat verilir. Beləliklə, kanal belə bir qanun əsasında təşkil olunur: girişdə x olarsa, çıxışda y olacaq (ehtimalların şərti paylanması $P(Y=y|X=x)$). Ən önəmli örnəklərdən biri ikilik simmetrik kanaldır (İSK) ki, burada giriş və çıxış ikilik ədədlərdir ($X=Y=\{0,1\}$) və $y=x$ olması ehtimalı (ikilik simvolun səhvsiz ötürülməsi ehtimalı) q-yə bərabərdir, yəni səhv ehtimalı $p=1-q$. Məlumatın belə kanalla etibarlı ötürülməsinin ən sadə yolu simvolların bir neçə dəfə təkrar edilməsidir (necə ki, telefon danışıqları zamanı dinləyən həmsöhbət sözü eşitmirsə, danışan sözü təkrar edir).

Tutaq ki, $p = 0{,}1$ və hər bir ikilik simvol $2m+1$ dəfə (məsələn, yeddi) təkrarlanır. Onda qəbuledici qərar verir ki, kanalın girişində daha tez-tez görünən ($4 = m + 1$ dəfədən az olmayaraq) simvol

HEMMİNQ KODLARI

Səhvləri düzəldən kodların tarixi yarım əsrdən bir qədər əvvəl – amerikalı gənc alim R.U.Hemminq ilk hesablayıcı maşınların birində məsələlər həll edərkən başlayıb. O dövrün kompüterləri etibarlı deyildi və tez-tez səhv edirdi. Artıq o zaman səhvləri aşkara çıxaran sadə üsul (cütlük yoxlaması, 0 və 1-dən ibarət olan hər bir maşın sözündəki vahidlərin sayı cüt olarsa həmin sözə 0-a bərabər olan, tək sayda olarsa 1-ə bərabər olan əlavə artıqlıq biti yazılır) xidmət edən heyətə hesablama prosesini yenidən başlatmağa imkan verirdi. Lakin istirahət günlərində, əsas maşın vaxtı Hemminqdə olanda kompüterə heç kim xidmət etmirdi və nəticədə çoxlu hesablamalar əbəs yerə aparılırdı. Həmin vaxt onun ağlına belə bir fikir gəlir ki, əgər kodlar səhvləri avtomatik aşkarlaya bilirsə, onda səhvləri düzəldə bilən kodlar da olmalıdır.

İndi yaxşı məlum olan və tək-tək səhvləri düzəldən [7,4]-Hemminq kodu belə yarandı. Bu kod 4 bit informasiyanı 7 bit informasiyanın köməyilə elə təsvir etməyə (kodlaşdırmağa) imkan verir ki, 7-bitlik təsvirdə olan ixtiyari tək-tək səhv düzəldiləcək və nəticədə ilkin 4 bitin düzgün bərpasına nail olunacaqdır.

4 bit informasiya $2^4=16$ müxtəlif məlumatı təsvir etdiyindən, Hemminq kodu, uzunluğu 7 olan 16 kod sözündən ibarətdir. Tək-tək səhvin düzəldilməsi xüsusiyyəti onu göstərir ki, istənilən bitin hər hansı kod sözündə və istənilən bitin (yaxud heç bir bitin) digərində təhrif olunması eyni bir 7-bitlik sözə gətirib çıxara bilməz. 7-bitlik $2^7=128$ sözdən ibarət çoxluq ("fəza") və onun daxilində 16 sözdən ibarət altçoxluq (kod) vardır. Hər bir kod sözünün ətrafında tək-tək səhvlər nəticəsində yaranan 7 sözdən ibarət "bulud" ("kürə") mövcuddur və bu "buludlar" ("kürələr") kəsişməməlidir. Tək-tək səhvlərin düzəldilməsinin sonuncu şərti istənilən iki kod sözünün ikidən artıq mövqedə fərqlənməsini nəzərdə tutur.

Hər bir kürədə 8 nöqtə (7 üstəgəl mərkəz), fəzadakı bütün sözlərin sayı isə 128 olduğundan, 128/8=16-dan çox olmayan sayda kəsişməyən kürələr yerləşdirmək olar, yəni *Hemminq kodu optimaldır*.

Hemminq kodu "$x_1, \dots, x_7$" yeddi məchullu dörd xətti tənlikdən ibarət həmcins sistemin (yəni sağ tərəf 0 olan) həllər çoxluğu kimi verilir. Bu sistemdə əmsallar matrisinin sütunları 0 və 1-dən ibarət sıfır olmayan 7 sütunun hamısında olur, sıfra bərabərlik isə iki moduluna görə bərabərlik kimi başa düşülür (yəni hesab əməllərinin nəticəsi adi tam ədədlərdə olduğu kimi cüt ədəd olmalıdır).

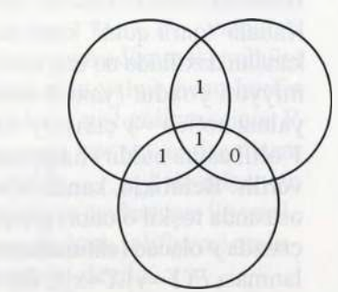
Tutaq ki, $x = (x1, \dots, x7)$ – kod sözü, $y = (y_1, \dots, y_7)$ isə kanaldan qəbul edilən və x-dən birdən çox olmayan mövqedə fərqlənən sözdür. Kod açma, yəni x-in y-ə görə bərpası belə baş verir. Əvvəlcə x_i-in yerinə y_i qoyulmuş dörd tənliyin sağ tərəflərinin qiymətləri kimi y sözünün $s(y) = (s_1, \dots, s_4)$ sindromu hesablanır. Sonra elə i mövqeyi axtarılır ki, orada tənliklər əmsalının müvafiq sütunu sindromla üst-üstə düşür və y sözündə i-ci mövqe düzəldilir (yəni çevrilir). Sindrom sıfra bərabərdirsə, deməli, tək-tək səhvlər yox imiş: $x = y$.

Bu konstruksiya sözün uzunluğu $n = 2^r - 1$, kodlaşdırılan bitlərin sayı isə $k = n - r$ olan hal üçün asanlıqla tətbiq olunur. Alınan Hemminq kodu tək-tək səhvləri düzəltməyə imkan verir və optimaldır. $2^r - 1$-dən fərqli olan kod uzunluqları üçün tək-tək səhvləri düzəldən optimal ikilik kodun qurulması axıra kimi həll edilməmiş mürəkkəb riyazi məsələdir. Məlumdur ki, kodda sözlərin maksimal sayı: $n = 7$ üçün 16 (1950-ci il); $n = 8$ üçün 20 və $n = 9$ üçün 40 (1980-ci il); $n = 10$ üçün 72 (1999-cu il), $n = 11$ üçün cavab məlum deyil (lakin o, n = 12-dən 15-ə qədər mövcuddur).

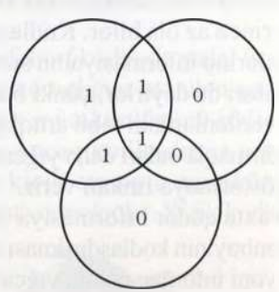
Təkrarlı səhvləri düzəldən kodların yaradılması daha mürəkkəb və maraqlı məsələdir. Hazırda belə kodların praktikada geniş şəkildə və uğurla tətbiq edilən bir neçə sinfi mövcuddur: BCH (Bose, Ray-Chaudhuri, Hocqueenghem), Rid-Solomon, Rid-Maller və s. Blok strukturunun təkrarlı səhvlərini düzəldən (məsələn, 128 baytdan ibarət sözlərdə istənilən beş səhv bayt düzəldilir) Rid-Solomon kodu daha geniş yayılmışdır. Bu kodlar optimaldır, çünki səhv blokların düzəldilməsi üçün onlara $r = 2t$ yoxlama blokları şəklində artıqlıq tələb olunur. Rid-Solomon kodlarından CD səsləndiricilərdə istifadə olunur və musiqinin yüksək keyfiyyətlə səsləndirilməsi təmin olunur. Səhv düzəldən kodlara daha bir misal: istər kosmosdan, istərsə də adi telefon kanalı vasitəsilə (məsələn, İnternetə seansla daxil olan zaman) verilənlərin ötürülməsi, mobil rabitə, verilənlərin kompüterdə saxlanması və s.



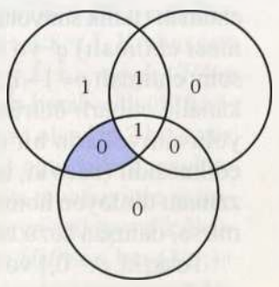
Riçard Uesli Hemminq.



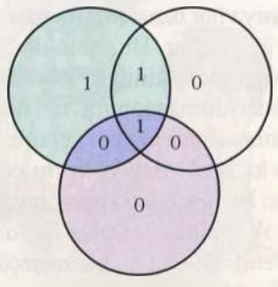
İnformasiya bitləri üç çevrənin kəsişməsi nəticəsində alınan dörd sahədə saxlanılır.



Üç cütlük biti qaydaya görə yazılır: hər bir çevrədə vahidlərin sayı cüt olmalıdır.

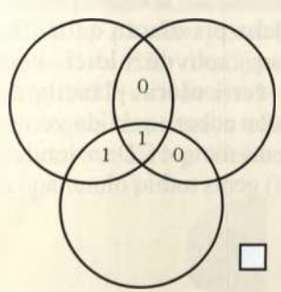


Aradan qaldırılabilən səhv yaranıb (rənglə seçdirilib).

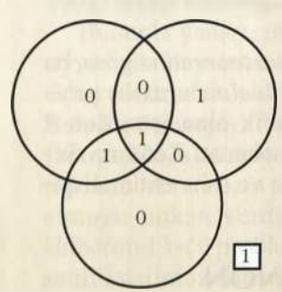


Səhv asanlıqla aşkarlanır.

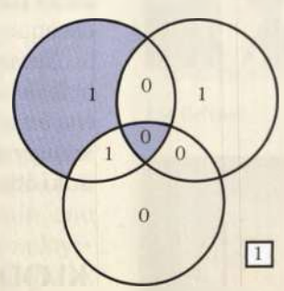
Tək-tək səhvlərin aradan qaldırılması (Venn diaqramları).



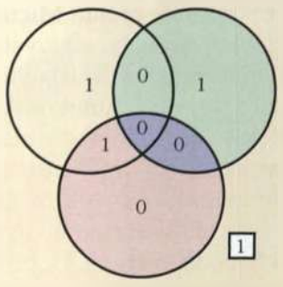
Bunun üçün daha bir cütlük yoxlaması biti istifadə olunur



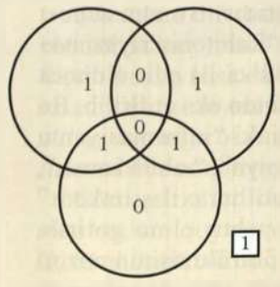
Vahidlərin tam sayı cüt olmalıdır.



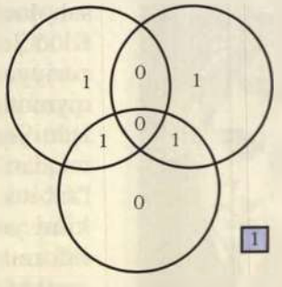
Tutaq ki, ikiqat səhv meydana çıxdı.



Səhv düzgün interpretasiya edilmədi.



Səhv düzgün düzəldilmədi.



Yardımçı bitli cütlük yoxlaması səhvi aşkarlayır.

İkiqat səhvin aşkarlanması.

"İnsan biliklərinin diferensasiyasının artdığı bizim əsrdə Klod Şennon mücərrəd riyazi fikirlərin dərinliyini texnikanın böyük problemlərinin geniş, eyni zamanda tamamilə konkret anlamı ilə birləşdirən son dərəcə gözəl örnəkdir. Onu eyni dərəcədə son onilliklərin həm birinci riyaziyyatçılarından, həm də birinci mühəndislərindən biri hesab etmək olar".

A.N.Kolmoqorov.

ötürülüb. Bu zaman kod açmanın səhv ehtimalı m artıqca sıfra yaxınlaşır (doğrudur, olduqca yavaş).

Bu metodun əsas çatışmazlığı ondan ibarətdir ki, onun ötürmə sürəti adlanan FİƏ-si də (FİƏ = $1/(2m+1)$) sıfra yaxınlaşır. İlk baxışdan elə təəssürat yaranır ki, guya bu çatışmazlıq "anadangəlmədir" və kod açmanın səhv ehtimalının sıfra yaxınlaşması tələbi informasiyanın müvafiq üsullarla ötürülməsi sürətinin sıfra yaxınlaşmasına aparıb çıxardacaq.

Şennonun ikinci teoreminə görə, bu belə deyildir və *kanalın buraxılış qabiliyyəti* C-dən kiçik olan istənilən R kəmiyyəti üçün məlumatı R ötürmə sürətilə ötürmək olar və səhv ehtimalı qabaqcadan verilmiş istənilən kiçik kəmiyyətdən kiçikdir. İkilik simmetrik kanal üçün buraxılış qabiliyyəti $C = p\log_2 p + q\log_2 q$ və o, 1/2-dən fərqli olan istənilən səhv ehtimalı üçün sıfırdan böyükdür. Xüsusən, İSK-nın $p = 0{,}1$ səhv ehtimalı ilə buraxılış qabiliyyəti təqribən 1/2-ə bərabərdir. Bu o deməkdir ki, k bit informasiyadan ibarət bloku İSK ilə ötürülən, təqribən $2k$ bit kod blokuna çevirən *səhvdüzəldici kod* mövcuddur. Ona görə də, hətta kanalda səhv olduqda da bir kod blokunun digərinə "qarışması" ehtimalı çox azdır.

Şennon öz teoremini təbii ola biləcək kodları göstərmədən, təsadüfi seçilən kodun demək olar ki, mütləq tələb olunan xassələrə malik olmasını göstərməklə isbat etmişdir.

Aparılan fəal tədqiqatlara baxmayaraq, hələ də aşkar qurulan kodların parametrləri təsadüfi kodların parametrlərindən daha pis olaraq qalır. Buna baxmayaraq, səhvdüzəldici kodlar texnikada (verilənlərin planetlərarası stansiyalardan etibarlı şəkildə verilməsindən tutmuş məişət CD səsləndiricilərinə kimi) geniş tətbiq olunmaqdadır.

KLOD ŞENNON

İnformasiya nəzəriyyəsi yaradılması vaxtı dəqiq məlum olan nadir elm sahələrindən biridir. Bu elm sahəsi Klod Şennonun "Rabitənin riyazi nəzəriyyəsi" (1948-ci il) adlı olduqca qiymətli elmi işində əks etdirilib. Bu işdə Şennon nəinki "informasiyanın miqdarı", "entropiya", "rabitə kanalı", "rabitə kanalının buraxılış imkanı" kimi yeni anlayışları elmə gətirib, informasiyanın ötürülməsinin nəzəri hədlərini göstərib, həm də informasiya nəzəriyyəsinin sonrakı bir neçə onillikdəki inkişaf istiqamətlərini qabaqcadan müəyyənləşdirib.



Klod Elvud Şennon.

Klod Elvud Şennon 1916-cı il aprelin 30-da Qeylord şəhərciyində (ABŞ) anadan olub. O, 1936-cı ildə Miçiqan Universitetini bitirərək, "riyaziyyatçı" və "elektrik mühəndisi" ixtisasları üzrə bakalavr dərəcəsi alır. Bundan iki il sonra isə "Rele və dəyişdirici açar sxemlərinin simvolik analizi" mövzusu üzrə dissertasiya müdafiə edərək magistr dərəcəsinə layiq görülür. Həmin dissertasiyasında Şennon göstərdi ki, bu cür sxemlərin (müasir kompüter texnikasının örnəyi) analizi simvolik (Bul) məntiqin riyazi aparatının köməyilə aparıla bilər. Onunla eyni vaxtda analoji nəticələri V.İ.Şestakov və A.Nakasima da əldə etmişlər. Şennon gözlənilmədən genetikaya müraciət edir və 1940-cı ildə riyaziyyat üzrə "Nəzəri genetika cəbri" mövzusunda dissertasiya (fəlsəfə doktoru dərəcəsinə) müdafiə edir. Təəssüflər olsun ki, həmin elmi iş vaxtında dərc edilməmiş və yalnız XX əsrin sonunda işıq üzü görmüşdür.

Şennon İkinci Dünya müharibəsinin əvvəlində AT&T Bell Telephones şirkətinin riyaziyyat laboratoriyasında işə başlayır, eyni zamanda ABŞ-ın Müdafiə Nazirliyinin Milli-Tədqiqat Komitəsində müşavir kimi fəaliyyət göstərir. "Bell Labs"da o, 1972-ci ilə qədər çalışır. Bu illər ərzində Şennon informasiya nəzəriyyəsini yaradır, idarəetmə sistemləri nəzəriyyəsinin əsasını qoyur və kriptoqrafiyaya dair cəmi bir məqalə yazmaqla, həmin fənni sənətdən elmə çevirir.

1956-cı ildə Klod Şennon ABŞ-ın Milli Elmlər Akademiyasına və Amerika İncəsənət və Elm Akademiyasına üzv seçilir, 1957-ci ildə isə Massaçusets Texnologiya İnstitutunun elektrotexnika və riyaziyyat üzrə professoru olur.



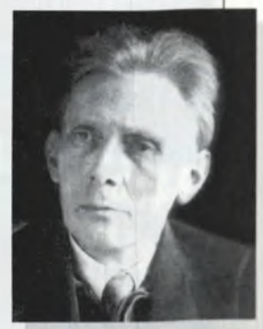
Şennonda "kabinet alimi" stereotipi yox idi. Onu həmişə "şahmat maşını", yaxud "şahmat proqramı" yaratmaq imkanı maraqlandırırdı. O, 1960-cı ildə Moskvaya gələrək, dünya çempionu Mixail Botvinnik ilə şahmat oynamağa nail olur. Onda sirk artistinə xas istedad (jonqlyorluq etmək, birtəkərli velosiped sürmək) və konstruktorluq bacarığı var idi (özü bir neçə avtomat düzəltmişdi, onların arasında jonqlyorluq edən maşın da var idi).

Müxtəlif ölkələrin (əsasən də, keçmiş SSRİ-nin və ABŞ-ın) alimləri iyirmi ilə yaxın müddət ərzində Şennon tərəfindən əldə edilmiş nailiyyətləri ümumiləşdirərək, onlara ciddi riyazi əsaslandırmalar verdilər. Bunların arasında A.Y.Xinçinin, İ.M.Gelfandın, A.N.Kolmoqorovun, A.M.Yaqlomun, R.L.Dobruşinin və M.S.Pinskerin işlərini xüsusi qeyd etmək lazımdır.

Bununla yanaşı, informasiya nəzəriyyəsinin ideyalarının riyaziyyatın özü üçün də faydalı olduğu aşkar edildi. Belə ki, "entropiya" anlayışından istifadə A.N.Kolmoqorova dinamik sistemlərin izomorfizm problemlərini həll etməyə imkan verdi, həmçinin onu Hilbertin 13-cü probleminin (çoxdəyişənli kəsilməz funksiyaların daha az dəyişənli kəsilməz funksiyaların superpozisiyası şəklində göstərilə bilməsi) həllinə sövq etdi.

Şennonun informasiya nəzəriyyəsi ehtimal nəzəriyyəsinə əsaslanır. 1960-cı illərin ikinci yarısında A.N.Kolmoqorov təsadüfilik kimi əsaslı anlayışa başqa cür yanaşmağa əsas verən alqoritmik informasiya nəzəriyyəsini yaratdı.

Görkəmli alim Klod Şennona XXI əsrə qədəm qoymaq nəsib olmuşdur. O, 85 il ömür sürmüş və 2001-ci ilin fevral ayının 24-də ABŞ-ın Massaçusets ştatının Medford şəhərində vəfat etmişdir.



A.Y.Xinçin.



İ.M.Gelfand.



A.N.Kolmoqorov.

◄ Elizabet Şennon ərinin Geylorddakı abidəsi yanında.



İNFORMASİYANIN KODLAŞDIRILMASI

RƏSMDƏN HƏRFƏ

Mənası bizə məlum olan ilk şəkillər – onlarla əsr bundan öncə mağara divarlarında qədim əcdadlarımız tərəfindən çəkilmiş heyvan təsvirləridir. Lakin rəsmlərlə yazının meydana gəlməsi arasında olduqca böyük zaman məsafəsi vardır.

İnformasiyanın ötürülməsi üçün ilkin olaraq əşya işarələrindən: balıqqulaqları, daşlar, sümüklər, quş lələkləri, düyünlər və s. istifadə edilirdi. Bu cür işarə, məsələn, analoji əşyaların tam sinfinin mövcudluğunu göstərə bilər (mağazalarda vitrinlərin doldurulması məhz bu prinsiplə həyata keçirilir, yəni hər çeşiddən yalnız bir neçəsi vitrinə düzülür). Digər halda əşya işarəsi anlayışın, əşyanın və ya bütöv bir vəziyyətin simvolu kimi çıxış edirdi. Məsələn, Afrika qəbilələrinin birində məhkumun edam edilməsinə dair xəbərdarlıq işarəsi kimi ona tutuquşu yumurtası göndərilirdi.

Bu cür işarələrin tipik funksiyaları mənsubiyyətin, statusun, mülkiyyətin göstərilməsilə, sehrbazlıq və hesabla bağlı idi. Çəpər – mülkiyyətin ilk işarələrindən biri, tac və ya hökmdar əsası – hakimiyyət və cəmiyyətdəki mövqe işarələri, büllur kürə – gələcəyi görmə işarəsi idi, xırda daşlar isə çoxdan hesab çötkəsinin aşığına çevrilmişdir.

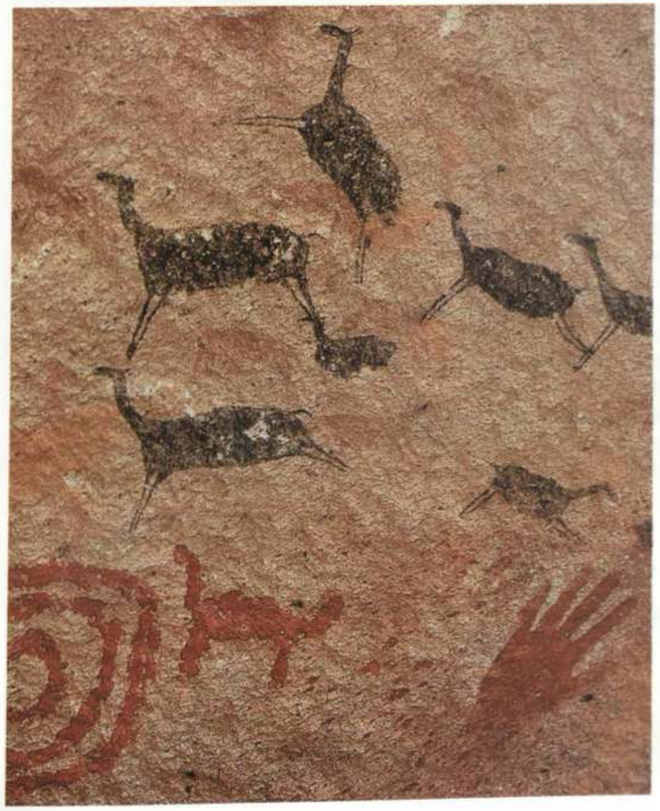
Ola bilsin ki, əşyadan görüntüyə keçən işarələr məhz sayma ilə bağlı olmuşdur. Bunlar borcu ifadə etmək üçün ağac üzərində qoyulan kərtiklər idi. Hər hansı bir səthin üzərində həkk edilən işarələr artıq təkcə hesablamalar üçün yox, həm də geniş məlumatların ötürülməsi üçün istifadə edilirsə, ideoqrafik, yəni rəsmlə işarəli yazı arasında olan şəkilli-təsvirli yazı meydana çıxır. Şəkil, qayaüstü rəsmlərdə olduğu kimi bütövlükdə hansısa vəziyyəti əks etdirə bilər, ancaq şəkildə vəziyyətin ayrı-ayrı elementlərini təsvir etməkdən çox, onları işarə edən elementlər seçilir.

Ayrı-ayrı işarələrin müəyyən anlayışlara bağlanması, onlardan müxtəlif vəziyyətləri təsvir etmək üçün istifadə imkanı şifahi və ya heroqlif yazılara keçidi göstərir. Onlarda çəkilmiş işarələr sözləri ifadə edir. Bununla belə, şəkillər daha çox sxematikləşdirilirdi, ilkin təsvirlə oxşarlığını itirirdi və artıq əşyanın özü ilə deyil, sözlə assosiasiya olunurdu. Artıq bilavasitə gerçəkliyi yox, insan nitqini kodlaşdıran yazıya keçid baş verdi.

Heroqlif yazının imkanları ideoqrafiyanın imkanlarından müqayisəyəgəlməz dərəcədə genişdir. Axı insanların düşündüklərini və bir-birlərinə deməyi vacib saydıqlarının hamısını heç də həmişə olduğu kimi təsvir etmək mümkün olmur. Daha təsvirə bağlı olmayan, bilavasitə dildəki sözü ifadə edən işarələr isə hətta ən ümumi, mücərrəd mənanı çatdırmağa imkan verir. Məsələn, təkcə hansısa "qocanı" deyil, bütövlükdə "qocalığı", müəyyən "soyuq günü" deyil, ümumiyyətlə ilin bütöv bir fəsli olan "qışı". Mənaları birləşdirmək də olar. Məsələn, rus dilində "su" və "düşmək" mənalarını verən "вода" və "падать" sözlərini göstərən işarələr birlikdə "şəlalə" ("водопад") mənasını verə bilər.

Heroqlif yazısı zəngin, inkişaf etmiş işarələr sistemidir ki, ondan bir çox nəhəng sivilizasiyalar istifadə etmiş və indi də istifadə edirlər (məsələn, Çində).

Yəqin ki, dünyada yalnız bir dil olsaydı, heroqlif yazısı daimi olaraq mövcud olardı. Məhz başqa dillərdəki sözlərlə rastlaşılma vəziyyəti heroqlif yazısının imkanlarının məhdudluğunu üzə çıxardı. Bir işarə kimi heroqlifdə ikiyə bölünmə yoxdur, o, sözlə bütövlükdə bağlanır, onu səslərlə vermir. Bəs belə şərtlərlə yeni sözləri, məsələn, əcnəbilərin adlarını necə ifadə etmək olar? Əgər o əcnəbi həmin ərazini ələ keçirərək, orada hökmranlıq etmişsə, yaxud o əcnəbi qadın hökmdarın arvadı olmuşsa, bu cür hadisələr mütləq öz əksini salnamələrdə tapmalıdır. Belə olduqca asan olmayan məsələni həll etmək üçün qədim mirzələr əcnəbi adlarını, səslənməsi ana dillərindəki bu və ya digər sözlərə oxşar hissələrə bölürdülər və müvafiq heroqliflərlə yazırdılar (məsələn, tutaq ki, müasir Azərbaycan yazısı heroqlif yazıdır və biz ABŞ-ın keçmiş prezidenti Bill Klintonun soyadını paya və not təsviri vasitəsilə çatdırmaq istəyirik: "klin" üstəgəl "ton"). Bu halda oxucuların heroqlifdən istifadənin, necə deyərlər, ciddi olmadığını, yalnız rebusun bir hissəsi olduğunu başa düşmələri üçün aşağıdakıları bildirən xüsusi işarələr qoyulurdu: "Mənasını deyil, yalnız səslənməsini qəbul et".



XƏZİNƏNİ AXTARAQ

Neçə addım atmaq lazımdır?
Şimala doğru on və on, şərqə doğru beş və beş, cənuba doğru iki və iki, qərbə doğru bir və bir və sonra aşağıya...

A.Konan Doyl. "Mesqreyvlər evinin ənənələri"

Yəqin, hamı xatırlayacaq ki, qədim sənəddəki bu göstəriş üzrə xəzinə necə tapılıb. Bu mətn məkandakı nöqtənin tapılması üçün koordinat sisteminin verilməsindən ibarətdir. Yol coğrafi koordinatlardan – şimal, cənub, qərb, şərqdən və uzunluq ölçüsündən – addımdan istifadə etməklə müəyyənləşdirilir. İstiqamətin günəşə görə müəyyənləşdirilməsi hər hansı məkandakı vəziyyətin müəyyən edilməsinin ən geniş yayılmış və ən əlverişli üsuludur. Qədim insanlar quru və su yolları ilə səyahətə çıxarkən günəşin, ayın və ulduzların köməyilə olduqları yeri təyin etməyi əla bacarırdılar. Kompasın meydana gəlməsinə qədər bütün naviqasiya bu cür üsullarla qurulurdu.

Yollar, tanış olmayan sahillər və ölkələr, hətta xəzinələrin gizlədildiyi yerlər haqqında informasiyaların saxlanması və ötürülməsinə xəritələr kömək edir. Kiməsə yolu izah etməliyiksə, bunu sözlə ifadə etməkdənsə, plan çəkmək daha asandır.

İlk xəritələr, görünür, qədim İkiçayarasında (Babilistanda) meydana gəlib: miladdan öncə XXIII əsrə aid saxsı lövhəciklər üzərində çəkilmiş babil xəritələri məlumdur. Orta əsrlərdə dənizçilər dəniz ticarət yollarının "portolan" adlanan və şüa formasında olan, bir mərkəzdən yayılan və bir nöqtədən digərinə olan qısa yolların proyeksiyasını göstərən xəritələrindən istifadə edirdilər. Kartoqrafiyanın inkişafındakı sonrakı nailiyyət Cerard Merkatorun (Flandridən olan) adı ilə bağlıdır. O, sferik və müstəvi səthdəki görüntülər arasında sistemli əlaqə quran silindrik proyeksiya metodunu yaratdı. Proyeksiyadan istifadə coğrafi koordinatları – en və uzunluq dairələrini sferik səthdən cüzi təhriflərlə müstəvi səthə keçirməyə imkan verdi.

Müasir xəritə – hər hansı məkanda yolları kodlaşdıran universal göstəricidir. Fiziki obyektlərin və onların konturlarının real təsvirinin həndəsi bənzəri xəritədə miqyaslaşdırma vasitəsilə yaradılır. Xəritədə həmçinin təsviri xüsusiyyət daşıyan (məsələn, göy rəngdən sututarların, yaşıl rəngdən meşələrin və düzənliklərin göstərilməsi üçün), yaxud sırf rəmzi xarakter daşıyan (məsələn, siyasi-inzibati xəritələrdə) rəngdən istifadə edilir. Bundan başqa, xəritədə yaşayış məntəqələrini, yolları, faydalı qazıntıların mövcud olduğu yerləri və ya heyvanların məskunlaşdığı əraziləri göstərən xüsusi rəmzi işarələrdən və nişanlardan istifadə edilir.



Müasir yazıya doğru sonuncu addım da sivilizasiya ilə mədəniyyətlərin qarşılıqlı təsirilə bağlıdır. Finikiyalılar, tacirlər və dəniz səyyahları misirlilərdən götürdükləri heroqliflərin təsvirlərini onların ilkin mənasına fikir vermədən, ancaq öz dillərində hecaları verməkdən ötrü öz ehtiyacları üçün sadələşdirirdilər. Hecalı fonetik yazı belə yarandı. Bu sistemdə sait və samitlərdən ibarət birləşmələri ayrıca işarələr verirdi, işarələr isə heroqlif yazıda olduğu kimi minliklərlə deyil, artıq onluqlarla hesablanırdı.

Nəhayət, yunanlar finikiyalıların yazısını götürərək, sonuncu, həlledici addımı atdılar: saitləri samitlərdən ayırdılar və hər bir hərfi ayrıca bir işarə ilə verməyə başladılar. Yazının ümumi tarixinin sonuncu mərhələsi kimi bizim hamımıza məlum olan səsli-hərfli və ya əlifbalı yazını da məhz yunanlar yaratdılar. Bütün tarixi yazılar bu yolla getdi.

Yazının işarə sisteminin inkişafı – təsvir prinsipinin, zahiri oxşarlığın aradan qaldırılması və onun funksional oxşarlıqla əvəz olunması yoludur. Bu yolda



Samuel Finli Briz Morze.

iki ən mühüm, inqilabi məqamı seçdirmək olar. Birincisi, heroqlifin, sözü ifadə etmək üçün xüsusi yazı işarəsinin yaradılması, ikincisi isə, məna vahidlərinin – sözlərin məna kəsb etməyən elementlərə – ayrı-ayrı işarələrlə göstərilən hecalara, yaxud səslərə ayrılmasıdır. Yazının təkamülü kodların sayının minlərlə heroqlifdən iyirmi-otuz hərfə endirilməsinə imkan verir. Kodun ixtisarının son həddi – iki işarədir. Bu məsələnin həllinə hamıdan çox Samuel Morze və Lui Brayl yaxınlaşa bilmişdir. Morze latın əlifbasının hərflərini iki elementar qrafik işarənin – nöqtə və tirenin (və daha birinin – pauzanın) kombinasiyası ilə, Brayl isə qabarıq nöqtələrlə və ya onların olmaması ilə şifrələdi. Lakin kodun ixtisarının əks tərəfi – mətnin həcminin böyüməsidir. Heroqlif yazısında cəmi üç işarə ilə yazılan cümlə səs-hərf yazısında, demək olar ki, bütöv bir sətri, Morze əlifbası ilə yazıldıqda isə bir neçə sətri tutur.

QARĞALARI SAYIRIQ

Saymanın əsasında müqayisə ideyası dayanır: sayılmalı əşyalar ardıcıl olaraq sayma vasitələrilə – çubuqlarla, daşlarla, əl barmaqları ilə və s. tutuşdurulur. Ən geniş yayılmış onluq say sisteminin əsasında məhz barmaq hesabı dayanır. Axı əl barmaqları saymaq üçün ən münasib vasitədir. Barmaqlara başqa sistemlər də müraciət edir, məsələn, qədim qallar və gürcülərin iyirmilik sistemi (görünür, onlar həm də ayaq barmaqları ilə sayırmışlar), beşlik sistem (bir əlin barmaqları ilə sayırdılar). Yeni Qvineyada telefol qəbiləsinin papuaslarında olduqca qeyri-adi barmaq sistemi var idi: onlar sağ əlin çeçələ barmağından başlayır, sonra dirsəkdən, çiyindən keçərək, burnun ucuna gəlib çıxırdılar və simmetrik olaraq sol ələ keçirdilər – nəticədə 27 say vahidi alınırdı.



Qortinadan (Krit adası) qədim yunan yazısının fraqmenti. E.ə. V əsrin ortaları.

Tutaq ki, biz qarğaları saydıq. Bəs bu informasiyanı necə ötürək? Əlbəttə, sayılan şeylərin, sadəcə, şəklini (məsələn, 6 qarğanın şəklini) çəkmək olar. Ancaq daha yığcam hərəkət etmək – bir qarğanın şəklini çəkib yanına 6 barmaq (çubuq) işarəsi qoymaq daha yaxşı olar. Adətən, qədimdə məhz belə edirdilər. Yəqin ki, elə əzəldən cizgilər barmaqları əks etdirib (rus sözləri olan "пальцы" və "пальма" eyni kökdəndir) və ola bilsin ki, işarəni bu cür əks etdirmək daha münasib olub. Bəzi xalqlarda (məsələn, mayyalarda) vahidlər dairəciklərlə və ya nöqtələrlə işarə edilirdi – bu da olduqca sadə işarədir, ola bilər ki, daş təsviridir.





HƏRFLƏRİN YERİNƏ NÖQTƏLƏR

Kod nə qədər qısa olarsa, yazılış bir o qədər uzun olur və əksinə. Kodun qısalığı ilə yazılışın qısalığı arasında optimal nisbət tapmaq məsələsini görməyən insanlar üçün əlifbanın yaradılması zamanı həll etmək lazım gəldi. Adət etdiyimiz hərfləri toxunmaqla oxumaq rahat deyil, bu hərflərin elementlərindən ibarət olan və görməklə asanlıqla qavranılan işarələr, məlum olur ki, lamisə (toxunma hissi) üçün yetərli deyil. Yaxud hərfləri elə böyük etmək lazım gəlir ki, bu halda da hər hansı böyük ölçülü mətnin yazılması və saxlanması əsl problemə çevrilir.

Körpə yaşlarında görmə qabiliyyətini itirən və müstəqil şəkildə oxuya bilmədiyinə görə çox əziyyət çəkən 15 yaşlı Lui Brayl 1824-cü ildə hərfləri işarə etmək üçün müəyyən hissələri qabarıq ola bilən altı mövqeli ($2 \cdot 3$) kombinasiyalardan istifadə etməyi düşünüb tapır. Asanlıqla hesablana bilər ki, belə kombinasiyaların sayı 64-dür və bu həm əlifbanın bütün hərflərini, həm də durğu işarələrini və rəqəmləri vermək üçün tamamilə yetərlidir. Qabarıq nöqtələr qalın kağızın üzərinə vurulur və əl ilə asanlıqla tanınır. Bundan başqa, kor adamın özü də itiuclu qələmlə nöqtələri möhkəm basmaqla bu əlifba vasitəsilə yaza bilər.

Brayl sistemində ədədin xüsusi kod-əlaməti və onu ləğv etmək üçün hərfin kod-əlaməti var idi, çünki rahat olsun deyə, rəqəmlər ilk on hərfin işarələrilə kodlaşdırılırdı. Bu, tarixdə ilk keçid kodları idi.

Sonradan rahatlıq üçün və kombinasiyalı yazıların uzunluğunu qısaltmaq üçün yerdə qalan boş kombinasiyalara daha çox istifadə olunan hərf birləşmələrini mənimsətdilər.

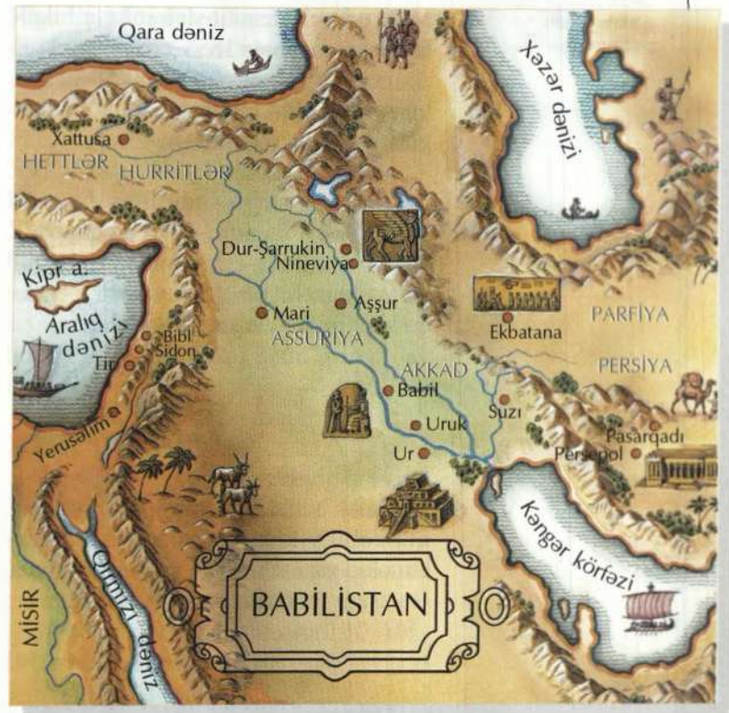
Bundan başqa, əgər işarə qonşularından böyük intervalla ayrılırsa, o, hərfi yox, tez-tez rast gəlinən sözləri əks etdirir: *y* əvəzinə *you*, *n* əvəzinə *not* və s.

Vahid üçün qəbul edilmiş işarə varsa, əslində, təkcə onunla da kifayətlənmək olar. Amma bu cür üsul, yəqin ki, yalnız zindanda günləri saymaq üçün əlverişli olar. Qalan hallarda yazının çox yer tutması onunla işləməyi əhəmiyyətli dərəcədə çətinləşdirə bilər. Ona görə də insanlar işarələrlə bütöv çoxluqları – beşlikləri, onluqları, düjinləri (12 ədədini) və s. göstərməyə başladılar. Məsələn, rum rəqəmi olan V, tədqiqatçıların fikrincə, ovuc təsviri kimi yaranıb. Böyük ədədləri belə üsulla ifadə etmək olduqca əlverişlidir, çünki ədədin ümumi qiyməti bu kəmiyyətləri toplamaqla alınır. Yığcamlıq üçün işarələrin düzülüş qaydası müəyyənləşdirilir və toplama ilə yanaşı çıxma üsullarından da istifadə etməyə başlayırlar: tutaq ki, işarədən sağdakı onluqları göstərən vahidlər onun üzərinə gəlinir, soldakılar isə çıxılır. Rum sistemi də bu cür qurulub: onun əsas işarələri latın əlifbasının hərfləri şəklində olan ideoqramlardır – I, V, X, L, C, M (1, 5, 10, 50, 100, 1000) və s. Yunanlar da, ərəblər də, ruslar da (əlifbanın böyük bir hissəsini yunanlardan götürərək), bir sıra başqa xalqlar da ədədləri hərflər vasitəsilə göstərirdilər. Bu üsul çox əlverişli idi və geniş şəkildə yayıldı. Aydındır ki, hansısa məqamda hərflər yetərli olmaya bilər, ancaq həmin keçmiş zamanlarda çox böyük ədədlərdən nadir hallarda istifadə edirdilər.

Yazının yaradılmasında finikiyalıların və yunanların atdığı mühüm addımı ədədlərin yazılması üçün hindlilər və ərəblər atdılar. Onlar sıfrı və ədədlərin mövqeli yazılış sistemini düşünüb tapdılar. Bu sistemin ideyası zahirən çox sadədir: əgər iki nəfər razılaşsa ki, birinci vahidləri sayacaq, ikinci onluqları, onda onlar əl barmaqlarından istifadə etməklə 20-yə deyil, 100-ə qədər saya bilərlər (üç nəfərsə 30-a deyil, 1000-ə qədər saya bilər və s.).

SAY SİSTEMLƏRİ

BABİLİSTANDA NECƏ SAYIRDILAR



Dəclə və Fərat çayları arasında yerləşən qüdrətli Babilistan dövlətində hesab və say sistemləri yüksək səviyyədə inkişaf etmişdi (bax: "Uşaqlar üçün ensiklopediya"nın "Riyaziyyat" cildində "Ədədlərin qədim yazı sistemləri" bölməsi). Babil ədədləri iki deyil, üç mixi (xətli) işarəsinin kombinasiyasından ibarətdir: vahidlər – 𒁹, onluqlar – 𒌋 və yüzlüklər – 𒐖.

Toplama, yaxud vurma prinsiplərindən istifadə etməklə bu işarələr vasitəsilə istənilən ədədi yazmaq olar:

$$𒌋𒁹 - 10 \cdot 100 = 1000,$$

$$𒌋𒁹 - 10 + 1 = 11.$$

Böyük ədədlər həmişə kiçik ədədlərdən əvvəldə gəlirdi. Ədədlərin yazılışının bu üsulundan başqa, mövqeli altmışlıq sistem də tətbiq olunurdu. Düzülüş qaydasından asılı olaraq, vahidləri göstərən işarə 1, 60, 60^2 kimi, onluqları göstərən işarə isə uyğun olaraq 10, $10 \cdot 60^2$ kimi və i.a. işarə oluna bilər:



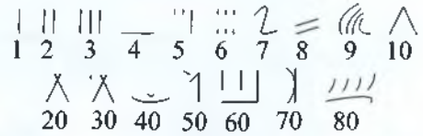
$$𒌋𒐗𒐗𒐗 - 1 \cdot 60 + 10 + 2 = 72.$$

Babillər "sıfır" üçün də oxşar işarəyə – 𒑚 (iki çəpəki vahid işarəsi) malik idilər, amma təəssüf ki, həmin işarə ədədin sonunda qoyulmurdu. Onlar sadə və altmışlıq kəsrlərdən (məxrəcləri 60, 60^2 və s. olan) istifadə etməyi də bacarırdılar və bu kəsrləri indi onluq kəsrlərin yazıldığı kimi yazırdılar. Hesablamalar zamanı babillər hazır vurma cədvəlindən istifadə edirdilər.

MİSİRDƏ NECƏ SAYIRDILAR

Misir ədədi sisteminə dair müasir bilgilərin əsas mənbəyi 1853-cü ildə tapılmış Axmes papirusudur (yaxud 1858-ci ildə bu sənədi əldə etmiş şəxsin soyadına görə – Raynd papirusu).

Misirlilər ədədləri yazmaq üçün heroqliflərdən istifadə edirdilər: *bir* – 𓏺, *on* – 𓎆, *yüz* – 𓍢, *min* – 𓆼 və bu qaydada 10 milyona kimi – 𓁨. Sonradan heroqlif yazısı iyeratik (*yun.* "heratikos" – "müqəddəs") yazıyla sadələşdirildi:



1 2 3 4 5 6 7 8 9 10

20 30 40 50 60 70 80

Qalan bütün ədədlər toplama əməlinin köməyilə bu işarələrdən istifadə etməklə yazılırdı. Əvvəlcə yüksək, sonra isə aşağı mərtəbəli ədəd yazılırdı:

𓍢𓍢𓎆𓏺𓏺𓏺𓏺=100+100+10+1+1+1+1=214.

Misirlilər vurma və bölməni ədədləri ardıcıl olaraq iki dəfə artırmaqla yerinə yetirirdilər – ikiyə xüsusi rol verilirdi.

19 · 31 misalında misirlilər 31 ədədini ardıcıl olaraq 2 dəfə artırırdılar. Sağ sütunda iki dəfə artırmanın nəticələrini, sol tərəfdə isə ikinin müvafiq dərəcəsini yazırdılar:

1	31
2	62
4	124
8	248
16	496





Sonra sol sütunun elə sətirlərini şaquli xətlərlə qeyd edirdilər ki, onları toplayaraq vuruğu tapmaq olurdu (19 = 1 + 2 + 16) və qeyd olunan sütunların sağ tərəfindəki ədədləri cəmləyirdilər (31 + 62 + 496 = 589).

Misir kəsrlərinin surəti həmişə vahidə bərabər olurdu (2/3 istisna idi) (bax: "Uşaqlar üçün ensiklopediya", "Riyaziyyat" cildi, "Hesablama üsulları" məqaləsinə). Kəsrlər natural ədədlər kimi yazılırdı, amma onların üstündə nöqtələr qoyulurdu, 1/2 və 2/3 üçün xüsusi işarələr var idi:

𓍣 – 1/10, 𓂝 – 2/3, 𓂞 – 1/2.

ROMADA NECƏ SAYIRDILAR

Rum ədədləri hamıya məlumdur, onları indi də bir çox yerlərdə, məsələn, qüllə saatlarının sferblatında görmək olar.

Rum say sistemində yeddi ədəd hərflərlə işarə olunur: I – I, 5 – V, 10 – X, 50 – L, 100 – C, 500 – D, 1000 – M, qalan ədədlər isə bu işarələrin kombinasiyaları ilə yazılırlar. Bu kombinasiyalardakı ədədlər böyükdən kiçiyə doğru düzülürsə, onlar toplanılır:

XXI – 10 + 10 + 1 = 21,

MMVI – 1000 + 1000 + 5 + 1 = 2006,

ədədlər kiçikdən böyüyə doğru düzülürsə, ədədin qiyməti növbəti hərfdən çıxılır:

IV = 5 – 1 = 4.

Belə sistemdə toplamaq və çıxmaq asan olsa da, vurmaq və bölmək çox mürəkkəbdir. Romalılar məxrəci 60 (Babilistanda olduğu kimi), 12, 24, 48 olan kəsrlərdən istifadə edirdilər. Pulun, ölçünün və çəkinin hesablanması zamanı kəsrlərdən istifadəyə ehtiyac yaranırdı. Roma sikkəsi ass misdən kəsilirdi, ağırlığı bir funt idi və 12 unsiyaya bölünürdü.

YUNANLAR NECƏ SAYIRDILAR

Yunanlar ədədləri yazmaq üçün bir neçə üsuldan istifadə edirdilər. İonik nömrələmə zamanı ədədlər əlifbanın hərflərilə ifadə edilirdi. Ədədi sözdən fərqləndirmək üçün ədədləri göstərən hərflərin üstünə xüsusi işarə ~ – *titlo* qoyulurdu. Bu yazı üsulunu Milet və İskəndəriyyə sakinləri tətbiq edirdilər. Afinalılar ədədləri göstərmək üçün onları ifadə edən sözlərin baş hərflərindən istifadə edirdilər:

Γ (Γεντε)	– beş,
Δ (Δεκα)	– on,
H (Ηκατον)	– yüz,
X (Χιλιασ)	– min,
M (Μυριασ)	– on min,
I, II, III, IIII	– müvafiq olaraq 1, 2, 3, 4,
ΔΔΔIIII	– 10 + 10 + 10 + 4 = 34.

Qədim Yunanıstanın sakinləri bu rəqəmlərin köməyilə istənilən ədədi yaza bilirdilər.

Dahi yunan riyaziyyatçısı İskəndəriyyəli Diofant kəsrləri təxminən indi qəbul olunduğu kimi yazırdı: surət məxrəcin üstündə olurdu, amma aralarında üfüqi xətt qoyulmurdu. Bu, qədim Yunanıstanda kəsrlərin yazılması üsullarından biri idi.

HİND RƏQƏMLƏRİ

Hazırda istifadə edilən rəqəmlər Hindistandan gəlib. Avropa xalqları onlarla ərəblərin sayəsində tanış olublar. İlk dəfə 1202-ci ildə riyaziyyatçı Leonardo Pizanski ərəb rəqəmlərinin adını çəkir. XVI əsrdə yeni nömrələmə geniş şəkildə yayılmağa başlayır; Rusiyaya o, XVII əsrdə gəlib çıxır və XVIII əsrin başlanğıcından əlifba sistemini tamamilə sıxışdırıb çıxarır. Onluq mövqeli sistem (ərəb sistemi) istənilən qədər böyük ədədi yazmaq üçün prinsipial imkanlar verir.

Onluq sistemdə on rəqəm var: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9.

Ədəd təkcə rəqəmdən deyil, həm də onun yerləşmə yerindən asılıdır. 44 ədədi vahidlərin və onluqların sayını ifadə edir, 0 rəqəmi isə rəqəmin mövqeyini göstərir, məsələn, 40.

10 ədədi və onun dərəcələri: 10, 10^2, 10^3 ... xüsusi rol oynayır.

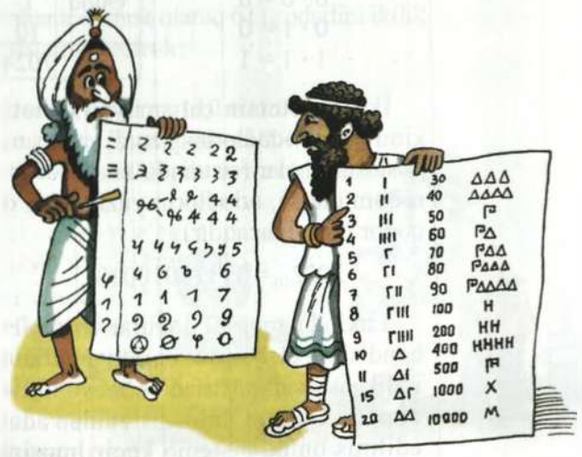
$2005 = 5 + 0 \cdot 10 + 0 \cdot 100 + 2 \cdot 1000,$

yaxud

$2005 = 5 + 0 \cdot 10^1 + 0 \cdot 10^2 + 2 \cdot 10^3,$

yaxud

$2005 = 2 \cdot 10^3 + 0 \cdot 10^2 + 0 \cdot 10^1 + 5 \cdot 10^0.$





İstənilən dördrəqəmli ədədi bu cür yazmaq olar:

$$N = a_3 \cdot 10^3 + a_2 \cdot 10^2 + a_1 \cdot 10^1 + a_0 \cdot 10^0,$$

burada a_0, a_1, a_2, a_3 – ədədin onluq rəqəmləridir (a_3 rəqəmi 0-dan fərqlidir). Düsturdakı 10 ədədi say sisteminin *əsası* adlanır. Başqa əsaslı say sistemləri də mövcuddur.

p say sistemində n ədədinin yazılışı:

$$n = a_n p^n + a_{n-1} p^{n-1} + \ldots + a_1 p^1 + a_0 p^0,$$

burada $a_i = 0, 1, 2, \ldots, p-1$ və $a_n \neq 0$.

Əgər p olaraq 2 ədədini götürsək, onda ikilik say sistemi alınar.

İKİLİK KODLAŞDIRMA

İkilik say sistemi kompüterlərin meydana gəlməsindən xeyli əvvəl yaradılıb. Hələ dahi alman riyaziyyatçısı Qotfrid Vilhelm Leybnis ikilik sistemdə xüsusi gizli mənanı görmüşdü. Həqiqətən də, ikilik sistem olduqca sadədir:

cəmi iki rəqəmdən istifadə edilir – 0 və 1;

vurma cədvəli cəmi üç sətirdən ibarətdir:

$0 \cdot 0 = 0$
$0 \cdot 1 = 0$
$1 \cdot 1 = 1$

təkcə hesablayıcı maşın yox, insan özü də yerinə yetirə bilər.

Rahat olsun deyə, ikilik ədədi dərəcələr üzrə cədvələ yazaq, növbəti sətirdə dərəcələri nömrələyək (0-dan başlayaraq, sağdan sola) və ikinin müvafiq dərəcələrini yazaq (0 dərəcəsindən başlayaraq, sağdan sola).

1111010011_2 ədədi üçün cədvəl belə görünəcək:

Ədəd	1	1	1	1	1	0	1	0	0	1	1
N	10	9	8	7	6	5	4	3	2	1	0
2^n	1024	512	256	128	64	32	16	8	4	2	1

İkilik sistemin çatışmayan cəhəti kimi yalnız ədədlərin "uzun" yazılışını göstərmək olar (sistemdə nə qədər az rəqəm olarsa, ədədlərin yazılışı bir o qədər uzun olacaqdır):

$$2002_{10} = 11111010010_2.$$

Lakin bu təəssüf doğuran fakt mühəndislərə elektron sxemləri qurarkən ikilik sistemdən istifadə etməyə mane olmadı. Ədədin ikilik sistemdən adət edilmiş onluq sistemə keçirilməsini

Cədvəlin birinci sətrindəki vahidlər ədədi almaq üçün ikinin hansı dərəcələrinin toplanmalı olduğunu göstərir:

$$1024 + 512 + 256 + 128 + 64 + 16 + 2 + 1 = 2003,$$

yəni $1111010011_2 = 2003_{10}$.

Ədədin ikilik sistemə keçirilməsi alqoritmi əvvəlkinə uyğundur, lakin qalıqlı bölmə əməliyyatının aparılmasını tələb edir. Ədəd ardıcıl olaraq ikiyə bölünür, nəticələr sütuna, qalıqlar isə (sıfırları

buraxmamaqla) qonşu sütuna yazılır. Sağ sütunda alınan ədəd aşağıdan yuxarı oxunmaqla cavabı əks etdirəcəkdir. 413 ədədi üçün bu, belə görünəcək:

413	1
206	0
103	1
51	1
25	1
12	0
6	0
3	1
1	1

Əgər indi biz sağ sütunda alınan nəticələri aşağıdan yuxarıya yazsaq, yekun cavab bu cür olacaq:

$$110011101_2 = 413_{10}.$$

Ədədin ikilik say sistemindən onluq say sisteminə və əksinə keçirilməsində mürəkkəb və sehirli bir şey yoxdur. Bu mürəkkəb olmayan elmi mənimsəyən hər bir kəs özünü EHM dilini anlayan hesab edə bilər. Buna baxmayaraq ədədi ikilik sistemdə yazmaq əvvəllər olduğu kimi əlverişli deyil, çünki bu cür yazı çox yer tutur. İkilik sistemə keçirməni də çətin ki, zehində yerinə yetirmək mümkün olsun. Buna görə də ikilik say sisteminə yaxın olan sistemlərdən istifadə edildi, çünki həmin sistemlərdə ədədlər kağız üzərində ikilik sistemdə olduğundan daha qısa yazılır, çevirmə alqoritmləri isə mürəkkəb hesablamalar tələb etmir.

SƏKKİZLİK SAY SİSTEMİ

Bu sistemdə səkkiz rəqəm var: 0, 1, 2, 3, 4, 5, 6, 7. Kiçik mərtəbədə yazılan 1 rəqəmi, onluq sistemdə olduğu kimi, sadəcə, vahidi ifadə edir, növbəti mərtəbədə o, 8_{10}, daha sonra 64_{10} və s. deməkdir.

100_8 ədədi 64_{10}-dur, $635_8 = 6 \cdot 64 + 3 \cdot 8 + 5 = 413_{10}$. Hesablamalarda ədədlər



PDP 11/70 kompüteri səkkizlik kodlaşdırmanı istifadə edirdi.

onluq sistemdə göstərilib və onluq sistemin göstəricisi olan 10 buraxılıb.

Səkkizlik sistemdən onluq sistemə keçid mürəkkəbliyinə görə ikilik sistemdən onluq sistemə keçiddən az fərqlənir. Səkkizlik sistemdən ikilik sistemə keçid isə çox sadədir. Bunun üçün üçlük (üç rəqəmdən ibarət) cədvəl tərtib etmək və onu vurma cədvəli kimi yadda saxlamaq kifayətdir.

$0_8 - 000_2$	$4_8 - 100_2$
$1_8 - 001_2$	$5_8 - 101_2$
$2_8 - 010_2$	$6_8 - 110_2$
$3_8 - 011_2$	$7_8 - 111_2$

Səkkizlik sistemdən ikilik sistemə keçirmə zamanı hər bir səkkizlik rəqəm bu cədvəldəki müvafiq üçlüklə əvəz olunur. Örnək olaraq 611_8 ədədini ikilik sistemə keçirək:

$611_8 = 110\ 001\ 001_2$
(6_8 rəqəmi 110_2 ilə,
1_8 isə 001_2 ilə əvəz olunub).





EHM VAX-da onaltılıq kodlaşdırmadan istifadə edilmişdir.

Əks əməliyyat üçün, yəni ikilik sistemdən səkkizlik sistemə keçmək üçün ikilik ədəd sağdan sola doğru üçlüklərə bölünür, sonra hər bir qrup bir səkkizlik rəqəmlə əvəz olunur. Tutaq ki,

$$101\ 111\ 111_2 = 577_8.$$

İlk tamqiymətli üçlük alınmırsa (rəqəmlər kifayət etmirsə), ona aparıcı sıfırlar əlavə edilir:

$$1\ 111\ 000_2 = 001\ 111\ 000_2 = 170_8.$$

ONALTILIQ SAY SİSTEMİ

Ədədin səkkizlik sistemdə yazılışı yetərincə yığcamdır, ancaq o, onaltılıq sistemdə daha yığcam alınır. İlk on rəqəm üçün adət olunmuş rəqəmlərdən: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, qalan altı rəqəm üçün isə latın əlifbasının ilk altı hərfindən istifadə olunur:

A — 10,
B — 11,
C — 12,
D — 13,
E — 14,
F — 15.

Səkkizlik sistemdə olduğu kimi, kiçik mərtəbədə yazılan 1 rəqəmi vahidi göstərir. Növbəti mərtəbədə həmin 1 rəqəmi 16_{10}, daha sonrakı mərtəbədə 256_{10} və s. ifadə edir. Kiçik mərtəbədə yazılan F rəqəmi 15_{10}, növbəti mərtəbədə $15_{10} \cdot 16_{10}$ göstərir və s. Məsələn,

$$100_{16} = 256_{10},$$
$$1AF_{16} = 1 \cdot 16^2 + 10 \cdot 16 + 15 = 431_{10}.$$

Onaltılıq sistemdən ikilik sistemə və əksinə keçid səkkizlik sistemlə olan əməliyyatlar kimidir. Yenə də çevirmə cədvəlini yadda saxlamaq lazımdır, burada fərq yalnız ondadır ki, üçlüklərin yerində dördlüklər dayanır:

$0_{16} - 0000_2$	$8_{16} - 1000_2$
$1_{16} - 0001_2$	$9_{16} - 1001_2$
$2_{16} - 0010_2$	$A_{16} - 1010_2$
$3_{16} - 0011_2$	$B_{16} - 1011_2$
$4_{16} - 0100_2$	$C_{16} - 1100_2$
$5_{16} - 0101_2$	$D_{16} - 1101_2$
$6_{16} - 0110_2$	$E_{16} - 1110_2$
$7_{16} - 0111_2$	$F_{16} - 1111_2$.

Deməli,

$$61A_{16} = 110\ 0001\ 1010_2$$
(6_{16} rəqəmi 0110_2 ilə,
1_{16} rəqəmi 0001_2 ilə, A rəqəmi isə 1010_2 ilə əvəz edilmişdir).

Əks əməliyyatın yerinə yetirilməsi zamanı ikilik ədədi sağdan sola doğru dörd-dörd bölmək lazımdır (əgər birinci dördlükdə rəqəm çatmırsa, ona aparıcı sıfırlar əlavə edilir). Sonra hər qrup bir onaltılıq rəqəmlə əvəz olunur:

$$11\ 1110\ 0111_2 = 0011\ 1110\ 0111_2 = 3E7_{16}.$$

BİTLƏR VƏ BAYTLAR

İkilik say sisteminin simvolları – 0 və 1-ə ikilik rəqəmlər, yaxud bitlər (*ing.* binary digit – ikilik rəqəm sözlərindəndir) deyilir.

Elektron sxemlərdə ikilik kodların emalı üçün yalnız iki dayanıqlı vəziyyət mövcuddur: "siqnal var" / "siqnal yoxdur" (yüksək gərginlik / aşağı gərginlik).

Müvafiq sxemləri texniki cəhətdən reallaşdırmaq çətin deyil. Ona görə də informasiyanın ikilik kodlaşdırılması zamanı ədədin yazılışı çox uzun alınsa da, belə kodlaşdırma mühəndisləri cəlb edir. Kompüterə azsaylı mürəkkəb elementlərlə işləməkdənsə, çoxsaylı sadə elementlərlə işləmək daha əlverişlidir.

İnformasiyanın kompüterin yaddaşında saxlanması zamanı hər bit bir yaddaş mərtəbəsində saxlanılır. Mərtəbələr müəyyən ölçülü yaddaş oyuqlarında (8, 16 və 32 mərtəbə) birləşir və "bayt", "söz" və "ikiqat söz" kimi xüsusi adlar daşıyırlar. Faktiki olaraq, bu, müasir fərdi kompüterlər üçün standartdır.

Ədədləri kodlaşdırarkən yadda saxlamaq lazımdır ki, məsələn, 1000_{10} ədədini bir bayta yerləşdirmək olmaz, çünki onun ikilik yazılışı (1111101000_2) on rəqəmlə ifadə olunur, bu da baytın ölçüsündən artıqdır. Əlbəttə, ikiqat sözə yerləşməyəcək qədər böyük ədədlər də mövcuddur.

Belə ədədləri böyük sayda baytlara yerləşdirmək olar, ancaq kompüter baytlarda və sözlərdə (bəzən ikiqat və dördqat sözlərdə) yerləşən ədədlərlə sürətli hesablamalar aparmağa qadirdir. Əgər ədəd böyük olarsa, onunla aparılan əməliyyatlar baytlarla, sözlərlə və s. ilə aparılan əməliyyatlara parçalanacaqdır.

Analoji olaraq bir baytda hər hansı mətnin 256 müxtəlif simvolunu kodlaşdırmaq olar. İlk baxışdan bu çoxdur: Azərbaycan əlifbasının böyük (baş) və kiçik hərfləri, durğu işarələri, on rəqəm, həmçinin hər hansı başqa əlifba (məsələn, rus əlifbası). Lakin daha hər hansı bir əlifbanı daxil etmək tələb olunarsa, onda yəqin ki, yer çatmayacaq. Onda mətnin simvolu baytla deyil, sözlə kodlaşdırılacaq ki, bu da avtomatik olaraq saxlanılan informasiyanın həcmini iki dəfə artıracaqdır.

Artıq qeyd edildiyi kimi, ikilik kodlaşdırma zamanı, ikilik rəqəmlərdən başqa, kompüterdə heç nə yazmaq olmaz. Bir baytda 0-dan 255-ə qədər bir mənfi olmayan tam ədəd saxlamaq olar, ancaq mənfi ədədlərin kodlaşdırılması haqqında da düşünmək lazımdır. Məsələn, bir baytda 128 mənfi tam ədəd (-128-dən -1-ə qədər), 0 və 127 müsbət tam ədəd (1-dən 127-ə qədər) saxlamaq olar. Bu halda da, kodlaşdırılmış simvollar kimi, ədədlərin sayı yenə də 256 olacaqdır.

Sıfır 8 sıfırın kombinasiyası ilə kodlaşdırılacaqdır: 00000000.

1-dən 127-dək müsbət ədədlər, adətən, ikilik sistemdə kodlaşdırılır və bu zaman soldan *qiymətli* olmayan sıfırlar artırılır:

1 — 00000001
2 — 00000010
3 — 00000011
...
126 — 01111110
127 — 01111111



Mənfi ədədlər *tamamlayıcı kodda* yazılır. Məsələn, –5 ədədi 256–5 ($256 = 2^8$) ədədinin ikilik yazılışı kimi saxlanılır. Yəni 5-ə 256-ya qədər tamamlama:

11111011

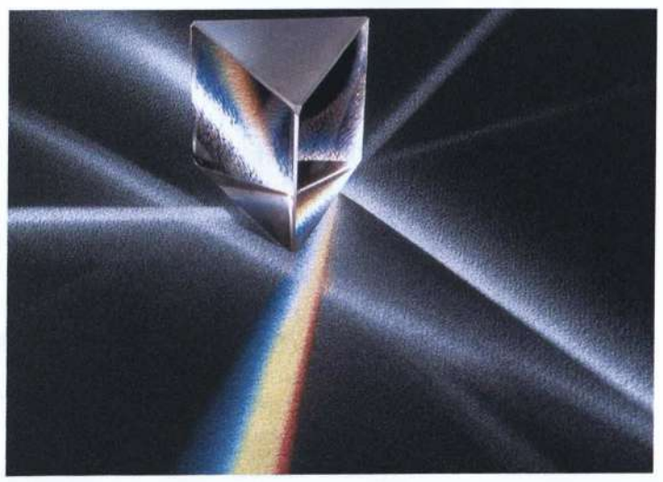
–5 + 256 > 127 olduğundan ikilik ədədin yuxarı mərtəbəsində vahid olacaqdır ki, bu da ədədin mənfi olmasını göstərir. Ona görə də oyuğun yuxarı mərtəbəsi, adətən, *işarə* mərtəbəsi adlanır. Mənfi ədədləri də bu cür kodlaşdırılır: –1, –2, –3, ... , –127, –128.

TƏSVİRLƏRİN KODLAŞDIRILMASI

İnsan informasiyanın çox hissəsini ətraf mühitdən görmə vasitəsilə əldə edir. İnsanın gözü işığı yalnız o zaman hiss edir ki, tor qişaya 360–750 nm diapazonunda şüalı enerji, yəni elektromaqnit rəqslərinin spektrlərinin görünən hissəsi düşür.

İnsan tərəfindən cisimlər rənglilikləri (rəngin tonu və rəngin dolğunluğu birlikdə) və parlaqlıqlarının müxtəlifliyinə görə dərk edilir. Cisim işıq mənbəyindən nə qədər uzaqlaşırsa parlaqlığı bir o qədər azalır. Cismin parlaqlığı həmçinin onun əksetmə qabiliyyətindən və formasından asılıdır. Ağ rəng ən mürəkkəbdir, belə ki, o, müxtəlif uzunluqlu dalğaların şüalarını birləşdirir. Hər bir cisim şüanı özünəməxsus şəkildə əks etdirir ki, bu da onlar arasında rəng fərqlərini izah edir.

Prizma və işığın spektral ayrılışı.



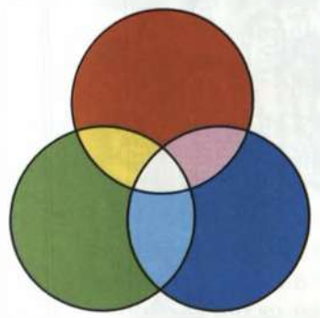
Rəngi hər bir dalğa uzunluğuna düşən enerjinin miqdarına – *spektral xarakteristikalara* görə qiymətləndirmək olar.

Lakin rəngin vizual qiymətləndirilməsi də tətbiq edilə bilər. 1756-cı ildə görkəmli rus alimi Mixail Vasilyeviç Lomonosov (1711-1765) ilk dəfə fikir yürütmüşdür ki, təbiətdə hər hansı bir rəngi yenidən yaratmaq üçün üç əsas rəngi (qırmızı, yaşıl, göy) müəyyən edilmiş nisbətlərdə qarışdırmaq kifayətdir. Rəngin *üçkomponentlik* nəzəriyyəsi çoxsaylı təcrübələrə söykənərək təsdiq edir ki, insanın görmə sistemində birbirindən asılı olmayan üç tip əsəb oyanışı baş verir. Görmə zamanı rənglərin hiss edilməsinin müxtəlifliyini üç tip oyanışın kombinasiyalarının müxtəlifliyilə izah etmək olar. Bu nəzəriyyə rəngli təsvirlərin yenidən yaradılmasının əksər praktiki üsullarının əsasını təşkil edir: fotoqrafiyada, televiziyada, kinoda və s.

Təsvirlərin kompüter kodlaşdırılması da bu nəzəriyyə əsasında qurulmuşdur. Şəkil şaquli və üfqi xətlərlə kiçik düzbucaqlılara bölünür. Alınmış düzbucaqlılar matrisi rastr, matrisin elementləri isə – piksellər (picture's element – "təsvir elementi") adlanır. Hər bir pikselin rəngi üç əsas rəngin intensivlik qiymətlərinin üçlüyü ilə verilir. Hər bir əsas rəng üçün ayrılmış bitlərin sayı nə qədər çox olarsa təsvirin hər bir elementi haqqında daha çox rənglər ahəngliyini saxlamaq mümkündür. True color (real rəng) adlanan standartda hər əsas rəngə 1 bayt olmaqla, rastrın hər bir nöqtəsinə 3 bayt sərf olunur. Beləliklə, qırmızı rəngin 256 parlaqlıq dərəcəsi, yaşıl rəngin 256 parlaqlıq dərəcəsi, göy rəngin 256 parlaqlıq dərəcəsi birlikdə təxminən 16,7 milyon müxtəlif rəng çalarlarını verir ki, bu da insan gözünün rəng qavrama qabiliyyətindən çoxdur.

Şəkli bütövlükdə saxlamaq üçün, piksellərin rənginin qiymətlər matrisini müəyyən qayda üzrə, məsələn, soldan sağa və yuxarıdan aşağıya yazmaq kifayətdir. Belə kodlaşdırmada şəkil haqqında informasiyanın bir hissəsi itir. Piksellər nə qədər kiçik olarsa, itki də bir o qədər az olar. Diaqonalı 15–17 düym olan müasir kompüter monitorlarının ekranında şəklin elementlərinin yerləşdirilməsi və keyfiyyəti arasında səmərəli kompromisi 768x1024 nöqtəli rastr təmin edir (məişət videomaqnitafonu – hər kadra 250 xətt, yəni şaquli üzrə 250 nöqtə, SECAM standartlı televizorlarda – hər kadra 625 xətt).

Təsvirin kodlaşdırılmasında məhz üç əsas rəng həmişə istifadə olunmur. Şəkillərin kağız üzərində çapı (həm də çəkilməsi) zamanı qırmızı ilə yaşıl rəngin qarışığından sarı deyil, qəhvəyi rəng alınır. Bu ona görə baş verir ki, rənglər özlüyündə işığı əks etdirmirlər, yalnız üzərlərinə düşən işıq selindən bəzi rəngləri udurlar. Odur ki, əsas kimi başqa rənglərdən istifadə olunur: mavi, yaşıl və sarı. Rəngin kodlaşdırılması metodu isə CMY (*ing.* cyan – "mavi", magenta – "yasəməni", yellow – "sarı") adlanır. Bu halda qırmızı rəng yasəməni və sarı, yaşıl isə sarı və mavi rənglərin cəmindən alınır.



Belə ki, rəngin vizual qəbulu üçkomponentli olduğundan istənilən rəng üçdən az olmayan parametrlə verilə bilər. Bir çox təsvirin emalı proqramlarında insanın qavramasına daha yaxın olan HSV (*ing.* hue – "rəng keyfiyyəti", saturation – "dolğunluq", value – "həcm", "parlaqlıq") kodlaşdırma sxemindən istifadə olunur.

Rəng keyfiyyəti – rəngin bu və ya digər rəng spektrlərinə yaxınlığıdır. Ağ rəngin spektrdə paylanmasını müşahidə edərkən insan gözü spektrin görmə duyğusuna görə kəskin fərqlənən yeddi sahəsini görür: qırmızı, narıncı, sarı, yaşıl, mavi, göy, bənövşəyi. Diqqətlə baxdıqda insan daha çox rəng sezir, lakin gözün rəng keyfiyyəti 200-ə yaxın rəngi ayırmağa qadirdir. Rəngin keyfiyyəti mövcud rəngə daha yaxın olan spektral şüalanmanın dalğa uzunluğu ilə xarakterizə olunur.

Dolğunluq – rəngdə rəng keyfiyyətinin ifadə dərəcəsidir, yəni mövcud spektral rəngə onunla eyni parlaqlığa malik olan ağ rəngin nə qədər qarışmasıdır. Əgər təmiz spektral rəngə ağ rəng

Nöqtənin rəngini üç ədəd təsvir edir – bu RGB də ola bilər, başqa üç rəng də. Televiziyada qəbul olunmuş rəngin kodlaşdırılmasında RGB-dən YC_bC_r cevrilməsi üçün düsturlar belə verilir:
$Y = 0{,}299R + 0{,}587G + 0{,}114B$.
$C_b = 0{,}564\ (B-Y)$.
$C_r = 0{,}649\ (R-Y)$.

Çox zaman HSV əvəzinə HSB (brightness) işarəsindən istifadə olunur.

əlavə edilərsə, o zaman dəyişməyən ümumi parlaqlıqda spektral rəngin dolğunluğu azalacaqdır. Maraqlıdır, təbiətdə rəngin dolğunluğuna görə fərqlənən şeylər azdır (ola bilsin ki, yalnız tropik flora və faunadır). Süni boyalar, məsələn monitorda təsvirlər daha yüksək dolğunluğa malik olduğuna görə insan gözünü sevindirir.

Parlaqlıq (aydınlıq) gözə təsir edən şüalanma dərəcəsiylə müəyyən edilir. Əgər obyektin bir hissəsi birbaşa, digər hissəsi həmin mənbədən səpələnmiş şəkildə işıqlanarsa, o zaman insan tərəfindən rəngin qavranması müxtəlif olur, baxmayaraq ki, bu hissələrdə rəngin keyfiyyəti və dolğunluğu eynidir.

Rəngin keyfiyyəti, dolğunluğu və parlaqlığı vasitəsilə kodlaşdırma rəngli şəkillərdən təsvir obyektlərinin parlaqlığını dəyişmədən monoxrom və ya başqa cür desək, təsvir obyektlərinin parlaqlığını dəyişdirmədən ağ-qara şəkilləri asanlıqla əldə etməyə imkan verir.

SƏSİN VƏ MUSİQİNİN KODLAŞDIRILMASI
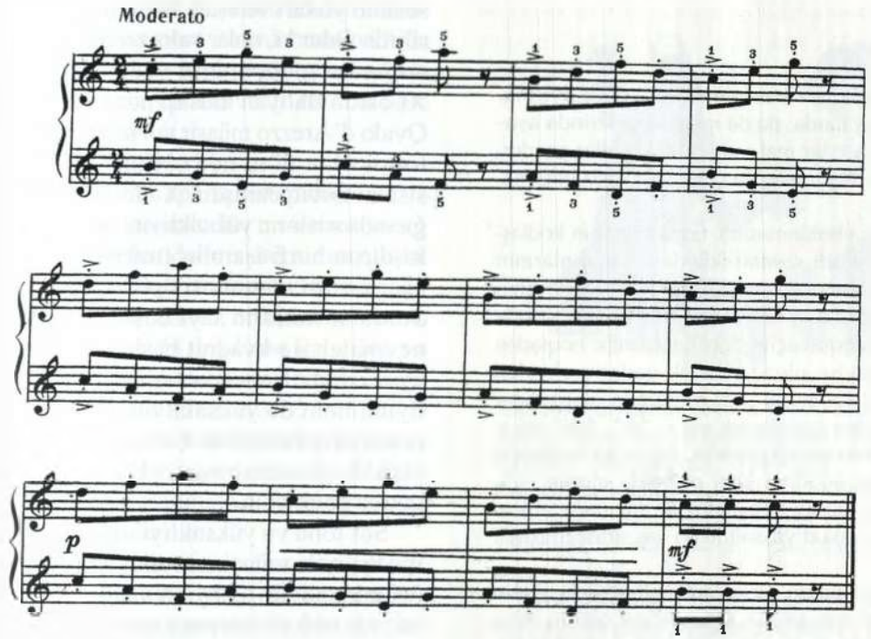
Məlumdur ki, səs özlüyündə havanın rəqsidir. Səs fasiləsiz siqnal olduğundan rəqslərin amplitudu daim dəyişir. Səsin kodlaşdırılması zamanı bu siqnalı sıfır və birlərin ardıcıllığı şəklində vermək lazımdır. Məsələn, mikrofondan istifadə etməklə səsi, rəqs amplitudunu bərabər zaman fasilələrində saniyədə onminlərlə dəfə ölçməklə elektrik cərəyanının rəqslərinə çevirmək olar. Hər bir ölçmə məhdud dəqiqliklə aparılır və ikilik şəkildə yazılır. Bu proses *diskretləşmə*, onun yerinə yetirilməsi üçün istifadə olunan qurğu isə, *analoq-rəqəm çeviricisi* (ARÇ) adlanır. Tipik səkkiz bitli ARÇ qurğusu -500-500 mV diapazonunda olan gərginliyi -128^{10}-127^{10} diapazonunda yerləşən səkkizmərtəbəli ikilik ədədə çevirir.

Əks proses – kodlaşdırılmış səsin səsləndirilməsi *rəqəm-analoq çeviricisi* (RAÇ) vasitəsilə həyata keçirilir. Bu zaman, ikilik rəqəm -128^{10}-127^{10} diapazonundan ikilik rəqəm -500-500 mV diapazonundakı gərginliyə çevrilir.

Alınmış pilləli siqnal əvvəlcə analoq süzgəc vasitəsilə hamarlanır, sonra isə gücləndirici və dinamikin köməyilə səsə çevrilir.

Kodlaşdırılmış səsin səsləndirilmə keyfiyyətinə əsasən iki parametr təsir edir: diskretləşmənin tezliyi və amplitudun qiymətinin yazılışına ayrılmış bit-





Stereosəslə işləyərkən diskretləşmə ayrı-ayrılıqda, sol və sağ kanallar üçün asılı olmadan aparılır.

lərin sayı. Məsələn, kompakt disklərdə (CD) yazı zamanı 16 mərtəbəli qiymətlərdən istifadə olunur, diskretləşmə tezliyi isə 44032 Hs-ə bərabərdir. Bu parametrlər musiqinin və nitqin əla keyfiyyətlə səslənməsini təmin edir.

Diskretləşmə tezliyinin seçilməsi onunla izah olunur ki, insanın eşidə biləcəyi səsin maksimal tezliyi 22 kHs-i aşmır. 22 kHs tezlikli rəqslər həmin tezlikli diskretləşmə zamanı sükutdan fərqlənmir. Diskretləşmənin hər bir periodunda iki qiymətin yazılması üçün iki dəfə böyük, məhz 44 kHs diskretləşmə tezliyi lazımdır. Yüksək keyfiyyət tələb olunmadıqda kiçik diskretləşmə tezliyindən istifadə etmək olar: 11 kHs, 5,5 kHs və s.

Hər növ səs kimi musiqi də səs dalğalarıdır və onları kifayət qədər dəqiq kodlaşdırdıqda, səsi sonradan səsləndirmək mümkündür. Ancaq belə üsul bəstəkara bəstələdiyi əsəri yazmağa, musiqiçiyə isə onu bir dəfə də olsun eşitmədən ifa etməyə imkan vermir.

Vəziyyətdən çıxış çoxdan tapılmışdır – *not yazısı*, yaxud *notasiya* (*lat.* "notatio" – "yazmaq", "qeyd etmək") – musiqini yazmaq üçün tətbiq olunan qrafik işarələr sistemidir. Orta əsrlərin əvvəlində xüsusi işarələrlə – xətlərdən, nöqtələrdən, vergüllərdən ibarət nevmalarla – ayrıca səslər, səs qrupları, səsin yüksəkliyini dəqiq təyin etmədən insan

Rus bəstəkarı Y.A.Şaporin yazırdı: "Mənə elə gəlir ki, dünyada musiqi kimi insanları doğmalaşdıran başqa bir incəsənət yoxdur. Onun dili hər kəsə aydındır..." O, nəzərdə tuturdu ki, milyonlarla insanlar musiqi alətlərinin adını, musiqi əsərlərinin yaranma prinsipini bilmədən musiqiyə qulaq asmaq və ondan həzz almaq qabiliyyətinə malikdir.

5 kHs tezlikdən yuxarı ikiyə vurma prinsipi pozulur. Yuksəkliyin bir oktava artırılmasını duymaq üçün tezliyi təxminən 10 dəfə artırmaq lazımdır. Hətta kamil musiqi eşitmə qabiliyyətinə malik insanlar da 5 kHs tezlikdən yuxarı səslər üçün notu təyin etməyə çətinlik çəkirlər. Bu 5 kHs və yuxarı səs pərdələrinin yüksəkliklərinin qəbulu mexanizmlərinin müxtəlifliyini göstərir.

RƏNGLƏR VƏ SƏSLƏR

Biz göy qurşağının yeddi rəngini və musiqidə yeddi notu ayırırıq. Təbiidir ki, nə səslənən melodiyalarda, nə də rənglər spektrində ayrı-ayrı səslər və rənglər arasında heç bir real sərhədlər, keçidlər yoxdur. Bu şərtilik musiqini və rəngi kodlaşdırmaq və onlar haqqında informasiyanı ötürmək üçün vacibdir.

Musiqi səsinin işarələrə necə bölünməsinə, fasiləsizliyinin kodlaşdırılmasına baxaq. Musiqi səsi adi səsdən (küydən) səs tonlarının olması ilə fərqlənir. Səs tonları yüksəkliyə görə (faktiki olaraq, havanın vibrasiya tezliyinə görə) dəyişir. Yüksəkliyin fasiləsiz dəyişməsində şərti olaraq hər hansı başlanğıc nöqtə seçilir ("do"). Başlanğıc nöqtədən bərabər və bir-birindən müqayisə oluna biləcək qədər aralı olan (dəyişilmə ölçüsü 1 və ya 1/2 səs tonlu şərti məsafədə olan) digər nöqtələr nizamlanır. Onların hər birinin adı var – "re", "mi" və s. Ardıcıl yerləşməklə onlar səs sırası əmələ gətirirlər. Səs sırası üzvlərinin havanı vibrasiya tezlikləri kəsr ədədləri kimi nisbətdə olurlar (4/3, 3/2 və s.), bu da akkordda harmonik həmahənglik duymağa imkan verir. Tezliyi iki dəfə azaldıb, yaxud yüksəltdikdə səs sırası müxtəlif oktavalarda təkrarlanır.

Musiqi səsləri üçün yüksəklikdən başqa səslənmə sürəkliyinin nisbəti vacib xüsusiyyətlərdən biri sayılır. Onu ardıcıl olaraq ikiyə bölünən tam etalon ilə tutuşduraraq qeyd edirlər: yarım, dörddə bir, səkkizdə bir, on altıda bir və s. (tamın sürəkliyi təxminən insan ürəyinin iki döyüntüsünə bərabər götürülür; əsas metrik vahid kimi dörddə bir çıxış edir; musiqi ölçüsü – zərbə hissəsindən başlayan və ritmi təşkil edən takt sürəkliyi məhz ona görə qeyd olunur).

Belə işarə sistemi kodlaşdırmanın dil formasından əlavə qrafik kodlaşdırma formasına – xüsusi simvolların köməyilə səsin sürəkliliyi və yüksəkliyinin işarə edildiyi musiqi savadına malikdir. Bu sistemin əsas işarəsi – not düşərgəsində yerləşən müəyyən mövqeli notdur; notun yüksəkliyinin onun mövqeyinə görə dekodlaşdırılmasına hesablama nöqtəsini verən açar (adətən, skripka və ya bas açarı) kömək edir. Not – həm səsin yüksəkliyini, həm də sürəkliyini kodlaşdıran mürəkkəb işarədir.

Müasir notlara başlanğıc verən vokal notasiyaların işarələri bizə kifayət qədər sonrakı Bizans yazılarından (bizim eranın IX əsrindən) məlumdur, ancaq musiqi yazıları sistemi xeyli əvvəl yaranmışdır. Məlumdur ki, qədim yunanlarda səsi və alətlərdə ifa olunan musiqiləri yazmaq üçün hərf sistemi mövcud idi. Qədim tibet ibadət kitablarında



səsinin yuxarı və aşağı hərəkəti göstərilirdi. Odur ki, onlar yalnız məlum olan nəğməni müğənninin yadına salırdı. XI əsrdə italiyan musiqi nəzəriyyəçisi Qvido d'Arezzo müasir not sətrinin prototipi olan dörd not xəttindən ibarət sistem tətbiq etmişdir. Xətlərin başlanğıcında səslərin yüksəkliyini müəyyənləşdirən hərfi işarələr (müasir musiqi açarlarının nümunəsi) yerləşdirilirdi. Sonralar xətlərin sayı beşə çatdırıldı, nevmalar isə kvadrat başlıqlı notlarla əvəz edildi. Nəticədə səsin həm sürəkliyini, həm də yüksəkliyini qeyd edən notasiya qəbul edildi. Çox da böyük dəyişiklik olmadan notasiya bizim günlərə qədər gəlmişdir.

Səs tonu və yüksəkliyi səs rəqslərinin tezliyilə müəyyən edilir. Hər bir not özünün məlum tezliyinə malikdir. Hal-hazırda birinci oktavanın notu "lya" əsas sayılır – 440 Hs (əvvəllər onun tezliyi 435 Hs-ə bərabər idi). Qalan notlar ona nəzərən təyin olunur, axı, insanın eşitmə qabiliyyəti üçün mütləq tezlik deyil, tezliklərin nisbəti vacibdir. Tezlikləri düz iki dəfə fərqlənən səslər oxşar, "qohum" kimi görünürlər. 262 Hs-dən $2 \cdot 262 = 524$ Hs-ə qədər olan tezliklərin intervalı 12 hissəyə bölünür; alınan 12 səs birinci oktava adlanır, onlardan yeddisini hamı tanıyır: *do, re, mi, fa, sol, lya, si*. Verilmiş 12 səsin tezliklərini iki dəfə artırdıqda ikinci oktavanı, növbəti ikiqat artırılmada isə üçüncünü alarıq.
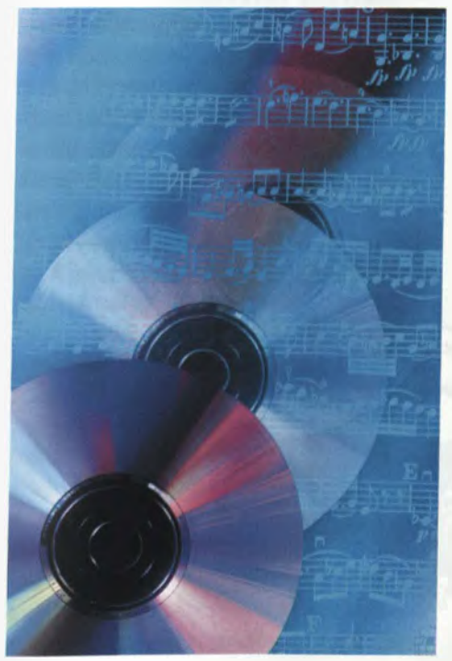
Musiqiçi üçün not yazısı komandalar ardıcıllığıdır. Məsələn, klavişi basmaq və onu saxlamaq, eyni vaxtda bir neçə klavişi basmaq, əvvəl basılmış klavişi buraxmaq və s. Bütün mümkün olan komandaları yazıb götürsək, onda xəyali ifaçının komandalar sistemi alınacaq.

XX əsrin 80-ci illərində bir şox mövcud musiqi alətlərinin səslərini, eləcə də tamamilə yeni səsləri canlandıran elektron musiqi alətləri – sintezatorlar meydana çıxdı. Elektron musiqi alətləri istehsal edən firmalar komandaları klavişlərdən yazan və sonra isə onları səsləndirən alət – *sekvensorlar* istehsal etməyə başladılar. Lakin yazının vahid formatı hələ mövcud deyildi.

1983-cü ildə elektron musiqi sintezatorlarının aparıcı istehsalçıları və kompüter istehsalçıları universal sintezatorun komandalar sistemi haqqında, belə komandaların hansı elektrik siqnalları ilə verilməsi, hətta kompüter və sintezatorları birləşdirən kabellər və birləşdirici yuvalar barəsində razılığa gəldilər. Bu razılaşma *MIDI standartı* (Musical Instrument Digital Interface – musiqi alətinin rəqəmli interfeysi) adını aldı. MIDI musiqinin kodlaşdırılmasının əlverişli üsulu oldu.

Lakin müxtəlif sekvensorların proqramı əvvəlki kimi MIDI-faylların müxtəlif formatları ilə işləyirdi. Hər bir firmanın yalnız özü oxuya bilən bağlı fayl formatı var idi.

1988-ci ildə Opcode firmasının təklifilə "Standard MIDI File" formatı





nəğmələr dalğavari xətlərlə işarələnirdi. Ehtimal olunur ki, bu işarələr səsin tondan-tona keçməsini göstərir. Rusiyada (XI əsr) yayılmış musiqi işarələri ümumilikdə "qarmaqlar" ("крюки"), yaxud "bayraqlar" ("знамёна") adlanırdı. Onlar içərisində sadə həndəsi işarələrdən – "qarmaqlar"dan, "çubuqlar"dan başqa daha mürəkkəb elementlər xarakterik adlarla – "kasa", "kasa altı", "parlaq bulud", "tutqun bulud" və s. seçilirdi.

Qədim notasiya sistemləri səsin konkret yüksəkliyini göstərmədən səs tonunun dəyişikliyini qeyd edirdilər, yəni onlar melodiyanı yazmaqla deyil, yada salmaqla ifaçıya sonrakı hərəkətlərini öyrədirdilər. Yalnız not dəsti daxil edildikdən və səsin yüksəkliyi qeyd edildikdən sonra müasir anlamda melodiyanın yazılışı mümkün oldu.

Spektrin rənglərinin ayrılması səslərin ayrılmasından daha şərtidir. Rəng spektri kəsiklərə bölünür, lakin işarələmə musiqi səslərində olduğu kimi parçanın daxilindəki nöqtəyə deyil, bütün parçaya aid edilir. Azərbaycan dilində danışan hər kəsə göy qurşağının rənglərinin ardıcıllığını yadda saxlamağa kömək edən "Qarı nənə səksən yaşında məxməri gülə bənzər" ifadəsi məlumdur. Amma ingilis və fransızlar üçün göy qurşağında rənglərin sayı yeddi deyil altıdır – mavi və göy rənglər bölünmürlər. Bu o demək deyil ki, onlar zəif görürlər, zərurət yarandıqda mavi rəng, məsələn, ingilis dilində "light blue" kimi qeyd olunur, lakin göy və qırmızı rənglərlə bərabərhüquqlu sistem elementi kimi qəbul edilmir.

Vyetnam və eləcə də yapon dillərində göy, mavi və yaşıl rənglər, ümumiyyətlə, bir sözlə ifadə edilir, yalnız ehtiyac yarandıqda əsas ada əlavə söz artırmaqla çaları dəqiqləşdirilir. Belə ki, Vyetnam dilində bu əsas rəngin – "xanh" rənginin 42 qəbul olunmuş rəng çaları var. Məsələn, dəniz dalğası "xanh"ı, düyü cücərtisi "xanh"ı. Tanqma papuas dilində (Yeni Qvineya adası) bütün əsas rəngləri göstərmək üçün yalnız 2 sözdən istifadə olunur: biri isti rənglər (sarı, narıncı, qırmızı), digəri isə soyuq rənglər (mavi, bənövşəyi, yaşıl) üçün.

Əsas rənglərin adlandırılmasının önəmi müxtəlif dillərdə fərqli olacaq. Rənglərin adlandırılması – şifahi ifadənin köməyilə ətraf aləmdə hadisələri kodlaşdıran universal təsnifatçı kimi, dilin roluna parlaq nümunədir. Dillərin, müxtəlif dillərdə danışanlara özünün xüsusi baxış tərzini qəbul etdirərək, dünyanı fərqli şəkildə parçalamaları fikri, müəyyən mənada, düzgündür.

1983-cü ildə ilk MIDI – Prophet 600 sintezatoru təqdim olundu. Çox keçmədən Roland sintezatorlu MIDI-interfeys PC çıxdı. 1985-ci ildə dünyada praktiki olaraq bütün elektron musiqi alətləri MIDI yuvasına malik idi.

qəbul edildi. Bu açıq fayl formatını MIDI-ni öz formatıyla yanaşı istifadə edən bütün proqram istehsalçıları dəstəklədi.

MIDI formatında musiqi əsərinin yazılışı – kodlaşdırılmış pauzalarla ayrılmış kodlaşdırılmış məlumatın sintezatora verilmə ardıcıllığıdır. Məlumat sintezator komandası ola bilər (müəyyən olunmuş klavişi basmaq, yaxud buraxmaq, səslənmənin yüksəkliyini və tembrini dəyişdirmək və s.); səsləndirmə parametrlərinin təsviri ola bilər (məsələn, klavişə düşən təzyiqin gücünün qiyməti); idarəedici məlumat da ola bilər (məsələn, məlumatı sinxronlaşdıran polifonik rejimin qoşulma komandası və s.).

Lakin belə kodlaşdırma ilə vokal əsərlərini yazmaq olmaz, çünki, ifaçının və ya xorun oxumalarının səslənməsi sintezatorun komandalar sisteminə daxil deyil. Qalan hallarda belə kodlaşdırma sistemi olduqca əlverişlidir, çünki, yazı çox yığcamdır və insanın eşitmə qabiliyyətinə uyğun gəlir.

FİLMLƏRİN KODLAŞDIRILMASI

Kinematoqrafiyanın kəşf edildiyi ildən – 1895-ci ildən bizim günlərədək filmlərin istehsalı üçün texniki vasitələr kinoçəkiliş aparatı və kinolentdir.

İnsanın görmə xüsusiyyəti hərəkətin ardıcıl fazalarının təsvir olunduğu kadrların tez-tez dəyişdirilməsi (saniyədə 15 dəfədən çox) hərəkət illüziyasını yaratmağa imkan verir. Kino da, televiziya da, filmlərin kompüter kodlaşdırılması da bu prinsipə əsaslanıb.

Təsvirin rəqəmli kodlaşdırılmasından (yazısından) son zamanlarda istifadə olunmağa başlanmışdır, çünki, ənənəvi üsulla bu məsələnin reallaşdırılmasına resurslar çatmırdı. 1 saniyəlik (ölçüsü 1024×768 olan 25 kadr) səssiz rəngli təsviri yazmaq üçün təqribən 60 Mbayt (25 kadr · 1024 · 768 · 3 bayt hər nöqtəyə = 58 982 400 bayt) tələb olunur. Bu zaman televiziyaya yaxın keyfiyyət təmin olunacaqdır.

Asanlıqla hesablamaq olar ki, iki saatlıq filmin yazılışı üçün 400 Gbayt gərəkdir. XXI əsrin başlanğıcında masaüstü kompüterin vinçesterinin orta hesabla həcmi bir neçə on giqabaytı aşmasına baxmayaraq, bu həcmdə informasiyanın saxlanması baha başa gəlir.

Görüntüyə səs əlavə edilsə, belə yazılış həcmi o qədər də böyüməyəcək. Kompakt-diskdə yazılışda olduğu kimi

Videokodlaşdırma sahəsindəki işlər Video-CD (videokompakt – disklər üçün formatların) və DVD-nin yaranmasına səbəb oldu. "Sıxlaşdırılmış" kinonun nəinki saxlanması, həm də onun göstərilməsi (yazılışı, göndərilməsi) baha başa gəlir. Belə ki, bunun üçün yüksək sürətli rabitə kanalları və yüksək məhsuldarlıqlı kompüterlər lazımdır.



1 san filmin yazılışı üçün təqribən 170 Kbayt lazımdır.

Səs görüntüdən asılı olmadan yazılır, lakin canlandırma zamanı sinxronluğu gözləmək gərəkdir.

Filmlərin yazılışları zamanı böyük həcmli informasiya problemini həll etmək üçün informasiya daşıyıcılarının yeni formaları işlənib hazırlanır, ağıllı alqoritmlər düşünülüb tapılır. Məsələn, kadrların özlərini deyil, yalnız onların dəyişmələrini yadda saxlamaq olar. Onda əgər şəkil statikdirsə, növbəti kadrın yazılışı üçün tələb olunan həcm kiçilir.

Buna baxmayaraq alimlər özlərinin başlıca uğurlarına insan gözünün daha bir xüsusiyyətini nəzərə almaqla nail oldular. İnsan gözü ayrıca bir nöqtənin rəngliliyindən çox, onun işıqlılığına daha həssasdır. Buna görə də onlar hər bir dörd nöqtə üçün 4 bayt işıqlılıq (hər bir nöqtə üçün müxtəlif) və 2 bayt rənglilik üçün yer saxlamağı təklif etmişlər. Beləliklə, zəruri olan həcm iki dəfə azaldı: 12 bayt (12 = 4 · 3) əvəzinə yalnız 6 bayt (4 · 1 + 2) yer ayrılırdı. Yazılışın keyfiyyəti korlansa da, bu insan gözü tərəfindən demək olar ki, seçilmir.

Montaj zamanı səsin və təsvirin sinxronlaşdırılmasını üzərində səhnə və dubl nömrəsi olan "şax-şax" şıqqıltısı təmin edir. Şıqqıltı anı növbəti dubldakı çəkilişin əvvəlində lentdə yaxşı görünür (həmçinin "şax-şax" səsi yaxşı eşidilir). Bu onu asanlıqla təsvirin və səsin əvvəlilə uyğunlaşdırmağa imkan verir.

MÜHASİBAT İNFORMASİYASININ KODLAŞDIRILMASI

Puldan söz düşəndə, demək olar ki, hər bir qadın ekspert kimi çıxış edir. Bir qayda olaraq, ailədə büdcəni qadınlar idarə edir: xərcləri planlaşdırır, gəlir, məsrəf və çıxarları nəzərə alır. Ailə mühasibatının düzgün aparılması xüsusi diqqət tələb edir, hər gün aparılan qeydlər ilin sonunda qalın bir dəftərə çevrilir. Bu dəftərdə, demək olar ki, hər şey – nəyə, nə vaxt, nə qədər xərcləndiyi də göstərilmişdir.

Mahiyyət etibarı ilə, müəssisənin mühasibat işi elə bu məqsədlərə qulluq edir. O, ailə mühasibatlığından müxtəlif tipli informasiyanın böyük həcmi və mühasibat uçotunun aparılması ilə bağlı müəyyən qaydaların olmasıyla fərqlənir.

Mühasibatlıqda pullar və qiymətli materiallar pul ekvivalentində göstərilməlidir. Müəssisə iki tip kapitaldan istifadə edir: öz kapitalından və cəlb olunan kapitaldan (*kredit borcu*). Bunlar müəssisənin iqtisadi resurslarını təşkil edir:



İqtisadi resurslar =
= Cəlb olunan kapital +
+ Özəl kapital.

Aktivlərə pul vəsaitləri, hazır məhsul (əmtəə ehtiyatları), torpaq sahələri, avadanlıqlar, hətta patentlər, müəlliflik hüquqları və s. kimi qeyri-maddi aktivlər də aiddir.

Passivlər müəssisənin borclarını, kreditə götürülmüş əmtəələri, borc alınmış pulları, müəssisənin öz əməkdaşlarına əmək haqqı və dövlət vergisi üzrə borcları əks etdirir.

Mühasibatlıqda iqtisadi resurslar – aktivlər, cəlb olunan kapital isə passivlər adlanır. Ona görə də əsas tənliyi aşağıdakı kimi yazmaq olar:
Aktivlər = Passivlər + + Kapital

Aktivlər		Passivlər	
Kassa	50	Nizamnamə kapitalı	85
Əsas vəsaitlər	01	Mal verənlərə borc	60
Hesablaşma hesabı	51	Əməyin ödənişi üzrə borc	70
Hazır məhsul	40	Ehtiyat kapitalı	80

Azərbaycanda hələlik, əski Sovetlər Birliyində olduğu kimi, sadəlik üçün, müəssisənin öz kapitalı da (müəssisə sahiblərinin nizamnamə ödənişləri daxil olmaqla) buraya daxil edilir. Mühasibatlığın ən vacib anlayışı balans anlayışıdır. Balans iki tərəfli bir cədvəldir. Bu cədvəlin sol tərəfi müəssisənin təsərrüfat vəsaitlərinin tərkibini, onların yerləşdirilməsini özündə əks etdirən *aktivi*, sağ tərəfi isə bütün daxilolmaları özündə qruplaşdıran *passivi* (və ya passiv + kapital) göstərir. Balansın aktivlərinin cəmi passivlərinin cəmilə üst-üstə düşməlidir, çünki bu faktiki olaraq, eyni vəsaitlərə iki müxtəlif baxışdır.

Müəssisədə idarəetmə və uçot üçün zəruri olan informasiya toplanmalı və həmişə müəssisə rəhbərinin əlinin altında olmalıdır. Mühasibat uçotunda informasiyanın saxlanması sistemi *hesablardan* təşkil olunur. Hesab hər növ aktiv, passiv və ya kapital üçün, gəlirlər və xərclər də daxil olmaqla yaradılır. Aktiv və ya passivlərin uçotu üçün uyğun hesablardan istifadə olunur. Hər hesabın öz şifrəsi var, bu şifrəni hesabın adının yerinə yazmaq olar. Kiçik firmaların hesablar planında bu hesabların sayı onlarladırsa, böyük şirkətlərdə minlərlədir.



İ.Q.Tişbeyn.Yohann Volfqanq Gete Roma şəhəri ətrafında. 1787-ci il.

Mühasibat uçotu əllə aparıldıqda hər hesab üçün ayrıca səhifə və ya kart ayrılır.

Böyük alman şairi Yohann Volfqanq Gete mühasiblərə ikrah hissilə yanaşaraq ikili mühasibatı "insan zəkasının ən görkəmli ixtiralarından biri" adlandırırdı ancaq bu fikirdə müəyyən qədər həqiqət də var.

İkili yazı sistemi ikitərəflilik prinsipinə əsaslanır. Bütün iqtisadi hadisələr, sanki enerjinin saxlanması qanununa tabe olaraq bir-birlərini kompensasiya edən iki aspektə – artma və azalmaya malikdir.

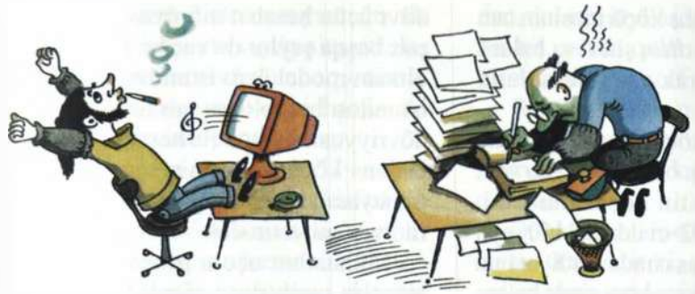
Bu sistemdə təsərrüfat həyatı faktı ən azı iki dəfə qeyd olunmalıdır: bir hesabın *debetində*, başqa hesabın *kreditində*. Belə ki, debet üzrə ümumi cəm, kredit üzrə ümumi cəmə bərabər olsun.

"T" hərfinə oxşar aşağıdakı cədvəldəki yazıda hesabın adı, debet (sol tərəf) və kredit (sağ tərəf) olur. Tutaq ki, kassaya pullar daxil olur (sol tərəf) və ordan ödənişlər (sağ tərəf) aparılır:

Kassa	
1000	500
300	120
100	
1400	620
Saldo	780

Aktiv hesablar vəsaitlərin qalığını göstərən debet saldosuna, passiv hesablar isə bütün daxilolmaları nəzərə alan kredit saldosuna malikdir. Passiv hesablarının kreditində bu mənbələrin artımı, debetində isə azalması əks olunur.

Hər bir təsərrüfat əməliyyatı bir hesabın debetində və başqa hesabın kreditində eyni məbləğlə göstərilir (balans köçürməsi). Bu da dediyimiz həmin o



ikili yazılış qaydasıdır. Hesablar arasında qarşılıqlı əlaqə yaranır: debet üzrə bir hesab, kredit üzrə başqa hesabla bağlıdır. Məsələn, əsas vəsaitlərin köhnəlməsi hesablanarkən 80 hesabı debetləşir, 02 hesabı isə kreditləşir. Bunu belə təsəvvür etmək olar: gəlirdən müəyyən məbləğ köhnəlmiş avadanlığın gələcək təmiri, yaxud təzəsilə əvəz olunması məqsədilə ayrılıb kənara qoyulur.

Əməliyyatlar zamanı hesablardakı məbləğlər dəyişir, amma aktiv və passivin bərabərliyi pozulmur. Hər bir hesab üzrə vəsaitlərin mədaxil və məxaricinin qeyd olunduğu cədvəl tərtib olunur. Hər ayın sonunda bu cədvəllər əsasında hər bir hesab və ya hesablar qrupu üçün vəsaitlərin mədaxil məxariclərini cəm şəklində özündə əks etdirən *jurnal-orderlər* tərtib olunur. Bu jurnal-orderlər əsasında hər hesabat dövrünün sonunda *Baş kitaba* qeydlər aparılır və *müəssisənin balansı* tərtib olunur.

Bu, həcm etibarilə çox böyük bir işdir: bəzən mühasibatlıqda yüzlərlə hesab aparıla bilər və ay ərzində bu hesabların hər biri üzrə balans köçürmələrinin sayı yüzlərlə, hətta minlərlə ola bilər.

Balansın aktiv və passivlərinin bərabərliyinin saxlanılması, jurnal-orderlərin doldurulması böyük səy tələb edir. Balans köçürmələri aparılan sənədlərin uçotunu, ödəniş tapşırıqlarının, əmək haqqı cədvəllərinin, mədaxil və məxaric orderlərinin tərtibini də bura əlavə etsək, mühasibat uçotunun aparılması işinin nə dərəcədə zəhmət tələb edən, məsuliyyətli bir iş olduğu aydın olar. Məhz bu işdə kompüter insana lazımi köməyi edə bilər. Bəs, kompüterə hansı mühasibat məsələlərini tapşırmaq daha yaxşı olar?

İlk növbədə kompüterdən, vəsaitlərin hərəkəti üzrə informasıyanın saxlanması üçün bir arxiv kimi istifadə etmək olar. Hər hesab üçün aşağıdakı informasiyanı yadda saxlamaq lazımdır: hesabın şifri, hesabın adı, tipi (aktiv və ya passiv), qalığı (hesabda vəsaitin cari miqdarı). Hesablar haqqında informasiya cədvələ yerləşdirilir. Tutaq ki, müəssisənin kassasında 1100 manat var. Onda cədvəl bu şəkildə olur:

Şifr	Ad	Tip	Qalıq
50	Kassa	Aktiv	1100

Debetləşən və kreditləşən hesablar istənilən kimi seçilə bilməz. Mühasibin istifadə edə biləcəyi, ciddi şəkildə müəyyən olunmuş balans köçürmələri vardır. Ona görə də, mühasibat informasiyası kodlaşdırılan zaman bu siyahını da mütləq yazmaq lazımdır, məsələn:

Tip	Ad	Debetin şifri	Kreditin şifri
1	Hazır məhsulun satışı	40	51
2	Hesablaşma hesabından pulun götürülməsi	51	50
3	Əsas vəsaitlərin köhnəlməsinin hesablanması	80	02

Burada tip, bir növ, balans köçürməsinin şifridir; başqa cədvəllərdən bu balans köçürmələri tiplərinin cədvəlinə istinad

"Debet" (*lat.* "debet" – "o borcludur") və "kredit" (*lat.* "credit" – "o inanır") sözləri artma və ya azalmanı göstərmək üçün deyil, sadəcə, hesabın sol və sağ tərəflərini göstərmək üçün mühasibat terminidir.

İkili yazı sistemi İntibah dövründə yaranmışdır. Hələ 1494-cü ildə fransiskan monaxı, Leonardo da Vinçinin dostu Luka Paçoli ilk dəfə olaraq ikili yazı sisteminin təsvirini vermişdi.



Luka Paçoli.

edilə bilər, ad – balans köçürməsinin tam adıdır, *debet və kreditin* şifri isə balans köçürməsində iştirak edən hesabların şifrələridir.

Hər bir balans köçürməsi üzrə bunları bilmək vacibdir: onun tipini, nə vaxt və nə qədər vəsaitin köçürülməsini. Əgər 31 dekabr 2002-ci ildə, 23450 saylı balans köçürməsi əsasında 1000 manat köhnəlmə hesablanmışdırsa, onda balans köçürməsi cədvəlində yazı bu şəkildə olacaq:

Nömrə	Tip	Tarix	Miqdar
23450	3	31.12.2002	1000

Əlbəttə, mühasibatlıqda saxlanılan informasiyanın həcmi qat-qat çoxdur. Balans köçürmələrinin yerinə yetirilməsinə əsas olan ilkin sənədlər, müəyyən dövr üçün hesabat informasiyası və bir çox başqa şeylər də vardır. Hətta təklif olunan model üzrə istənilən dövr üçün istənilən hesab üzrə vəsaitlərin yekun dövriyyəsini asanlıqla hesablamaq olar. Balans köçürmələrinin sayından asılı olmayaraq kompüter bu işi səhvsiz olaraq yerinə yetirəcək!

Mühasibat uçotu proqramı mühasibatlıq işçilərinin gündəlik əməyini yüngülləşdirməyə qadirdir. Məsələn, kassadan pul verilərkən balans köçürməsindən əlavə məxaric orderi də doldurulmalıdır. Bu işi də kompüter görür: mühasib məbləği göstərir və kompüter balans köçürməsini apararaq orderi çap edir. Bu təkcə vaxta qənaət etmir, həm də orderdəki məbləğin balans köçürməsindəki məbləğlə üst-üstə düşməsinə tam zəmanət verir.

İNFORMASİYANIN SIXILMASI YOLLARI

Təkcə təbii resurslara deyil, ümumiyyətlə resurslara qənaət etmək lazımdır. Bunu bütövlükdə, informasiyanın saxlanması və ötürülməsi məqsədilə istifadə olunan EHM-in yaddaşına da şamil etmək olar. Çox vaxt kodlaşdırılmış informasiyanı elə çevirmək olar ki, son nəticədə o, yaddaşda daha az yer tutsun. Bu prosesə informasiyanın qablaşdırılması və ya sıxılması deyilir.

Burada məqsəd yalnız yerə qənaət etməkdir, informasiyanın gizlədilməsi və şifrələnməsi deyildir.

İnformasiyanın sıxılması texnologiyası iki böyük sinfə bölünür:

– informasiyanın itkisiz sıxılması – bu zaman ilkin informasiyanı qablaşdırılmış informasiya əsasında tam dəqiqliyilə bərpa etmək olar;

– informasiyanın itkili sıxılması – bu halda bərpa olunmuş informasiya ilkin informasiyadan fərqlənəcək. Sonra görəcəyik ki, belə itki heç də həmişə faciəvi olmur, yəni itmiş informasiyanı nəzərə almamaq da olar. Ancaq əksər hallarda informasiyanın təhrif olunması arzuolunmazdır və ya yolverilməzdir (məsələn, proqram və mətnlər üçün).

ÜMUMİLƏŞMİŞ RLE-KODLAŞDIRMASININ ALQORİTMİ

```
alq RLE kodlaşdırması
verilib mətn
gərəkdir RLE kodlarını çap etmək
baş tam sayğac, lit simvol1, simvol2
| sayğac := 0
| simvol1:= mətndən oxumaq
| db nə qədər ki mətnin sonu
| | simvol2 := mətndən oxumaq
| | əgər simvol1 = simvol2 və sayğac < 255
| | | onda sayğac := sayğac + 1
| | tamam
| | çap et sayğac
| | çap et simvol1
| | simvol1 := simvol2
| | sayğac := 0
| ds
son
```


Bu günə qədər informasiyanın itkisiz sıxılması üçün çoxlu sayda alqoritmlər işlənib hazırlanmışdır, ancaq onların alqoritmlərin, demək olar ki, hamısı iki sadə ideyadan birinə əsaslanır.

Birinci ideya ilk dəfə 1952-ci ildə Devid Haffman tərəfindən təklif olunmuşdur və bu ideya adi mətndə müxtəlif simvolların rastgəlmə tezliyinin müxtəlif olmasına əsaslanır. Mətnin standart kodlaşdırılmasında hər bir simvol bir baytla kodlaşdırılır. Bir simvol üçün bir baytdan istifadə mətnin emalını asanlaşdırır. Amma uzun müddət saxlanıma (və ya rabitə kanalı ilə ötürülmə) üçün daha mürəkkəb kodlaşdırmaya əl atmaq olar. Haffman metodu ilə qablaşdırma zamanı tez-tez rast gəlinən simvollar qısa bitlər ardıcıllığı ilə (8-dən az), az rast gəlinənlər isə uzun bitlər ardıcıllığı ilə (ola bilər 8-dən çox) kodlaşdırılır. Nəticədə, orta hesabla, bir simvola 8 bitdən az düşür.

Bunu bir misalda aydın şəkildə göstərək. Tutaq ki, 1000 baytdan ibarət bir mətndə 50% boşluq simvolu var. Onda bu mətni ikilik 0 və 1 ardıcıllıqları şəklində kodlaşdırmaq olar.

Əgər mətndə boşluq simvoluna rast gəlinirsə, ardıcıllığın sonuna 0 yazılır, əks halda – 1 və həmin simvolun ikilik kodu yazılır, yəni 1+8 bit. Mətnin yarısı boşluq simvollarından təşkil olunduğundan, onların kodlaşdırılmasına 500 dəfə bir bit, yerdə qalan 500 simvolun hər birinə isə 9 bit sərf olunacaq.

Beləliklə, bütün mətnə $500 + 9 \cdot 500 = 5000$ bit sərf olunacaq ki, bu da ilkin mətndəki 8000 bitdən əhəmiyyətli dərəcədə azdır.

Qablaşdırmanın ikinci əsas ideyası ondan ibarətdir ki, məlumatda tez-tez ardıcıl yerləşən eyni baytlara tez-tez rast gəlinir, bəzi baytlar ardıcıllığı isə sonsuz sayda təkrarlanır. Birinci hala daha çox qrafik informasiyaların, ikinci hala isə daha çox mətnlərin qablaşdırılmasında rast gəlinir. Məsələn, rəsmdə mavi səma eyni rəngdən ibarət olduğundan, kodlaşdırma zamanı, bu rəngin çox sayda təkrarlanması nəzərə alınır. Qrafik informasiyanın saxlanılması üçün xeyli çox yer tələb olunduğundan onun sıxılması məsələsi çox önəmlidir. Ümumiyyətlə, qrafik informasiya çox az hallarda kompüterdə sıxılmamış şəkildə saxlanılır.

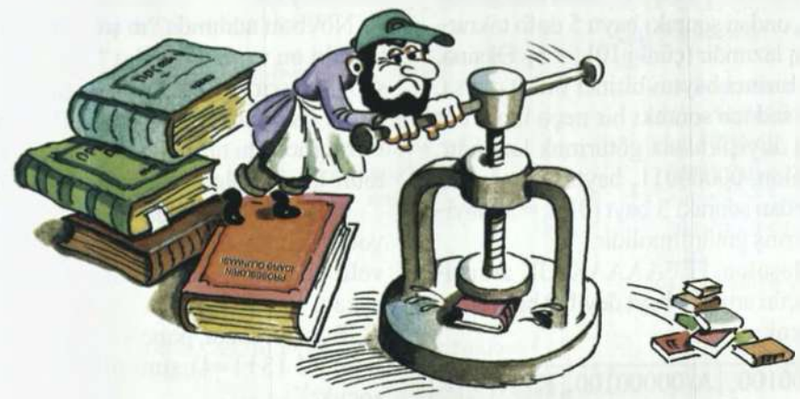
Haffmanın ideyalarından CCITT (Teleqraf, telefon və faks verilişi üçün Beynəlxalq Komitənin abbreviaturası) sıxılmasında istifadə olunur.



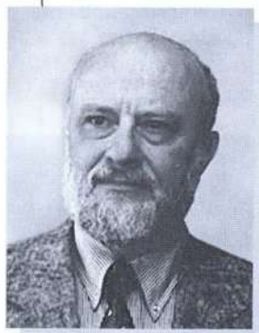
Devid Albert Haffman.

QRUPLARLA KODLAŞDIRMA

RLE – kodlaşdırma (*ing.* Run-Length Encoding – təkrarlanma sayına görə kodlaşdırma). Bu üsul nəinki təsvirlərin, istənilən məlumatların kodlaşdırılmasına tətbiq oluna bilər, həm də eyni baytlardan ibarət uzun ardıcıllıqları yığcam şəkildə kodlaşdırmağa imkan verir.



LZSS alqoritminin dəyişdirilmiş forması mövcuddur ki, burada əgər simvolun göstəricisi baytdan az yer tutursa, adətən triadada simvolun əvəzinə onun göstəricisi saxlanılır.



Abraham Lempel.



Cekob Ziv.

Məsələn:

AAAAAAAAAAAAAAA

Belə qrup, adətən, iki bayt vasitəsilə kodlaşdırılır. Birinci bayt qrupdakı simvolların sayını, ikinci bayt isə təkrarlanan simvolu göstərir:

15 A

Başqa bir misal:

FFFAAAAKUUUUK qrupunun kodlaşdırılması üçün artıq 2 deyil, 10 bayt lazım gələcək:

3	F	4	A	1	K	4	U	1	K

Beləliklə, təkrar olunmayan baytlardan ibarət ardıcıllıq RLE qablaşdırılmasında öz ilkin tutduğu yerdən daha çox yer tutacaq.

RLE metodunun belə bir modifikasiyasını təklif etmək olar.

Tutaq ki, birinci bayt özündə təkcə təkrarlanma sayı haqqında deyil, həm də təkrarlanmanın olub-olmaması haqqında informasiyanı saxlayır. Bu baytın birinci biti (ən solda duran) 1-ə bərabərdirsə, onda sonrakı baytı qablaşdırma zamanı yerdə qalan 7 bitdə yazılmış ədəd qədər təkrarlamaq gərəkdir.

Məsələn, 10000101_2 baytı onu göstərir ki, ondan sonrakı baytı 5 dəfə təkrarlamaq lazımdır (çünki $101_2 = 5$). Əksinə, əgər birinci baytın birinci biti 0 olarsa, onda sadəcə sonrakı bir neçə baytı hər hansı dəyişikliksiz götürmək lazımdır. Məsələn, 00000011_2 baytı onu göstərir ki, ondan sonrakı 3 bayt ($011_2 = 3$) dəyişilməmiş götürülməlidir.

Məsələn, FFFAAAAKAJK ardıcıllığı üçün artıq 12 bayt deyil, 9 bayt lazım gələcək:

10000011_2	F	10000100_2	A	00000100_2	K	A	J	K

RLE-kodlaşdırma alqoritmləri çox sadə və sürətlidir, ancaq sıxılmanın effektivliyi, yəni ilkin ardıcıllıqdakı baytların sayının, sıxılmadan sonra alınan ardıcıllıqdakı baytların sayına nisbəti, nəyin sıxılmasından çox asılıdır. Kitab səhifəsindən daranmış (skanerdən keçirilmiş) ağ-qara təsviri sıxlaşdırmağa cəhd etsək effektivlik çox yüksək olacaq, çünki ağ rəng qaradan xeyli çoxdur, yəni təsvirdə bir rəngin kəsilməz böyük hissələri var. Amma, çox sayda rəngləri olan mürəkkəb təsvir pis kodlaşdırılır, çünki, eyni rəngə malik qrupların sayı az olur.

LZW KODLAŞDIRMASI

Təsvirlərin sıxlaşdırılmasında Lempel-Ziv-Uelç (Lempel-Ziv-Welch – LZW) sıxlaşdırması ən geniş yayılmış sxemlərdən biridir. 1977-ci ildə Abraham Lempel və Cekob Ziv tərəfindən ilk LZ77 alqoritmi yaradıldı. Bu alqoritm təqribən belə işləyirdi. Sıxılma, əvvəlki ardıcıllıqda rast gəlinmiş eyni simvollar qrupunun üçlüklə (triada) əvəz olunması hesabına baş berir. Bu uçlük, qrup göstəricisindən, üst-üstə düşən qrupun uzunluğundan və qrupdan sonra gələn ilk fərqli simvoldan ibarətdir.

F fraqmentinin maksimal uzunluğu 10–20 simvola bərabər götürülür. Mətnə baxılan "sürüşkən" pəncərə N* ilə işarə olunmuşdur.

Növbəti addımda "ın şəri" fraqmentindəki ən uzun ardıcıllıq "ın xeyrindən sabahın şəri" buferində axtarılır. Bu halda üst-üstə düşən simvollar qrupu F fraqmentini tam örtə də bilər, amma, təbii ki, onunla üst-üstə düşə bilməz.

Göstərdiyimiz misalda tapılmış simvollar qrupu – "ın_" (sonda boşluq simvolu ilə), (7, 3, Ş) triadası ilə əvəz olunacaq.

Bundan sonra, pəncərə qrupun uzunluğu +1 (3+1=4) simvol qədər sürüşəcək:**

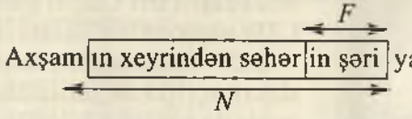
Çox güman ki, buferdə hətta fraqmentin başlanğıcı tapılmayacaq, yəni fraqment, sadəcə, buferdə olmayan simvolla başlayır. Onda alqoritmin qaydasına görə bu simvol (0, 0, olmayan simvol) triadası ilə əvəz olunacaq və pəncərə 1 simvol sürüşəcək.

Verilmiş misalda F =7, N =25 olduğundan, triadanın ikilik kodlaşdırılmasına cəmisi 2 bayt tələb olunacaq: bir bayt – simvola, o biri bayt isə – iki ədədə. Qrupun uzunluğuna ən çoxu 7, yəni 3 bit, göstəriciyə isə N-F =25–7 =18, yəni 4 bit. Cəmi, 3+4 =7 bit yəni, bir baytdan azdır.

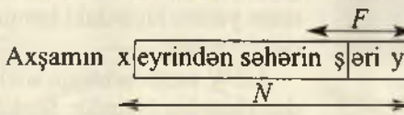
Baxış pəncərəsinin saxlanması üçün isə N bayt lazım gələcək. Bu da onu göstərir ki, LZ77 metodu EHM yaddaşını səmərəli istifadə edir.

Lempel və Ziv göstərdilər ki, N-in kifayət qədər böyük qiymətlərində onların alqoritmi çox yaxşı nəticələr verir.

*



**



Bir ildən sonra onlar, ayrıca sözlüklərdən istifadə edən yeni LZ78 alqoritmini təqdim etdilər. Burada təkrar olunan qruplar lüğətə yerləşdirilərək, sıxılmış fayla əlavə olunurdu. Kodlaşdırılan mətnə isə, bu qrupların göstəriciləri yerləşdirilirdi. Bu alqoritm 1984-cü ildə Unisus firmasının

LZW ALQORİTMİ

LZW metodu adətən 4096 elementi olan sözlükdən istifadə edir. 0-255 nömrəli elementlər 8-bitli kodlaşdırmadan bütün 256 simvolunu saxlayır. Sözlüyün bu elementləri həmişə sabitdir, 256-4095 nömrəli yerdə qalan elementlər başlanğıcda boş olurlar, lakin iş prosesində aşağıdakı alqoritmlə təyin edilən sözlərlə doldurulurlar:

```
alq LZW kodlaşdırması
verilib sözlükdə 256 simvol
gərəkdir LZW kodlarını çap etmək
baş lit söz, simvol
| söz := ""
| db
| | simvol := Mətndən simvol
| | əgər Sözlükdə var (söz + simvol)
| | | onda söz := söz + simvol    | simvolun sözə
| | |                               əlavə edilməsi
| | | əks halda çap et Kod (söz) | sözlükdən kod
| | | Sözlüyə əlavə etmək (söz + simvol)
| | | söz := simvol
| | tamam
| ds
son
```

Sözlük aşırı dolursa, Uelç alqoritmi sanki işi yenidən başlayaraq onu boşaldır.

Bu alqoritmin, aşırı dolma zamanı bütün sözlüyü deyil, yalnız onun hər hansı bir hissəsini (məsələn, birinci yarısını) təmizləyən modifikasiyaları mövcuddur.

Qablaşmadan qurtaran alqoritm də bu prinsiplə, lakin bir qədər mürəkkəb işləyir.

```
alq LZW dekodlaşdırması
verilib sözlükdə 256 simvol
gərəkdir dekodlaşdırılmış simvolları çap etmək
baş lit söz, simvol, tam k, k0
| k0 := Kodu oxumaq
| çap Sözlük(k0)
| db
| | k := Kodu oxumaq
| | söz := Sözlük(k0)
| | əgər Sözlük(k) <> Boşdur   | Söz sözlükdə var ?
| | | onda çap et Sözlük(k)
| | | simvol := Sözlük(k)da birinci simvol
| | | əks halda simvol := söz[1] | birinci simvol
| | tamam
| | Sözlüyə əlavə etməli (söz + simvol)
| | k0 := k
| ds
son
```



Benua Mandelbrot.

əməkdaşı Terri Uelçin yaratdığı yeni LZW alqoritminin əsasını təşkil etdi.

LZ77 alqoritmindən fərqli olaraq bu alqoritm, işləmə prosesində, xüsusi qayda ilə düzəlmiş sözləri toplayan bir lüğət yaradır.

Əgər mətndə rast gəlinən qrup lüğətdəki sözlə üst-üstə düşürsə, onda onun yerinə lüğətdəki həmin sözün indeksi yazılır.

LZW metodu başqa sözlük qablaşdırıcılarından üstündür. Sözlük həm qablaşdırma zamanı, həm də qablaşdırmadan açma zamanı yaradılır. Bu ailədən olan alqoritmlərin hamısı sözlüyün yaradılması qaydasına və sözlük dolub daşarkən davranışına görə fərqlənir.

LZW metodu geniş yayılıb, ancaq, onun patenti Unisys Corporation firmasına məxsus olduğundan ondan proqramlarda istifadə etmək üçün uyğun lisenziyanın alınması tələb olunur.

FRAKTAL SIXILMA

Fraktal, müxtəlif miqyaslı və ölçülü oxşar forma və rəsmlərdən təşkil olunmuş strukturdur. İlk dəfə olaraq bu termini Benua Mandelbrot, bir çox strukturlarda rast gəldiyi təkrar olunan rəsmləri təsvir etmək üçün işlətmişdir. O, müəyyən etdi ki, fraktalları riyazi təsvir etmək mümkündür və onları çox sadə alqoritmlərlə yaratmaq olar.

Əgər hər hansı bir döşəməyə nəzər salsaq, orada çoxlu sayda oxşar, təkrar olunan fraqmentlərə rast gəlmək olar. Bu halda, döşəmənin səthinin necə olması heç bir əhəmiyyət kəsb etmir. Döşəmənin parketdən, betondan olmasından, xalçayla döşənməsindən və s. asılı olmayaraq, şəkillər ən kiçikdən tutmuş, lap böyüyə qədər təkrarlanacaq. Əgər döşəmənin səthinin bir hissəsinin üzünü köçürsək, demək olar ki, eynilə bu hissədən döşəmənin başqa yerində də tapmaq olar. Bundan başqa, əgər bir hissənin ölçülərini dəyişsək, fırlatsaq və ya güzgü əksini alsaq, onda oxşar hissələrin sayı daha da çox olacaq. Onda bütün oxşar sahələri və kopyanın üzərində aparılmış bütün (riyazi) çevirmələri saymaq olar. Alınmış verilənlərə (tənliklər sisteminə), adətən, *fraktal kodlar* deyilir. Belə halda deyirlər ki, onlar səthi öz fraktal xüsusiyyətlərilə təsvir edirlər. Onlardan istifadə etməklə döşəmənin səthini yaratmaq olar.

Fraktal görüntü.



Görüntüdə fraktal rəsmlərin axtarılıb tapılması artıq çox hesablamaların aparılmasını tələb edir. Bununla belə, fraktal sıxılma itkilərlə müşayiət olunur, çünki fraktalların müqayisəsi prosesi dəqiq uyğunluğun tapılmasını nəzərdə tutmur. Bunun əvəzinə kodlaşdırma vaxtına, keyfiyyətcə orijinala uyğunluğu və alınmış verilənlərin ölçüsünə görə, mümkün variantlardan ən yaxşısı axtarılır. Bir qayda olaraq, nəticənin "gözəyarı", praktik olaraq orijinaldan fərqlənməməsi tələb olunur.

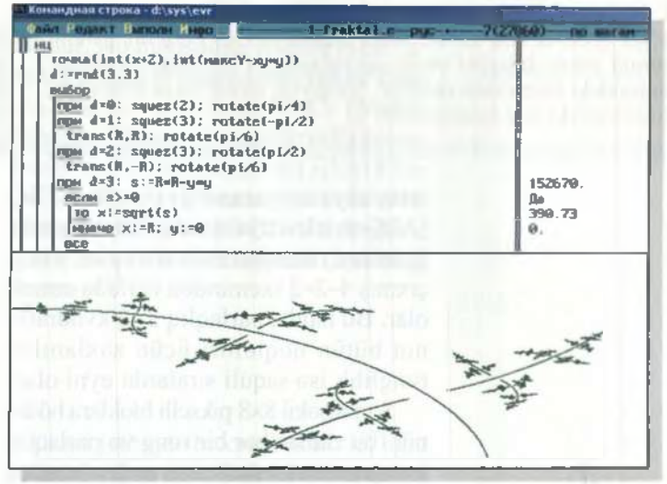
Ancaq nəticələr böyük səmərə verir. Fraktal sıxılmış görüntünün qablaşdırmadan açılmasına sərf olunan vaxt, qablaşdırmaq üçün sərf olunan vaxtdan qat-qat azdır. Bəzən fraktal təsvir yaddaşda öz orijinalından 100 və daha çox dəfə az yer tutur. Bundan başqa fraktalları detalların itkisi olmadan və əlavə detallar (artefaktlar) olmadan çox asan miqyaslaşdırmaq olur.

Lakin həmişə olduğu kimi, əksər fraktal sıxılma alqoritmləri patentləşdirilib və onlardan proqramlarda sərbəst istifadəyə icazə verilmir.

JPEG SIXILMASI

JPEG abbreviaturası Joint Photographic Experts Group (Fotoqrafiya üzrə birləşmiş ekspertlər qrupu) standartlaşma komitəsinin adından götürülmüşdür. LZW sıxma alqoritmi görüntülərin sıxılması üçün o qədər də yaramır, çünki rəqəmli fotoqrafiyalarda, daranmış şəkillərdə təhriflər olur və rəngin dərinliyi (1 pikseldəki bitlərin sayı) artdıqca, onların da sayı artır.

Faktiki olaraq JPEG təkcə sıxma alqoritmi deyil, o, bir sıra sıxma üsullarının toplusudur. Fraktal sıxılmada olduğu kimi, burada da verilənlərin gözlə görünməsi çətin olan itkisi baş verir. Rəngin kiçik dəyişmələri gözlə o qədər də hiss olunmur, amma parlaqlığın cüzi



KuMirdə proqram fraktal çəkir.

dəyişməsi, əksinə asanlıqla hiss olunur. Buna əsaslanaraq, JPEG sxemi yarımtonunu saxlayır, ancaq rənglə daha "sərbəst" hərəkət edir. Sıxılma o zaman effektiv olur ki, qonşu piksellər arasındakı fərq çox cuzi olur. JPEG, dərinliyi bir rəng kanalına (RGB) 5 bitdən çox olan təsvirlərlə daha yaxşı işləyir. Bu zaman standartın özü 8 bit dərinliyi üçün təsvir olunur.

Sıxılma prosesi bir neçə mərhələdə baş verir. Birinci mərhələdə rəngli görüntünü standartdan, məsələn, RGB-dən (qırmızı, yaşıl, göy) YUV-ə (parlaqlıq/rənglilik) və ya YC_bC_r-ə (parlaqlıq və iki fərqli rəng komponenti), çevirmək gərəkdir.

Sonrakı emal zamanı parlaqlığı mümkün qədər səliqəli saxlamaq lazımdır, rənglilik isə az önəmlidir. İnsan gözləri xüsusi həssaslığa malikdir, onlar rəng kanalları üçün piksellərin sayını azalda bilər. Bu hadisəyə subdiskreditləşdirmə deyilir. Hər iki rəng kanalı üçün subdiskreditləşdirmə zamanı, vaxtilə rəngli

Əgər siz portağal şirəsi olan paketə portağallar yığınının ZIP-faylı kimi baxırsınızsa, onda siz informasiyanın sıxılma nəzəriyyəsinə yiyələnmisiniz.

Hələ 1860-cı ildə alman alimi Qustav Fexner belə bir qanun (onun adını daşıyan) verib: qıcıqlandırıcı ilə insan hissiyatı arasındakı əlaqə xətti deyildir. "Hissiyyat stimul loqarifmlərinin nisbətləriylə düz mütənasibdir".

televiziyanın yarandığı illərdə NTSC (ABŞ-ın televiziya standartları üzrə milli komitəsi) standartında sınaqdan yaxşı çıxmış 4-2-2 sxemindən istifadə etmək olar. Bu halda, parlaqlıq 2×2 kvadratının bütün nöqtələri üçün saxlanılır, rənglilik isə şaquli sıralarda eyni olur.

Sonra şəkil 8×8 pikselli bloklara bölünür (bu zaman hər bir rəng və parlaqlıq komponenti bir-birindən asılı olmadan emal olunur). JPEG, Furye çevirməsinin bir növü olan diskret kosinus – çevirməsinə (DCT) əsaslanır. O, görüntünün fəza təqdimatından onun, musiqi mərkəzləri olanlara yaxşı tanış olan, spektral təqdimatına keçməyə imkan verir (spektroanalizatorun kiçik ekranında balaca sütunlar musiqi ritminə uyğun qalxıb-düşür).

Əks kosinus çevirməsi uyğun olaraq aşağıdakı əks əməli yerinə yetirir: tezlik təsvirindən fəza təsviri alınır.

DCT-çevirməsi 8×8 blokuna tətbiq olunur. DCT-hesablamalar olduqca mürəkkəbdir və bu JPEG sıxılmasında ən çox zəhmət tələb edən bir mərhələdir. Bu mərhələdən sonra təsvir, yüksək tezlikli və aşağı tezlikli informasiyalara bölünmüş olur. Yuvarlaqlaşdırma xətaları istisna olmaqla bu mərhələdə itkilər nəzərdə tutulmur.

Matrisin elementləri (yeni blokun) aşağıdakı düsturla hesablanır (u və v-nin 0-dan 7-yə qədər qiymətləri üçün):

$$F(u,v)=\frac{1}{4}C_uC_v\sum_{i=0}^{7}\sum_{j=0}^{7}f(i,j)\times$$

$$\times\cos\frac{(2i+1)u\pi}{16}\cos\frac{(2j+1)v\pi}{16},$$

$$C_u,C_v=\begin{cases}1, & u,\ v\neq 0 \text{ olduqda}\\ \frac{1}{\sqrt{2}} & u,\ v=0 \text{ olduqda}\end{cases}$$

8×8 fraqmentinin əmsalları bir-birlərinə yaxın olduqda, DCT-çevirməsi nəticəsində alınan matrisin elementlərinin mütləq qiymətləri, sol yuxarı küncdən başlayaraq sağ aşağı künc kimi azalma tendensiyasına malik olacaq.

Sonrakı mərhələdə, böyük həcmdə informasiyanı atmaqdan qabaq, alınmış matrisin bütün əmsallarını nəticəni tama qədər yuvarlaqlaşdırmaqla *kvantlaşdırılma* əmsallarına bölmək lazımdır. Kvantlaşdırma matrisi, sağ aşağı küncdən başlayaraq sol yuxarı künc istiqamətində artan müsbət tam əmsallardan ibarət bir matrisdir.

Kvantlaşdırma əmsalları nə qədər böyük olursa, bir o qədər çox verilənlər itirilir. Matrisin 64 elementindən hər birinin öz kvantlaşdırma əmsalı var. Parlaqlıq verilənlərinin kvantlaşdırma əmsalları, rəng verilənlərinin kvantlaşdırma əmsallarından kiçik olur, yəni parlaqlıq verilənləri daha dəqiqliklə saxlanılır. JPEG sıxılmanın "keyfiyyəti" birbaşa kvantlaşdırma matrislərindən asılıdır.

Beləliklə, kvantlaşdırma nəticəsində çoxlu sıfırları və sol yuxarı küncündə bir neçə sıfırdan fərqli elementləri olan matris alınır. İkinci sıxılma üsulu adlanan son mərhələdə, matrisin sol yuxarı küncündən başlayaraq diaqonal zolaqlarla hərəkət edərək, yaxşı tanış olduğumuz Haffman metodu vasitəsilə sıfıra bərabər əmsallar yox edilir:



Onda ardıcıllığın əvvəlində, əsasən sıfır olmayan əmsallar, sonunda isə ancaq sıfır olan əmsallar olacaq. Belə ardıcıllıqları Haffman üsulu yaxşı sıxa bilir.

Görüntünü bərpa etdikdə isə bütün proseslər əks istiqamətdə gedir. Əvvəlcə, Haffman metodu vasitəsilə dekodlaşma

həyata keçirilir, sonra isə matrisin elementləri kvantlaşdırma matrisinin əmsallarına vurulur. Daha sonra əks kosinus – çevirməsi tətbiq olunur və RGB sxeminə keçid hesablamaları aparılır. Görüntünün bərpası zamanı JPEG-dekodlaşmadakı hesablamaların həcmi, JPEG-kodlaşdırmadakı hesablamaların həcmilə müqayisə oluna bilər, ona görə də bu əməliyyatların sürətləndirilməsi üçün aparat tezləşdiricilərindən istifadə olunur. Bu məqsədlə, İntel firması bazara Pentium prosessorunu çıxaran kimi, elə həmin vaxta onun MMX (Multi-Media eXtension) – multimedia genişlənməsini də xüsusi olaraq işləyib hazırladı. Bu genişlənmə 8x8 matrisləri üzərində düz və əks DCT-ni tezləşdirən çevirmələri bəzi əməliyyatları aparmağı "bacarır".

SƏSİN SIXILMASI

MIDI şəklində kodlaşdırılması mümkün olan rəqəmli səslər (musiqi olmayan) şəkillərin sıxılmasında olduğu kimi, sıxmaq üçün əlverişli deyil. Səs siqnalları nadir hallarda artıqlığa, yəni təkrar olunan sahələrə malik olur (əsasən, küylərin olmasına görə). Bu səbəbdən də onlar itkiyə yol verməyən LZW və ya Haffman üsuluna analoji sıxılma alqoritmləri vasitəsilə pis sıxılır.

Səsləri, *kompandlaşdırma* (*ing.* "compound" – "birləşmə", "düzəltmə") kimi qəribə adı olan metod vasitəsilə iki dəfə sıxmaq olar.

Bu metod psixoloqlar tərəfindən kəşf olunmuş bir qanuna əsaslanır: əgər qıcıqlandırıcının intensivliyi həndəsi silsiləylə dəyişirsə, onda insanın qəbul etmə intensivliyi ədədi silsiləylə dəyişir.

Səsə tətbiqdə bu onu bildirir ki, əgər səsin ucalığını 2, 4, 8 və s. dəfə artırsaq, insan qulağı onu intensivliyin xətti artımı kimi qəbul edəcək. Yəni, insan səsin ucalığının 1-dən 2-yə və yaxud 100-dən 200-ə artımını eyni cür qəbul edəcək, amma səsin ucalığının 100-dən 101-ə dəyişməsini isə insan praktik olaraq hiss etməyəcək.

Ona görə də kompandlaşdırma zamanı səsin amplitud qiymətləri həmin qiymətlərin loqarifmlərilə əvəz olunur.

Səsin 8-bitlik kodlaşdırılması zamanı $|a|<2^7$ olduğundan, $\log_2|a|<7$ olur. Deməli, səs 8 əvəzinə 4 bitlə kodlaşdırıla bilər: 3 bit səsin kodlaşdırılmasına və 1 bit də amplitudun işarəsinə sərf olunur.

Beləliklə, kompandlaşdırma zamanı, keyfiyyətdə bir az itirmək şərtilə 8-bitlik səsi iki dəfə sıxmaq olur.



Rəqəmli səsləndirmənin etalonu kimi qəbul olunmuş CD-də olan səslər impuls-kod modulyasiyası (PCM – Pulse Code Modulation) üsulu ilə kodlaşdırılmışdır. Səsləndirmənin vaxtı və amplitud diapazonu, uyğun olaraq, bərabər aralıqlara – rəqəmləşdirmə parametrləri olan – diskretləşdirmə tezliyinə və diskretləşdirmə səviyyəsinə (kanala düşən bitlərin sayı) bölünür. Nəticədə CD-də (təqribən 44 kHs tezliklə və kanala 16 bit olmaqla) 1 saniyə stereosəsin yazılması 1,4 Mbit ikilik verilənlər, yaxud axınla səsləndirmə (yazılış) adlanan 1400 kbit/s tələb edirdi.

ADAPTİV FƏRQ SIXLAŞDIRMASI

Xüsusi növ səsin – insan nitqinin sıxılma üsulları, insanın necə eşitməsinə yox, onun necə danışmasına əsaslanır.

Audioverilənlərin həcminin azaldılması üsullarından biri – verilənlərin özlərinin deyil, iki qonşu qiymətlər arasında fərq böyük olmadıqda, həmin fərqin saxlanılmasıdır. Bir qayda olaraq, bu, insan danışığı üçün doğrudur. Bu halda fərq, ilkin verilənlərə nisbətən daha az bitlə kodlaşdırılır. Bu üsula DPCM (Delta Pulse Code Modulation) üsulu deyilir.

Başqa, daha qənaətli ADPCM (adaptive DPCM) üsulunda yaxud adaptiv fərq sıxlaşdırmasında fərqlərin qiymətləri daha kobud kvantlaşdırmaya (mümkün qiymətlərin sayı) məruz qalır ki, bu da tərtibi 16 bitdən 4 bitə endirməyə imkan verir, həm də kvantlaşdırma addımı siqnalın cari dəyişmə qiymətindən asılı olaraq seçilir. Fərq dalbadal bir neçə dəfə 16-ya (4 bit fərqin kvantlaşdırmasına ayrılıb) çatırsa o artır, fərq bir neçə dəfə 4-dən az olursa, azalır. Üsulun adında olan "adaptiv" sözü məhz bunu göstərir. Bununla 16 bitlik verilənlərin: ADPCM üsulu ilə sıxılma əmsalının 4:1 olmasını demək qəbul olunub.

MPEG (Monitor Pictures Experts Group) standartı üzrə sıxılma alqoritmlərinin əsasında keyfiyyətcə başqa prinsiplər durur.

MP3

Təbii olaraq belə bir sual yaranır: qrafikada da JPEG-çevirməsinə analoji olan bir üsuldan istifadə etmək olmazmı? Axı şəkilləri sıxarkən, çox zaman gözlə seçilməyən təhriflərin olması təsvirin ölçülərini güclü surətdə azaltmağa imkan verir. Tutaq ki, kompressiya və dekompressiya nəticəsində alınan səs orijinalla eyni deyil, ancaq qarşıya ilkin səs reqslərinin formasını olduğu kimi bərpa etmək məqsədini qoymaq da lazım deyil. Əsas məsələ, minimum eşidilən (və ya heç eşidilməyən) təhriflər olmaqla səs siqnalının maksimum sıxılmasıdır. İnsanın eşitmə qabiliyyəti bir sıra xüsusiyyətlərə malikdir ki, bu da, səsin keyfiyyətində subyektiv – minimal itkilərə yol verməklə səs verilənlərinin kompressiyası üçün kifayət qədər effektiv alqoritmlərdən istifadə etməyə imkan verir. Stereosəsin atası sayılan görkəmli amerikan fiziki Harvi Fletçer 1940-cı ildə, insanın eşitmə qabiliyyətini tədqiq etmək



məqsədiylə çoxlu sayda insan üzərində sınaqlar apardı. O, eşitmənin ən son həddinin səs siqnalının tezliyindən necə asılı olduğunu, yəni hansı amplitud qiymətində müəyyən tezlikli səsin insanın eşitmədiyini təhlil etdi. Təcrübələr zamanı qurulan əyrilər nəticəsində müəyyən olundu ki, insanın eşitmə diapazonunun maksimal qiymətləri gözlənildiyi kimi, təqribən 20 Hs - 20 kHs intervalında, minimum qiyməti isə 5 kHs yaxındır. Ancaq onun başlıca diqqət yetirdiyi o idi ki, insan qulağı yeni səslərə uyğunlaşma qabiliyyətinə malikdir və bu uyğunlaşma özünü, eşitmə həddinin artmasında göstərirdi. Başqa sözlə desək, səslərin bəziləri o birilərini eşitməz edir ki, buna da bir səsin başqa səslə maskalanması deyilir.

Maskalanma həm tezlik (və ya tezlik-dinamik) xüsusiyyətinə (uca səslər tezliyi onunkuna yaxın olan sakit səsləri, aşağı tezlikli səslər yüksək tezlikli səsləri maskalayır), həm də zaman xüsusiyyətinə (maskalanmış səs kəsildikdən sonra, müəyyən müddət effekt davam edir) malikdir.

Eşitmənin bu sonuncu xüsusiyyəti ona imkan verir ki, kompessiya zamanı yüksək səs siqnalından sonra, müəyyən kiçik vaxt ərzində səs verilməsin, yəni heç bir səs saxlanılmasın. Bu effekti, kobud da olsa, belə bir misalla nümayiş etdirmək olar: topdan atəş açıldıqda insan ümumiyyətlə müəyyən müddət heç nə eşitmir. Yəni 0,1 san müddətli gur çıqqıltı, özündən sonra gələn 0,5 san müddətli sakit səsləri maskalaya bilər ki, onları da saxlamağa gərək yoxdur. Belə səsin dəqiq səsləndirilməsi üçün ilkin siqnalın kiçik bir hissəsini – çıqqıltını səsləndirmək yetərlidir, çünki onsuz da yerdə qalan hissə eşidilmir. Müəyyən edilib ki, bu misalda kompressiya əmsalı

$$6:1=\frac{0,5+0,1}{0,1},$$

Tezliyi 1 kHs, ucalığı 60 dB olan maskalayıcı tonla, tezliyi 1,1 kHs, ucalığı 40 dB olan test tonunu eyni zamanda səsləndirsək, test tonu eşidilməyəcək. Maskalayıcı tonu tamam aradan götürüb, test tonunu qısa müddət ərzində o, eşidilənə qədər səsləndirməkdə davam etmək olar. Təcrübə göstərəcək ki, bu zaman təqribən 5 ms təşkil edir.

Təbiidir ki, səs spektrinin mütləq eşitmə həddindən aşağıda yerləşən hissələrini (insan qulağının qəbul edə biləcəyi səs təzyiqinin minimal qiymətini) nə kodlaşdırmağa, nə də ötürməyə gərək yoxdur.

təsvir olunan sıxma proseduruna isə, adətən, *zaman sahəsinə görə maskalama* deyirlər.

Tezliklər sahəsində maskalama zamanı sinusoidal siqnal, tezliyə görə yaxın olan sakit siqnalları, eləcə də amplitudu dəfələrlə kiçik olan sinusoidal siqnalları maskalayır. Bu da siqnalın ilkin spektrində, praktiki olaraq eşidilməyən hissələri ayırmağa imkan verir, insanın eşitmə qabiliyyətinin xüsusiyyətlərinə əsaslanaraq siqnal spektrini müxtəlif enli zolaqlara bölmək daha yaxşı olar: qulaq ilbizi, hər biri müəyyən tezlik zolağına cavabdeh olan sahələrdən ibarətdir (spektrin aşağısındakı 100 Hs enindən, yuxarı hissədəki 3500 Hs eninə qədər). Adətən,





1987-ci ildə Erlangen Universitetinin (Almaniya) professoru Diter Seytzerin bilavasitə iştirakı ilə rəqəmli yayım üçün (Digital Audio Broadcasting, DAB) audio-kodlaşdırma alqoritmi işlənib hazırlanarkən, adətən MP3 adlandırılan məşhur ISO-MPEG Audio layer 3 standartı yarandı.

27 böhran zolağını (*ing.* "critikal band") seçdirirlər: 0-cı 50-95 Hs, 1-ci 95-140 Hs..., 26-cı 20245 Hs və yuxarı.

Sıxma alqoritmini yerinə yetirmək üçün, ilkin siqnal tezlik təhlilinə məruz qalan kadrlara bölünür. Sıxma alqoritmi təqribən belədir:

• Siqnal xüsusi alqoritmlər (məsələn, cəld Furye çevirməsi və ya başqa analoji alqoritmlər) vasitəsilə 32 bərabər spektr zolağına bölünür. Bu zaman alınmış bir zolağa bir neçə böhran zolağı düşə bilər.

• Psixoakustik modeldən istifadə edərək (bura tezlik maskalanması da daxildir) hər bir zolağın qonşu zolaqlarla maskalanma səviyyəsi müəyyən olunur.

• Zolaqda səviyyə hesablanmış həddi aşmadıqda onun qiyməti sıfra bərabər hesab edilir və saxlanılmır. Əksinə, maskalanmamış səviyyə çıxış verilənlərinə yazılır.

Məsələn:

Zolaq №	...	7	8	9	...
Səviyyə (dB)	...	10	60	35	...

Cədvəldən göründüyü kimi 8-ci zolağın səviyyəsi 60 dB-ə bərabərdir. Psixoakustik model əsasında belə nəticə çıxarılır ki, bu zolaq, səviyyəsi 12 dB olan 7-ci zolağı və səviyyəsi 15 dB olan 9-cu zolağı maskalaya bilər. Cədvələ baxsaq görərik ki, 7-ci zolağı saxlamaq lazım deyil, çünki 10 < 12, 9-cu zolağı isə əksinə saxlamaq lazımdır, çünki 35 > 15.

Sonra hər sıfır olmayan səviyyəni təxmini göstərmək üçün yetərli olan sayda bit ayrılır. Belə ki, spektrin insan qulağının aşağı eşitmə həddinə malik olduğu hissəsində informasiya 16 bitlə, qıraqlarda – qulağın təhriflərə daha az həssas olan hissəsində isə 6 və ya daha az bitlə kodlaşdırılır. Alınmış bitlər selinə, məsələn, Haffman sıxma alqoritmini tətbiq etmək olar.



▸ Jan Batist Jozef Furye.

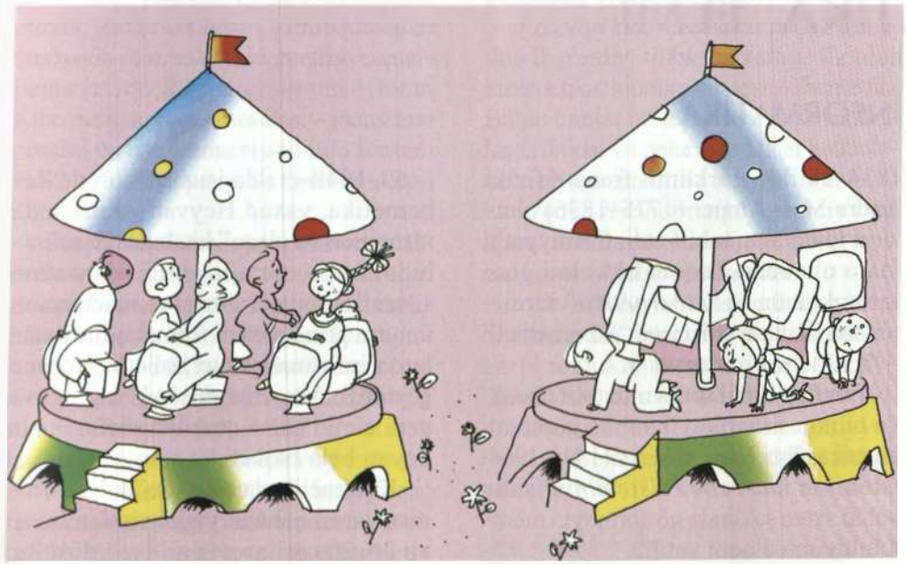
Səsin MPEG-sıxılma alqoritminin üç versiyasını fərqləndirirlər. Hər versiyada verilənlər kadrlara bölünür, yəni ayrıca kadr, hər birində 12 qiyməti olan 32 zolaqdan ibarətdir: 32 · 12 = 384.

MPEG layer 1 ("təbəqə 1") versiyasında tezlik filtrində bir kadrdan və diskret kosinus çevirməsinə (DCT) əsaslanan alqoritmlərdən istifadə olunur. Psixoakustik model ancaq tezlik maskalanmasını hərəkətə gətirir. Alqoritm 1:4 nisbətində, 384 kbit/san selilə yığcamlaşdırmağa imkan verir.

MPEG layer 2 tezlik filtrində üç kadrdan (əvvəlki, cari və sonrakı) istifadə edir. Onların ümumi həcmi, hər birində 12 qiyməti olan 3 kadrlı 32 zolaqdan ibarətdir (32 · 12 · 3 = 1152). Bu model zaman maskalanmasından da istifadə edir. 1:6 nisbətindən 1:8 nisbətinə kimi yığcamlaşdıra bilir.

MPEG layer 3 (və ya MP3) müxtəlif ölçülərə malik zolaqları olan tezlik filtrindən istifadə edir; psixoakustik modelə stereosiqnalın zaman maskalanması daxildir. Keyfiyyətdə kiçik itkilərə yol verməsinə baxmayaraq yüksək sıxılma (1:10–1:12) səviyyəsinə malikdir.

MP3 səsin sıxılmasında ən uğurlu standartlardan sayıla bilər. O, bazarda təkcə Sony firmasının mini – disk (MD) audioməhsullarını sıxışdırmadı, həm də hərəkət edən (mexaniki) hissələri olmayan pleyerlər üçün əsas yazı formatına çevrildi. Bir CD diskinə MP3 formatında 6 saatdan 12 saata qədər musiqi yazmaq olar.



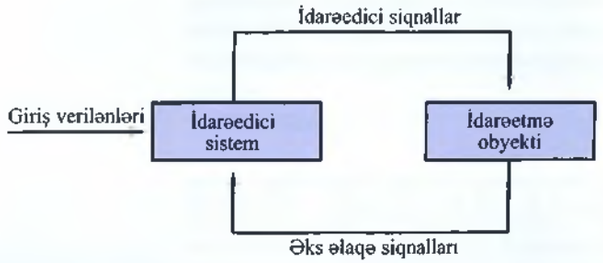


İNFORMATİKA ELMİ

İNFORMATİKA

1834-cü ildə görkəmli fransız fiziki Andre Mari Amper (1775-1836) elmlərin təsnifatını təklif etdi. Cəmiyyətin idarə olunması haqqındakı elmi göstərmək üçün o, "*kibernetika*" terminindən istifadə etdi (*yun.* "kibernetika" – "idarəetmə məharəti").

Bundan 100 il sonra müxtəlif texniki və bioloji sistemləri öyrənən görkəmli amerika alimi Norbert Viner (1894-1964) istənilən *idarəetmə sisteminin* işinin vahid sxem şəklində göstərməyin mümkünlüyünə diqqət yetirdi.



O, 1948-ci ildə işıq üzü görən "Kibernetika, yaxud Heyvan və maşında idarəetmə və əlaqə" kitabında öz müşahidələrini ümumiləşdirdi, idarəetmə sistemlərinin qurulması və işləməsinin ümumi prinsiplərini formalaşdırdı, onlarda informasiyanın həlledici rolunu göstərdi. Amperin düşünüb tapdığı və yeni elmin adına çevrilən sözün ikinci həyatı belə başladı.

Kibernetikada *əlaqə* dedikdə, informasiyanın qəbulu, yadda saxlanılması və ötürülməsi prosesləri başa düşülür. *İdarəetmə – idarəedici sistemin idarəolunan obyektlərə* təsir göstərərək qarşıya qoyulmuş məqsədlərə çatmaq üçün onun işini təmin etməsi prosesidir. O, idarəedici sistemin hazırladığı *idarəedici siqnalların* köməyilə həyata keçirilir. Bu siqnallar idarəolunan obyektin tələb olunan hərəkəti haqqında informasiya daşıyır. O, *giriş verilənlərinin* (xarici mühitin vəziyyəti haqqında informasiyanın) analizi və *əks-əlaqə siqnalları* (idarəolunan obyektin vəziyyəti haqqında informasiya daşıyan) əsasında yaradılır. Əks-əlaqə kanalları ilə daxil olmuş informasiyaların yenidən işlənməsinə *nəzarət* deyilir.

Verdiyi tərifin natamamlığını hiss edən Viner sanki Amperin ardınca gedərək 1954-cü ildə "Kibernetika və cəmiyyət" adlı kitabını nəşr etdirdi ki, burada yeni elmin təsir dairəsini sosial sistemlərə qədər genişləndirdi. İngilis alimi Uilyam Ross Eşbi (1903-1972) "Kibernetikaya giriş" (1956) kitabında Vinerin ideyalarını sistemləşdirərək kibernetikanın təbliğatına böyük töhfələr verdi.

Kibernetika maşın və canlı orqanizmlərin müəyyən informasiyaları qəbul etmək, yadda saxlamaq, rabitə kanalları vasitəsilə ötürmək və idarəetmə siqnallarına çevirmək qabiliyyətlərini öyrənir. Kibernetikanın əsas məsələsi – idarəetmə prosesi və idarəetmə sisteminin ümumi xassələrini seçdirmək və öyrənməkdən ibarətdir.

Kibernetika öz tədqiqat predmeti (idarəetmə sistemləri) olan müstəqil elmdir. Başqa elmlərdə olduğu kimi onda da mühüm bölməni – *nəzəri kibernetikanı* ayırmaq olar. O, müxtəlif təbiətli idarəetmə sistemlərinin öyrənilməsi üçün gərəkli olan aparat və tədqiqat üsulları işləyib hazırlayır. Nəzəri kibernetika riyaziyyatın bir neçə bölməsini özündə birləşdirir: riyazi məntiq, alqoritmlər nəzəriyyəsi, informasiya nəzəriyyəsi, kodlaşdırma nəzəriyyəsi. Bəzi yeni elmi istiqamətlər kibernetikanın öz çərçivəsində yaranmışdır: avtomatlar nəzəriyyəsi, formal dillər nəzəriyyəsi, formal qrammatikalar nəzəriyyəsi və qeyri-səlis riyaziyyat.

Beləliklə, kibernetika əhəmiyyətli dərəcədə riyazi əsasda qurulur, ancaq onun özünün də xüsusi tədqiqat üsulu var. Bu, obyektləri deyil, onların təsvirini (modelini) öyrənməyə imkan verən *kompüter modelləşdirməsidir*. Burada modelləşdirmə fizika, kimya, başqa texniki və təbiət elmlərindəki eksperimentin rolunu oynayır. Əsas vacib amil ondan ibarətdir ki, tənlik, yaxud düsturla ifadə olunmayan obyektləri də kompüterdə modelləşdirmək mümkündür. Bu isə kibernetikanı (riyaziyyat kimi) xüsusi vəziyyətə qoyur, çünki belə üsul ən müxtəlif elm sahələrində tətbiq edilə bilir. Uyğun bilik sahələrinin hər birinin öz adı var:



Andre Mari Amper.

• *Texniki kibernetika*, nəzəri kibernetikanın nəticə və yekunlarından istifadə edərək – sadə avtomatik tənzimləmə sistemlərindən superkompüterlər əsasında qurulmuş mürəkkəb avtomatlaşdırma idarəetmə sistemlərinədək müxtəlif texniki idarəetmə sistemlərini işləyib hazırlayır və tədqiq edir.

• *Bioloji kibernetika* bir neçə bölmədən ibarətdir: tibbi, psixoloji, fizioloji kibernetika, bionika və neyrokibernetika. Bütün bunlar bioloji sistemlərin modelləşdirilməsi və onların tədqiqat nəticələrinin riyazi emalı ilə məşğul olur.

• *İqtisadi kibernetika* iqtisadiyyatın idarə olunma proseslərini öyrənir, iqtisadi sistemləri modelləşdirir.

• *Hərbi kibernetikada* ordunun idarə olunmasının ümumi məsələlərinə və hərbi texnikanın tətbiqinin effektivliyinin artırılması üsullarına baxılır.

• *Sosial kibernetika* insan cəmiyyətində gedən proseslərin modelini tədqiq edir.



U.R.Eşbinin "Kibernetikaya giriş" kitabı.

İNFORMATİKA SSRİ-də

İkinci Dünya müharibəsindən sonrakı illərdə kibernetika SSRİ-də təqib edildiyindən hesablama texnikasının istehsalına görə SSRİ inkişaf etmiş kapitalist ölkələrindən çox-çox geridə qalmışdı.

Bu və ya digər dərəcədə hərbi sahədə istifadə oluna bilən bütün fundamental tədqiqatların nəticələri avtomatik olaraq "məxfidir" qrifi alırdı. Bu hal görkəmli amerikan riyaziyyatçısı Norbert Vinerin "Kibernetika, yaxud heyvan və maşında idarəetmə və əlaqə" (1948 il) kitabı ilə də baş vermişdir: SSRİ-yə bu kitab gələn kimi, dərhal onu məxfiləşdirdilər.

Müəllifin irəli sürdüyü ideyalar sovet cəmiyyətinin rəsmi doktrinası ilə ziddiyyət təşkil edirdi. Vinerin canlı həyat ilə (insan və heyvanlar) və maşınlar aləmi (ictimai həyatda və iqtisadiyyatda modelləşdirmə) arasında çəkdiyi paralellər "ali formanın" "sadə formaya" çevrilə bilməməsi təlqin olunan sovet vətəndaşları üçün hökumət əleyhinə suiqəsd idi. Məhz buna görə də bu kitab xüsusi mühafizəyə düşdü.

Təsəvvür etmək çətindir ki, öz karyeraları və isti yerləri üçün qorxaraq onları təqib edən insanlar olmasaydı, nə qədər gənc və məşhur insanlar əsl alim ola bilər və öz doğma ölkəsinə xeyir gətirə bilərdilər. Rütbə və vəzifə talant və ya elmi nailiyyətlərə görə deyil, kommunist partiyasına və fəhlə-kəndli mənşəyinə mənsubiyyətinə görə verilirdi.

Təsədüfü deyil ki, 50-ci illərin fəlsəfə lüğətində kibernetika İkinci Dünya müharibəsindən sonra ABŞ-da əmələ gəlmiş və başqa kapitalist ölkələr arasında geniş yayılmış yalançı, mürtəce elm kimi qələmə verilmişdir. 1953-cü ildə "Fəlsəfənin sualları" jurnalında "sifariş"lə "Kibernetika kimə xidmət edir" adlı məqalə çıxır. Burada cəld işləyən "arifmometr"lərin inkişafı və istehsalı tövsiyə edilirdi. Bununla yanaşı qeyd edilirdi ki, modelləşdirilmə və simvolların işlənməsi üçün maşınların istifadəsi marksizm-leninizm nəzəriyyəsinə ziddir:

"Ən müasir konstruksiyalı hesablama maşınlarının işləmə prinsiplərini onların keyfiyyət özəlliklərini nəzərə almadan ən müxtəlif təbiət və cəmiyyət hadisələrinə tətbiqinə çalışan kibernetika nəzəriyyəsi idealizmə çevrilən mexanisizmdir". Bu, idrakın xüsusiyyətlərindən birinin həddən artıq və birtərəfli şişirdilməsi nəticəsində bilik ağacında yaranan boşluq idi. Məqalənin müəllifi "materialist" ləqəbilə imza qoymuşdu. Birdən MK-nın göstərişi dəyişilər və kibernetikanın inkişafına yaşıl işıq verilər? Məhz buna görə də məqalədə kompüterqayırmanın inkişafı belə ehtiyatlı verilmişdir. Təəssüf ki, belə davranış norma sayılırdı.



A.İ.Kitov.

Lakin alimlər arasında həmişə elə insanlar tapılırdı ki, ideya, vətəndaşlıq mövqeyi və elm naminə sona qədər getməyə hazır idilər. İndi kibernetikanın təşəkkülündə əməyi olan bütün insanları sadalamağın mənası yoxdur. "Fəlsəfə məsələləri" (1955) jurnalında *İnformasiya* (*lat.* informatio – "izah etmə", "şərh etmə"), idarəetmə kimi kibernetikanın əsas anlayışı olub materiya və enerjilə yanaşı kainatın fundamental təməllərindəndir. EHM-lərin (informasiyanın toplanması və emalı üçün qurğuların) meydana gəlməsi kibernetikanın yaradılması üçün bir təkan olmasına baxmayaraq, tezliklə məlum oldu ki, bu elm hesablama texnikasının inkişafı ilə bilavasitə əlaqədar bir çox nəzəri və praktiki suallara cavab vermir. Buna görə də XX əsrin 40-50-ci illərində ingilisdilli ölkələrdə "Computer Science" (kompüter elmi) adını almış yeni bir elm formalaşmağa başladı. Fransada bu elmi, "Information" və "Automatique" sözlərini birləşdirib "Informatique" adlandırdılar.

SSRİ-də yeni elmin taleyi başlanğıcda o qədər də asan olmadı. Axı kibernetika inkişafının ilk mərhələlərində xeyli dərəcədə sosial və fəlsəfi bir elm idi. Bu elmin ictimai həyatın bəzi hadisələrini izah etmək cəhdi, sovet kommunist ideoloqlarının kəskin marksizm monopoliyasına qarşı çıxması kimi qəbul olunurdu. Buna görə də, məsələn, "Qısa fəlsəfə lüğəti"ndə (1954) deyilirdi: "Kibernetika – İkinci Dünya müharibəsindən sonra ABŞ-da meydana gəlmiş mürtəce yalançı elmdir... təkcə imperializmin ideoloji silahı deyil, həm də onun aqressiv hərbi planlarının həyata keçirilməsi üçün bir vasitədir". Buna baxmayaraq, tezliklə kibernetikaya münasibət yaxşıya doğru dəyişdi. Bu öz-özünə baş vermirdi – kibernetikanın təbliği və inkişafında bir çox məşhur sovet alimləri böyük rol oynadılar: A.A.Lyapunov (1911-1973), A.İ.Berq (1893-1979), İ.A.Poletayev (1915-1983) və b. Bu elmin necə çətinliklə özünə yol açmasını təsdiqləyən belə fakt var. 1952-ci ildə A.A.Lyapunovun həmfikiri A.İ.Kitov Vinerin kitabından yaranmış təəssüratlarla kibernetikanın rəsmi şərhinə etiraz edərək, məqalə yazdı. Bu məqaləyə 3 il müxtəlif instansiyada

baxıldıqdan sonra dərc olunmasına icazə verildi. "Kiçik Sovet Ensiklopediyası"nda (1959-cu ildə) kibernetikanın "burjuaziya pozğunluğuna" xidmət etməsi haqqında olan ovsunların hələ də təkrarlanmasına baxmayaraq, ona artıq "xalqın yaradıcı qüvvələrinin azadlığı"na xidmət edən elm – "sovet kibernetikası"na qarşı qoyulurdu.

Kibernetikaya münasibət, onun bəzi nəticələrinin praktik problemlərin həllində tətbiqindən sonra kəskin dəyişdi. SSRİ-də kibernetika bir elm kimi rəsmən qəbul edildikdən sonra EHM-in yaradılması və istifadəsilə bağlı bütün məsələlər də daxil olmaqla çox geniş anlamlarda yozmağa başladılar. Lakin tədricən bu məsələlərin ayrılması prosesi baş verdi. Nəticədə kibernetika informasiya anlayışı ilə bağlı məsələləri informatikaya verdi, idarəetmə problemlərinin həllini isə özündə saxladı.

Buna baxmayaraq, kibernetikanın müxtəlif sahələrində proqram sistemləri, yüksək məhsuldarlıqlı kompüterlər, şəbəkə texnologiyaları kimi praktiki nailiyyətlərindən geniş istifadə olunur.

İnformatikanın nəzəri əsasını riyaziyyat, ilk növbədə diskret riyaziyyat təşkil edir. Onun məzmunu əhəmiyyətli dərəcədə nəzəri kibernetikanın məzmunu ilə kəsişir. Kompüter vasitəsilə məsələlərin həlli üsullarının yaradılmasına yönəldilmiş hesablama riyaziyyatı və sistemli analiz də mühüm rol oynayır. İnformatika kibernetikadan hipotezlərin formalaşması, modellərin yaradılması, eksperimentlərin aparılması, verilənlərin toplanması və nəticələrin analizindən ibarət olan kompüter modelləşdirilməsinin eksperimental üsulunu da götürmüşdür.

Verilənlərin manipulyasiya proseslərinin öyrənilməsilə yanaşı, informatika onların reallaşdırılması və insan fəaliyyətinin müxtəlif sahələrində tətbiqilə də məşğul olur. Onun başlıca vəzifəsi bütün bu proseslərin *avtomat-*

çap olunmuş S.L.Sobolev, A.İ.Kitov və A.A.Lyapunovun elmi müdafiə etdikləri "Kibernetikanın əsas xüsusiyyətləri" məqaləsini qeyd etmək kifayətdir. Bu sübut edirdi ki, kibernetika partiya xəttinə zidd deyil, sadəcə olaraq elmi mühitdəki funksionerlərin mövqeyini sarsıdırdı.

Sovet alimləri (ilk növbədə hərbiçilər) gec də olsa, kompüterqayırma və kibernetikanın (informatikanın) inkişafının mühüm əhəmiyyət kəsb etdiyini sübut etdilər.

Sovet EHM-ləri qərb analoqlarından bir qədər gec buraxılsalar da, bir çox parametrlərinə görə onlardan geri qalmırdı. BESM, "Setun" və başqa hesablayıcı maşınlar elm və texnikada əbəs yerə aşkar iz qoymamışlar. SSRİ-də informatika da riyaziyyat kimi güclü bir məktəb oldu. Lakin SSRİ hesablama texnikasının istehsalı, xüsusən də, kütləvi kompüterqayırma sənayesi üzrə Qərbdən çox geri qalırdı.

Hesablama texnikasının ETİ-lərdə və KB-lərdə işlənib hazırlandığı vaxtlarda bir çox maraqlı, qeyri-adi elmi qərarlar qəbul edildi. Amma bunları həyata keçirən zaman ya istehsal çətinliklərilə üzləşirdilər (əsasən, keyfiyyət sahəsində), ya da müəssisə maraqları toqquşurdu. Bir sıra nazirliklər çox vaxt parametr və göstəricilərinə görə eyni olan hesablama texnikasını işləyib hazırlayır və istehsal edirdilər. Burada, kapitalizmdə olduğu kimi rəqabətin sərt qanunları işləyirdi. Mübarizənin nəticəsi kimi nəinki köhnə texnika istehsal edilir, hətta Qərb korkoranə təqlid edilirdi. Sənaye miqyasında xarici hesablama texnikasının surəti çıxarılmağa başladı. Əlbəttə ki, rəqabət aparan dövlətlərin təcrübəsini öyrənmək pis olmazdı. Lakin tək uğurlarını deyil, həm də, səhvlərini təkrarlayaraq, onların yedəyində getmək – bu şübhəsiz ki, səhv strategiyadır.

80-ci illərin axırları göstərdi ki, fərdi kompüterləri ildə onlarla yox, minlərlə, on minlərlə, yüz minlərlə istehsal etmək lazımdır. Milli kompüter sənayesi hələ buna hazır deyildi. İnanmaq istərdik ki, ötən əsrin dərsləri bizim üçün hədər getməyəcək. Ola bilər ki, bizim zavodlar qərb zavodlarından geri qalmasın, hətta onlardan üstün olsun. Çox təəssüf ki, elm dünyası dəyişiklərə az meyllidir, kibernetikada baş verənlər yenidən təkrarlana bilər.

BURJUA İDEOLOGİYASININ TƏNQİDİ

Kibernetika kimə xidmət edir

Kapitalizmin müdafiəsinə istiqamətlənən müasir burjuaziya sosioloji nəzəriyyələri arasında, texnikanı fetişləşdirərək onu ictimai inkişafın əsas aparıcı qüvvəsi olduğunu təsvir etməyə cəhd göstərən "nəzəriyyə" axırıncı yeri tutmur. Bəzi burjua alimləri kapitalizm cəmiyyətində mövcud olan ziddiyyətləri, bütün bəlaları, bədbəxtlikləri texnikada görməyə meyllidirlər. Onlar müharibələrin, işsizliyin və böhranın səbəblərini texnikanın "mistik gücündə" görürlər. Bu "sosioloqlar" texnikanı qırıb dağıtmağa, sosial toqquşmaların (mübahisələrin), maşınların olmadığı bir zamana və dinc (sakit) ibtidai həyat dövrünə qayıtmağa çağırırlar. Həmin düşərgənin digər "sosioloqları" texnikanı guya kapitalist qurluşunun bütün ziddiyyətlərini aradan götürə biləcək, müsbət qüvvə kimi fetişləşdirirlər. İmperializmin nökər alimlərinin bu uydurmalarının elmlə heç əlaqəsi yoxdur və müasir burjua elminin cırlaşmasına sübutdur.

"Fəlsəfə məsələləri" jurnalından məqalə. 1953-cü il.

SƏNƏDŞÜNASLIQ

Bir çox dərsliklərdə oxumaq olar ki, "informatika" sözü XX əsrin 70-ci illərində fransız dilindən götürülmüşdür. Lakin bu, heç də belə deyil.

Həmin vaxtadək bu adda elm sahəsi mövcud idi və dörddə bir əsrdən çox vaxt ərzində uğurla inkişaf edirdi! Bu sahə, elmi informasiyanın təbiət, cəmiyyət və düşüncənin obyektiv qanunauyğunluqlarını əks etdirən verilənlərin ümumi xassələrini və quruluşunu tədqiq edirdi. Bundan əlavə, o, konfransda şifahi ünsiyyətdən tutmuş, məqalə və monoqrafiya şəklində nəşrlərə qədər, elmi informasiyanın mübadiləsi prosesini də öyrənirdi. Praktikada informatika elmi informasiyanın toplanması, saxlanması, axtarılması və yayılması üsullarını hazırlayırdı. Onun bilavasitə sələfi 100 il öncə belçikalı alim Pol Otle tərəfindən yaradılmış *sənədşünaslıq* – sənədlərin yığılması, yenidən işlənməsi, saxlanılması, axtarılması və yayılması haqqında elm idi.

İnformatikanın kargüzarlıqla, kitabxana və arxiv işlərilə kitabşünaslıqla və s. əlaqəsini görmək çətin deyil. İnformatika (İnformation Science) 40-50-ci illərdə bunların əsasında meydana gəldi, lakin kompüterlərin ortaya çıxmasından sonra daha sürətlə inkişaf etdi.

Odur ki, fransız dilindən götürülmüş bu söz yeni deyildi, sadəcə, onun daha geniş mənası köhnəsini sıxışdırıb çıxardı.

Bu gün bütövlükdə informatikanı verilənlərin yaradılması, təsviri (yazılışı), saxlanılması, yenidən işlənilməsi və ötürülməsi proseslərini sistemli öyrənən bir elm kimi başa düşürlər.

laşdılırmasıdır (yəni insanın iştirakının istisna edilməsidir). Buna görə də informatikanın mühəndis hissəsini yaranan problemlərin həlli üçün nəzərdə tutulmuş qurğu və sistemlərin yaradılması (layihələndirilməsi) təşkil edir. Qurğu və sistemlərdən danışarkən, onların aparat tərkibini və proqram təminatını da nəzərə almaq lazımdır.

İnformatikanın xarakterik xüsusiyyəti – onun nəzəri və mühəndis aspektlərinin sıx qarşılıqlı əlaqəsidir. Bununla o, başqa elmlərdən fərqlənir. Məsələn, kimyada nəzəriyyə ilə kimyəvi maşınqayırma arasında sərhəd aydın ifadə olunub. İnformatikada təcrübəni təmin edən vasitələr (kompüterlər, kompilyatorlar, tətbiqi proqramlar), eyni zamanda işləyib hazırlanmanın məqsədi və nəzəriyyənin tətbiqi obyektidirlər.

İnformatikanın başqa bir fərqləndirici xüsusiyyəti – effektivlik anlayışının fundamental xarakterindədir. Bu, informasiya proseslərinin bütün mərhələlərində mütləq təmin olunmalıdır.

Əvvəllər kibernetikada olduğu kimi, informatikanı da bəzən onun istifadə sahələrindən asılı olaraq iqtisadi, hüquqi, tibbi və s. kimi sahələrə ayırırlar. Çox vaxt bölgü konkret sahələrin xüsusiyyətlərini nəzərə almır və söhbət, sadəcə, kompüterlərin, verilənlər bazasının, tətbiqi proqram paketlərinin istifadəsindən gedir (bu, mühüm nəticələr əldə olunmuş və sürətlə inkişaf edən bioinformatika və neyroinformatikaya aid deyil).

İnformatikanın daxili strukturunu nəzərdən keçirmək daha maraqlı olardı. İnformatikanın əhatə etdiyi hadisələr sferası yekcins deyil və olduqca mürəkkəbdir. Bununla yanaşı, yetərincə cavan olmasına baxmayaraq, informatika dinamik inkişaf edən bir elmdir. Mövcud olduğu dövr ərzində onun bu və ya digər bölmələri dəfələrlə meydana gəlmiş və yox olmuşdur.

İnformatikanın inkişafının müxtəlif dövrlərində onun strukturunun dəyişməsinə baxmayaraq bir neçə dayanıqlı sahələri qeyd etmək olar. Onlardan hər biri dəqiq qeyd olunmuş fəaliyyət dairəsinə, məsələlərin həllinə və tədqiqat üsullarına öz formalaşmış yanaşmalarına malikdir. Bunlara alqoritm və verilənlərin strukturu, kompüterlərin arxitekturası və təşkili; proqramlaşdırma dilləri; əməliyyat sistemləri; proqramlaşdırma texnologiyası; insan və kompüterin əlaqəsi aiddir.

Proqramlaşdırmanın texnologiyası, əsasən, böyük proqram sistemlərinin yaradılması üsullarını, onların tələbata uyğun gəlməsini və işin etibarlılığının təmin olunmasını öyrənir və həyata keçirir. Üsullar özləri də daim inkişaf edir, təkmilləşir, lakin, informatikanın məqsədi dəyişməz qalır. Digər adı çəkilən bölmələr haqqında da eyni sözləri demək olar.



"Şatl" kosmik gəmisinin modeli.

İnformatikanın qalan sahələri daha çox dəyişiklərə məruz qalmışdır. Hələ 90-cı illərin əvvəlində informatika aşağıdakı sahələrə bölünürdü: verilənlər bazası, informasiyanın axtarışı və bərpası; süni intellekt və robototexnika.

Onların informatikada əhəmiyyəti və xüsusi rolu bu gün də şübhə doğurmur, ancaq son zamanlar bu sahələr məzmunca elə güclü dəyişilmişdir ki, onları başqa cür adlandırmaq məqsədəuyğundur: informasiyanın idarə edilməsi; intellektual sistemlər.

Yeni mühüm sahələr yarandı: şəbəkə texnologiyaları və hesablamaları; kompüter qrafikası və vizuallaşdırma.

Bəşəriyyət özünün növbəti inkişaf mərhələsinə – informasiya cəmiyyətinə daxil olur. Kompüter texnologiyalarının tətbiqi sosial münasibətlərin xarakterini



və yaşayış mühitini sürətlə dəyişdirir. Ona görə də informatikanın sosial və professional problemlərinə ayırılmış sahələri xüsusi məna kəsb edir. Bunların siyahısına qəbul edilmiş qərarların nəticələrinə görə məsuliyyət, vətəndaş azadlığı və şəxsi həyatın toxunulmazlığını təmin etmək üsulları, kompüter cinayətkarlarına qarşı mübarizə, intellektual mülkiyyətin qorunması daxildir.

Yekun nəticəyə gələrək demək olar ki, böyük nailiyyətlərinə baxmayaraq, informatika çox gənc bir elmdir və öz inkişaf yolunun başlanğıcındadır.

CON FON NEYMAN

Görkəmli riyaziyyatçı və fizik Con Lüis fon Neyman 28 dekabr 1903-cü ildə Budapeşt şəhərində doğulub. Anadan olanda valideynləri ona Yanoş adını vermişdi. İsveçrə və Almaniyada oxuduğu və işlədiyi illərdə o, özünü İohann, ABŞ-a köçdükdən sonra isə Con adlandırmışdır. Con fon Neyman imkanlı macar bankirinin böyük övladı idi. Onun qeyriadi qabiliyyəti çox erkən üzə çıxmışdı. 6 yaşında bir neçə əcnəbi dil bilirdi (qədim yunan dilində sərbəst danışır və bu dildə atası ilə tez-tez zarafat edirdi), səkkiz rəqəmli ədədləri beynində bölürdü, 8 yaşında isə ali riyaziyyatın əsaslarını mənimsəmişdi. 12 yaşında olarkən bir neçə səhifəlik mətni asanlıqla yadında saxlayırdı.

İbtidai təhsilini Budapeşt şəhərində almışdır. Onun 17 yaşı tamam olanda atası belə qərara gəlir ki, oğlu riyaziyyatdan daha sərfəli bir işlə məşğul olsa daha yaxşıdır və onu kimyanı öyrənməyə inandırır. Con kimyanı öyrənmək üçün əvvəl Almaniyaya, daha sonra isə İsveçrəyə gedir. Eyni zamanda Budapeştdə riyaziyyatı öyrənməkdə davam edir. 1926-cı ildə Fon Neyman kimyaçı diplomunu və riyaziyyat doktoru dərəcəsini alır. Bir müddət Almaniyada dərs deyir, 1930-cu ildə isə onu Priston universitetinə (ABŞ) dəvət edirlər.

1933-cü ildə Pristonda Perspektiv Tədqiqatlar İnstitutu açıldığı zaman Con fon Neyman institutun əməkdaşı olur və həmişəlik ABŞ-da məskunlaşır. Con çox ünsiyyətcil adam idi: çox bilikli, məlumatlı və söhbətcil olduğundan istənilən məclisin canı olurdu.

Con fon Neyman riyaziyyat və fizikanın bir sıra sahələrinin yaranması və inkişafına böyük töhfələr vermişdir. O, riyazi məntiq, qruplar nəzəriyyəsi, operatorlar cəbri, kvant mexanikası, atistik fizika ilə bağlı əsaslı tədqiqatlar aparmışdır.



Con fon Neyman.

Con fon Neyman həmkarlarının böyük hörmət və sevgisini qazanmışdı. Teoremlərin isbatı zamanı o, yazı lövhəsini düsturlar ilə yazıb doldurardı və sonra onları elə tez silərdi ki, həmkarları onun bu hərəkətini zarafatla "silmə metodu ilə isbat" adlandırırdılar.



1944-cü ildə fon Neyman iqtisadçı Oskar Morgenştern ilə birgə yazdıqları "Oyunlar nəzəriyyəsi və iqtisadi davranış" kitabında nəinki riyazi oyunlar nəzəriyyəsini, eyni zamanda onun iqtisadi, hərbi və digər elm sahələrinə tətbiqini inkişaf etdirmişdir. Con fon Neyman Monte-Karlo metodunun – riyazi məsələlərin təsadüfi kəmiyyətlərin modelləşməsinə əsaslanan ədədi həlli metodunun yaradıcılarından biridir. Bundan başqa, Neyman atom bombasının yaradılması üzrə gizli layihədə də iştirak etmişdi.

İkinci dünya müharibəsi dövründə amerika alimləri hərbi sifarişləri yerinə yetirməli olmuşdular. Con fon Neyman riyazi məsələlər üzrə məsləhətçi kimi ENİAK-ı (elektron-hesablama maşınları) hazırlayan qrupa göndərilmişdi. Konstruksiyanı diqqətlə öyrəndikdən sonra onda EHM-lərin yeni tip məntiqi qurulması ideyası yaranmışdı:

- informasiyanın giriş-çıxış qurğusu;
- kompüterin yaddaşı;
- idarəetmə qurğusu (İQ) və hesabi-məntiqi qurğudan (HMQ) ibarət olan prosessor.

1946-cı ildə o, German Qoldsteyn və Artur Berks ilə birgə "Elektron hesablama maşınının məntiqi konstruksiyasının ilkin müzakirəsi" adlı hesabat hazırlayır. Kitabda çoxlu maraqlı ideyalar irəli sürmüş, məsələn, yaddaşa paralel müraciətin prinsipləri verilmişdir. EHM-lərin birinci iki nəslinin arxitekturası proqramda komandaların ardıcıl icrası ilə "EHM-lərin fon-neyman arxitekturası" adını almışdır.

Con fon Neyman.



Fon Neymanın 1954-cü ildə layihələndirilməsində və hazırlanmasında iştirak etdiyi EHM-lərdən biri hidrogen bombasının yaradılması üçün informasiya emalında əsas rol oynamışdır. Öz iti ağlı və hazırcavablığı ilə tanınan Con fon Neyman zarafatla bu maşını "Manyak" (MANIAC – Mathematical Analyzer Numerical Integrator and Computer) adlandırmışdı.

Müharibədən sonrakı dövrdə fon Neymanın ən böyük elmi nailiyyəti avtomatlar nəzəriyyəsinin yaradılması oldu. O, 1952-ci ildə "Etibarsız elementlərdən etibarlı orqanizmlərin sintezi və ehtimal olunan məntiq" adlı işində etibarlı EHM və digər avtomatların yaradılması yolunu göstərmişdir (Fon Neymanın nəzəriyyəsinə görə, etibarsız elementlərdən qurulmuş sistemin etibarlılığı elementlərin və onlar arasındakı bağlantıların sayının artırılması hesabına yüksəldilə bilər).

1954-cü ildə Con fon Neyman ABŞ-ın atom enerjisi üzrə Komissiyasının üzvü təyin edilir və 1955-ci ilin yazında Prinston şəhərindən Vaşinqtona köçür. Çox keçmir həkimlər onda sümük xərçəngi xəstəliyi aşkar edirlər. Alim dözülməz ağrılardan əzab çəkirdi. Ömrünün son anlarınadək gizli layihələr üzərində işlədiyi hərbi idarənin təhlükəsizlik xidməti fon Neymanın sayıqlama hallarında hər hansı dövlət sirrini verməməsinə ciddi nəzarət edirdi. Con fon Neyman 8 fevral 1957-ci ildə Vaşinqton şəhərində vəfat etdi.

Fon Neyman üzərində işlədiyi "Hesablama maşını və beyin" adlı sonuncu kitabını axıra çatdıra bilmədi.

EHM-lərin Con fon Neyman tərəfindən verilmiş arxitektura prinsipləri uzun müddət demək olar ki, dəyişməz qalmış və yalnız 20-ci əsrin 70-ci illərinin axırlarında superkompüter və matris prosessorların arxitekturasında bu prinsiplərdən kənara çıxıldı.

NORBERT VİNER

Təbiətşünaslığın problemlərilə yaxından maraqlanan bu insanın adı 1948-ci ildə "Kibernetika" adlı kitabı ilə bütün dünyaya tanındı. Norbert Viner rabitə nəzəriyyəsi sahəsində böyük xidmətləri olan, ilk hesablama maşınlarının yaradılmasında iştirak edən tanınmış amerika riyaziyyatçısıdır.

Heç nə kitabın belə sensasiya doğuracaq uğurunu qabaqcadan xəbər vermirdi. Viner 1946-cı ildə Parisdə riyaziyyat üzrə konfransda olarkən, naşirlərdən biri ona avtomatlar (maşınlar) və insanın sinir sistemi arasında əks-əlaqə haqqında kitab yazmağı təklif edir.

Kitab üzərində işləməyə başlayarkən, Viner uzun müddət onun adı haqqında düşünür. O, elə bir söz tapmaq istəyirdi ki, bu söz idarəetmə, informasiya və rabitə ideyalarını özündə birləşdirsin... Öz seçimini o, yunan sözü olan "kibernetika" üzərində saxlayır (bu sözü ilk dəfə qədim yunan filosofu Platon işlətmişdir; Amper XIX əsrdə insan cəmiyyətinin idarə olunması haqqında elmi belə adlandırmağı təklif etmişdi).

"Kibernetika" böyük xətalarla çap edilmişdi (həm də ona görə ki, Vinerin korrektura etməyə imkanı olmamışdı) və naşir kitabın kommersiya baxımından gələcəyini yüksək qiymətləndirməmişdi. Lakin bu kitab həqiqi bestseller oldu, XX əsrin ən populyar kitablarından birinə çevrildi. "Kibernetika" sözü qeyri-adi populyarlıq qazandı, Vinerin ideyaları isə insan düşüncəsində iz qoydu.

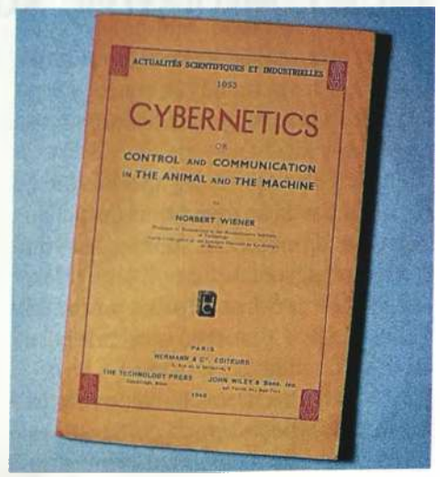
Norbert Viner 26 noyabr 1894-cü ildə Kolumbiya şəhərciyində (Missuri ştatı, ABŞ) doğulmuşdur. Atası poliqlot, slavist (slavyan filologiyası mütəxəssisi), ədəbiyyat bilicisi, tərcüməçi idi. Viner kiçik yaşlarından qədim dilləri, təbiət elmlərini öyrənir, çox mütaliə edirdi. O, fövqəladə istedada malik idi: orta məktəbi 11, ali məktəbi isə 14 yaşında bitirmişdi (onda elmlər bakalavrı dərəcəsini almışdı). 17 yaşında incəsənət magistri olmuş, 18 yaşında isə riyazi məntiq sahəsində fəlsəfə doktoru adını almışdır. Harvard universitetinin təqaüdünü alan Viner Kembric və Gyottingen universitetlərində təcrübə keçirdi. O, Bertran Rassel, Qodfri Hardi, Edmund Landau, David Hilbert və başqa tanınmış riyaziyyatçılar ilə tanış idi.



Norbert Viner.

Birinci Dünya müharibəsindən sonra (1918) Viner riyaziyyatda görkəmli nailiyyətlər əldə edir. O, XX əsrin 20-ci illərində Broun hərəkətini xarakterizə edən *Viner prosesini* müəyyənləşdirmiş (bax: "Uşaqlar üçün ensiklopediya"nın "Fizika" cildində "Atomlardan molekullara" məqaləsinə) və 30-cu illərdə riyaziyyatda böyük hadisə olan Helfondun Banax cəbri nəzəriyyəsi üçün təməl daşlarından biri olan harmonik analiz üzrə mühüm işlər nəşr etdirmişdi. 1932-ci ildən Viner Massaçusets Texnologiya İnstitutunun professoru olmuşdu.

İkinci Dünya müharibəsi dövründə (1939-1945) Norbert Viner əsasən, aviasiya və zenit artilleriyasının ehtiyacları ilə bağlı tətbiqi problemlərlə məşğul olmuşdur. O, xidməti istifadə



"Kibernetika"nın 1-ci buraxılışının fotoşəkli.



Viner mühazirə zamanı.

(məxfi olduqlarına görə) üçün nəşr edilmiş işlərdə müharibədən əvvəl A.N.Kolmoqorov tərəfindən yaradılmış təsadüfi proseslərin ekstrapolyasiyası və filtrasiyası nəzəriyyəsini tamamlamışdır. Viner ənənəvi mühəndis təhsilinə daxil olmayan riyazi aparatdan istifadə etmişdir. Ona görə də sarı cilddə dərc edilmiş kitab ünvanlandığı mühəndislər arasında "sarı təhlükə" adını almışdı. Amerikan riyaziyyatçısı Klod Şennon kitabı daha sadə mühəndis dilinə "çevirmiş" və o zamandan etibarən 20-ci əsrin bu iki böyük elm xadimi arasında daimi əlaqələr yaranmışdır.

Müharibə illərində Viner Prinstonda riyaziyyatçılarla yanaşı neyrofizioloqların, hesablama texnikası və rabitə nəzəriyyəsi üzrə mütəxəssislərin də iştirak etdiyi seminar təşkil etmişdi. Bu seminar, faktiki olaraq, sonralar "kibernetika" adını almış elmi istiqamətin başlanğıcı olmuşdur.

SSRİ-də kibernetika ideyaları həddən artıq mənfi qarşılanmışdı (bax: "İnformatika" məqaləsinə).

Lakin artıq 1958-ci ildə Norbert Vinerin hər iki kitabı – "Kibernetika, yaxud Heyvan və maşında idarəetmə və əlaqə" və "Kibernetika və cəmiyyət" rus dilində dərc olunur. Həmin ildə Böyük Sovet Ensiklopediyasının 51-ci əlavə cildində A.N.Kolmoqorovun "Kibernetika" məqaləsi nəşr edilir. Burada qeyd edilir ki, "bu yeni elmi istiqamətin məsələləri 1948-ci ildə amerika alimi N.Viner tərəfindən formalaşdırılmışdır".

Vinerin adı həddən artıq məşhurlaşır. O, ömrünün son illərində öz ideyalarını təbliğ edərək çoxlu səyahət edir. Moskvada da olmuş, orada riyaziyyatçı, biokimyaçı və psixoloqlarla görüşmüş, Politexnik muzeyində məruzə ilə çıxış etmişdir.

Viner 19 mart 1964-cü ildə vəfat etmişdir.

ANDREY NİKOLAYEVİÇ KOLMOQOROV

Andrey Nikolayeviç Kolmoqorov (1903-1987) XX əsrin ən böyük riyaziyyatçılarından və məşhur rus alimlərindən biri olmuşdur.

A.N.Kolmoqorov 25 aprel 1903-cü ildə Tambovda anadan olmuşdur. O doğulandan bir neçə saat sonra anası vəfat etmiş və uşağı onun bacısı – Vera Yakovlevna Kolmoqorova tərbiyə etmişdi. Onun uşaqlığı Yaroslavl ətrafında babasının mülkündə keçmişdir.

Riyaziyyata həvəs onda çox tez üzə çıxır. Dörd-beş yaşında o, "Весенние ласточки" ailə uşaq jurnalında riyaziyyat bölməsini aparır. Burada onun özünün fikirləşdiyi məsələlər və riyazi "kəşflər"i (məsələn, ilk *n* sayda tək ədədin cəminin n^2-na bərabər olması) dərc olunurdu.

Andrey 1910-cu ildən etibarən Moskvada yaşayır. O, Moskvanın ən yaxşı özəl gimnaziyalarının birində oxuyur. Onun gənclik maraqları çox geniş idi. O, ciddi şəkildə biologiyaya, fizikaya, riyaziyyata maraq göstərirdi. 14 yaşında ikən Brokhauz və Yefronun ensiklopediya lüğətindən differensial və inteqral hesabını müstəqil şəkildə öyrənmişdi.

1920-ci ildə Kolmoqorov Moskva universitetinə daxil olur və öz ətrafında Moskva riyaziyyat məktəbinin ən yaxşı nümayəndələrini toplamış əfsanəvi riyaziyyatçı N.N.Luzinin şagirdi olur.

Elə o vaxt Kolmoqorovun humanitar maraqlarının dairəsi yaranir: o, məşhur rus tarixçisi, Moskva universitetinin professoru S.B.Baxruşinin seminarlarına gedir və özünün Novqorod mülkədarlığına həsr olunmuş ilk elmi məruzəsini edir.

1922-ci ildə, hələ tələbə ikən, 19 yaşında Kolmoqorov triqonometrik sıralar nəzəriyyəsində nəzərə çarpan nəticə alır və onun adı bütün riyaziyyat aləmində tanınır. Elə həmin vaxt o, riyazi məntiq sahəsində də fundamental nəticələr alır.

1924-cü ildən başlayaraq, 40 il ərzində A.N.Kolmoqorov ehtimal nəzəriyyəsi sahəsində tədqiqat aparır. O, bu elmdə şərtsiz lider mövqeyini qazanır. Onun "Ehtimal nəzəriyyəsinin əsas anlayışları" kitabı Yakob Bernulli və Pyer Simona Laplasın klassik memuarları ilə bir sırada qızıl fonda daxil olub. Bu kitabda, o cümlədən, ehtimal nəzəriyyəsinin hamının qəbul etdiyi aksiomatikası yaradılmışdır.

1929-cu ildə Kolmoqorov aspiranturanı bitirir və həmin vaxtdan ömrünün sonunadək Moskva universitetinin mexanika-riyaziyyat fakültəsində işləyir.

Otuzuncu illər onun yaradıcılıq uğurlarının qeyri-adi yüksəliş dövrü idi. O, Albert Eynşteynin, Maks (Karl Ernest Lüdviq) Plankın, Marian Smoluxovskinin və Norbert Vinerin səylərini tamamlayaraq Markov təsadüfi proseslər nəzəriyyəsini yaradır; riyazi statistika üzrə köklü nəticə alır; bir sıra riyazi elm sahələrinin (topoloji cəbr, həndəsə, topoloji vektor fəzaları nəzəriyyəsi) əsasını qoyur; topologiyanın ən vacib anlayışlarından birini (yuxarı homologiyalar anlayışı) daxil edir; təsadüfi prosesslərin ekstrapolyasiyası nəzəriyyəsini qurur (bir qədər sonra eyni nəticələri alan kibernetikanın yaradıcısı Norbert Viner, onları özünün ən yüksək yaradıcılıq nailiyyəti hesab edirdi).



A.N.Kolmoqorov.

1939-cu ildə, 36 yaşında A.N.Kolmoqorov Elmlər Akademiyasının həqiqi üzvü seçilir.

XX əsrin 40-cı illərində Andrey Nikolayeviç yeni elm sahəsinə – turbulentlik lokal nəzəriyyəsinə müraciət edir və bu sahədə əsl çevriliş yaradır. Onun 3 qısa qeydindən yaranmış elmi istiqamətlərin əhatə dairəsini indi hətta təsəvvür etmək mümkün deyil. Böyük Vətən Müharibəsi zamanı hərbi təşkilatların tapşırığı ilə o, atəşin statistik nəzəriyyəsini yaradır və müharibədən sonrakı ilk illərdə məhsula statistik nəzarət nəzəriyyəsini işləyib hazırlayır.

Onun yaradıcılığının çiçəklənməsi dövrü 50-ci illərə təsadüf edir. Kolmoqorov tərəfindən göy cisimləri mexanikasında nəzərəçarpacaq nəticələr alınmışdır. Bu nəticələri Laplas və Puankarenin nailiyyətlərilə müqayisə edirlər. Bundan başqa, o, V.İ.Arnoldla birlikdə Hilbertin 13-cü problemini həll edir.

A.N.Kolmoqorov SSRİ-də informatikanın və kibernetikanın inkişafının başlanğıcında dayanan alimlərin sırasında olmuşdur və özünün yaradıcılığı



A.N.Kolmoqorov.

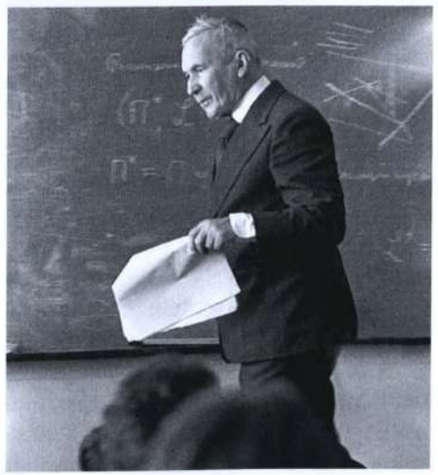
Kolmoqorova görə məlumatın mürəkkəbliyi, bütün mümkün şifrələmə metodlarına çəkilən minimal xərcdir. Bu, məlumatda olan informasiyanın daxili miqdarının ən məşhur təriflərindən biridir.

və nüfuzu ilə onların təşəkkül tapmasına imkan yaratmışdır.

50-ci illərin əvvəllərində Kolmoqorov informatikanın əsas anlayışını – alqoritm anlayışını inkişaf etdirmək haqqında fikirləşir və bu anlayışın dəqiq, hamı tərəfindən qəbul olunan tərifini verir. 60-cı illərin əvvəllərində o, inkişaf edən və özü-özünü yaradan avtomatlar ideyasını inkişaf etdirərək avtomatın ümumi konsepsiyası üzərində düşünür.

O, *informasiya nəzəriyyəsinin* yaradıcısı Klod Şenonun əsərlərinin riyaziyyat elmi üçün əhəmiyyətini dərk edən ilk rus alimlərindən biri olmuşdur. Bu nəzəriyyənin əsas anlayışını – *entropiya* anlayışını Kolmoqorov ümumiləşdirdi və riyaziyyatın çox uzaq sahələrinə: funksional analiz, dinamik sistemlər nəzəriyyəsi və approksimasiya nəzəriyyəsinə tətbiq etdi. Bu isə bir sıra yeni elmi istiqamətlərin yaranmasına gətirib çıxardı. Lakin Kolmoqorov Şennonun entropiya anlayışını *informasiya* anlayışı ilə birləşdirərək, ona geniş riyazi məna verdi.

Kolmoqorov kibernetikanın fəlsəfi aspektləri üzərində, "həyat", "təfəkkür", "idarəetmə" anlayışları üzərində, bəşəriyyətin gələcək inkişafı problemləri üzərində çox düşünürdü. Məqalələrinin birində Andrey Nikolayeviç özünü "fədakar kibernetik" adlandırmışdı. "Maşınların özləri özlərini yarada bilərmi və özünüyaratma prosesində proqressiv təkamül alına bilərmi; maşınlar emosiyaları (sevinmək, kədərlənmək, bir şeydən narazı olmaq, bir şey istəmək) hiss edə bilərmi; konstruktorların maşınların qarşısında qoymadığı məsələləri maşınlar öz qarşılarına qoya bilərlərmi?" suallarına o, müsbət cavablar vermişdi.



Kolmoqorov mühazirə oxuyarkən.

Kolmoqorov yaradıcılıq fəaliyyətinin son dövrlərində informasiya və ehtimal nəzəriyyəsinin alqoritmik əsaslarının yaradılması kimi böyük niyyətini inkişaf etdirməyə başladı. Məşhur Kolmoqorov mürəkkəbliyi anlayışı ona məxsusdur.

Kolmoqorov böyük elmi məktəb yaratmışdı, onun bir çox şagirdləri tanınmış alimlər olmuş, dünyada şöhrət qazanmışlar, onlardan 12-si Elmlər Akademiyasının həqiqi və müxbir üzvləri seçilmişlər.

Ömrünün son 25 ilini Kolmoqorov məktəb riyaziyyatının tədrisi problemlərinə həsr etmişdir.

A.N.Kolmoqorov "Успехи математических наук", "Теория вероятностей и ее применение", "Квант" (gənclər üçün jurnal) kimi riyazi nəşrlərin yaradıcılarından olmuş və onlara başçılıq etmişdir.

Öz dostunu xarakterizə edərək akademik Pavel Sergeyeviç Aleksandrov yazırdı ki, "son onilliklərdə nəinki belə geniş spektrli, hətta riyazi zövqlərə və riyaziyyatın inkişafına bu dərəcədə təsir edən riyaziyyatçı tapmaq çətindir".

Kolmoqorov universal alimin parlaq nümunəsidir. O, həm görkəmli məntiqçi, həm həndəsə alimi, həm analitik, həm naturfəlsəfəçi, həm "riyaziyyat üçün riyaziyyatçı" və həm də dərin humanitar maraqları olan insan idi.

A.N.Kolmoqorov həmişə həqiqəti axtarmış və onu müdafiə etmişdir. O, müstəsna, dahi şəxsiyyət olmuşdur. Bu cür fövqəladə qüvvəyə, geniş və dərin düşüncəyə malik alim tapmaq çətindir.

ALAN MATİSON TÜRİNQ

Bu gün Alan Matison Türinq bizə Kompüter Elminin Banisi, XX əsrin sonlarında aparıcı texnologiyanın yaradıcısı kimi təqdim olunur. Bu sözləri sağlığında heç kim demədi və gələcəkdə onu başqa səpgidə görə bilərlər.

Endryu Hocs. "Alan Türinq: Tapmaca"



Alan Türinq.

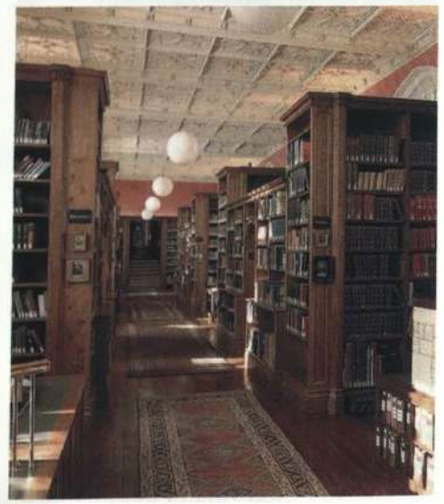
Alan Matison Türinq 1912-ci il iyunun 23-də dünyaya göz açıb. O, Culius Matison Türinq və Etel Sara Stounların ailəsində ikinci uşaq idi.

Türinqin valideynləri Hindistanda tanış olub, kəbin kəsdirmişdilər. Bu, ciddi imperiya ənənələri yaşayan, çox da varlı olmayan, aristokrat ingilis ailəsi idi. Onların çoxsaylı qohumları arasında elementar elektrik zərrəciyi anlayışını elmə gətirən, "elektron" terminini düşünüb tapan irlandiyalı fizik və riyaziyyatçı Corc Conston Stoney də var idi.

Uşaqlıqda Alan və onun böyük qardaşı Con valideynlərini çox az hallarda görürdülər. Atası Hindistanda xidmət etdiyi müddətdə uşaqlar İngiltərədə qalırdılar və ailənin dostlarının himayəsilə yaşayırdılar. Ənənəvi İngilis tərbiyəsində təbiət elmlərinin əsaslarının öyrənilməsi nəzərdə tutulmurdu. Amma Alan çox ağıllı olduğuna və hər şeylə maraqlandığına görə tərbiyəçilər ona elmi-populyar kitablar oxumağa icazə verirdilər (o, oxumağı 6 yaşında müstəqil olaraq öyrənmişdi). O vaxt, qohumlarından biri ona Edvin T.Brüsterin "Hər bir uşağın bilməli olduğu təbiət möcüzələri" adlı populyar kitabçasını bağışlamışdı. Elə hər şey də bundan başladı.

Alan 11 yaşında olarkən yosunlardan yod almaq məqsədilə kimyəvi təcrübələr aparırdı, bu isə anasını çox narahat edirdi. O, ehtiyat edirdi ki, oğlunun ümumi qəbul olunan qaydalardan kənara çıxan bu marağı ona Public Schools (zadəgan oğlan uşaqları üçün məcburi sayılan qapalı, özəl ingilis məktəbi) məktəblərindən birinə daxil olmağa mane olar. Ananın narahatçılığı əbəs idi: 1926-cı idə Alan nüfuzlu Şerborn məktəbinə (Sherborn Public School) qəbul olundu, ancaq tezliklə, onun bu məktəbi qurtara bilməməsi qorxusu yarandı.

Sinif jurnalı Alanın dərsə davamiyyət dərəcəsini aydın göstərirdi: ingilis dilindən sinifdə axırıncı, latın dilindən axırıncıdan qabaqkı, qalan fənlərdən bir az yaxşı. Bütün müəllimlər onun haqqında yekdil fikirdə idilər: "ümidsiz gerilik, qeyri-kafi davamiyyət" ...Məktəbin direktoru yazırdı: "Bu oğlan istənilən məktəb və cəmiyyət üçün böyük problemdir". Bununla belə məktəb jurnalında başqa yazılar da var idi: "Əgər o, yalnız elmi işçi olmaq istəyirsə, Public School'da əbəs yerə vaxt itirir... Yəqin ki, o, riyaziyyatçı olacaq. Belə şagirdlər 200 ildə bir dəfə doğulur".



Kembric universitetinin Kral kollecinin kitabxanası.



Alan Türinq tələbəlik illərində.

15 yaşında Alan müstəqil şəkildə nisbilik nəzəriyyəsini mənimsəyir, lakin klassik Britaniya məktəbində mövcud vəziyyət və tədrisin aparılma qaydası belə maraqların inkişafına şərait yaratmırdı. Gənc oğlan məcburi proqram fənlərinə qarşı tam laqeyd idi və nəticədə ümumiyyətlə attestat almamaq təhlükəsilə üzləşdi.

1928-ci ildə Türinq çoxdan gözlədiyi həmdəm – Kristofer Mork adlı, çox istedadlı, qabiliyyətli yeni sinif yoldaşı tapdı. Axır ki, Alan kiminləsə öz fikirlərini bölüşə bilərdi. Gənclər ayrılmaz dostlara çevrildilər. Məktəbi bitirdikdən sonra hər ikisi Kembric Universitetinə daxil olmağa hazırlaşırdı. Hazırlıq imtahanı vermək üçün Alanın ilk cəhdi uğursuz oldu. Ancaq sınaqdan müvəffəqiyyətlə çıxan və təqaüd alan Kristoferə görə ürəkdən sevinən Alan buna görə çox da heyfsilənmədi. O, dostu ilə birlikdə oxumaq üçün ikinci cəhddə müvəffəqiyyət qazanmağa ümid edirdi.

1930-cu il fevralın 13-də dostu Kristoferin qəfil ölümü 17 yaşlı Türinqi sarsıtdı: insan zəkası necə maddiləşir və ölümdən sonra azad olurmu? – sualı onu düşündürdü. Alan bu sualların cavabını Artur Eddinqtonun (1882-1944) "Fiziki aləmin təbiəti" kitabında, insanın seçim azadlığının izahını isə kvant mexanikasında axtardı.

1931-ci ilin oktyabrında Alan Kembric Universitetinin Kral kollecinin tələbəsi oldu. Tezliklə, xeyli uğurlar müşahidə olundu: əla davamiyyət, intensiv elmi iş, 1934-cü ildə fərqlənmə ilə Kembric bakalavr dərəcəsi, 1935-ci ildə Kral kollecinin magistr dərəcəsi və aspirant təqaüdü, 1936-cı ildə ehtimal nəzəriyyəsinə aid işinə görə Smit mükafatı. İnsan zəkasının təbiətinə dair düşüncələri tələbə Türinqi 1932-ci ildə Con fon Neymanın kvant mexanikasının məntiqi əsasları haqqında çox mürəkkəb işini öyrənməyə və formal məntiq sahəsində tədqiqatlara gətirib çıxartdı. 1935-ci ildə o, gözlənilmədən həllin mümkünlüyü haqqında məşhur Hilbert problemilə maraqlandı və onun üzərində işləməyə başladı. 1936-cı ilin aprelində Alan işini artıq başa çatdıranda Amerika məntiqçisi Alonzo Çerçin həmin nəticə ilə, ancaq tamam başqa metodla alınmış məqaləsi çıxdı. Hər iki tədqiqatçının gəldiyi nəticə mənfi idi, lakin Alan bu nəticəyə çox orijinal yolla gəlmişdi. Əvvəlcə Alan kağız lentdən simvol-komandaları oxuyan xəyali maşının iş prosesi şəklində metod müəyyənləşdirdi. Sonra o, isbat etdi ki, onun maşını formal məntiq çərçivəsində insan ağlının qadir olduğu istənilən hesablamaları aparmaq gücündədir. Baxmayaraq ki, Çerçi məşhur problemi həll edən ilk adam hesab edirlər, "Həllin mümkünlüyü probleminə tətbiq etməklə hesablana bilən ədədlər haqqında" Türinqin işi ona "hesablama alqoritmi" və "universal kompüter" anlayışlarının ilk yaradıcısı kimi şöhrət gətirdi.

Türinq 4-illik təhsil kursunu müvəffəqiyyətlə başa vurdu. Sanki onu sırf riyaziyyatla məşğul olan, qismən eksentrik Kembric professoru kimi uğurlu karyera gözləyirdi. Lakin 1936-cı ilin sentyabrında Alan Prinston (ABŞ) Universitetindən 2-illik aspirant təqaüdü aldı və Con fon Neymanın yanına gəldi. Orada o, cəbr və ədədlər nəzəriyyəsi sahəsində işlədi və hesablanabilmə haqqında Çerçin tərifilə öz tərifi arasındakı qanunauyğunluğu isbat etdi. O, həmçinin hesablama məntiqi barəsində öz fikirlərini inkişaf etdirdi və bu Türinqin ən mürəkkəb və dərin riyazi işinin – doktorluq dissertasiyasının mövzusu oldu. Türinq elektromaqnit relesindən istifadə etməklə şifrləmə maşınını düzəltdi və şifrləmə elmini öyrənməyə başladı.



Türinq kross yarışında iştirak edir. 1946-cı il.

1938-ci ildə doktorluq dissertasiyasının müdafiəsindən dərhal sonra Alana Prinston universitetində professor vəzifəsi təklif etdilər. Lakin o, İngiltərəyə – Kembricə qayıtdı. 1939-cu ilin sentyabrında Kod və şifrlər üzrə hökumət məktəbində işə başladı.

İkinci Dünya müharibəsi başladı. Almaniya radioməlumatları kodlaşdırmaq üçün "Eniqma" ("Tapmaca") elektromexaniki şifrləmə maşınını tətbiq etməyə başladı ki, onun da şifrlərini açmaq mümkün olmurdu. Əyləncəli kriptoqrafik döyüşün baş qəhrəmanı Alan Türinq oldu. İngilislər "Eniqma"ya qarşı özlərinin deşifrləyici maşını olan "Bomba"nı qoydular, lakin o, bu işin öhdəsindən gəlmədi. Türinq maşını kifayət qədər təkmilləşdirə bildi ki, bunun da nəticəsində lyuftvaffda (almanların hərbi hava qüvvələri) istifadə olunan şifr sındırıldı. Bundan sonra ingilislər Gerinqin hava donanmasının planlaşdırdığı əməliyyatların bütün detallarından xəbərdar oldular.

Ancaq 1942-ci il fevralın 1-də, ABŞ müharibəyə qoşulduqdan az sonra alman submarinləri mürəkkəbləşdirilmiş şifrləməyə keçdilər və "Bomba" müvəffəqiyyətsizliyə uğradı.

Alan "Koloss" adlı yeni maşın düzəltməyə başladı ki, burada da elektromexaniki rele əvəzinə 2 min elektron lampa istifadə olunurdu. "Koloss" çox asanlıqla "Eniqma"nı məğlub etdi və hətta "balıq dili"nin də öhdəsindən gəldi.

1942-ci ilin noyabrından 1943-cü ilin martına kimi Türinqi ABŞ-a ezam etdilər. O, burada Ruzveltlə Çerçill arasındakı danışıqların elektron şifrlənməsinin təmin olunmasında Britaniyanı təmsil edirdi (sonralar Alan Çerçillin təşəbbüsü ilə Britaniya imperiyası ordenilə təltif olundu, ancaq çox az adam bildi ki, bu hansı xidmətə görədir).

Türinq nəzəri tədqiqatları da unutmamışdı. Onun ilkin ideyalarından biri – xəyali "Türinq maşını" ilə insan ağlının işində analogiya – "ağıllı kompüter" (süni intellekt) konsepsiyasına transformasiya olundu ki, onun da IQ-sı (intellekt əmsalı) indi Türinq testi adlanan xüsusi işlənmiş məntiqi prosedurun köməyilə yoxlanılırdı.

David Hilbert (1862-1943) belə bir sual qoymuşdu: "Prinsipcə elə bir üsul və ya proses varmı ki, onun köməyilə istənilən riyazi məsələni həll etmək mümkün olsun?"

Müharibədən sonrakı illərdə Alan hər bir işi görməyə qabil olan: sayan və mürəkkəb cəbri hesablamalar aparan, şifrləri açan və şahmat oynayan, universal proqramlaşdırıla bilən elektron kompüter yaratmaq ideyası ilə məşğul olmağa başladı. O, proqramlar kitabxanası, terminallar şəbəkəsi olan vahid milli kompüter mərkəzi arzulayırdı.

1949-cu ildə Türinq Mançestr Universitetinə keçdi və orada elektron kompüter üçün proqramlar hazırladı. Manchester Mark I və ya MADAM (Manchester Automatic Digital Machine) adlanan kompüter meydana gəldi. MADAM layihəsinin inkişafı nəticəsində dünyada ilk tam iş qabiliyyətli kompüterlərdən biri – Ferranti Mark I işə salındı.

1950-ci ildə Türinq "Hesablama maşınları və intellekt" (60-cı illərdə rus dilində "Maşın fikirləşə bilərmi?" adlı ayrıca kitab şəklində çıxmışdır) fəlsəfi məqaləsini – süni intellekt haqqında gələcək elmin klassik əsərini yazdı. 1951-ci ildə onun elmi xidmətləri etiraf olundu: alimi London Kral elmi cəmiyyətinə üzv seçdilər. Türinq bitkilərdə və balıqqulaqlarında yeni forma alınmasının qeyri-xətti modelləşdirilməsi üzrə hesablama eksperimentlərinə başlayır. O, canlı orqanizmlərin necə böyüməsini və nə üçün öz formalarını almasını tədqiq edir. Alan uşaqlıqdan onu narahat edən sualların cavablarını, həllərini kompüterdə modelləşdirməklə, kimyəvi reaksiyaların tənliklərinin qeyri-xəttiliyində və diffuziyada tapır. Onun tədqiqatları göstərdi ki, elektron kompüterlər elmi məqsədlər üçün istifadə oluna bilər.



"Eniqma" şifrləmə maşını.

Özünün kimya məşğələlərində Alan Türinq onun özü tərəfindən ixtira olunmuş "Kimsəsiz ada" oyununu oynayırdı. Oyunun məqsədi əlinin altında olan vasitələrdən istifadə etməklə müxtəlif faydalı kimyəvi maddələrin (paltaryuyan toz, qab yumaq üçün vasitələr, mürəkkəb və bu kimi məişətdə istifadə olunan kimyəvi maddələrin) alınması idi.

Zadəgan ailəsindən çıxmasına baxmayaraq, Alan heç vaxt estet olmamışdı: Kembric siyasi və ədəbi dərnəkləri ona yad idi. O, sevdiyi riyaziyyatla məşğul olmağa, boş vaxtlarında isə kimyəvi reaksiyalar aparmağa, şahmat məsələləri həll etməyə, qo oynamağa üstünlük verirdi.

Tələbələr pıçıldaşırdılar ki, zəngli saatın əqrəblərini çəkərkən Alan dəqiq vaxtdan istifadə etmirdi, ancaq ulduzlara baxıb onun yalnız özünə məlum olan hesablamalar aparırdı.

Alanın, demək olar ki, dostu yox idi. Onun bir qədər səliqəsiz geyim stili, ekssentrik saç düzümü və kəskin cır səsi (bundan əlavə o hərdən güclü kəkələyirdi) çoxlarını ondan uzaqlaşdırırdı.

Türinq qibtə olunacaq sağlamlığı ilə seçilirdi, yaxşı idmançı idi, marafon qaçışına və avarçəkməyə üstünlük verirdi. Boş vaxtlarında Valton idman klubunda məşğul olurdu. Tez-tez səhərlər 10 km (evdən laboratoriyaya qədər) qaçmaqla iş yoldaşlarını təəccübləndirirdi. Onda allergiya yaradan bitkilər çiçəkləyən dövrdə isə əleyhqaz taxıb velosipedlə gedərdi. Buna görə onu polislər dəfələrlə saxlamışdılar. Alan özünün gərgin məşqlərini işlə bağlı stressləri aradan qaldırmaq zərurətilə izah edirdi. 1948-ci ildə kross qaçışında onun göstərdiyi nəticə tezliklə Olimpiya oyunlarının gümüş mükafatçısı olacaq idmançının nəticəsindən yaxşı idi. təsadüfi zədə Alana Olimpiya oyunlarında iştirak etməyə imkan vermədi.

Bu hərtərəfli istedadlı insan həyatdan tez getdi. Alan Türinq 1953-cü il mart ayının 31-də həbs edildi və homoseksualizmdə təqsirləndirilərək məhkəmə qarşısında durdu (bu gün qeyri-ənənəvi seksual yönüm adlandırılan əməl müharibədən sonrakı dövrdə İngiltərədə cinayət sayılırdı). Ona seçim təklif olundu: həbsxana, yaxud müalicə. O, müalicə olunmağı seçdi. Bütün bu yaşananlar onun psixoloji sağlamlığında pozuntulara səbəb oldu. Dərin depressiya (ruh düşkünlüyü) şəraitində öz axırıncı kimyəvi təcrübəsini apardığı günədək o, elmi işlərini davam etdirdi.

1954-cü il iyun ayının 8-də xidmətçi onu ölmüş halda tapdı. Axıra qədər yeyilməmiş almada və alimin barmaqlarında kalium sianid aşkar olundu. Müstəntiqin gəldiyi nəticə qəti idi: intihar. Elə həmin zəhər onun bir gün əvvəl kimyəvi təcrübə apardığı stolun üstünə dağılmışdı. Ancaq Türinqin anası və həmkarlarının çoxu buna inanmadı. Alan proqramlaşdırma nəzəriyyəsi, biologiya, kvant fizikası sahələrində yeni tədqiqatlara təzə başlamışdı, baxmayaraq ki, məhkəmədən sonra ona gizli kompüter laboratoriyalarında işləmək qadağan olunmuşdu... Türinqin 41 yaşı vardı.



A.P.Yerşov.

ANDREY PETROVİÇ YERŞOV

Akademik Andrey Petroviç Yerşov (1931-1988) – nəzəri və sistem proqramlaşdırmasının banilərindən biri, Sibir informatika məktəbinin yaradıcısıdır. İnformatikanın elmin yeni bir sahəsi və ictimai həyatın yeni fenomeni kimi təşəkkül tapmasında onun önəmli töhfəsi postsovet məkanında və xaricdə geniş surətdə qəbul edilmişdir.

Andrey məktəbi qurtararkən, o dövrün bir çox gəncləri kimi, nüvə fizikası ilə məşğul olmağı arzulayırdı. 1949-cu

ildə o, Moskva universitetinə daxil olur, lakin artıq sənədlərini verəndə fizik-nüvəçi karyerasından imtina etmək lazım gəldi: Böyük Vətən müharibəsi illərində 11-12 yaşlı uşaq olmuş Andrey faşistlər tərəfindən işğal olunmuş ərazidə yaşamışdı və bu kifayət edirdi ki, o, dövlət sirri olan elm sahələrinə buraxılmasın.

Hələ MDU-nun tələbəsi olarkən A.A.Lyapunovun təsiri altında Yerşov hər addımı yaradıcı və maraqlı axtarış olan proqramlaşdırmaya vuruldu və ömürlük ona bağlandı. Onun müəllimləri Ünvanların avtomatik verilməsi proqramının – (*rus.* программа автоматического присвоения адресов – ПАПА) öyrənilməsilə məşğul olurdular (o, ilk hesablama mərkəzlərində tətbiq edilirdi və aşağı səviyyəli dillərin inkişafına – avtokoda gətirib çıxardı). Andrey isə başqa bir istiqaməti tədqiq edirdi. Onun diplom işi "Proqramlaşdıran proqram" – PP adlanırdı. Burada söhbət ilk yüksək səviyyəli sistem proqramlaşdırması translyatorundan gedirdi. Bu iş gənc alimin həyat seçimini çox şeydə müəyyən etdi.

Universiteti bitirdikdən sonra Andrey Petroviç Elmlər Akademiyasının Dəqiq mexanika və hesablama texnikası İnstitutuna – ilk sovet proqramçı kollektivlərindən birinin yarandığı təşkilata işə girdi.

1957-ci ildə onu SSRİ EA-nın yeni yaradılan Hesablama Mərkəzində proqramlaşdırmanın avtomatlaşdırılması şöbəsinə müdir təyin edirlər. SSRİ EA-nın Sibir şöbəsinin (SŞ) yaradılması ilə əlaqədar Riyaziyyat İnstitutunun direktoru akademik S.L.Sobolevin xahişilə Andrey Petroviç öz üzərinə bu institutun proqramlaşdırma şöbəsinin təşkilatçısı və faktiki rəhbəri vəzifəsini götürür, daha sonra isə REA SŞ-nin Hesablama Mərkəzinə keçir. Amma "direktorun xahişilə" – bu, daha çox rəsmi ifadə idi.



162 saylı məktəb. Novosibirsk. 1984-cü il.



A.P.Yerşov Ümumittifaq yay məktəbinin şagirdləri arasında.

Əslində isə o, ilk təşəbbüsçü könüllülər sırasında onun üçün gerçəkdən doğma olmuş, yeni Sibiri öyrənməyə – Akademşəhərcik tikməyə getmişdi. Yerşov Akademşəhərciyin elmi və ictimai həyatında yaxından iştirak edirdi. O, yüksək məsuliyyət hissilə elmi konfranslar təşkil edir, dünyanın hər yerindən qonaqları və dostları qəbul edir, SSRİ EA-nın SŞ-nin Rəyasət heyəti yanında Elmi Şuranın işində fəal iştirak edirdi.

A.P.Yerşovun proqramlar sxemləri və kompilyasiya nəzəriyyəsi sahəsindəki fundamental tədqiqatları onun çoxsaylı şagirdləri və ardıcıllarına nəzərə çarpacaq dərəcədə təsir göstərdi.

A.P.Yerşovun "Sürətli elektron hesab maşını üçün proqramlaşdıran proqram" adlı kitabı – proqramlaşdırmanın avtomatlaşdırılması üzrə dünyada ilk monoqrafiyalardandır.

A.P.YERŞOV ALİM VƏ İNSAN KİMİ

Andrey Petroviç Yerşovun ətrafında həmişə həyat qaynayırdı – onu tələbə və məktəblilər, aspirantlar və gənc tədqiqatçılar, xarici alimlər və Sibir jurnalistləri əhatə edirdilər. Andrey Petroviç nə vəzifəsinə görə administrator, nə təhsilinə görə tərbiyəçi deyildi, lakin bununla belə ona məsləhət, xahiş üçün daim müraciət edirdilər. Həm xoş gündə gəlirdilər: "Oğlum olub!", həm də dar gündə: "Nə edim? Müdafiəm baş tutmadı". Onun aspirantları arasında özbək və yəhudilər, rus və tatarlar, ukraynalılar və litvalılar dostcasına və həvəslə işləyirdilər. O hər bir adam üçün lazımi söz və düzgün ton tapa bilirdi. Söhbət son dərəcə işlə yüklənmiş və hər bir dəqiqəsini qiymətləndirib saya bilən bir adamdan gedir. Lakin ondan heç kəs heç vaxt sanki çox təbii bir replikanı eşitməzdi: "Mən məşğulam! Vaxtım yoxdur!". Yüksək səviyyəli ziyalı olan bu adam həmkarlarının, dostlarının və qohumlarının xatirində həm də əsl insan kimi əbədi olaraq qalmışdır.

Akademik A.P.Yerşov alimlərin informasiya təminatı probleminə çox diqqət yetirirdi. Kitabı çox sevir və ömrü boyu onları toplayırdı. Onun şəxsi kitabxanasında praktiki olaraq bütün Avropa dillərində 30 mindən çox kitab, jurnal, konfransların işləri, ayrı-ayrı məqalələrin ottiski qorunub saxlanırdı. Ölümündən sonra onun davamçıları kitabxananı artıq o vaxtlar Hesablama Mərkəzindən ayrılmış İnformatika sistemləri İnstitutuna təhvil verdilər. İndi bu, A.P.Yerşov adına Xatirə kitabxanasıdır.

Xidməti maşından çox vaxt istifadə etməzdi: Akademşəhərciyin meşə cığırları ilə sübh tezdən işə və gecə yarıdan keçmiş evə piyada getmək onun gündəlik həyatının bir hissəsi idi. Andrey Petroviç Yerşov təkcə zəkalı alim, müəllim və öz ideyaları uğrunda mübariz deyildi, həm də görkəmli, hərtərəfli istedadlı şəxsiyyət idi. O, musiqini, poeziyanı sevir, özü şeir yazır və ingilis şairlərinin şeirlərini rus dilinə tərcümə edirdi. Onun çox gözəl tərcümə etdiyi Kiplinqin "Nə vaxt" ("Oğluma") şeirini mütəxəssislər ən yaxşı tərcümə hesab edirlər (axı S.Y.Marşak və M.L.Lozinski kimi poeziya sütunlarının tərcümələri də vardır). Alimlər Evində musiqi gecələri onun ən sevimli istirahət yeri idi.

Andrey Petroviç gitarada çox gözəl çalır və oxuyurdu. Ob sahilindəki şam meşəsində tonqal başında keçən gecəni xatırlamaq xoşdur. Gündüzlər "Proqramlaşdırmada insan və estetik faktorlar haqqında" adlı mühazirələr dinləyən məktəblilər əlinə gitara almış sevimli müəllimlərini üzük qaşı kimi dövrəyə alaraq onun oxuduqlarını sakitcə təkrar edirlər. Akademik Yerşov ilə ünsiyyətdə olmaq xoşbəxtliyinə nəsib olan adamlar onun parlaq ideyalarını, böyük nailiyyətlərini və qeyri-adi xeyirxahlığını heç zaman unutmayacaqlar.

O çox işləyirdi. Əlbəttə, iş kabineti və universitet kitabxanası müqəddəs idi. Lakin əgər təyyarə kreslosunun qatlanan stolunda və ya şəhərlərarası Novosibirsk-Barnaul avtobusunda yazmaq lazım gəlirdisə, böyük alim bunda bir çətinlik görmürdü. Kiçik həmkarları və şagirdləri onun iş üslubundan hətta kiçik xırdalıqlara qədər nümunə götürməyə çalışırdılar. Əlyazmasının hər səhifəsində tarix, saat və dəqiqə qeydləri vardı. Çox nadir fikir yığcamlığı və məqsədyönlülük ona imkan verirdi ki, geniş diapazonda olan işlərini yaddan çıxartmasın.

Hətta ölümündən bir az əvvəl Yerşovla Onkoloji Mərkəzin palatasındakı görüşü belə xəstəyə baş çəkmək adlandırmaq olmaz. Xəstəxana çarpayısının yanında üstündə kitab və əlyazmalar olan stol vardı. Andrey Petroviç daim işləyir, görüşə gələn adamla ona əzab verən xəstəlikdən yox, iş və qayğılardan danışırdı.

1983-cü ildə A.P.Yerşov qarışıq hesablamalar nəzəriyyəsinə əsaslı töhfəsinə görə akademik A.H.Krılov adına mükafatla təltif edilmişdir.

"Alfa" proqramlaşdırma dili və optimallaşdırıcı "Alfa"-translyatoru, "Aist-0" ilk sovet vaxtın bölüşdürülməsi sistemi, "Məktəbli qız" tədris informatika sistemi, "Rubin" çap məhsullarının hazırlanması sistemi, "Mramor" çoxprosessorlu işçi stansiyası – bütün bu mühüm layihələr A.P.Yerşovun təşəbbüsü ilə yaradılmış və onun rəhbərliyilə yerinə yetirilmişdir.

Unikal elmi uzaqgörənliyi sayəsində Andrey Petroviç elmin və cəmiyyətin tərəqqisində hesablama texnikasının həlledici rolunu ilk başa düşən alimlərdən biri olmuşdur. Onun parlaq ideyaları SSRİ-də paralel proqramlaşdırma və süni intellekt kimi elmi istiqamətlərin inkişafının əsasını qoymuşdur.

A.P.Yerşov 1980-ci ildə orta məktəbdə proqramlaşdırmanın tədrisi üzrə eksperimentlərə başladı. Bu, məktəb proqramına informatika və hesablama texnikası kursunun daxil edilməsilə nəticələndi.

Bu gün geniş istifadə olunan "məktəb informatikası" termini EA SŞ-nin Hesablama mərkəzində, Yerşovun kabinetində yaranmışdır. Onun bilavasitə müəllif kimi iştirakı ilə (məktəb alqoritmik dilinin təsvirini təklif etmişdir) informatikaya aid ilk sovet məktəb dərsliyi hazırlanmışdır. Bundan başqa, bu kurs üzrə o, bir sıra tədris televerilişləri də yaratmışdır.

A.P.Yerşov Novosibirsk Akademşəhərciyində müntəzəm olaraq keçirilən Gənc proqramçıların Ümumittifaq yay məktəblərinin daimi və ən əziz qonağı idi.

Akademik Yerşovun elmdə təşkilatçı kimi rolu böyük idi: o, müxtəlif beynəlxalq elmi konfrans və konqreslərin hazırlanmasında çox aktiv iştirak edir,

həm rusca çıxan "Микропроцессорные средства и системы", "Кибернетика", "Программирование", həm də beynəlxalq "Acta Informatica", "Information Processing Letters", "Theoretical Computer Science" jurnallarının redaktoru və redaksiya heyətinin üzvü idi.

Rusiya EA SŞ İnformatika sistemləri institutu və əsas məqsədi informatikanı ixtiraçılıq, yaradıcılıq və sənətkarlıq kimi inkişaf etdirmək olan A.P.Yerşov adına xeyriyyə Fondu onun adını daşıyır.



A.P.Yerşov 1986-cı ildə. Fiziki təcrübələr.

UŞAQLARA PROQRAMLAŞDIRMANI NƏ ÜÇÜN VƏ NECƏ ÖYRƏTMƏLİ

PROQRAMÇILARIN ÜÇ KATEQORİYASI

Hesablama maşınlarının sayı ildən-ilə çoxalır. Kompüterin köməyilə həll olunan məsələlərin sayı ondan da böyük sürətlə artır. Sanki perspektiv aydındır: tezliklə hamı alqoritmik dillərdə proqram yazmalı olacaq.

Hesablama texnikasının inkişafının ilk dövrlərindən proqramçılar arasında əməyin bölgüsü nəzərdə tutulub. *Sistem proqramçılar* – professional proqramçılardır. Onların yaratdıqları proqramlar (əməliyyat sistemləri, yüksək səviyyəli dillərin translyatorları, verilənlər bazasını idarəetmə proqramları və s.) bütün kompüter istifadəçilərinin işini asanlaşdırır. Sistem proqramçılar ali, bir qayda olaraq, universitet təhsili alırlar.

İkinci "qat" proqramçılar – *tətbiqi proqramçılardır*. İqtisadiyyat, geologiya, poliqrafiya, meteorologiya, tibb və s. üzrə mütəxəssis olan bu adamlar proqramlaşdırma nəzəriyyəsini, konkret kompüterlərdə proqramlaşdırmanın xüsusiyyətlərini bilmədən öz sahələri üzrə məsələni ən münasib proqramlaşdırma dillərində verə bilirlər. Tətbiqi proqramçıları, adətən, profilli ali məktəblər hazırlayır.

Sistem və tətbiqi proqramçıların səyilə məsələlərin maşında həlli hazır prosedurlara parametrlərin qiymətlərinin verilməsi şəklinə gətirilə bilər.

Məhz bu yolla daha çox insanlar – *parametrik istifadəçilər* üçün kompüterlərə yol açılır. Onların nə kompüterin qurğularını, nə həll edilən məsələnin alqoritmini, nə də proqramın yazıldığı dili bilməyinə gərək yoxdur. Bu istifa-



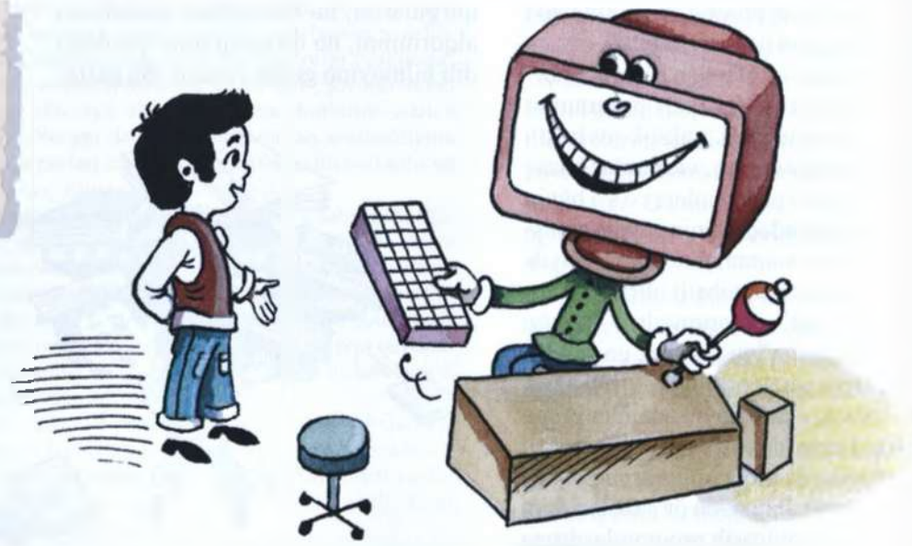


Kompüter sinfi.

dəçilər dəmiryol kassasının kassiri, bank operatoru, poçt xidmətçiləridir.

Sistem və tətbiqi proqramçıların sayı və artım sürəti müqayisədə çox deyil. "Sadə" parametrik istifadəçilərin sayı isə artır və sürətlə artmaqda davam edəcək. İndi tamamilə aydındır ki, tezliklə yer kürəsi əhalisinin böyük hissəsi istifadəçiyə çevriləcək. Lakin kompüterdən uğurla istifadə üçün hətta parametrik istifadəçilər də müəyyən bacarıq və vərdişlərə yiyələnməlidirlər.

Qol saatınızda "Pentium IV"dən aşağı olmayan prosessor dururs, onda siz əsl proqramçısınız.



PROQRAMÇI NƏ EDƏ BİLƏR

Aşağıda sadalananlar hər bir parametrik istifadəçiyə lazım ola biləcək bacarıq və vərdişlərin bir hissəsidir.

Birincisi, məhdud vasitələr çərçivəsində işləri planlaşdırmağı öyrənmək.

İkincisi, bütün obyektləri formal və tam olaraq verilənlər və ya nəticələr kimi göstərə bilmək, onların qarşılıqlı əlaqəsini və ya proqramçıların dililə desək, məsələnin informasiya modelini qurmağı bacarmaq.

Üçüncüsü, lazımi informasiyanı necə tez tapmağı bilmək (axı, maşında məsələnin həllinə sərf olunan vaxtın böyük hissəsini informasiyanın axtarışı tutur).

Dördüncüsü, maşın üçün komandaları düzgün qurmaq çox əhəmiyyətlidir: əgər kompüter zəngin prosedurlar yığımına (və ya sistem proqramçısının dediyi kimi, yaxşı proqram təminatına) malikdirsə, onda onunla böyük informasiya porsiyaları – prosedurlar vasitəsilə ünsiyyətdə olmaq olar; lakin əgər əvvəlcədən belə prosedurlar yazılmamışsa, onda tələb olunan əməliyyatları daha kiçik informasiya porsiyaları – operator və ya

direktivlər vasitəsilə təsvir etməklə maşına müraciət etmək olar.

İndiyədək göstərilən bacarıq və vərdişlər hesablama texnikasından yalnız effektiv istifadə zərurətilə əlaqələndirilirdi. Bununla belə, onların hər biri daha geniş məna daşıyır.

Məsələn, planlaşdırmağı bacarmaq təkcə kompüter istifadəçisinə deyil, həm də mühəndisə, aqronoma, zabitə, müəllimə, iqtisadçıya və başqalarına lazımdır.

Modelləşdirməyi bilmək, yəni yaradılan konstruksiyanın xüsusiyyətlərini əvvəlcədən görmək, yaradıcı işçilərə – texnoloqlara, mühəndislərə, fiziklərə, astronomlara daha gərəklidir.

Axtarışın düzgün təşkili – lazımi məlumat almaq üçün insan biliyinin müxtəlif saxlanma yerlərinə: kitabxanalara, arxivlərə, informasiya sistemlərinə müraciət edilən hər bir kəsin uğurlu işinin əsasıdır. Əbəs yerə demirlər ki, bəşəriyyətin topladığı bütün bilikləri şagirdə məktəb illərində vermək imkanı olmadığından müəllim ilk növbədə ona oxumağı, daha doğrusu, ona lazımi informasiyanı axtarıb tapa bilməyi öyrətməlidir.

Nəhayət, istənilən insan kollektivində ünsiyyət intizamı kompüterlə "söhbət"dən az əhəmiyyət kəsb etmir. Burada məsələ yalnız ifadə dəqiqliyində və nəzakətdə deyil. Söhbətin işgüzar alınması üçün həmsöhbətinin, məsələn, bilik səviyyəsini nəzərə almalısan.

PROQRAMLAŞDIRMA KÖMƏYƏ GƏLƏR

Planlaşdırma, modelləşdirmə, informasiya axtarmaq, ünsiyyət qurmaq bacarığı o qədər fundamentaldır ki, öyrənməyə mümkün qədər tez başlamaq lazımdır. Lakin nə fizika, nə də ədəbiyyat, nə hətta riyaziyyat bunun üçün vacib olan alətlərə (tam həcmdə) malik deyil. Proqramlaşdırmada isə belə alətlər var! Həqiqətən, kim ki, prosedur, şərti operator, dövr nə olduğunu başa düşür, planlaşdırma bacarığına da asanlıqla yiyələnə bilər. İnformasiya modelləşdirməsi üçün müasir proqramlaşdırma dillərində olan verilənləri təsvir vasitələri – çoxluqlar, cədvəllər, kataloqlar, fayllar çox münasibdir. Eləcə də

MAĞAZALARI GƏZİRİK...



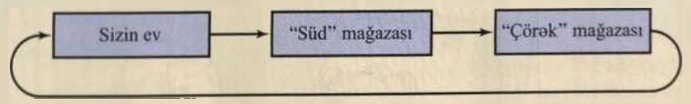
Biz hər tərəfdən informatika ilə əhatə olunmuşuq. Hərdən adama elə gəlir ki, informatika bilikləri bizə anadangəlmə verilir. Əksər pedaqoqları belə bir sual narahat edir: alqoritmik düşünməyi öyrənmə təbii şərtlər daxilində mümkündürmü? Təəssüf ki, yox. İnformatika müəllimlərini uzun illər heyrətləndirən fakt budur.

İnformatika kursu başlayandan bir həftə sonra, "maraq peyvəndi" adlandırılan dərsdə çox sadə bir məsələ təklif olunur: "Çörək və süd alıb evə qayıtmalısınız. Bu zaman heç bir qeyri-adi və əlavə şərtlər fikirləşmək lazım deyil. Sadəcə almaq və sonra isə geri dönmək lazımdır."

Heç bir sinifdə heç bir şagird əvvəl çörək, sonra süd almadı, tərsinə elədi, yəni boş yerə gücünü sərf elədi. Hamı südü çörək dükanına daşıdı və əksinə.

Təəssüflə deyə bilərik ki, planlaşdırma (və ya onu proqramlaşdırma adlandıraq) adi (yəni təbii) və yayılmış düşüncə tərzi deyildir. Alqoritmik düşüncə, məsələn, ya tanış hərfləri oxuyarkən, ya da gəzərkən müvazinəti saxlamaq kimi "avtomatik" deyil. Belə çıxır ki, bu bacarığı (hətta onu öyrətsən belə) hər bir halda xüsusi olaraq "işə salmaq" lazımdır. Məsələnin birbaşa həllindəki uğursuzluq "işə salmaq" üçün siqnaldır.

Maneələr çıxmırsa, hərəkətlər ardıcıllığı həmişə primitivdir (əvvəlcə süd, sonra – çörək) və çox vaxt effektiv deyildir.





axtarışın təşkili üçün müxtəlif axtarış mexanizmləri (kataloqlar, çoxsəviyyəli indekslər, düzülüş funksiyaları və s.), ünsiyyət prosesinin təşkili üçün isə sintaktik təsvir və makrovasitələr istifadə edilir.

"Uşaqları necə öyrətməli?" sualına cavab almağa dərs proqramlarının yaradılması ilə məşğul olan müəllim və proqramçıların təcrübəsi kömək etdi. Onlar hesab edirlər ki, ilk növbədə hər biri konkret, eyni zamanda dərs məsələsi üçün nəzərdə tutulan sadə icraçı-robotları idarə etməyi öyrənmək vacibdir.

İbtidai məktəb şagirdlərinin informatika dərslərində ekranda LOGO və ya Pascal dillərində yazılmış proqramların sətirləri deyil, icraçılar – Daşıyıcı, Cəmləyici, Cizgiçi, Kenquru, Xoruz... görünür.

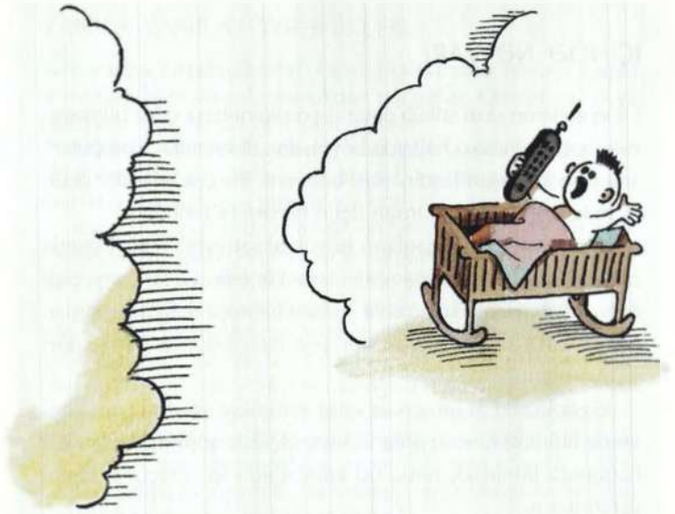
Uşaqlar oynaya-oynaya öyrənirlər. Və yalnız icraçılarla oyunda proqramlaşdırma üçün lazım olan əsas anlayışları bilib (komanda, komandalar sistemi, informasiya obyektinin adı, budaqlanmalar, təkrarlar) əvvəl təhsil, sonra isə professional proqramlaşdırma dillərinin mənimsənilməsinə başlamaq olar.

PROQRAMLAŞDIRMA İKİNCİ SAVADDIR



İndi Azərbaycan məktəbində informatika kursunun məzmunu bu kursun daxil edildiyi vaxtdan bu günə qədər əsaslı surətdə dəyişilmişdir: məktəbliləri populyar Basic və Pascal dillərindən birində proqramlaşdırma vasitəsilə alqoritmləşdirmənin məcburi şəkildə öyrədilməsindən katibə-makinaçı yetişdirmək üçün Microsoft Word ilə mətnin redaktəsinin öyrədilməsinədək. Özünün (və ya başqasının) hərəkətlərini əvvəlcədən planlaşdırmaq bacarığı, proqram tərtib etmək (sonrakı işlərin planı) hər yerdə istifadə olunur. Məsələn, müəyyən vaxtda televizordan videomaqnitofona kinofilmi yazmaq üçün əvvəlcə videomaqnitofonun proqramlaşdırma dilini göstərən istismar təlimatını öyrənmək lazımdır:





Yazı tarixini təyin etmək = bu gün;
Yazının başlanğıcını təyin etmək = 10.00;
Yazının sonunu təyin etmək = 11.25;
Telekanalın nömrəsini təyin etmək = 4;
Yazının sürətini təyin etmək = LP.

Doğrudan da, videomaqnitofonun proqramlaşdırılması o qədər də mürəkkəb proses deyil. Lakin müasir proqramlaşdırıla bilən ev cihazlarının sayı günbəgün artır. Bu həm nömrələrin yadda saxlanması proqramlaşdırılan, müxtəlif parametrlərin qoyulduğu mobil telefonlardır, həm də mikrodalğalı sobalardır, mətbəx qəhvə dəmləyənləri, çörək sobalarıdır və s. Kompüter də belə ev cihazlarındandır, ancaq daha mürəkkəb və universaldır.



Universal proqramlaşdırma dilindən istifadə edərək beyində və ya kağızda proqram tərtib etmək olar. Proqramı kompüter, insan və ya dili başa düşən və *icraçıları* komandaları yerinə yetirməyə məcbur edə bilən abstrakt *universal işgörən* yerinə yetirə bilər. Ana uşağa nənəsinə zəng etməyi belə öyrədir:

"2 klavişini bas,
3 klavişini bas,
2 klavişini bas..."

Burada uşaq proqramın işgöreni, telefon isə – icraçısıdır.

Riyaziyyat dərslərində də proqramların yerinə yetirilməsi baş verir, məsələn, $ax^2+bx+c=0$ kvadrat tənliyinin həlli üçün $D=b^2-4ac$ diskriminantını hesablamaq lazımdır:

əgər $D<0$, *onda* köklər yoxdur,
əgər $D>0$, *onda* köklər üst-üstə düşür,
əgər...

Qol saatınızda düymələrin sayı noutbukunuzun klaviaturasındakından az deyilsə, siz əsl proqramçısınız.

Konserv bıçağınızda USB bağlayıcısı varsa və ondan MP3-pleyer kimi istifadə edirsinizsə, siz anadangəlmə proqramçısınız.

İÇİNDƏ NƏ VAR?

Uşaq elektron saatı söküb onun iş prinsipini başa düşə bilməyəcək (həmçinin bəzi hallarda böyük də). Beləcə də "kompüter" adlı obyektin əsatirləşdirilməsi baş verir. Bir çox adamlar üçün o, möcüzədir. Və bu, insan üçün tamamilə təbiidir.

Digər tərəfdən, uşaqların böyük əksəriyyəti 9-10 yaşında mexaniki saatın necə işləməsini təsəvvür edir. Onlara yay, dişli çarx, əqrəb haqqında müxtəlif yollarla bir neçə dəfə danışmışlar. Bəziləri isə saatı sökərək hər şeyi öz gözlərilə görmüşlər.

Nə alınır?

Əgər biz iki əqrəb çıxan səthi görürüksə və dişli çarx haqqında biliriksə (ancaq onların sayı, ölçüsü, qoşulma ardıcıllığı haqqında bilmirik), onda bu kifayət edir ki, möcüzə effekti yaranmasın.

Məlum nəticəni çıxaraq.

İstənilən obyekt üzərindən əsatir pərdəsini götürmək üçün onun haqqında (məsələn, elektron saat və ya kompüter haqqında) çox da danışmaq lazım deyildir. Mütləq kompüterin quruluş prinsipini və onun üçün proqram tərtibini izah etmək lazımdır (və yalnız iş əsnasında kompüteri şəbəkəyə qoşmağı öyrətmək). Başqa sözlə, informatika kursunda saatı necə qurmağı göstərmək və onun cıqqıltısına qulaq asmaq yox, dişli çarx haqqında danışmaq lazımdır.



Bu halda şagird mahiyyətcə öz komandalar yığını ilə birlikdə icraçıdır.

Kompüter çox vaxt özündə bütöv bir icraçı dəstini birləşdirdiyi və özü də proqramın işgörəni olduğu kimi şagird də tez-tez müxtəlif proqramlaşdırma dillərində yazılan proqramların işgörəni rolunda çıxış etməli olur. Həqiqətən, müəllimlər riyaziyyat və fizika dərslərində, biologiya və kimyadan laboratoriya məşğələlərində proqramların ifadə edildiyi dili (hərəkətlər ardıcıllığını) heç vaxt formallaşdırmırlar. Şagirdlər bu və ya digər dərəcədə onu başa düşməyi öyrənirlər.

İcra keyfiyyəti bəzən şagirdin bacarıq və fərasətindən yox, ifadə dilini formallaşdırmağı bacaran və şagirdlərə onu öyrədən pedaqoqun talant və təcrübəsindən asılı olur. Bax burada alqoritmləşdirmə vərdişləri kara gələrdi! Əlbəttə, müəllimlər onları bir kəlmədən başa düşən və bütün tapşırıqları yerinə yetirən, yəni proqramlaşdırma dilini anlayan və proqramları icra edən şagirdləri ideal hesab edir.

Dilin uğurlu formallaşdırma örnəkləri mövcuddur. Hərbçilərin nizamnaməsində hər şey yazılmışdır – idarəetmə dilindən (hərbçilərin proqramlaşdırılması) proqrama (əsgər və zabitlərin davranışı) qədər. Demək olar ki, "dəftərxana" dili formallaşdırılıb. Müəssisələrdə bu dildə yazışma aparılır, əmr və göstərişlər verilir.

Uşağın ibtidai siniflərdə aldığı biliyin həcmi imkan verməlidir ki, o bütün məktəb fənləri üzrə proqramları şüurlu surətdə yerinə yetirə bilsin. Riyaziyyatın və başqa fənlərin gücü, mürəkkəb məişət texnikasının istismar təlimatının gücü məhz proqramlaşdırma dilini təsvir etmədən insandan işgörən kimi istifadə edən metodik üsullardadır.

Orta sinif məktəblisi üçün universal işgörənin dilində proqram tərtib etməyi öyrənmək əhəmiyyətlidir. Fəndən fənnə

icraçılar qrupu dəyişir – hesabi əməliyyatlardan kimyəvi və ya fiziki təcrübələrədək. Şagird müəllimlərin proqram işgörəni kimi müxtəlif dillərdə proqramları başa düşmək və yaratmaq iqtidarındadır.

Yuxarı siniflərdə məsələlərin çətinliyi artır, inkişaf etmiş struktura malik dillər və mürəkkəb işgörənlər meydana gəlir. Bu müəyyən proqramlaşdırma texnologiyasını (proqram tərtibini) öyrənməyi tələb edir. Nəticədə şagird konkret məsələnin həlli üçün vacib olan işgörən və icraçının proqramlaşdırma dilini layihələndirməyi bacarmalıdır. Bu artıq gündəliyin doldurulması və istifadəsi deyil, belə gündəliyin layihələndirilməsidir.

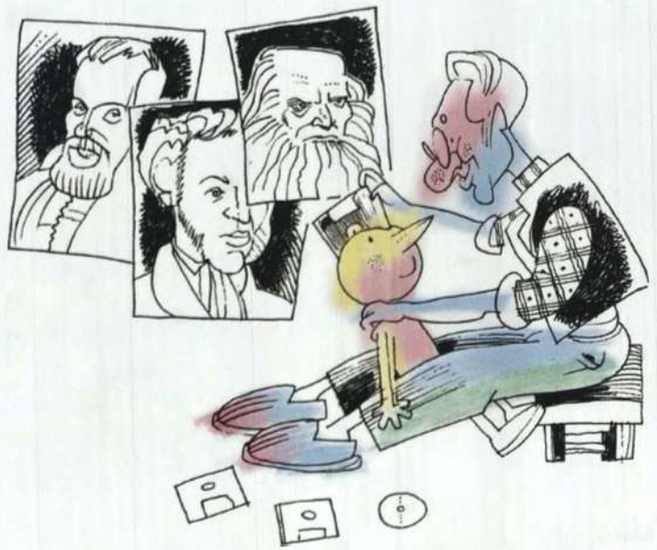
YANILA-YANILA ÖYRƏNİRLƏR...

Səhvə görə cəzalandırırlar. Ona görə də səhv etməyi nəinki sevmirlər, həm də səhv etməkdən qorxurlar. Qorxudan isə ya fəaliyyətsizlik, ya da yeni səhv yaranır.

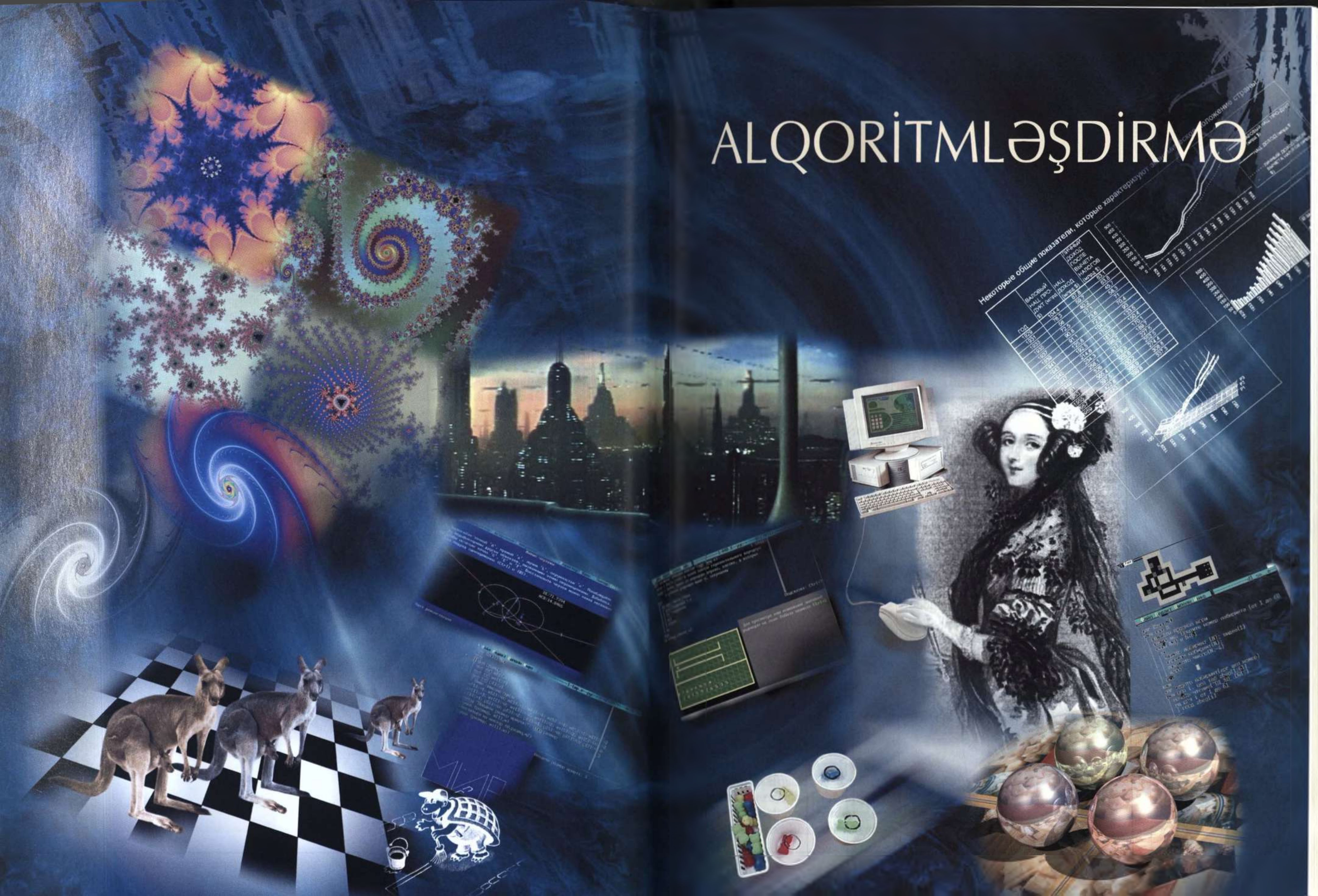
Lakin biz bilirik elə adamlar var ki, işə bu fikirlə başlayırlar: "İndi mən hər şeyi səhv yazacağam". Və onlar bundan qorxmurlar. Bu adamlar – proqramçılardır.

İnformatikanı öyrətmə prosesində təkcə səhv etməkdən qorxmamağı deyil, həm də səhvlərə sevinməyi öyrənmək olar, çünki hər bir tapılan səhv hələ tapılmamış səhvlərin sayını azaldır.

Səhv mövzusunu davam edərək, qeyd edək ki (əgər məşğələ kompüterdə işləməklə müşayiət edilirsə), iş əsnasında öz proqramını düzgün və tez sazlamağı (başqa sözlə, "simptomlara görə diaqnoz qoymağı") öyrənmək olar. Bu isə müstəntiq, həkim, siyasətçi, jurnalist, geokəşfiyyatçılara və başqalarına vacibdir, çünki proqramda səhvlərin axtarışı qəzanın səbəbini müəyyən etməyə və onu təkrar-təkrar yoxlamağa imkan verir. Lakin bunun tədrisi alqoritmik düşüncənin öyrədilməsilə heç cür kəsişməməli olan ayrıca bir məsələdir.



ALQORİTMLƏŞDİRMƏ





ALQORİTM

ALQORİTM VƏ PROQRAMLAR

ALQORİTM ANLAYIŞI

Hər bir insan öz gündəlik həyatında, təhsilində və ya işində həll edilməsi vacib olan müxtəlif çətinlikli çoxlu sayda məsələlərlə qarşılaşır. Onlardan bəziləri o dərəcədə çətin olur ki, onların həllini tapmaq üçün (bəzi hallarda isə həlli tapmaq mümkün olmur) uzun müddət düşünmək tələb olunur. Digərləri isə əksinə, o qədər sadə və adi olur ki, onlar



avtomatik həll edilir. "İnstituta qəbul olmaq" kimi bir məsələnin həlli "Dondurma almaq" kimi bir məsələnin həllindən yetərincə çətindir və onun həlli üçün daha çox sayda çətin əməliyyatların yerinə yetirilməsi tələb olunur. Bu zaman, hətta ən sadə məsələnin həlli belə adətən bir neçə ardıcıl mərhələ və ya *addımda* həyata keçirilir. Əgər bu məsələlərin həlli yollarını analiz etsək, görərik ki, məsələnin həlli üçün yerinə yetirilməsi lazım olan hər bir əməliyyat, dəqiq şəkildə ifadə olunmalı və yazılmalıdır.

Məsələn, nahar hazırlanması prosesini belə təsvir etmək olar:

1) şorba bişirmək;
2) ikinci yeməyi hazırlamaq;
3) desert hazırlamaq.

Belə bir yazılış (bunu *təlimat* adlandıraq) praktika üçün çox az əhəmiyyətlidir. Aydındır ki, addımlar daha sadə olmalıdır, ona görə də onları bir qədər ətraflı

izah etmək lazımdır (hələlik birinci bəndlə kifayətlənək):

1) tərəvəzləri yumaq, təmizləmək və doğramaq;
2) pilətəni yandırmaq;
3) qazanda suyu qaynatmaq;
4) əti qaynanmış suya tökmək;
6) ət bişdikdən sonra üzərinə tərəvəzləri tökmək;
7) tərəvəzlər bişdikdən sonra pilətəni söndürmək.

Burada göstərilən əməliyyatlar ardıcıllığına əməl etməklə hər bir insan şorba hazırlaya bilər. Buna analoji olaraq orta məktəb riyaziyyat kursundan məlum olan bir çox məsələlərin həlli prosesini addımlar ardıcıllığı şəklində təsvir etmək olar: kəsirlərin ortaq məxrəcə gətirilməsi, xətti tənliklər sisteminin məhsulların ardıcıl yox edilməsi üsulu ilə həlli, üçbucağın verilmiş üç tərəfinə görə pərgar və xətkeş vasitəsilə qurulması və s. Məsələnin həllində addımların bu cür ardıcıllığı *alqoritm* adlanır.

Hər bir əməliyyat ayrılıqda – *alqoritmin addımıdır* ("tərəvəzləri təmizləmək", "*a*-nı *b*-yə vurmaq" və s.). Alqoritmin addımlarının ardıcıllığı ciddi qeyd olunur, başqa sözlə addımlar *nizamlı* olmalıdır. Düzdür, bu qaydanın gözlənilmədiyi *paralel alqoritmlər* də mövcuddur. Belə alqoritmlərdə addımların yazılış ardıcıllığının uyğun əməliyyatların yerinə yetirilməsi ardıcıllığıyla üst-üstə düşməsi vacib deyildir.

Alqoritm anlayışı *üsul* (xətti tənliklər sisteminin həlli üçün Qaus üsulu), *qayda* (pərgar və xətkeş vasitəsilə üç tərəfinə görə üçbucağın qurulması qaydası) və hətta *resept* kimi anlayışlara yaxındır. *Prosedur* (məhkəmə proseduru), *proqram* (hökumətin iqtisadiyyatı böhrandan çıxarmaq üçün fəaliyyət proqramı), *proses* (Yaxın Şərqdə sülh prosesi) kimi digər anlayışlar da alqoritmlə çoxlu oxşar xüsusiyyətlərə malikdir. Buna görə də məhz alqoritmlərin fərqli xüsusiyyətlərini dəqiq ifadə etmək lazımdır.

Hər şeydən əvvəl, alqoritm giriş verilənlərinin və müəyyən nəticələrin olmasını nəzərdə tutur. Deməli, alqoritm – dəqiq müəyyən edilmiş təlimatdır ki, onu giriş verilənlərinə ardıcıl tətbiq edərək məsələnin həllini almaq olar. Bir çox alqoritmlər elə xassəyə malikdir ki, müxtəlif giriş verilənləri üçün dəyişməz qalır (şorbanın hazırlanma qaydası istifadə edilən müxtəlif tərəvəzlər üçün eynidir; düzbucaqlının sahəsinin hesablanması qaydası onun *a* və *b* tərəflərinin istənilən qiymətlərində eynidir). Alqoritmin bu cür xassəsini *kütləvilik* adlandırırlar. Bu xassə məsələnin həlli üçün eyni bir alqoritmi çoxlu sayda tətbiq etməyə imkan verir. Bu zaman nəzərə almaq lazımdır ki, hər bir alqoritm üçün mümkün giriş verilənləri kimi götürülə bilən obyektlər çoxluğu mövcuddur. Məsələn, *a* və *b* yalnız həqiqi müsbət ədədlər ola bilər; həqiqi ədədlərin bölünmə alqoritmində bölünən istənilən ədəd ola bilər, bölən isə sıfıra bərabər olmamalıdır.

Alqoritmi mümkün giriş verilənləri çoxluğundan olan müəyyən verilənlər yığınına tətbiq etməklə həmişə nəticə almaq mümkündürmü? Həmişə mümkün olmur, belə ki, 7-ni 3-ə bölərkən hər





iki tam ədəd mümkün olan ədəddir, amma nəticə 2,333... sonsuz onluq kəsirdir. Bu növ alqoritmlər üçün deyilir ki, onlar *potensial həll edilə biləndir*, amma sonsuz sayda addımlardan ibarətdirlər.

Bununla əlaqədar olaraq alqoritmin sonluğu anlayışını daxil etmək lazımdır. Onun mənası bundan ibarətdir ki, alqoritmin yerinə yetirilməsi mütləq onu sona çatdırmalıdır. Eyni zamanda, praktikada geniş şəkildə tətbiq edilən sonsuz alqoritmlərə nümunələr göstərmək olar. Məsələn, meteoroloji məlumatların toplanması sisteminin iş alqoritmi bu sistemin fəaliyyət dövrü ərzində müəyyən tezliklə fasiləsiz (hər dəqiqə, hər saat və ya hər gün) təkrarlanan "Havanın temperaturunu ölçmək" və "Atmosfer təzyiqini müəyyən etmək" kimi əməliyyatlar ardıcıllığından ibarətdir.

Praktikada adətən, alqoritmin sonlu vaxt ərzində sonlu sayda addımlardan sonra verdiyi nəticə maraq doğurur (doğrudan da, sabaha olan hava proqnozunu bir həftəyə hesablayan alqoritmin nə faydası var). Sonluq xassəsi xüsusi əhəmiyyət kəsb edir, çünki tez-tez alqoritmin potensial deyil, məhz *real* yerinə yetirilməsinin mümkünlüyü tələb olunur.



Bölmə misalına qayıdaq. Əgər prosesi hər hansı bir addımda dayandırsaq, onda alınan ədəd, ümumiyyətlə, qismət olmayacaq, bununla belə onu konkret məsələ üçün müəyyən dəqiqliklə təqribi nəticə kimi qəbul etmək olar.

Alqoritm anlayışı ilə *alqoritmin icraçısı* haqqında təsəvvür sıx surətdə bağlıdır. Bu rolda kimin çıxış etməsi vacib deyildir: insan (aşpaz, abituriyent və ya yuxarıda göstərilən nümunələrdəki riyaziyyatçı) və yaxud müəyyən texniki qurğu (o cümlədən kompüter). Əsas şərt – alqoritmin icraçısı onun hər bir addımını yerinə yetirməyi bacarmalıdır.

Buradan da alqoritmin növbəti vacib xassəsi meydana çıxır – *müəyyənlik*. Alqoritmin hər bir addımında, bu addımı yerinə yetirmək üçün icraçının kifayət qədər məlumatı olmalıdır. Bundan başqa, alqoritmin icraçısı onun hansı yolla yerinə yetirildiyini dəqiq bilməlidir. Asanlıqla görmək olar ki, yuxarıda göstərilən şorba hazırlanması alqoritminin bəzi addımlarının da öz növbəsində dəqiqləşdirilməsinə ehtiyac vardır (məsələn, "Qaz pilətəsini yandırmaq" bəndi, "Kibrit qutusunu götürmək", "Kibrit dənəsini yandırmaq", "Qaz pilətəsinin ventilini burmaq", "Qazı yandırmaq" kimi əməliyyatlar ardıcıllığının yerinə yetirilməsini tələb edir). Deməli, müəyyənlik xassəsinin yerinə yetirilməsi üçün təlimatdakı addımlar kifayət qədər sadə, *elementar* olmalı, icraçı isə alqoritmi təşkil edən əməliyyatlar ardıcıllığında hər bir addımın məğzini *birmənalı* şəkildə başa düşməlidir (düzbucaqlının sahəsini hesablayarkən alqoritmi yerinə yetirən hər bir icraçı mütləq vurmanı bilməli, eləcə də "x" əməl işarəsini toplama kimi yox, vurma kimi başa düşməlidir). Burada vacib bir problem ortaya çıxır: bu şərtin yerinə yetirilməsi üçün alqoritm *hansı şəkildə, hansı formada* göstərilməlidir.

İLK ALQORİTMLƏR

Riyazi alqoritmlərdən bəziləri artıq neçə minilliklərdir ki, məlumdur. Belə hesab edilir ki, trivial (adi) olmayan ən qədim alqoritm iki tam ədədin ən böyük ortaq böləninin tapılması üsuludur. O, təxminən 2300 il əvvəl kəşf edilmiş və Evklid alqoritmi kimi tanınmışdır. Bu alqoritm qədim yunan riyaziyyatçısının "Başlanğıclar" əsərinin yeddinci kitabında əks olunmuşdur (düzdür, belə fərziyyə var ki, Evklid alqoritmi Evdoksun 75 il ondan əvvəl təklif etdiyi alqoritmin interpretasiyasıdır).

Evklid alqoritmi iki müsbət, tam ədədin ən böyük ortaq böləninin (ƏBOB-un) tapılmasından ibarətdir. "Tutaq ki, *A* və *C* verilmiş iki müsbət tam ədədlərdir. Onların ən böyük ortaq bölənini tapmaq lazımdır. Əgər *C* ədədi *A*-nı bölürsə, onda *C* ədədi *A* və *C* ədədlərinin ortaq bölənidir, çünki *C* özünə də bölünür. Aydındır ki, *C* ən böyük ortaq bölən olacaq, çünki *C*-dən böyük olan və *C*-ni bölən başqa ədəd yoxdur. Əgər *C* *A*-nı bölmürsə, onda ardıcıl olaraq onların ən böyüyündən kiçiyini çıxırıq. Bu əməliyyat o vaxta qədər davam edir ki, alınan ədəd əvvəlki çıxılanı bölsün. Gec-tez belə bir ədəd alınacaqdır, çünki əgər fərq 1-ə bərabər olarsa, onda 1 əvvəlki çıxılanı bölməkdir.

Tutaq ki, *E* *A*-nın *C*-yə bölünməsindən alınan müsbət qalıqdır. *F* isə *C*-nin *E*-yə bölünməsindən alınan müsbət qalıqdır. Tutaq ki, *F* *E*-ni bölür. *F* *E*-ni böldüyü üçün və *E* də (*C* – *F*)-i böldüyü üçün *F* (*C* – *F*) fərqini də bölür, lakin *F* öz-özünü böldüyü üçün buradan alınır ki, *F* *C*-ni bölür. Lakin *C* (*A* – *E*)-ni böldüyü üçün *F* həm də (*A* – *E*)-ni bölür. O həm də *E*-ni bölür, buna görə də o həm də *A*-nı bölür. Buradan alınır ki, *A* və *C* ədədlərinin ortaq böləni *F*-dir.

İndi göstərək ki, o, həm də ən böyük ortaq böləndir. Doğrudan da, əgər *F* *A* və *C* ədədlərinin ən böyük ortaq böləni deyilsə, onda F-dən böyük bir ədəd onların hər ikisini bölməcək. Bu ədədi *G* ilə işarə edək.

Bir halda ki, *G* *C*-ni bölür, *C* isə öz növbəsində (*A* – *E*)-ni bölür, onda *G* də (*A* – *E*)-ni bölür. *G* həm də *A*-nı bölür, nəticə etibarilə o, *E* qalığını da bölür. Lakin *E* (*C* – *F*)-i bölür. Ona görə də, *G* (*C* – *F*)-i bölür. *G* həm də *C*-ni bölür və deməli o, *F* qalığını da bölür. Yəni böyük ədəd kiçiyi bölür. Bu isə mümkün deyil.

Buna görə də *F*-dən böyük olan elə bir ədəd yoxdur ki, *A* və *C* ədədlərini bölsün. Deməli, *F* onların ən böyük ortaq bölənidir".

Qədim Misir papiruslarında hesablama prosedurlarının daha qədim nümunələrini tapmaq olar. Belə ki, iki dəfə artırmağa və toplamaya əsaslanan vurma üsulu misirlilərə məlum idi (yeri gəlmişkən, geniş istifadə olunan bəzi müasir alqoritmlər də bu ideyaya əsaslanır). Qədim Misirdən fərqli olaraq Babilistanda məsələnin həllindən əlavə ümumi şəkildə yararlı olan hesablama prosesinin izahının verilməsi də qəbul olunmuşdu. Məsələn, əgər vuruqlardan biri vahidə bərabər olanda vurma yenə də yerinə yetirilirdi. Babilistanlılar yunanlardan 1500 il əvvəl alqoritmlərdən istifadə etdiklərinə baxmayaraq, tarixçi alimlər Evklidin öncül olmasını yekdilliklə qəbul edirlər. Evklid alqoritmi kifayət qədər mürəkkəb quruluşa malikdir: o, *iterasiyalı* alqoritmdir (bəzi əməliyyatları bir neçə dəfə təkrarlanan alqoritmləri belə adlandırırlar). Babilistan alqoritmləri yalnız tarixi maraq daşıyır, lakin Evklid alqoritmi indiyə qədər öz praktiki əhəmiyyətini itirməmişdir.



Evdoks.

Praktikada geniş istifadə edilmiş daha bir qədim alqoritm: **Əl-Kaşi alqoritmi**. x^n qiymətinin hesablanması, burada *n* – müsbət ədəddir.

Addım 1. Üç kəmiyyət daxil edirik – *N*:= *n*; *y* := 1; *z*: = *x*.
Bu halda $x^n = y \times z^n$ münasibəti doğru olur.
Addım 2. *N*-i tam olaraq 2-yə bölək. *N*:=*N div 2*; eyni zamanda *N*-in qiymətinin bundan əvvəl cüt olub-olmamasını müəyyənləşdiririk.
Əgər N cütdürsə, 5-ci addıma keçirik.
Addım 3. *y*-i *z*-ə vururuq, *y*:=*y* × *z*.
Addım 4. Əgər *N* sıfıra bərabərdirsə, onda cavab *y*-ə bərabərdir.
Addım 5. *z*-i öz-özünə vururuq, *z*:=*z* × *z*.
2-ci addıma qayıdırıq.

Bu alqoritm XV əsrin əvvəllərində təklif edilmişdir (ancaq eramızdan təxminən 200 il əvvəl "Çanda-sutra" adlı hind riyazi traktatında analoji üsulun təsviri verilmişdir). Məhz belə prosedurdan Misir riyaziyyatçıları hələ eramızdan əvvəl XVIII əsrdə vurma üçün istifadə etmişlər.

Addım 1. Üç kəmiyyət daxil edək: – *N*: = *n*; *y*: = 0; *z*: = *x*
Addım 2. *N*-i tam olaraq 2-yə bölürük. *N*: = *N* div 2; eyni zamanda *N*-in bundan əvvəl cüt olub-olmamasını müəyyənləşdiririk.
Əgər *N* cütdürsə, 5-ci addıma keçirik.
Addım 3. *y*-i *z* qədər artırırıq *y*: =*y* + *z*.
Addım 4. Əgər *N* sıfıra bərabərdirsə, onda cavab *y*-ə bərabər olacaq.
Addım 5. *z*-i öz-özüylə toplayırıq *z*: =*z* + *z*.
2-ci addıma qayıdırıq.

3-cü və 5-ci addımlarda vurma əməli toplamayla əvəz edilmişdir və 1-ci addımda *y* 1-ə deyil sıfra bərabər olur. Bu cür dəyişdirilmiş alqoritmin yerinə yetirilməsi nəticəsində iki *n* və x ədədin hasilini alırıq: $y = n \times x$.

Praktiki olaraq "əllə" hesablamalar üçün bu son dərəcə əlverişli vurma üsuludur ki, bu üsul daha sadə 2-yə vurma, yarıya bölmə və toplama əməllərinə gətirilir. Məhz bu cür üsuldan çötkədə hesablama zamanı istifadə olunur – Avropada o ənənəvi olaraq "rus kəndli üsulu" adını daşıyır.



Urben Jan Jozef Leverye.

Faktiki olaraq, söhbət alqoritmin hansı *dildə* yazılmasından gedir.

Alqoritmin qəbulunun birqiymətli olmaması çox zaman onun mahiyyətiylə deyil, təsvir edilmə xüsusiyyətlərilə bağlı ola bilər. Hətta ətraflı təlimatda belə dəqiq müəyyənləşdirilməmişdir ki, "ət bişdi" və ya "tərəvəzlər bişdi" nə deməkdir. Alqoritmin bu və ya digər icraçısının xüsusiyyətlərindən (tutaq ki, aşbazın təcrübəsinə görə) asılı olaraq bu addımlar müxtəlif şəkildə yerinə yetirilə bilər. Məsələn, ət alaçiy bişmiş ola bilər, tərəvəzlər isə əksinə, həddən artıq bişmiş ola bilər. Beləliklə, alqoritmin hər addımı elə təsvir olunmalıdır ki, fərqli mənalar ortaya çıxmasın.

Alqoritmin müəyyənlik xassəsi daha vacib xassə olan *determinantlıq* xassəsilə əlaqədardır. Determinantlıq xassəsini belə başa düşmək olar: əgər alqoritm təkrar-təkrar eyni giriş verilənləri yığımına tətbiq edilərsə, onda o həmişə eyni nəticə verməlidir.

Beləliklə, alqoritm – alqoritmin icraçısına aydın olan bir dildə təsvir edilmiş qaydalar sistemidir. Bu qaydalar sistemi mümkün ilkin verilənlərdən müəyyən nəticəyə keçid prosesini müəyyən edir və kütləvilik, sonluluq, müəyyənlik və determinantlıq xassələrinə malikdir.

Alqoritmin bu cür tərifi ciddi deyildir (heç olmasa ona görə ki, onda dəqiq müəyyən olunmamış terminlərdən istifadə edilir, məsələn, "qayda"dan).

Əsrlər boyu alqoritm anlayışı ədədlərlə və onlar üzərindəki nisbətən sadə əməliyyatlarla bağlı olmuşdur. Elə riyaziyyatın özü daha çox hesablamalar haqqında elm, tətbiqi elm idi. Alqoritmlər daha çox riyazi düsturlar şəklində təsvir olunurdu. Alqoritmin elementar addımlarının ardıcıllığı mötərizələrin qoyulması ilə verilirdi. Addımların özləri isə hesablama əməliyyatlarının və müqayisə əməliyyatlarının yerinə yetirilməsindən ibarət olurdu (bərabərliyin, bərabərsizliyin yoxlanılması və s.). Düsturlar kifayət qədər çox yer tuturdu, əl hesablamaları isə son dərəcə çətin idi (fransız astronomu Jan Jozef Leverye ömrünün on ilini Günəş sistemi planetlərinin orbitlərinin hesablanmasına sərf etmiş, nəticədə əvvəllər naməlum olan Neptun planetini kəşf etmək ona nəsib olmuşdur), ancaq hesablama prosesinin özünün mahiyyəti tamamilə aydın qalmışdı. Riyaziyyatçılarda alqoritmin başa düşülməsinə və ciddi şəkildə təyin olunmasına, ümumiləşdirilməsinə tələbat yaranmamışdı.

Lakin riyaziyyatın inkişafı ilə yeni anlayışlar meydana gəlirdi ki, alimlər onlardan istifadə etməli olurdu: vektorlar, matrislər, qraflar, çoxluqlar və başqaları. Onların birmənalı olmasını necə müəyyən etməli və ya alqoritmin sonluluğunu necə təyin etməli, hansı addımlar elementar sayıla bilər (məsələn, matrisin tərsinin və ya iki çoxluğun kəsişməsinin tapılması elementar sayıla bilərmi)?

Alqoritmik həll edilə bilməyən, yəni həll prosedurunun tapılması mümkün olmayan problemlərin mövcudluğu ideyası meydana çıxdı. Deməli, müvafiq alqoritmin yoxluğu faktının *riyazi baxımdan ciddi isbatının* verilməsini öyrənmək lazım idi. Bunu yalnız alqoritmin ciddi tərifinin verildiyi halda etmək olar. Belə tərifin verilməsi cəhdləri XX əsrin 20-30-cu illərində *alqoritmlər nəzəriyyəsinin* meydana gəlməsinə gətirib çıxardı.

Demək olar ki, həmin vaxtlar mürəkkəb hesablamaların aparılmasına praktiki tələbatı da yarandı. Elmi və mühəndis sahələri üzrə müxtəlif məsələlərin həllinin tapılması zərurəti gündəmə gəldi: radiolokasiyalar, təyyarəqayırma, fiziki proseslərin modelləşdirilməsi (xüsusilə, nüvə fizikası, ballistika və bir çox başqa sahələrdə). İnsanın hesablama imkanları olduqca məhdud idi, mexanikləşdirmə, xüsusilə də hesablamanın avtomatlaşdırılması vasitələri praktiki olaraq mövcud deyildi. Hesablama maşınlarının ilk layihələrinin meydana çıxması alqoritmlərin *praktiki* olaraq tətbiq olunma imkanlarını artırmaq üçün stimul yaratdı. Belə alqoritmlərin mürəkkəb olması səbəbindən onların qabaqlar istifadəsi mümkün deyildi. Hesablama texnikasının sonrakı sürətli inkişafı alqoritmlərin öyrənilməsinin həm nəzəri, həm də praktiki aspektlərinin inkişafını müəyyən etdi.

ALQORİTMİN DAHA BİR FORMAL TƏSVİRİ



V.A.Uspenski.

Alqoritm əməliyyatların təsviridir; alqoritmin təkcə yazılışını öyrənməklə onun dinamikasını başa düşmək olmaz. Belə öyrənilmə zamanı alqoritmin yerinə yetirilməsinin nəticəsi heç də həmişə aydın olmur.

Kolmoqorov-Uspenski sxemi alqoritmlərin formal yerinə yetirilmə mexanizmlərinin qurulması cəhdlərindən biri idi. Bu sxemdə alqoritm və onun icraçısı – insan və ya hər hansı bir qurğu vəhdətdə baxılır. Burada icraçının hərəkətləri dəqiq müəyyənləşdirilmişdir, onların sayı isə məhduddur.

Nəzərdə tutulur ki, insan tərəfindən emal olunan məlumat kağız vərəqlərinə yazılır, bu zaman qeyri-məhdud təmiz vərəq ehtiyatı və yazılmış vərəqlərin saxlanılması üçün qeyri-məhdud yer ehtiyatı vardır. İnsan tərəfindən gerçəkləşdirilən məlumatın emalı ayrı-ayrı diskret addımlara bölünür. Hər addımda insan yazılmış vərəqlərin sayını gözdən keçirir və yazılanların məzmunundan asılı olaraq onlarda verilmiş ciddi qaydalar üzrə dəyişikliklər edir. Dəyişikliklər üç növ ola bilər:

– baxılan bütün məlumatın və ya onun bir hissəsinin silinməsi;
– baxılan vərəqlərə yeni məlumatların yazılması;
– baxılan vərəqlər yığımının dəyişdirilməsi.

Buna baxmayaraq, verilən model kifayət qədər formallaşdırılmamışdı və onun konkret alqoritmlərin reallaşdırılmasına tətbiqi çətindir. Bu sxem Post və Türinqin maşınlarına çox oxşar olmasına baxmayaraq, rahatlığına görə onlardan xeyli geri qalır.

ALQORİTMLƏR NECƏ YAZILMALIDIR

İki müsbət tam ədədin ən böyük ortaq böləninin tapılması üçün Evklid alqoritminin təsviri üç addımdan ibarətdir və cəmi bir neçə sətir yer tutur:

Addım 1. m-i n-ə bölək, burada $m > n$. Tutaq ki, p – bölmədən alınan qalıqdır.

Addım 2. Əgər p sıfıra bərabərdirsə, onda n axtarılan ən böyük ortaq böləndir.

Addım 3. Əgər p sıfıra bərabər deyilsə, onda m-i n ilə, n-i isə p ilə əvəz edərək birinci addıma qayıdırıq.

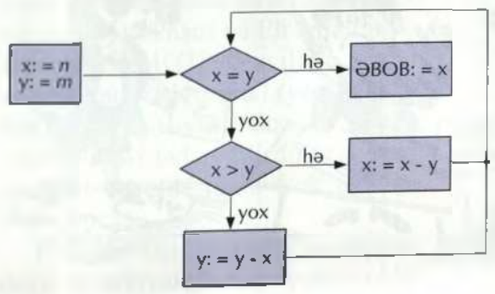
Bununla belə hətta Evklid tərəfindən verilmiş, kifayət qədər "sadələşdirilmiş" təsvir təxminən bir səhifəlik mətndir (!) və onun təsvirindən məhz nəyin və nə üçün edildiyini başa düşmək heç də asan deyildir.



Evklid.

Alqoritmlərin yazılışı üçün müəyyən bir dil gərəkdir. Bu zaman, məhz hansı dilin seçilməsi son dərəcə vacibdir. Azərbaycan dilinin (eləcə də, istənilən başqa bir təbii dilin) bu məqsədlər üçün münasib olmaması aydındır. Sözlərin ən müxtəlif məna çalarları ola bilər, amma bütün mətnin hərtərəfli analizi olmadan onları başa düşmək bəzən sadəcə mümkün olmur. Alqoritmlərin qrafik üsulla, *bloksxem* şəklində yazılması geniş yayılmışdır. O, oxlarla birləşdirilmiş elementlər (bloklar) yığınından ibarətdir. Hər bir element alqoritmin bir hissəsidir. Bloksxemin elementləri iki formada olur: əməliyyatları (hərəkətləri) düzbucaqlılarla, şərtin yoxlanılmasını isə romblarla göstərirlər. Düzbucaqlılardan həmişə yalnız bir ox çıxır (lakin bir neçə ox girə bilər), romblardan isə iki ox çıxır. Birincisi "hə" sözü ilə, ikincisi isə "yox" sözü ilə işarə olunur (onlar şərtin yerinə yetirilib yetirilmədiyini göstərir).

Şəkildə ən böyük ortaq bölənin (ƏBOB) tapılması alqoritminin bloksxemi göstərilmişdir:



Belə üsul hazırlanmış alqoritmin analizi üçün, onun işinin məntiqi düzgünlüyünün yoxlanılması üçün əlverişlidir. Şəkildəki hər blok alqoritmin bir elementar addımına uyğun gəlir. Əgər yazılan alqoritm nisbətən mürəkkəbdirsə, onda belə bir sual ortaya çıxır: onun lazımsız təfərrüatlarından necə xilas olmaq olar?

Bunu alqoritmin addımlarını iriləşdirməklə etmək olar. Onda hər bir blok kifayət qədər çətin əməliyyatlardan ibarət olacaq və hətta tam alqoritmi (daha dəqiqi, altalqoritmi) təsvir edəcək. Məsələn, blok daha əvvəl yazılmış və artıq yoxlanılmış alqoritmi əvəz edə bilər. Əgər bu blokun işinin düzgünlüyünə inam varsa, onda onun təfərrüatlarına baxmamaq olar. Blokda *nə* etməyin lazım olduğu bilinirsə, amma *necə* etmək hələ məlum deyilsə, onda "iri bloklu" belə təsvir alqoritmin işlənib hazırlanmasının ilkin mərhələsində olduqca səmərəlidir. Blokun alqoritminin dəqiqləşdirilməsini daha sonra etmək olar.

Blok-sxemin cəmisi bir neçə tipli elementlərdən qurulması onların kifayət qədər sadə şəkildə kompüter proqramlarına çevrilməsinə və prosesin formallaşdırılmasına imkan yaradır.

Alqoritmlərin yazılışının başqa əyani üsulları da məlumdur: ilk növbədə A.A.Lyapunov tərəfindən təklif edilmiş operator sxemləri bu üsullardan biridir.



ƏBOB alqoritmini təsvir edən Lyapunov sxemi aşağıdakı kimidir.

$$A_1 \downarrow^{4} P_1(x=y) \uparrow^{1} A_2 \uparrow^{2} \downarrow^{1} P_2(x>y) \uparrow^{3} A_3 \uparrow^{3} \downarrow^{4} A_4 \uparrow^{4} \downarrow^{2} S$$

Burada, A_i vasitəsilə hesablamaları yerinə yetirən operatorlar işarə edilmişdir (məsələn; A_4 $y := y - x$ mənimsəməsinə uyğundur), P_i isə məntiqi operatorları göstərir (P_1 $x = y$ şərtinin yoxlanılmasına uyğundur). ↑ və ↓ oxları keçidi bildirir, onlara blok-sxemdəki oxun başlanğıcı və sonu uyğun gəlir. Əgər yoxlanma mənfi nəticə verərsə, onda ox üzrə keçid baş verir, əks halda növbəti operator yerinə yetirilir. S – dayanma operatorudur. Hesablayıcı operatorlara düzbucaqlılar, məntiqi operatorlara isə blok-sxemin rombları uyğun gəlir. Operator sxemləri blok-sxem formasında yazmaq asandır, lakin operator sxemlər o qədər də aydın deyildir. Bu isə onların praktikada istifadəsini çətinləşdirir.

İndi Evklid alqoritmini Pascal proqramlaşdırma dilində yazaq:

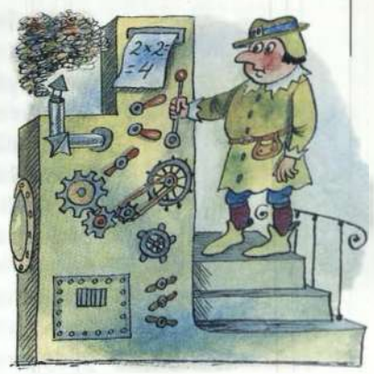
```
var
  n, m, x, y, ƏBOB: integer;
  begin
    x:=n; y:=m;
    while x<>y do
      if x>y then x:=x – y
             else y:=y – x;
    ƏBOB :=x;
  end;
```

Həm blok-sxem şəklində, həm də Lyapunovun operator sxemləri şəklində alqoritmlərin yazılışlarında oxşarlıq aşkardır. Əlbəttə, kompüter hələ də bu mətni birbaşa qəbul etməyə qadir deyil, onun xüsusi proqramın – translyatorların köməyilə ilkin emalı lazımdır. Buna baxmayaraq alqoritmin yazılışının başqa formaları ilə müqayisədə alqoritmin belə yazılışlarının reallaşdırılması əhəmiyyətli dərəcədə asandır. Bundan başqa, o, insan üçün də kifayət qədər rahatdır.

PROQRAM

"Proqram" sözünün bir neçə mənası vardır: fəaliyyət və ya iş planı, tədris fənninin məzmununun qısa şərhi, konsert və ya sirk tamaşasının məzmunu və başqaları. Onların arasında biri ayrıca olaraq xüsusi məna ilə seçilir: *proqram* – məsələnin həlli alqoritminin kompüter dilində təsviridir.

Məhz sözün bu mənası alqoritmlərin və proqramların sıx əlaqəyə malik olduğunu göstərir. Bu əlaqəyə ilk diqqət yetirənlərdən biri, 1945-ci ildə ingilis riyaziyyatçısı Alan Türinq olmuşdur. O, hesab edirdi ki, proqramlar (o, onları "təlimat cədvəlləri" adlandırırdı) hesablama işi təcrübəsinə malik olan və yaradıcılığa ("mürəkkəb məsələlərin həllinə") meylli olan riyaziyyatçılar tərəfindən yaradılmalıdır. Türinq məlum prosesin (başqa sözlə alqoritmin) təlimat cədvəlləri formasına çevrilməsinin əhəmiyyətini xüsusilə qeyd edirdi. Bu zaman, o, şübhə etmirdi ki, belə çevirmə – "heyranedici məşğuliyyətdir".



Ümumiyyətlə, demək olar ki, proqram – alqoritmin icraçıya aydın dildə yazılışıdır. Alqoritm icraçılarının əsas tipi – kompüterdir. Buna görə də, bir tərəfdən yaradıcılara alqoritmləri yazmağa imkan verən, digər tərəfdən də kompüterə yazılanı – *proqramlaşdırma dillərini* və ya *alqoritmik dilləri* başa düşmək imkanı verən xüsusi vasitələrin yaradılması vacibdir.

İlk proqramlaşdırma dillərindən biri olan Algol-60 ən əvvəl təkcə proqramların deyil, həm də alqoritmlərin yazılma vasitəsi kimi yaradılmışdı. Bunu dilin öz adı da göstərir: *algorithmic language* ingilis sözlərini "alqoritmik dil" ("proqramlaşdırma dili" sözlərinin sinonimi) kimi və "alqoritmlərin yazılması üçün dil" kimi başa düşmək olar. Bir çox illər ərzində Algol-60 alqoritmlərin yazılışı üçün faktiki standart olmuşdur.

"ALQORİTM" SÖZÜNÜN MƏNŞƏYİ HAQQINDA

Məlumdur ki, orta əsrlərin ilk dövrlərində "algorism" sözündən hesablama lövhələrindən (abaklardan) istifadə etmədən kağız üzərində arifmetik hesablamaları adlandırmaq üçün istifadə edirdilər. Məhz belə məna ilə o, bəzi Avropa dillərinə daxil oldu, məsələn, ingilis dilinin 1957-ci ildə nəşr edilmiş nüfuzlu "Webster's New World Dictionary" lüğətində o, "köhnəlmiş" işarəsiylə qeyd edilmiş və hesab əməliyyatlarının ərəb rəqəmlərinin köməyilə yerinə yetirilməsi qaydası kimi verilmişdir.

"Alqoritm" terminin nə vaxt yarandığının məlum olmasına baxmayaraq, dilçilər onun mənşəyini müxtəlif şəkildə şərh etməyə cəhd etmişlər. Bəziləri algorism sözünün *yun.* "alqiros" – "xəstə" və "arifmos" – "ədəd" sözlərindən əmələ gəldiyini deyiblər. Doğrusu, aydın deyil ki, niyə ədədlər "xəstədir?".

Başqaları alqoritm sözünü "King Algor of Castil" adlı hansısa əfsanəvi qədim ispan hökmdarı ilə əlaqələndirərək qeyri-adi izaha meyl edirdilər. Öz variantını Brokhauz və Efronun Ensiklopediya lüğəti (1890-cı il) də təqdim edir. Burada "alqorifm" (yeri gəlmişkən, inqilabdan əvvəl "alqoritm" yazılışından da istifadə edilirdi) ərəb sözü olan "Əl-Qorətm" (yəni "kök") sözündən alınır.

Həqiqəti ortaya çıxarmaq isə dilçilərə deyil, riyaziyyatçılara müyəssər oldu.



əl-Xarəzmi.

Onlar sübut etdilər ki, bu sözün mənşəyi böyük Orta Asiyalı alimin, IX əsrin birinci yarısında yaşamış, bir çox əsrlər ərzində məşhur olan riyaziyyat dərsliyinin müəllifi olan əl-Xarəzminin adı ilə bağlıdır. Latın transkripsiyasında onun adı Abu Abd Allah Muhammad ibn Musa al Khwarismi kimi yazılır və mənası "Abdullanın atası Musanın oğlu Məhəmməd Xarəzmli" deməkdir. Xarəzm – indiki Özbəkistan ərazisində yerləşən, paytaxtı qədim Xivə şəhəri olmuş tarixi vilayətdir.

Demək olar ki, bütün lüğətlər eyni fikirdədir ki, sözün başlanğıc forması "algorismi" olub və yalnız bir müddət keçəndən sonra bu söz son hərfini itirərək Avropa dillərində daha rahat tələffüz olunan "algorism" formasını alıb. Daha sonralar o da öz növbəsində dəfələrlə təhriflərə məruz qalaraq yunan mənşəli "arithmetic" sözüylə əlaqəli olmuşdur. Artıq yeni yazılışda bu söz: XVIII əsrdə, 1747-ci ildə Leypsiqdə nəşr edilmiş alman riyazi lüğətlərindən biri olan "Vollstandiges mathematisches Lexicon" lüğətində rast gəlinir. Algorithmus termini bu lüğətdə dörd hesab əməli haqqında anlayış kimi izah edilir. Belə izah yeganə deyildi, axı riyaziyyat elminin terminologiyası o zamanllar hələ yenicə formalaşırdı. Xüsusi halda, "algorithmus infinitesimalis" latın ifadəsi sonsuz kiçik kəmiyyətlərlə olan əməliyyatların yerinə yetirilməsi üsullarına tətbiq edilirdi.

"Alqoritm" sözünün bütün qədim mənaları tədricən sıradan çıxdı. XX əsrin başlanğıcında "alqoritm" sözü artıq "ciddi müəyyən olunmuş qaydalarla yerinə yetirilən hər cür hesabi və ya cəbri proses" mənasını verirdi. Böyük Sovet Ensiklopediyasında (1926-cı il) o, məhz bu cür izah edilir.

"Alqoritm" sözünün rus dilinə keçməsi prosesini izləmək olar. O, nə V.İ.Dalın "Толковый словарь живого великорусского языка" (1863-cü il; bax, orada "cəbr" sözü işlədilib), nə də D.N.Uşakovun redaksiyası ilə buraxılan "Толковый словарь русского языка" (1935-ci il) ("Rus dilinin izahlı lüğəti"ndə) yox idi. Lakin "alqorifm" sözü həm Qranat qardaşlarının inqilabdan əvvəlki məşhur ensiklopediya lüğətində, həm də artıq adını çəkdiyimiz Böyük Sovet Ensiklopediyasının birinci nəşrində istifadə olunub. Orada yazılır ki, orta əsrlərdə "dörd hesab əməlinin hər hansı birinin onluq say sistemində yerinə yetirildiyi qayda"nı alqoritm adlandırmışlar. "Riyaziyyat Ensiklopediyası" isə onun daha geniş izahını verir: təkcə mövqeli onluq say sistemində deyil, həm də elə onluq say sisteminin özündə hesablama məharəti kimi verilir.

Alqoritmin son dərəcə xüsusi və az əhəmiyyətli termin kimi qəbul edilməsini daha kiçik həcmli nəşrlərdə uyğun məqalələrin olmaması da təsdiq edir. O cümlədən, bircildlik ensiklopediya lüğətləri bir yana qalsın, heç oncildlik Kiçik Sovet Ensiklopediyasında (1957-ci il) da o, yoxdur. Ancaq 10 ildən sonra BSE-nin (1969-cu il) üçüncü nəşrində "alqoritm" artıq daha "sadə anlayışların terminləri vasitəsilə formal tərifə malik olmayan, lakin təcrübədən bilavasitə abstraktlaşdırılan əsas riyazi anlayışlardan (kateqoriyalardan) biri" kimi xarakterizə olunur. 40 il ərzində alqoritm yalnız mütəxəssislərin kiçik dairəsinə tanış olan anlayışdan riyaziyyatın əsas anlayışlarından birinə çevrildi və bu sözün ensiklopediyalara yox, lüğətlərə daxil edilməsi bunun təsdiqi oldu. Məsələn, o, məhz riyaziyyat sahəsinin termini kimi akademik "Rus dili lüğəti"nə (1981-ci il) salınmışdır.

Alqoritm anlayışının inkişafı ilə bir vaxtda tədricən onun riyaziyyatdan başqa sahələrə keçirilməsi də baş verirdi. Bunun başlanğıcını isə kompüterlərin yaranması qoydu və bunun nəticəsində "alqoritm" sözü yeni həyat qazandı. Məsələn, "Uşaq Ensiklopediyası"nın (1959-cu il) III cildində hesablama maşınlarından az danışılmır, ancaq onlar hələ sadəcə işıqlı gələcəyin bir atributu kimi qəbul edilir. Buna görə də alqoritmlər onun səhifələrində bir dəfə də olsun xatırlanmır. Ancaq keçən əsrin 70-ci illərinin başlanğıcında artıq kompüterlərin ekzotik əcayib şey olmağına son qoyulduqdan sonra "alqoritm" sözü gündəlik həyata sürətlə daxil olmağa başladı. "Kibernetika Ensiklopediyası"nda (1974-cü il) "Alqoritm" məqaləsində o, hesablama maşınlarında gerçəkləşdirmə ilə əlaqələndirilir, "Hərbi Ensiklopediya"da (1976-cı il) isə hətta ayrıca "Elektron hesablama maşınlarında məsələnin həlli alqoritmi" məqaləsi verilir.

XX əsrin son 15-20 ili ərzində kompüter həyatımızın ayrılmaz atributuna çevrilmiş, kompüter leksikası da hamı tərəfindən işlədilən olmuşdur. Bu gün "alqoritm" sözü hər kəsə tanışdır. O, danışıq dilinə elə qətiyyətlə keçmişdir ki, indi biz "davranış alqoritmi", "uğur alqoritmi", "xəyanət alqoritmi" kimi ibarələrə qəzet səhifələrində, siyasətçilərin çıxışlarında tez-tez rast gəlirik. Söz yaşayır, yeni-yeni anlam və məna çalarları ilə zənginləşir, yəqin ki, bundan sonrakı lüğətlərdə o, heç vaxt "köhnəlmiş" qeydilə verilməyəcək.



Fibonaççi.

ALQORİSMİKLƏRİN VƏ ABASİSTLƏRİN MÜBAHİSƏSİ

Bir çox əsrlər boyu abak faktiki olaraq yeganə praktiki hesablama vasitəsi olmuşdur. *Abasist* və *alqorismik* düşərgələrinin gərgin qarşıdurmaları riyaziyyat tarixinə daxil olub. Bəzən birincilər *qerbekistlər*, ikinciləri isə *kossistlər* adlandırırdılar (*ital.* "cosa" – "şey", tənlikdə naməlum kəmiyyəti belə adlandırırdılar). Bu, 1202-ci ildə Leonardo Pizanskinin (Fibonaççi) məşhur "Abak kitabı" meydana çıxdıqdan sonra başladı. Bu kitabda ilk dəfə olaraq deyilirdi ki, barmaqlarda və abakdan başqa həm də kağız üzərində hesablama aparmaq mümkündür. Alqorismiklər şərq riyaziyyatçılarının əsərlərinin tərcümələri sayəsində Avropaya nüfuz etməyə təzəcə başlamış ərəb rəqəmlərindən hesablamalarda istifadə etməyi təklif edirdilər. Amma yeni hesablama üsulu tamamilə təsdiqini tapana qədər üç yüz il keçdi. Bu qədər vaxt hamı tərəfindən qəbul edilmiş işarələri tərtib etməyə, hesablama üsullarının kağız üzərində yazılışını təkmilləşdirməyə və uyğunlaşdırmağa sərf olundu. Luki Paçolinin "Riyaziyyat, həndəsə, münasibətlər və mütənasibliklər haqqında biliklər toplusu" traktatı (1494-cü il) bu mübahisənin üstündən xətt çəkməsinə baxmayaraq, çötkə (abak) hələ uzun müddət istifadədən çıxmadı.



Luki Paçoli.



Xanım Arifmetika alqorismiklərin və abasistlərin mübahisəsini həll edir. Qravüra, 1504.



PROQRAMLAŞDIRMANIN ƏSASLARI

SADƏ PROQRAMLAR

İstənilən sadə və ya mürəkkəb kompüter proqramına xarici aləmlə – bu proqramdan istifadə edən istifadəçilə əlaqə çox vacibdir. Yaxud köməkçi vasitə kimi bu proqramdan istifadə edən digər proqramlarla da həmçinin.

Proqrama münasibətdə xarici olan mühitlə bu əlaqə giriş verilənləri kimi istifadə olunan hər hansı qiymətlərin, başqa sözlə kəmiyyətlərin proqrama ötürülməsilə ifadə olunur.



Hətta giriş verilənlərinin ötürülməsi formal olaraq olmadığı hallarda belə (misal üçün, əvvəlcədən proqramlaşdırılmış şəkil və məlumatları nümayiş etdirən proqramlarda) proqram o zaman işə başlayır ki, xarici aləmdən proqramın çağırılmasını təmin edən komanda şəklində siqnal daxil olsun.

Beləliklə, adi proqramın zəruri elementi kənardan (istifadəçi, yaxud digər proqram vasitəsilə) daxil edilən qiymətin ötürülməsinə və yaddaşda lazım olan yerdə saxlanılmasına xidmət edən daxiletmə komandasıdır. Məsələn,

```
daxilet a
```

komandası istifadəçi tərəfindən daxil ediləcək qiymətlər (məsələn, klaviaturada yığılan) yığını tam daxil edilənədək proqramın işini qeyri-məhdud müddət ərzində dayandırır. Klaviatura

ilə işləyən zaman daxiletmənin sona çatması Enter klavişinin basılması ilə qeyd olunur. Enter klavişi basıldıqdan sonra proqram öz işini davam etdirir. Bu zaman onun birinci işi daxil edilmiş verilənləri daxilet komandasında *a* hərfilə işarə olunmuş yaddaş sahəsində yerləşdirmək olur (yəni *a* – sonra proqramda istifadə olunacaq kəmiyyətin adıdır).

İki *a* və *b* kəmiyyətlərinin ədədi ortasını hesablayan sadə proqram nümunəsinə baxaq:

```
daxilet a
daxilet b
c := (a + b) / 2
```

İlkin verilənlər *a* və *b* daxil edildikdən sonra proqram onların ədədi ortasını hesablayaraq nəticəni *c* adlanan oyuqda yerləşdirir. Özündə := simvolunu saxlayan komanda *mənimsətmə* adlandırırlar.

Proqramın nəticəsi digər proqramlara lazımdırsa onu sona çatmış hesab etmək olar (bu proqramlar bilirlər ki, onlara lazım olan nəticə *c* oyuğunda yerləşir). Lakin əgər nəticə istifadəçiyə lazımdırsa və o, nəticəni ekranda, yaxud printerdə çap olunmuş formada görmək istəyirsə, onda proqramı *c* kəmiyyətinin çıxış komandası ilə tamamlamaq lazımdır:

```
daxilet a
daxilet b
c := (a + b) / 2
cepet c
```

Proqramın istifadəçiyə verdiyi nəticələr heç də həmişə ədədi qiymətlər, yaxud da, mətn tipli məlumatlar olmur. Tez-tez proqramın yerinə yetirilməsi onun işlədiyi mühiti dəyişdirən əməliyyatların həyata keçirilməsindən ibarət olur. Belə proqramlar çıxışa heç bir qiymət vermir.

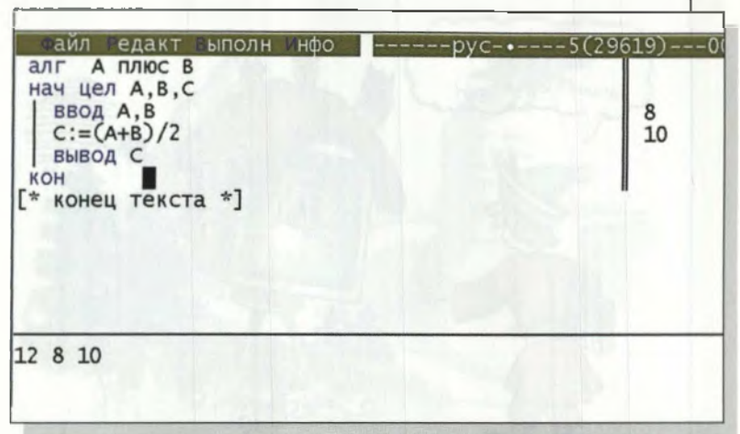
Məsələn, LOGO dilində yazılmış proqram kvadrat marşrut üzrə Tısbağanı hərəkət etdirir və hətta ekranda bu kvadratı çəkir, amma bu proqramın yerinə yetirilməsi nəticəsində heç bir qiymət yaxud məlumat verilməyəcək:

```
daxilet ədəd : tərəf
daxilet ədəd : bucaq
qələmi endir
irəli : tərəf
sağa : bucaq
irəli : tərəf
sağa : bucaq
irəli : tərəf
sağa : bucaq
irəli : tərəf
sağa : bucaq
```

Bir neçə izahat:

• Bilavasitə kəmiyyətlərin adları qarşısında LOGO-da probelsiz iki nöqtə (:) qoyulması qəbul edilmişdir;

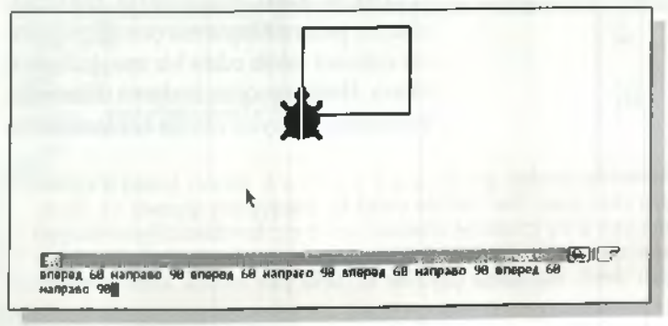
• "Daxilet" sözündən sonra "ədəd" sözünün yazılması proqramın daxil ediləcək kəmiyyətin ədəd olduğunu başa düşməsi üçün lazımdır (həmçinin,



Aşağıdakı proqram yerinə yetirildikdən sonra "bir" kəmiyyətinin qiyməti necə olacaqdır, o şərt daxilində ki, istifadəçi klaviaturadan birinci daxiletmə komandası üçün 5, ikinci üçün isə 6 qiymətini yığır:

```
daxilet iki
daxilet üç
bir := iki + üç - 4
```

Cavab: bir = 7.





mətndən ibarət qiymətlərin daxil edilməsi üçün "daxiletmə mətn" komandası da vardır);

• Nəhayət, burada daxiletmə üçün təklif olunmuş üsulun əvəzinə LOGO-da, adətən, zəruri qiymətlər bilavasitə komandada göstərilir. Beləliklə, yuxarıdakı proqramın daha ənənəvi yazılışı belə olacaqdır (əgər yerdəyişmə 20 addımla, dönmə isə 90° ilə yerinə yetirilərsə):

```
irəli 20
sağa 90
irəli 20
sağa 90
irəli 20
sağa 90
irəli 20
sağa 90
```

Nəticəni sonradan digər proqramlarda istifadə etmək üçün hesablayan proqramlar *funksiyalar* adlandırılır.

Yekun qiymətləri hasil etməyən proqramlar isə *proseduralar* adlandırılır.

Burada nümunələr kimi təqdim olunmuş proqramlar *xətti* olduqları üçün ən sadə proqramlar sırasına aid edilir. Komandalar əvvəlcədən qeyd olunmuş ardıcıllıqla (giriş verilənlərinin qiymətlərindən asılı olmayaraq), yazıldıqları qaydada (ikinci komanda birincidən sonra, üçüncü ikincidən sonra və s.) yerinə yetirilən proqramlar xətti proqramlar adlanır.

Əksər proqramlar mürəkkəb quruluşa malikdir. Onlarda komandaların yerinə yetirilmə sırasını dəyişdirən budaqlanmalara və ayrı-ayrı komandalar qrupunun çoxsaylı təkrarlanmalarını təşkil edən dövrlərə rast gəlmək mümkündür.

İDARƏEDİCİ STRUKTURLAR – BUDAQLANMALAR

Sadə proqramlar xəttidir: bu proqramların komandaları yazıldıqları sıraya uyğun olaraq, yəni bir-birinin ardınca yerinə yetirilirlər. Bəzən ardıcıl yerinə yetirilən belə komandalar qrupunu *seriya* adlandırırlar.

Təkcə xətti proqramlar mövcud olsaydı proqramlaşdırma çox darıxdırıcı və zəhmət tələb edən bir məşğuliyyət olardı. Bütün proqramlaşdırma dillərində proqramlar işləyən zaman komandaların yerinə yetirilmə ardıcıllığını dəyişməyə imkan verən idarəedici komandalar nəzərdə tutulur. Proqramın yerinə yetirilməsi ardıcıllığını müəyyən edən komandalar *idarəedici strukturlar* adlanır.

İdarəedici strukturların üç əsas tipi vardır:

• *seriyalar;*
• *budaqlanmalar* (şərti komandalar);
• *dövrlər* (təkrarlanma komandaları).

Şərti komandaların ümumi forması belədir:

```
əgər <şərt>
 | onda <komanda-1>
 | əks halda <komanda-2>
tamam
```

Şərtlərə nümunələr: $a = 15$, $a + 1 < c$, $15 = 5$. Birinci ancaq o zaman doğru olur (yəni "hə" cavabı verir) ki, kəmiyyətin qiyməti 15 olsun. İkinci şərt a və c kəmiyyətlərinin $a + 1 < c$ bərabərsizliyini ödəyən bütün qiymətləri üçün doğrudur. Ancaq üçüncü şərt heç zaman doğru deyil. Bu halda deyirlər ki, belə şərt eynilik kimi yalandır.

Altından xətt çəkməklə dilin açar sözləri seçdirilir, künc mötərizələrdə isə konkretləşdirmə – təyin olunmalar və izahlar tələb edən dil kateqoriyası göstərilir.

Şərt adətən mühakimə formasında olur. Onun yoxlanması birqiymətli cavab – "hə", yaxud "yox" (0 və ya 1, doğru və ya yalan) verir.

Şərti komandada birinci yerinə yetirilən əməliyyat şərt-mühakimənin doğruluğunun yoxlanılmasıdır. Onun qiyməti şərti komandanın iki *budağından* birinin seçilməsi üçün istifadə olunur. Bəzən <*komanda-1*>-in dayandığı budağı onda budağı, digər budağı <*komanda-2*>-ni əks halda budağı adlandırırlar.

Məsələn,

```
a := 10
 əgər a > 5
 | onda çapet "hər şey
                qaydasındadır"
 | əks halda çapet "qəza vəziyyəti"
 tamam
c := a + 1
```

$a > 5$ şərtinin doğruluğu şərti komandadan əvvəlki $a := 10$ mənimsətmə komandasında təmin olunduğu üçün proqram ekrana "hər şey qaydasındadır" məlumatını çıxaracaqdır. Növbəti yerinə yetirilən komanda $c := a + 1$ mənimsətmə komandası olduğundan, bu zaman "qəza vəziyyəti" məlumatı ekrana çıxarılmayacaqdır.





Xüsusi halda Robot təlim icraçısının idarə olunması üçün istifadə olunan KuMir proqramlaşdırma dilində iki fiziki kəmiyyətin qiymətlərini yoxlayan şərtlər nəzərdə tutulur. Bu kəmiyyətlər icraçının işlədiyi mühiti xarakterizə edən parametrlər – radiasiya və temperaturdur. Bu kəmiyyətlərin cari qiymətləri "radiasiya" və "temperatur" açar sözlərilə adlanan yaddaş sahələrində (oyuqlarda) yerləşirlər. Ona görə də, bu dildə, məsələn, radiasiya < 0,7, temperatur > 100 şərtləri mümkündür.

Həm onda, həm də əks halda budaqlarında təkcə bir yox, bir neçə komanda vermək olar. Ancaq, əgər şərti komandanın budaqlarında istifadə olunan komandalar qrupundan proqramda dəfələrlə istifadə olunursa, onda bu əmrlər qrupundan prosedur yaratmaq tövsiyə olunur. Bu halda uyğun budaqda <*komanda*> ifadəsində prosedurun çağırılması, başqa sözlə, sadəcə prosedurun adı yazılır.

```
алг цел Факториал(арг цел x)
 дано x>=0
 надо
нач
 | если x>1
 | | то   знач:=Факториал(x-1)*x
 | | иначе знач:=1
 | все
кон
|■ конец текста =1
```

```
5
Да

Нет
120
1

Подсказка:
```

```
x = 5

Факториал = 120
```



Qnda və əks halda budaqlarının uyğun əgər açar sözündən sağa doğru bir az sürüşdürülmək, şərti komandanın sonunu bildirən tamam sözünün isə ciddi olaraq əgər sözü ilə eyni mövqedə, soldan eyni məsafədə yazmaq tövsiyə olunur. Bu tövsiyə (dilin tələbi yox!) onunla əlaqədardır ki, <komanda-1> və <komanda-2> ifadələrinin yerinə, öz növbəsində *iç-içə* şərti komandalar zəncirini əmələ gətirən budaqlanmalar yazıla bilər. Bu halda şərti komandaların budaqlarının səliqəli iç-içə yazılması proqramı oxuyan insana, yorucu blok-sxemlərdən istifadə etmədən ən mürəkkəb budaqlanmaları belə başa düşməyə imkan verir. *a*, *b* və *c* ədədlərinin maksimumunu tapan proqramın təsviri aşağıdakı kimidir:

```
əgər a > b
| onda
| əgər a > c
| | onda max := a
| | əks halda max := c
| tamam
| əks halda
| əgər b > c
| | onda max := b
| | əks halda max := c
| tamam
tamam
```

Şərti komandalar bəzən qısaldılmış formada yazılır. Məsələn, əgər onda budağında olan komandanı ancaq şərtin ödənilməsi halında yerinə yetirmək lazımdırsa və əks halda heç bir əlavə əməliyyat yerinə yetirmək lazım deyilsə, şərti komandanı belə yazmaq olar:

```
a := 1
əgər a < 3
| onda b := "uyğun nəticə"
tamam
c := 10
```

Burada şərti komandada əks halda budağı yoxdur, *b* kəmiyyəti isə "uyğun nəticə" qiymətini yalnız şərt doğru olduqda alır.

Aşağıdakı proqram havaya görə paltar seçmək üçün məsləhət verir:

```
çıxarma "köynək geyin"
əgər t < 0
| onda çapet "kürk geyin"
tamam
```

Proqram ixtiyari temperaturda köynək geyməyi məsləhət görür, lakin soyuq günlərdə əlavə bir məsləhət də verir: kürk geyin.

Bir şərti komandanın digər şərti komandanın daxilində yerinə yetirilməsinə nümunə göstərilib. Mümkün variantlar içərisindən yalnız birinin seçilməsi lazım gəlirsə, oxşar hallara çox təsadüf olunur.

Proqramın layihələndirilməsinə aid növbəti misalda həftənin günlərinin kodları aşağıdakı cədvəldəki kimi daxil olunur (məsələn, klaviaturadan):

bazar ertəsi	1
çərşənbə axşamı	2
çərşənbə	3
cümə axşamı	4
cümə	5
şənbə	6
bazar	7

Bu kodu gün kəmiyyətinin mətnlə ifadə olunmuş qiymətinə çevirmək lazımdır.

Əlbəttə, belə proqramı klassik əgər komandasının köməyilə yazmaq olar. Bu halda proqram mürəkkəb quruluşlu iç-içə şərti komandalardan ibarət olacaqdır.


```
daxilet kod
əgər kod = 1
| onda gün := "bazar ertəsi"
| əks halda
| əgər kod = 2
| | onda gün := "çərşənbə axşamı"
| | əks halda
| | əgər kod = 3
| | | onda gün := "çərşənbə"
| | | əks halda
| | | əgər kod = 4
| | | | onda gün := "cümə axşamı"
| | | | əks halda
| | | | əgər kod = 5
| | | | | onda gün := "cümə"
| | | | | əks halda
| | | | | əgər kod = 6
| | | | | | onda gün := "şənbə"
| | | | | | əks halda
| | | | | | əgər kod = 7
| | | | | | | onda gün := "bazar"
| | | | | | | əks halda çapet "kod səhvdir"
| | | | | | tamam
| | | | | tamam
| | | | tamam
| | | tamam
| | tamam
| tamam
tamam
```

Proqramlaşdırmada oxşar vəziyyətlərə tez-tez rast gəlindiyindən bir çox proqramlaşdırma dillərində bir neçə mümkün haldan yalnız birinin seçilməsinə imkan verən xüsusi şərti komandalar nəzərdə tutulur.

Seçim komandasının köməyilə yuxarıdakı proqramı daha qısa və aydın formada yazmaq olar:

```
seçim
|kod =1 hal : gun := "bazar ertəsi"
|kod =2 hal : gun := "çərşənbə axşamı"
|kod =3 hal : gun := "çərşənbə"
|kod =4 hal : gun := "cümə axşamı"
|kod =5 hal : gun := "cümə"
|kod =6 hal : gun := "şənbə"
|kod =7 hal : gun := "bazar"
|əks halda çapet "kod səhvdir"
tamam
```

Mürəkkəb şərtləri olan proqramların layihələndirilməsi zamanı diqqətli olmaq lazımdır: ola bilər ki, hər şey ilk baxışda göründüyündən daha sadə həll edilir. Kompüterin təklif olunmuş dörd cavabdan birini düzgün cavab kimi tanımalı olduğu testə baxaq. Cavablar 1, 2, 3 və 4 ədədlərilə kodlaşdırılmışdır. Proqram məhz bu ədədlərdən birini, daxilet komandasından sonra klaviaturadan kodu daxil edən istifadəçidən gözləyir. Fərz edək ki, baxılan testdə düzgün cavab 3 ədədilə kodlaşdırılmışdır. Həftənin günlərinin kodlaşdırılması məsələsilə xarici oxşarlıqlarına baxmayaraq bu məsələ daha sadə yolla həll edilir:

```
daxilet kod
əgər kod = 3
| onda çapet "cavab düzgündür"
| əks halda çapet "cavab düzgün deyil"
tamam
```



İDARƏEDİCİ STRUKTURLAR – DÖVRLƏR



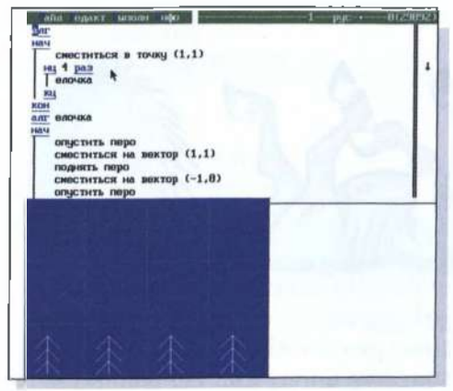
Proqramlaşdirma dillərində dəfələrlə təkrarlanan əməliyyatların yerinə yetirilməsi üçün *təkrarlanma* komandaları və yaxud *dövrlər* mövcuddur.

Proqramlaşdırma dillərində (bəzən müxtəlif formada yazılan) dörd növ dövrə rast gəlmək olar, ancaq onların hər birində məcburi elementlər vardır:

- *dövrün başlığı;*
- *dövrün sona çatması şərti;*
- *dövrün gövdəsi;*
- *təkrarlanma sayının sayğacı.*

Təkrarlanma komandasının başlanğıcını göstərən açar söz dövrün başlığıdır. Hər dövr növünün öz açar sözü olur.

Dövrün sona çatması şərti, adətən, budaqlanma komandasının şərtinin yazıldığı formada olur. Şərt – yoxlama tələb edən mühakimədir; nəticədə iki mümkün cavabdan biri alına bilər – "hə", yaxud "yox" (0, yaxud 1, doğru, yaxud yalan). Məsələn, $a > 10$ – şərt-mühakiməsi a-nın 0, 3, 7 qiymətlərində yalan və 12, 100, 105 qiymətlərində isə doğrudur.



Dövrün hər bir təkrarlanmasında yerinə yetirilən komandalar qrupu (seriyası) dövrün gövdəsini təşkil edir. Gövdədə digər idarəedici strukturlar – budaqlanma və təkrarlanma komandaları da iştirak edə bilər. Əgər bir dövrün govdəsində başqa bir dövr də yerləşərsə, belə təkrarlanma komandalarını *iç-içə dövrlər* adlandırırlar. Dövrün gövdəsinin sərhədləri mötərizələrlə qeyd olunur. Məs., KuMir proqramlaşdırma dilində açılan mötərizə, yaxud dövrün başlanğıcı – db açar sözü, bağlanan mötərizə, yaxud dövrün sonu isə ds sözüdür.

Nəhayət, bəzi dövr tiplərində olmayan element dövrün təkrarlanmaların sayını göstərən sayğacıdır. Tələb olunan təkrarlanmaların sayının dəqiq qeyd olunması vacib olduqda sayğac proqramçı və istifadəçi üçün çox gərəkli olur.

Beləliklə, proqramlaşdırma dillərində istifadə olunan dörd dövr tipi aşağıdakılardır:

- təkrarlanmaların sayı əvvəlcədən məlum olan dövrlər;
- şərti başlanğıcda yoxlanılan dövrlər;
- şərti sonda yoxlanılan dövrlər;
- dövr dəyişənini verilmiş addımlarla dəyişən dövrlər.

Təkrarlanmaların sayı əvvəlcədən məlum olan dövrlər ən sadə dövrlər hesab olunurlar. Onların ümumi şəkli belədir:

```
db <ifadə> dəfə
| <seriya>
ds
```

Dövr işləyərkən əvvəlcə *<ifadə>* hesabi ifadəsinin ədədi qiyməti müəyyən edilir (o tam olmalıdır), sonra isə tapılmış n qiymətindən təkrarlanmaların sayını qeyd etmək üçün istifadə olunur. Beləliklə, bu tip dövrlərdə dövrün sona çatması şərti formal olaraq olmur, lakin komandanın yerinə yetirilməsində onun iştirakı aşkardır: dövr o zaman sona çatır

ki, qeyri-aşkar şəkildə yaradılan və dövrün gövdəsinin hər bir təkrarlanmasında bir vahid artan sayğacın qiyməti n-lə müqayisə olunur.

Aşağıdakı dövr m massivinin birinci 22 qiymətini, $m[1]$-dən $m[22]$-dək cəmləyir:

```
a := 10
b := 12
i := 1
s := 0
db a + b dəfə
| s := s + m[i]
| i := i + 1
ds
```

Dövrün gövdəsində (*<seriya>*) yeganə komandanın mümkün olduğu Kukaraça icraçısının idarəolunması dilində təkrarlanma sayı əvvəlcədən məlum olan dövr tamamilə sadə görünür:

```
TƏKRARLA n <komanda>
```

Dörd komandanı – YUXARI, AŞAĞI, SOLA, SAĞA başa düşən və yerinə yetirə bilən Kukaraçanı (komandaların yerinə yetirilməsi zamanı verilmiş istiqamətdə bir xana yerini dəyişir) 10x10 ölçülü kvadrat sahə boyunca gəzməyə məcbur etmək olar:

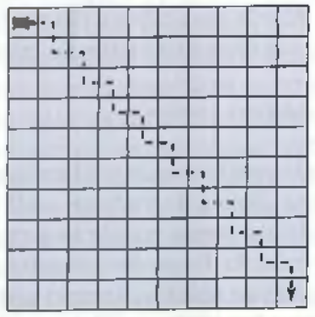
```
TƏKRARLA 9 SAĞA
TƏKRARLA 9 AŞAĞI
TƏKRARLA 9 SOLA
TƏKRARLA 9 YUXARI
```



Bəs, belə dövrün gövdəsində bir neçə komandanı yerinə yetirmək lazım gəldikdə nə etməli? Bunun üçün iki imkan vardır:

- əgər təkrarlanma komandasının gövdəsində yerinə yetirilməsi arzu olunan komandalar qrupu proqramda dəfələrlə lazım olursa, bu komandalar qrupundan ibarət prosedur təşkil etmək tövsiyə olunur və bu zaman uyğun budaqda *<komanda>* ifadəsini prosedurun çağırışı kimi yazmaq lazımdır. Pilləkənlə aşağı enmə məsələsini Kukaraça belə həll edir:

```
BU ENİŞ
  TƏKRARLA 9 PİLLƏKƏN
SON
BU PİLLƏKƏN
  SAĞA
  AŞAĞI
SON
```



- əgər belə komandalar qrupu dövrün gövdəsində bir dəfə istifadə olunursa, onda heç prosedur yaratmağa ehtiyac



yoxdur, qrupu fiqurlu mötərizələrə – { } (mahiyyətcə elə db və ds kimi) almaq kifayətdir; açılan və bağlanan mötərizələrlə əhatələnmiş bütün komandalar kompüter tərəfindən vahid bir komanda kimi qəbul olunur:

```
BU ENİŞ
  TƏKRARLA 9
  {
    SAĞA
    AŞAĞI
  }
SON
```

Şərti başlanğıcda yoxlanılan dövrlərdə gövdənin təkrarlanma sayını əvvəlcədən asanlıqla müəyyənləşdirmək həmişə mümkün olmur. Heç buna gərək də yoxdur – dövrün sona çatması dövrün başlığında yazılmış şərt vasitəsilə izlənilir:

```
db nə qədər ki <şərt>
  | <dövrün gövdəsi>
ds
```

Belə əmr komanda yoxlanılması ilə başlayır. Əgər o doğrudursa, onda dövrün gövdəsi yerinə yetirilir və şərt yenidən yoxlanılır. Bu proses şərt yalan olana qədər davam etdirilir. Axırıncı şərt təkrarlanma komandasının dayandırılması əlaməti olur. Əməliyyatların belə ardıcıllığı nəzərdə tutur ki, şərtdə istifadə olunan dəyişənin qiyməti dövrün gövdəsində dəyişdirilsin. Əgər şərt həmişə doğru olarsa, dövr sonsuz olaraq yerinə yetirilər, əksinə, əgər şərt başlanğıc anda yalan olarsa, dövr bir dəfə də olsun yerinə yetirilməz və idarəetmə dövrdən sonrakı komandaya keçər.

Yuxarıda tanış olduğumuz, *m* massivinin birinci 22 qiymətinin, *m*[1]-dən *m*[22]-dək cəmlənməsi məsələsini şərti başlanğıcda yoxlanılan dövrlə də həll etmək olar:

```
a := 10
b := 12
i := 1
s := 0
db nə qədər ki i < a + b
  | s := s + m[i]
  | i := i + 1
ds
```

Şərti sonda yoxlanılan dövr aşağıdakı formada olur:

```
db
  | <dövrün gövdəsi>
ds <şərt> olduqda
```

Bu dövr komandası əvvəlkindən onunla fərqlənir ki, şərt gövdənin əvvəlində deyil, sonunda yoxlanılır. Bu onu göstərir ki, istənilən halda (şərtdən asılı olmayaraq), dövrün gövdəsi ən azı bir dəfə yerinə yetiriləcəkdir. Şərti başlanğıcda yoxlanılan dövrdən istifadə etməklə yazılmış yuxarıdakı proqramın şərti sonda yoxlanılan dövrlə yazılan variantı belə olacaqdır:

```
a := 10
b := 12
i := 1
s := 0
db
  | s := s + m[i]
  | i := i + 1
ds i < a + b olduqda
```

Dövr dəyişənini verilmiş addımlarla dəyişən dövrlərdə daha iki anlayış meydana çıxır – dövr *dəyişəni* və dövr gövdəsinin hər bir təkrarlanmasında onun (dövr dəyişəninin) dəyişmə *addımı*. Belə dövrdə sayğaca başlanğıc qiymət verməyə gərək yoxdur, çünki bu dəyişənin başlanğıc, son qiymətləri və onun dəyişmə addımı aşağıdakı formalı təkrarlanma komandasında verilə bilər:

```
db <dəyişən> üçün <ifadə-1> dən
  | <ifadə-2> dək
  | [addım <ifadə-3>]
  | <dövrün gövdəsi>
ds
```

Ümumiyyətlə, dövr dəyişənilə sayğacın üst-üstə düşməsi vacib deyildir. Buna baxmayaraq bu şəkildə dövr dəyişənindən daha çox istifadə olunur. Bu dövrün ümumi təsvirində başlıq elementi kvadrat mötərizələrlə əhatələnmişdir. Bu onu göstərir ki, belə elementlər komandanın konkret yazılışında olmaya da bilər. Belə ki, [addım <ifadə-3>] ifadəsinin olmaması təkrarlanma komandasını məcbur edir ki, gövdənin hər bir yerinə yetirilməsi zamanı dövr dəyişəninin qiymətini 1 vahid artırsın. Beləliklə, susqunluqla <ifadə-3> ifadəsinin qiyməti 1-ə bərabər götürülür. Addımı əvvəlcədən verilmiş dövr bizim sadə misal üçün belə yazıla bilər:

```
a := 10
b := 12
s := 0
db
i üçün 1 dən a + b dək
  | s := s + m[i]
ds
```

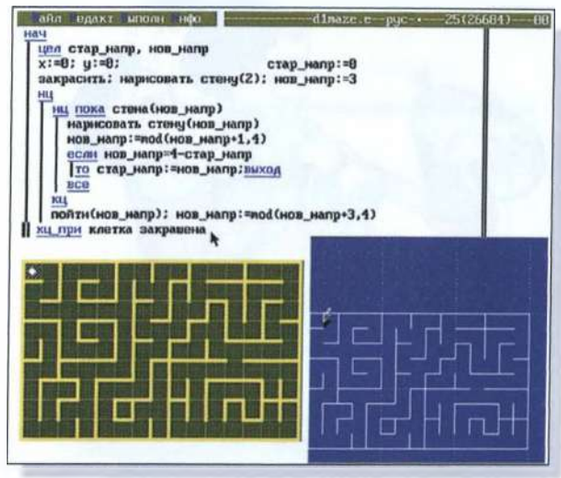
Burada dövr dəyişəni *i* kəmiyyətidir. Dövrün gövdəsində sayğacın dəyişdirilməsinə ehtiyac yoxdur, belə ki, *i* := *i* + 1 komandası dövrün təyinində nəzərdə tutulmuşdur (bu misalda 1 vahid dəyişmə addımı susqunluqla qəbul edilmişdir).

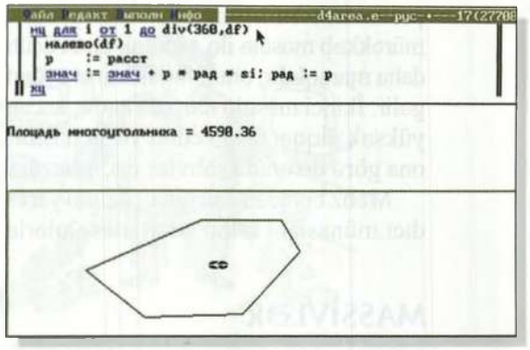
Axırıncı dövr tipi daha yığcamdır. Əgər nəzərə alsaq ki, <ifadə-1>, <ifadə-2> və <ifadə-3> elementlərinin əvəzinə mürəkkəb ifadələri də yazmaq olar, aydın olar ki, bu tip təkrarlanma komandasının geniş imkanları vardır.

Qeyd edək ki, təkrarlanma komandalarına aid bütün nümunələrdə dövrün gövdəsi bir şaquli xətt boyunca yerləşdirilmiş db və ds komandalarına nəzərən sağa sürüşdürülmüşdür. Bu tövsiyə (dilin tələbi yox!) onunla əlaqədardır ki, dövrün gövdəsində öz növbəsində bir-birinin daxilində yerləşdirilmiş dövrlər və budaqlanmalar kimi idarəedici strukturların zənciri gələ bilər.

Bu halda təkrarlanma və şərti komandaların səliqəli sürüşdürülməsi proqramı oxuyan hər kəsə çox yer tutan və artıq blok-sxemlərdən istifadə etmədən ən mürəkkəb budaqlanmaları araşdırmağa kömək edir.

Əgər riyaziyyatçını qələm və kağızla silahlandıraraq ondan iki məsələni (mürəkkəb inteqralın hesablanması və 300 ədədin toplanması) həll etməyi xahiş etsək, nəticə təəccüblü alınacaqdır: birinci məsələni, əlbəttə ki, riyaziyyatçı həll edəcək, amma ikincidə çox güman ki, səhv buraxacaqdır. Baxmayaraq ki, hər bir ayrıca toplamanı kiçik yaşlı məktəbli də asanlıqla yerinə yetirə bilər. Ona görə də, riyaziyyatçıya həll etmək üçün bu iki məsələdən birini seçmək imkanı verilsə,







şübhəsiz, o birincinin üzərində dayanar: mürəkkəb məsələ ilə işləmək onun üçün daha maraqlıdır, onun intellektinə uyğun gəlir. İkinci məsələ darıxdırıcıdır, ancaq yüksək diqqət tələb edən yorucu işdir, ona görə də orada səhvlər qaçılmazdır.

Məhz buradan – orijinal, özünə yaradıcı münasibət tələb edən məsələlərlə yeknəsəq və çox zəhmət tələb edən məsələlər arasında insanın və maşının imkanlarının sərhədləri keçir.

Maşın yorulmadan, gərginlik keçirmədən, mənfi emosiyalarsız nəinki üç yüzlərlə eyni tipli əməliyyatları, həmçinin üç minlərlə, üç milyonlarla belə əməliyyatları da asanlıqla yerinə yetirir.

MASSİVLƏR

İnformatikada tətbiq olunan fundamental verilənlər strukturundan biri də *massivlərdir*. Bu anlayışa tez-tez digər sinonim terminlərdə – "vektorlar", "matrislər" və b. rast gəlinir. Məsələn, KuMir proqramlaşdırma dili (burada bu dildən müxtəlif proqramlaşdırma mexanizmlərini nümayiş etdirmək məqsədilə istifadə olunur) "cədvəl", yaxud "cədvəl kəmiyyətləri" terminilə, yəni eynitipli elementlərin sonlu nizamlanmış çoxluğu ilə işləyir.

Massiv-cədvəllərə misallar:

1) həq cəd *maaş* [1:20]

(cədvəl kəmiyyəti *maaş* adlandırılıb, onun elementləri 20 həqiqi ədəddir);

2) lit cəd *qaçış* [1:7]

(*qaçış* cədvəlinə mətn, yaxud liter kəmiyyətlərdən ibarət yeddi element daxildir; orada yüngül atletik qaçış iştirakçılarının soyadları toplanıb);

3) lit cəd *kross* [1:4, 1:7]

(*kross* cədvəli dörd sətir və yeddi sütundan ibarətdir; orada yüngül atletika krosunun dörd qaçışı üzrə bölünmüş və hər birində 7 qaçışçı olmaqla bütün iştirakçılarının soyadları sadalanıb).



Verilmiş cədvəllərin təsviri (KuMir proqramlaşdırma dilinin qaydalarına uyğun olaraq) aşağıdakı qaydada qurulmuşdur:

1) öncə cədvəli təşkil edən elementlərin tipi göstərilir; birinci cədvəl həqiqi ədədlərdən (həq), ikinci liter kəmiyyətlərdən (lit) təşkil olunmuşdur;

2) sonra cədvəl kəmiyyətinin tipi – cəd açarsözü gəlir;

3) ondan sonra təsvir olunan cədvəl kəmiyyətinin adı verilir (*maaş, qaçış, kross*);

4) təsviri cədvəlin ölçülərini təyin edən konstruksiya tamamlayır.

Cədvəl *sadə* deyil, *mürəkkəb* verilənlər strukturuna aid olduğundan məhz cədvəlin ölçülərini verən konstruksiya onun təsvirini səciyyələndirir. Kvadrat mötərizələrdə iki nöqtə ilə ayrılmış tam ədədlər cütü verilmişdir. Bu cütdə birinci ədəd cədvəlin birinci elementinin, ikinci ədəd isə onun axırıncı elementinin nömrəsini göstərir (sıra ilə ardıcıl nömrələnmə nəzərdə tutulur). Bu onu göstərir ki, cütün birinci elementi 1-dirsə, onun ikinci elementi cədvəlin elementlərinin sayını ifadə edir.

Yuxarıda verilmiş misalların ilk ikisində vəziyyət belədir: *maaş* cədvəlində 20, *qaçış* cədvəlində 7 element vardır.



Göstərilən cədvəllər *xətti*, yaxud *birölçülü* cədvəllərdir. Bunu kvadrat mötərizələrdə ancaq bir cüt tam ədədin olması göstərir. Xətti cədvəldə elementin nömrəsinə onun *indeksi* deyilir.

Məsələn, qaçış cədvəli aşağıdakı şəkildədir:

lit cəd *qaçış* [1:7]

Elementlərin qiymətləri	Elementlərin indeksləri
Əmirli	1
Səmədzadə	2
Məmmədli	3
Babanlı	4
Ağayev	5
Tahirli	6
Əlibəyli	7

Ağayev soyadı *qaçış* cədvəlinin 5-ci elementinin qiymətidir. İndeksin köməyilə bu fakt belə yazılır:

qaçış [5] = "Ağayev"

Kross cədvəli isə *ikiölçülüdür*: onda dörd sətir və hər bir sətirdə yeddi element vardır. Bu parametrlər *kross* cədvəlinin adından sonra kvadrat mötərizələrin içində duran iki tam ədəd cütü şəklində asanca oxunur. Üçölçülü və daha yüksək ölçülü cədvəllər də analoji qaydada təsvir olunur. *n*-ölçülü cədvəlin elementini göstərmək üçün bir deyil *n* indeksdən istifadə etmək lazımdır. Beləliklə, ikiölçülü *kross* cədvəlində:

kross [3, 1] = "Qafarlı",
kross [2, 4] = "Salayev",
kross [4, 5] = "Göyçəli".

Cədvəldə hər bir element birqiymətli olaraq indekslərin köməyilə tanındığından cədvəl elementləri üzərində uyğun tip kəmiyyətlərə tətbiq olunan bütün əməliyyatlar aparıla bilər. Məsələn,

maaş [17] := *maaş* [1] + *maaş* [3]

komandasına əsasən kompüter *maaş* cədvəlindən *maaş* [1] və *maaş* [3] ədədlərini seçəcək, onları toplayacaq və nəticəni indeksi 17 olan elementə mənimsədəcək,

çapet *kross* [4, 7]

komandasına görə isə Azadxanlı sözünü çap edəcəkdir.

Cədvəl elementlərinin nizamlı olması qrup şəklindəki əməliyyatları effektli yerinə yetirməyə imkan verir. Məsələn, dövrdən istifadə etməklə birdəfəyə *maaş*

lit cəd *kross* [1:4, 1:7]

Əliyev	Həsənov	Dəmirov	Vəkilov	Salahlı	Ramazanov	Aslanov
Əmirli	Köçərli	Bədəlbəyli	Salayev	Cəfərli	Calallı	Həmidov
Qafarlı	Həsənoğlu	Salmanova	Qaraçaylı	Borçalı	Səfərli	Kərimzadə
Seyidov	Güllüoğlu	Şamilli	Məmmədov	Göyçəli	Həmzəyev	Azadxanlı
1	2	3	4	5	6	7

cədvəlinin bütün elementlərini 9% artırmaq olar:

```
alq artır
(arq nət həq maaş [1:20])
baş tam i
| db i üçün 1 dən 20 dək
| | maaş [i] := maaş [i] * 1.09
| ds
son
```

burada, *i* – tam tipli kəmiyyətin adıdır (dövrün idarəedici dəyişənidir). Bu misalda *i* dövrün gövdəsinin 1-dən 20-dək hər yerinə yetirilməsi zamanı 1 qiymət artır. *Maaş* cədvəli bu alqoritmdə eyni zamanda həm arqument, həm də nəticədir.

Yuxarıda təsvir olunmuş üç misalda cədvəllərin qeyd olunmuş sayda elementləri vardır. Bununla belə yaxşı olardı ki, bu kəmiyyət qiymətləri kimi dəyişən kəmiyyətlərdən istifadə olunsun. Məsələn, əgər *maaş* cədvəlinin təsvirində qeyd olunmuş məhdudiyyətin əvəzinə iki nöqtədən sonra tam *n* dəyişəni qoyulsa, belə cədvəldən alqoritmdə elementlərin ixtiyari sayı üçün istifadə etmək olar:

```
alq artır
(nət həq maaş [1:n])
baş tam i
| db i üçün 1 dən n dək
| | maaş [i] := maaş [i] * 1.09
| ds
son
```

Elementlər çoxluğunda verilmiş əlamətlərə görə axtarış zamanı, ən böyük və ən kiçik qiymətlərin təyinində, cürbəcür sıralamalarda cədvəllərdən istifadə etmək əlverişlidir. Misal: müsbət ədədi elementlərdən ibarət nizamlanmış çoxluqda, yəni cədvəldə ən böyük elementi və onun indeksini – bu çoxluqda elementin nömrəsini tapmalı.

```
alq maksimum
(arq həq maaş [1:n],
nət həq mak, tam ind)
baş tam i; mak := 0; ind := 1
| db i üçün 1 dən n dək
| | əgər maaş [i] > mak
| | | onda mak := maaş [i]; ind := i
| | tamam
| ds
son
```

VERİLƏNLƏRİN STRUKTURU

"Alqoritmlər + verilənlərin strukturu = = proqram" – proqramlaşdırma üzrə məşhur İsveçrə mütəxəssisi, parlaq pedaqoq, Pascal, Modula-2, Oberon proqramlaşdırma dillərinin müəllifi Niklaus Virtin kitabı belə adlanır. Proqramın Virt tərəfindən qeyd olunan hər iki hissəsi eyni dərəcədə önəmlidir. Təkcə qeyri-mükəmməl alqoritm deyil, verilənlərlə işin uğursuz təşkili də proqramın sürətini onlarla, bəzən isə milyonlarla dəfə azalda bilər. Digər tərəfdən, proqramlaşdırma nəzəriyyəsini bilmək və sistematik olaraq onu praktikada tətbiq etməyi bacarmaq effektli və həm də gözəl proqramlar yaratmağa imkan verir. Verilənlərlə işləmə məharəti proqramlaşdırma sənətinin olduqca vacib tərkib hissələrindən biridir.



Niklaus Virt.

MASSİV BAZA STRUKTURU KİMİ

Məlumdur ki, kompüterin iki başlıca tərkib hissəsi prosessor və operativ yaddaşdır. Kompüterin operativ yaddaşı hər birinin öz ünvanı olan elementlərdən ibarətdir. Müasir kompüterlərdə ünvanlana bilən ən kiçik element baytdır. Ardıcıl yerləşmiş bir neçə baytdan ibarət olan və bir tam ədədi işarə edən maşın sözü anlayışı da vacibdir. Müasir prosessorlarda maşın sözünün məbtəbəsi (bitlərin sayı) 32, yaxud 64 bitdir, uyğun olaraq maşın sözü 4, yaxud 8 bayt olur. Prosessor bütün əməliyyatları maşın sözü ilə icra edir (maşın sözü prosessor üçün ən təbii verilənlər elementidir). Maşın sözünün ünvanı olaraq onu təşkil edən baytla-

rın ünvanlarından ən kiçiyi götürülür. Beləliklə, prosessorun mərtəbəsindən asılı olaraq maşın sözünün ünvanı dördə, yaxud səkkizə tam bölünür.

Proqramçı baxımından operativ yaddaş elementlər massividir. Massivin hər bir elementinin öz ünvanı olur ki, onu da adətən, indeks adlandırırlar. Massivin ixtiyari elementini bir elementar əməliyyatla ani olaraq oxumaq, yaxud yazmaq olar.

İxtiyari elementinə birbaşa müraciət mümkün olan verilənlər strukturu birbaşa, yaxud ixtiyari irişimli verilənlər strukturu adlanır (*ing.* random access). Massiv (onu cədvəl də adlandırırlar) belə struktura nümunədir. Təkcə bir, yaxud bir neçə elementinə birbaşa müraciətin mümkün olduğu verilənlər strukturu (qalan elementlərinə müraciət etmək üçün əlavə əməliyyatlar aparmaq gərəkdir) ardıcıl irişimli verilənlər strukturu adlanır. Belə struktura nümunə kimi mahnılar yazılmış maqnitofon kasetini göstərmək olar; istənilən anda yalnız bir musiqiyə qulaq asmaq olar. Başqa musiqi fraqmentlərini səsləndirmək üçün kasetin lentini irəliyə, yaxud geriyə fırlatmaq lazımdır.

Kompüterin üçüncü çox önəmli tərkib hissəsi olan maqnit diski də məntiqi baxımdan massivdir. Onun üçün elementar oxuma və yazma vahidi blokdur. Blokun ölçüsü konkret diskin qurulşundan asılıdır, adətən o, 512 ədədinin misli olur. Bir elementar əməliyyat ilə ünvanı verilmiş bir bloku oxumaq, yaxud yazmaq olar.

Beləliklə, kompüterin ən vacib yaddasaxlama qurğusu olan operativ yaddaş və maqnit diski massivdir. Riyaziyyatçı üçün tam ədəd necə bir mövcudluqdursa, proqramçı üçün də massiv elə bir mövcudluqdur. Massivin elementlərilə iş son dərəcə sürətlə yerinə yetirilir, onlara müraciət etmək üçün qabaqcadan hər hansı əməliyyat gərək olmur.



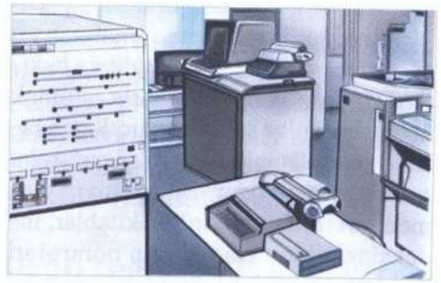
Bununla belə, effektiv proqramların yazılması üçün təkcə massivlər yetərli deyildir. Gerçək məsələlərin əksəriyyəti verilənlərlə daha mürəkkəb iş tələb edir. Massivlər ancaq baza elementləri rolunda çıxış edir ki, onlardan da daha mürəkkəb strukturlar qurulur.

Massivə örnək olaraq kitabxananı göstərmək olar. Proqramlaşdırma baxımından kitabxana kitablarla işləmək üçün verilənlər strukturudur. Kitabxanada nömrələnmiş rəfləri olan qəfəsələr massivin analoqdur. Bu massivin elementləri rolunu rəflərdəki kitablar, indekslər rolunu isə rəflərin nömrələri oynayır. Amma kitabxana təkcə qəfəsələr yığını deyil. Onun strukturuna (əlifba və sistematik) kataloqlar daxildir. Bundan başqa kitab anbarına sorğuların çatdırılması (məsələn, pnevmatik poçt) və kitabların oxuculara çatdırılması sistemləri, nəhayət, oxucuların sorğularını yerinə yetirən, həmçinin kitabxana fondunun qorunmasını və doldurulmasını həyata keçirən kitabxana işçiləri mövcuddur.



ƏN SADƏ VERİLƏNLƏR STRUKTURLARI

Sadə verilənlər strukturlarından ən vacibləri *stek* və *növbədir*. Bu strukturlara proqramlaşdırmada addımbaşı, ən müxtəlif hallarda rast gəlinir. Olduqca gözlənilməz tətbiqləri olan stek xüsusilə maraqlıdır. XX əsrin 70-ci illərinin başlanğıcında IBM 360 kompüterlər seriyasını yaradarkən IBM firması stekin aparat həllini qabaqcadan nəzərə almayaraq dramatik səhv buraxdı. Bu seriyada çoxlu başqa uğursuz həllər də vardı, ancaq, təəssüf ki, həmin yanlışlıqlar Sovet İttifaqında EC ƏBM adıyla buraxılan maşınlarda təkrarlandı.



▸ EC ƏBM 1022 seriyalı maşın.

NÖVBƏ

Proqramlaşdırmada növbə gündəlik həyatdakı növbələrə çox bənzəyir. O, birbirinin ardınca düzülərək zəncir əmələ gətirən elementlərdən ibarətdir. Növbənin başlanğıcı və sonu vardır. Adi növbəni istənilən an tərk etmək mümkün olsa da, proqramda növbənin ortasından elementləri uzaqlaşdırmaq olmaz.



Növbəni boru formasında da təsəvvür etmək olar. Borunun bir ucuna (növbənin sonuna) kürəciklər – növbənin elementləri əlavə edilir, o biri ucundan (növbənin başlanğıcından) isə onlar çıxarılır. Növbənin ortasındakı elementlər, yəni borunun içindəki kürəciklər irişilməzdir. Beləliklə, borunun ucları simmetrik deyil, kürəciklər onun içərisində ancaq bir istiqamətə hərəkət edir.

Boruyla analogiyanı davam etdirsək, aydın olar ki, prinsipcə elementləri onun hər iki ucuna əlavə etmək və hər iki ucundan götürmək olar. Oxşar verilənlər strukturu proqramlaşdırmada da vardır. Bu struktur "dek" (*ing.* Double Ended Queue – "iki uclu növbə") adlanır, ancaq növbəylə müqayisədə dekdən çox az istifadə olunur.

Növbənin proqramlaşdırmada rolu, demək olar ki, onun adi həyatdakı roluna uyğun gəlir. Növbə praktiki olaraq həmişə sorğuların eyni vaxtda yerinə yetirilməsinin mümkün olmaması halı ilə bağlı olur. Məsələn, istifadəçi kompüterdə mətn yazarkən klaviaturanın klavişini basırsa (yəni kompüterdən nəyisə yerinə yetirməsini xahiş edirsə), onda kompüterin işi mətnə bir simvolu əlavə etməkdən, yaxud ekranın müəyyən sahəsini yenidən çəkməkdən, pəncərənin işçi sahəsini fırlatmaqdan, abzası yenidən formatlamaqdan və s. ibarət olur.

İxtiyari, hətta ən sadə əməliyyat sistemi də həmişə bu və ya digər dərəcədə çoxtapşırıqlıdır. Bu o deməkdir ki, istifadəçi klavişi basan anda əməliyyat sistemi hansısa başqa bir işlə məşğul ola bilər. Buna baxmayaraq əməliyyat sistemi heç bir halda yeni tapşırığa etinasız qalmamalıdır. Bu səbəbdən də klaviş basıldıqda əvvəlki tapşırığın yerinə yetirilməsi müəyyən müddətə dayandırılır. Maşın işdə cari vəziyyəti yadda saxlayaraq sorğuya xidmət etməyə başlayır. Bu əməliyyat çox qısa olmalıdır ki, başqa tapşırıqların yerinə yetirilməsi də pozulmasın. Buna

görə də klavişin basılması ilə verilən komanda icrasını gözləyən sorğular növbəsinin axırına əlavə edilir. Bundan sonra kompüter klavişin basılması ilə yarımçıq kəsilmiş işə qayıdır. Növbəyə qoyulmuş sorğu dərhal deyil, növbəsi çatanda yerinə yetiriləcək. Hər proqramın öz sorğular növbəsi olur. İşlər yerinə yetirildikcə o, növbəti sorğunu növbənin başlanğıcından götürüb icra edir. Növbəni əməliyyat sistemi idarə edir.

Lazım olan prosedurların birbaşa başladılmasından deyil, sorğular növbəsinə əlavə olunan məlumatların göndərilməsindən ibarət olan belə yanaşma çoxlu üstünlüklərə malikdir və XX-XXI əsrlərin qovşağında geniş yayılmış obyektyönlü proqramlaşdırmanın xarakteristikalarından biridir. Məsələn, proqrama hansısa səbəbdən öz işini bitirmək lazımdırsa, o, təhlükəli olan sonlama komandasını dərhal çağırmır, çünki həmin komanda proqramın məntiqini pozur və verilənlərin itməsinə səbəb ola bilər. Əvəzində proqram özü özünə işin başa çatdırılmasının gərəkliyi haqqında məlumat göndərir, bu sorğu növbəyə qoyulur və ondan qabaq daxil olmuş sorğulardan sonra yerinə yetirilir.

NÖVBƏNİN MASSİV BAZASINDA REALLAŞDIRILMASI

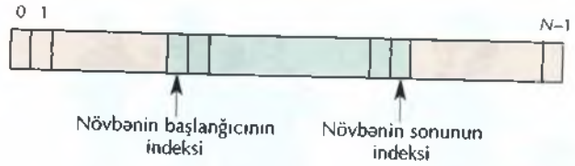
Massivi ilkin verilən kimi qəbul etsək, qalan bütün verilənlər strukturları onun bazasında reallaşdırılmalıdır. Bir verilənlər strukturunun başqasının bazasında reallaşdırılması onun işinin hər hansı məlum baza strukturunun terminləri əsasında təsvir olunması deməkdir. Reallaşdırmada mütləq "reallaşdırma ideyası"nın (başqa sözlə, reallaşdırılan strukturun elementlərinin baza strukturunda necə saxlanılması, hansı əlavə dəyişənlərin lazım olması) təsviri və hər biri reallaşdırılan strukturun müəy-



yən təlimatlarını baza strukturunun təlimatlarının köməyilə modelləşdirən altproqramlar toplusu olur.

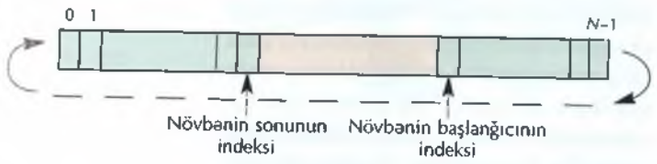
Reallaşdırma çoxmərhələli ola bilər, massiv isə heç də həmişə bilavasitə reallaşdırmanın bazası kimi çıxış etmir. Ümumiyyətlə, verilənlər strukturlarının bütün reallaşdırmaları iki sinfə bölünür – kəsilməz və istinadlı. Kəsilməz reallaşdırmalarda verilənlər strukturunun elementləri massivdə heç bir aralıq olmadan ardıcıl saxlanılır və beləliklə də massivin kəsilməz hissəsini tutur. İstinadlı, yaxud siyahılı reallaşdırmalarda isə verilənlər strukturunun elementləri massivdə, yaxud proqramçıların "qalaq"adlandırdıqları dinamik operativ yaddaşda nizamsız şəkildə saxlanılır. Bu zaman hər bir elementlə birlikdə qonşu elementlərə istinadlar da saxlanılır. Siyahılarda belə istinadlar iki (sonrakı və əvvəlki elementə), yaxud bir (sonrakı elementə istinad) ola bilər. Siyahının elementləri bir növ əlaqəli zəncir təşkil edir və siyahının ortasındakı elementlərə çatmaq üçün bu zəncir üzrə birinci elementdən başlayaraq keçmək lazımdır.

Növbələrdə iki reallaşdırma daha populyardır – massiv bazasında kəsilməz reallaşdırma (bunu dairəvi bufer bazasında reallaşdırma da adlandırırlar) və istinadlı, yaxud siyahılar bazasında reallaşdırma. Növbənin kəsilməz reallaşdırılması zamanı baza rolunda müəyyən N uzunluqlu massiv çıxış edir. Beləliklə, növbə məhduddur və onun N-dən çox elementi ola bilməz. Massivin elementlərinin indeksi 0-dan N-1-dək dəyişir.



Növbənin reallaşdırılmasında massivdən əlavə üç sadə dəyişən də var: növbənin başlanğıcının indeksi, növbənin sonunun indeksi və növbənin elementlərinin sayı. Növbənin elementləri massivin başlanğıc indeksindən sonuncu indeksinədək olan parçasında saxlanılır.

Növbənin sonuna yeni elementin artırılması zamanı sonun indeksi öncə bir vahid artırılır, sonra isə yeni element massivin oyuğuna bu indekslə yazılır. Eyni qaydada, növbənin başlanğıcından elementin çıxarılması zamanı massivin növbənin başlanğıc indeksli oyuğu əməliyyatın yekunu kimi yadda saxlanılır, sonra isə növbənin başlanğıc indeksi bir vahid artırılır.



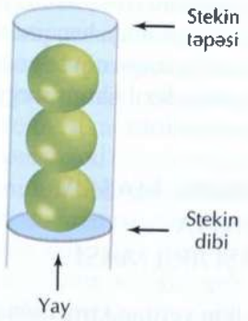
Növbənin reallaşdırılmasında əsas ideya ondan ibarətdir ki, massiv sanki qapanaraq dairə əmələ gətirir. Massivin sonuncu elementdən sonra onun birinci elementi gəlir (yada salaq ki, sonuncu elementin indeksi N-1, birincinin isə 0 olur). Növbənin sonunun indeksi massivin sonuncu elementini göstərən halda sağa sürüşdürülərsə o, massivin birinci elementinə keçir.



STEK

Stek proqramlaşdırmada ən populyar və bəlkə də ən vacib verilənlər strukturudur. Stek, elementləri artırıldıqlarının əks ardıcıllığıyla oxunan yaddasaxlama qurğusudur. Bu, sanki düzgün olmayan növbədir, ona axırıncı dayanana birinci xidmət göstərilir. Proqramlaşdırmaya aid ədəbiyyatda növbə və stekin işləmə qaydalarını göstərmək üçün abbreviaturalardan istifadə olunur. Növbənin işləmə qaydası – FIFO "birinci gəldin – birinci gedəcəksən" (*ing.* "First In First Out") anlamını verir. Stekin işləmə qaydası – FILO isə "birinci gəldin – axırıncı gedəcəksən" (*ing.* "First In Last Out") mənasını verir.

Stekə örnək olaraq masa üzərindəki kağız qalağı, yaxud boşqab qalağı ola bilər. "Stek" sözünün özü ingilis dilindən tərcümədə "topa, yığın" mənasını verir. Boşqablar qoyulduqları ardıcıllığın əksinə götürülür. Ancaq axırıncı, yəni stekin "yuxarısındakı" boşqabı götürmək olar.

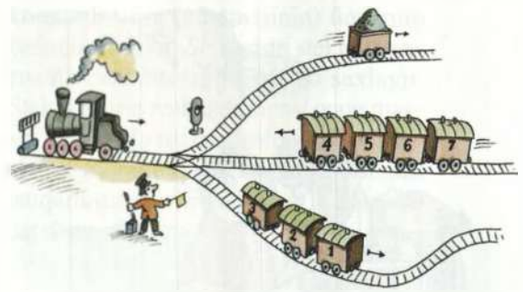


Steki dibində yay olan, şaquli yerləşdirilmiş boru kimi də təsəvvür etmək olar. Borunun yuxarı ucu açıqdır və oraya elementlər əlavə etmək olar. Hər bir yeni element stekə ondan qabaq yerləşdirilmiş elementləri bir mövqe aşağı itələyir. Elementi stekdən çıxararkən sanki qalan elementlərin hamısı yuxarı qalxır (ümumi qəbul edilmiş ingilis terminləri burada çox ifadəlidir: elementin stekə əlavə edilməsi "itələyib salmaq", "tıxamaq" mənalarını verən "push" sözü ilə, stekdən elementin çıxarılması isə "atıb qurtarmaq" mənasını verən "pop" sözü ilə işarə olunmuşdur).

STEKİN PROQRAMLAŞDIRMADA İSTİFADƏSİ

Stekdən çox tez-tez, həm də ən müxtəlif vəziyyətlərdə istifadə olunur. Onları ümumi məqsəd – hələ sonadək yerinə yetirilməmiş işdə cari anın yadda saxlanılması və lazım gəldikdə başqa tapşırığa keçid tələbi birləşdirir. Nə üçün məhz dayandırılmış tapşırıqların "dondurulmasında" stekdən istifadə olunur? Fərz edək ki, kompüter *A* tapşırığını yerinə yetirir. Onun yerinə yetirilməsi prosesində *B* tapşırığını etməyə zərurət yaranır. *A* tapşırığının həllinin cari anı yadda saxlanılır və kompüter *B* tapşırığını yerinə yetirməyə keçir. Burada da kompüter *C* tapşırığına keçə bilər və *C* tapşırığına keçməzdən əvvəl *B* tapşırığının vəziyyəti yadda saxlanılmalıdır. Sonradan, *C* tapşırığının yerinə yetirilməsi başa çatdıqda, əvvəlcə *B* tapşırığının cari vəziyyəti, *B* başa çatdıqdan sonra isə *A* tapşırığının cari vəziyyəti bərpa olunacaq. Beləliklə, bərpaetmə prosesi yadda saxlamaya əks ardıcıllıqda baş verir ki, bu da stekin iş qaydasına uyğun gəlir.

Stek rekursiya təşkil etməyə, yəni altproqramların özü-özlərinə ya birbaşa, ya da başqa çağırışlar zənciri vasitəsilə müraciət etməsinə imkan verir. Məsələn, tutaq ki, *A* altproqramı *X* giriş parametrindən və ola bilsin ki, ümumi verilənlərin vəziyyətindən asılı olan alqoritmi yerinə yetirir. *X*-in ən sadə qiymətləri üçün alqoritm birbaşa reallaşdırılır. *X*-in daha mürəkkəb qiymətləri halında alqoritm həmin alqoritmin *X*-in sadə qiymətlərinə tətbiqi vasitəsilə həyata keçirilir. Bu zaman *A* altproqramı zəruri parametr kimi *X*-in daha sadə qiymətlərini ötürməklə özü-özünə müraciət edir. Bu halda *A* altproqramı dəyişənlərin stekdə saxlanılan əvvəlki yığınını məhv etmədən yeni yığına keçir. Bu aralıq əməliyyatların sonunda *X* giriş parametrinin başlanğıc qiyməti stekdən bərpa olunur və altproqram öz-özünə müraciət zamanı yarımçıq kəsildiyi yerdən işini davam etdirir.



Bu zaman altproqramın lokal dəyişənlərini xüsusi olaraq stekdə saxlamaq lazım gəlmir, çünki bu avtomatik baş verir: altproqram işinin lap başlanğıcında aparat stekində öz lokal dəyişənlərinə yer tutur; yaddaşın bu blokunu, adətən, "altproqramın lokal dəyişənlər bloku" (*ing.* frame – "kadr") adlandırırlar. İşin sonunda altproqram stekdən öz lokal dəyişənlər blokunu uzaqlaşdıraraq yaddaşı azad edir.

Lokal dəyişənlərdən başqa aparat stekində altproqramların çağırılması zamanı dönüş ünvanları saxlanılır. Fərz edək ki, *A* proqramının hansı anındasa *B* altproqramı çağırılır. Onun çağırılmasından əvvəl *B*-nin çağırılması göstərişindən sonra gələn göstərişin ünvanı stekdə saxlanılır. Bu, *A* proqramına dönüş ünvanı adlandırılan ünvandır. *B* altproqramı işini bitirdikdən sonra stekdən *A* proqramına dönüş ünvanını çıxarır və idarəetməni bu ünvan üzrə ötürür. Beləliklə, kompüter *A* proqramının yerinə yetirilmə-





"FORTRAN 90"ın müasir standartı artıq əvvəlki versiyaların çatışmazlıqlarını aradan qaldıraraq, stek yaddaşından istifadə edir.

sini altproqramın çağırıldığı göstərişdən sonrakı göstərişdən başlayaraq davam etdirir. Prosessorların əksəriyyətində dönüş ünvanını əvvəlcədən stekdə yerləşdirən, həmçinin stekdən çıxarılan ünvana görə altproqramdan dönüşü təmin edən xüsusi komandalar olur. Adətən, çağırış komandası "call", dönüş komandası isə "return" sözü ilə işarə edilir.

Altproqramın, yaxud funksiyanın çağırılmasından qabaq onun parametrləri də stekdə yerləşdirilir. Onların stekdə yerləşdirilməsi ardıcıllığı yüksək səviyyəli dillərdə qəbul olunmuş razılaşmalardan asılıdır. Belə ki, C və ya C++ dilində stekin yuxarısında funksiyanın birinci arqumenti, onun altında ikinci və s. yerləşir. Pascal dilində hər şey tərsinədir – stekin yuxarısında funksiyanın axırıncı arqumenti yerləşir.

Ən əski və uğurlu proqramlaşdırma dillərindən biri olan FORTRAN IV-də arqumentlər stekdə yerləşməyən xüsusi yaddaş sahəsindən ötürülürdü, çünki XX əsrin 60-70-ci illərində hələ stekin aparat reallaşdırılması olmayan (məsələn, IBM 360, yaxud EC ƏBM kompüterlər var idi. Eyni qaydayla, dönüş ünvanları da stekdə deyil, yaddaşın hər bir altproqram üçün qeyd olunmuş oyuqlarında saxlanılırdı. Proqramçılar belə yaddaşı "statik" adlandırırlar, ona görə ki, statik dəyişənlər proqramın işinin ixtiyari anında həmişə yaddaşın eyni yerini tuturlar. Yalnız belə yaddaşdan istifadə etməklə rekursiya mümkün deyil, çünki hər yeni çağırışda lokal dəyişənlərin əvvəlki qiymətləri pozulur.

STEKİN MASSİV BAZASINDA REALLAŞDIRILMASI

Stekin massiv bazasında reallaşdırılması proqramlaşdırmanın "klassika"sıdır. Bəzən hətta stek anlayışının özü yanlış olaraq bu reallaşdırma ilə eyniləşdirilir.

Reallaşdırmanın bazası N ölçülü massivdir, yəni maksimal dərinliyi N parametrini aşmayan, məhdud ölçülü stek reallaşdırıla bilər. Massivin oyuqlarının indeksi 0-dan $N-1$-dək dəyişir. Stekin elementləri massivdə aşağıdakı qaydada saxlanılır: stekin dibində olan element massivin başlanğıcında, yəni 0 indeksli oyuqda yerləşdirilir. Stekin ən aşağıdakı elementinin üzərində yerləşən element 1 indeksli oyuqda saxlanılır və s. Stekin yuxarısı təxminən massivin ortalarında yerləşir. Stekin yuxarısındakı elementin indeksi xüsusi dəyişənlə işarə olunur ki, ona adətən, stekin göstəricisi (*ing.* Stack Pointer, yaxud *SP*) deyirlər.

Stek boş olduqda onun göstəricisinin qiyməti -1 olur. Stekə element artırıldıqda göstərici öncə bir vahid artır, sonra massivin stekin göstəricisinə uyğun indeksli oyuğuna artırılan element yazılır. Elementin stekdən çıxarılması zamanı, massivin stekin göstəricisinə uyğun indeksli oyuğunun tərkibi əməliyyatın nəticəsi kimi müvəqqəti dəyişəndə saxlanılır, sonra stekin göstəricisi bir vahid azalır. Təsvir olunmuş reallaşdırmada stek massivin oyuqlarının indekslərinin artması istiqamətində artır.



Tez-tez başqa variantdan – stekin vektor bazasında reallaşdırılmasından da istifadə olunur. Bu halda stekin dibi massivin axırıncı oyuğuna, yəni $N-1$ indeksli oyuğuna yerləşdirilir. Stekin elementləri, indeksi *SP*-də saxlanılan massiv oyuğundan başlayaraq axırıncı oyuğa qədər, onun kəsilməz parçasını tutur. Bu variantda stek indekslərin azalması istiqamətində artır. Stek boşdursa, onun göstəricisinin qiyməti N olur.

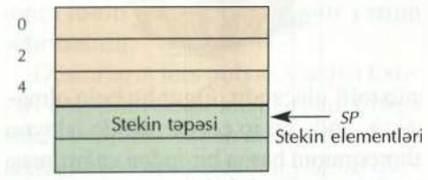


STEKİN APARAT REALLAŞDIRILMASI

Əslində kompüterdə heç bir aparat steki yoxdur. Yalnız ya baytlar massivi, ya da maşın sözləri massivi kimi baxıla bilən operativ yaddaş vardır. Aparat steki operativ yaddaş bazasında reallaşdırılır.

Maşın sözü özündə yaddaş oyuğunun ünvanını yerləşdirir. Məsələn, 32-mərtəbəli arxitekturada baytların ünvanı 0-dan $(2^{32}-1)$-dək dəyişə bilər. Maşın sözlərinin ünvanları 4-ün misilləridir və 0, 4, 8, ... , $2^{32}-4$ qiymətlərini alırlar.

Operativ yaddaşdan əlavə, prosessor özünün daxili və çox sürətli yaddaşına – prosessorun *registrlərinə* də malikdir. Onların arasında tam ədədləri, yaxud yaddaş oyuqlarının ünvanlarını özündə saxlayan ümumi təyinatlı registrlər və həqiqi ədədləri saxlayan sürüşkən registrlər vardır (proqramçı slenqində həqiqi ədədlərə sürüşkən ədədlər deyilir). Ucuz prosessorlarda sürüşkən registrlər olmaya da bilər, ancaq ümumi təyinatlı registrlərin olması vacibdir. Onların ikisi xüsusi rol oynayır: *PC* (*ing.* Program Counter – komandalar sayğacı, bəzən *IP* – Instruction Pointer də işlədilir) və *SP* (Stack Pointer – stekin göstəricisi). *PC* registri növbəti addımda yerinə yetiriləcək maşın komandasının (göstərişinin) ünvanını özündə saxlayır. *SP* registri stekin yuxarısının cari ünvanını özündə saxlayır. Stekin aparat reallaşdırılması onun massiv bazasında reallaşdırılması ilə (massivin oyuqlarının indekslərinin azalması istiqamətində stekin artdığı variantda) üst-üstə düşür.



APARAT STEKİ VƏ ALTPROQRAMIN LOKAL DƏYİŞƏNLƏRİ

Aparat steki, sadəcə, operativ yaddaş sahəsi üstəgəl stekin təpəsinin ünvanını saxlayan *SP* registridir. Aparat stekində proqramların adi dəyişənlərini yerləşdirmək olar. Lokal dəyişənlərin stekdə yerləşdirilməsi dəyişənlərin operativ yaddaşın qeyd olunmuş oyuqlarında statik yerləşdirilməsilə müqayisədə ciddi üstünlüklərə malikdir. Məsələn, bu rekursiyanın təşkilinə imkan verir. Müasir arxitekturalarda ümumi statik dəyişənlər üzərində işləyən paralel proseslərin dəstəklənməsinin böyük önəmi vardır. Bunlar yüngülçəkili adlandırılan proseslər, yaxud bir proqram çərçivəsində paralel işləyən iplərdir (Thread). Müxtəlif iplər ümumi statik dəyişənlər üzərində paralel işləyir və beləliklə, hər hansı birgə işi yerinə yetirirlər. Bu zaman eyni bir altproqram müxtəlif iplərdən çağırıla bilər. Bütün iplər üçün ümumi olan statik dəyişənlərdən fərqli olaraq, proqramların lokal dəyişənləri mütləq hər bir ip üçün ayrılmış stekdə yerləşməlidir. Onda müxtəlif iplərdən çağırılmış eyni altproqramın lokal verilənlərinin toplusu müxtəlif steklərdə yerləşdikləri üçün

Microsoft Windows sistemində bütün qrafik proqramların işi iplərdən istifadəyə əsaslanıb.



müxtəlif olacaqdır. Əgər bu belə olmasaydı, onda bir ip çərçivəsində işləyən altproqramın başqa bir ipdən çağırılması bu altproqramın lokal dəyişənlər yığınını dağıdardı.

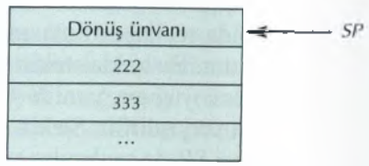
Altproqramın lokal dəyişənləri stekdə necə yerləşirlər? C dilində altproqramlara funksiyalar deyilir. Funksiya arqumentlərə və lokal dəyişənlərə, yəni yalnız funksiyanın yerinə yetirilməsi prosesində mövcud olan dəyişənlərə malik ola bilər. Tutaq ki, funksiya tam tipli iki *x* və *y* giriş arqumentlərindən asılıdır və tam tipli üç *a*, *b* və *c* lokal dəyişənlərindən istifadə edir. Funksiya tam qiymət qaytarır.

```
int f(int x, int y) {
int a, b, c;
...
}
```

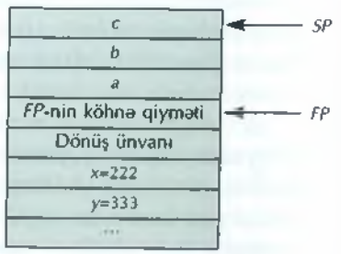
Tutaq ki, proqramın müəyyən *f* yerindən *f* funksiyası *x* = 222, *y* = 333 arqumentlərilə çağırılır.

```
z = f(222, 333)
```

Çağıran proqram *f* funksiyasının *x* və *y* arqumentlərinin faktiki qiymətlərini stekə yerləşdirir. Bu zaman stekin təpəsində funksiyanın birinci arqumenti, ondan altda ikinci arqumenti yerləşir. Funksiyanı çağıran komandanın yerinə yetirilməsi zamanı stekdə dönüş ünvanı da yerləşdirilir. *f* funksiyasının işinin başlanğıcında stek aşağıdakı şəkildə olur:



Yerinə yetirilməzdən əvvəl *f* funksiyası özünün *a*, *b*, *c* lokal dəyişənləri üçün stekdə yaddaş sahəsi tutmalıdır. C dilində belə bir razılaşma qəbul edilib: funksiyanın lokal dəyişənlər blokunun ünvanı funksiyanın işləməsi anında prosessorun *FP* (*ing.* "Frame Pointer" – "kadr göstəricisi") adlanan xüsusi registrinə yerləşdirilir. İlk növbədə *f* funksiyası stekdə *FP* registrinin əvvəlki qiymətini saxlayır. Ardınca stek göstəricisinin qiyməti *FP* registrinə köçürülür. Sonra *f* funksiyası özünün *a*, *b*, *c* lokal dəyişənləri üçün stekdə üç maşın sözü ölçüsündə yaddaş sahəsi tutur. Bunun üçün *f* funksiyası, sadəcə, *SP* registrinin qiymətini 12 vahid azaldır (üç maşın sözü 12 bayta bərabərdir). Lokal dəyişənlər kadrı tutulduqdan sonra stek aşağıdakı kimi görünür:



f funksiyasının arqumentləri və lokal dəyişənləri *FP* registrinə nəzərən ünvanlanırlar. Belə ki, *x* arqumentinin ünvanı *FP*+8, *y* arqumentinin ünvanı isə *FP*+12 olur. *a* dəyişəni *FP*–4, *b* dəyişəni *FP*–8, *c* dəyişəni isə *FP*–12 ünvanına malik olur.

f funksiyası işini qurtaran kimi özünün *a*, *b*, *c* lokal dəyişənlərini stekdən uzaqlaşdıraraq, stekin göstəricisini 12 vahid artırır. Sonra FP-nin köhnə qiyməti stekdən çıxarılaraq *FP*-yə yerləşdirilir (yəni *FP* registri *f* funksiyasının çağırılmasına qədərki qiymətini bərpa edir). Sonra

çağıran proqrama qayıdış baş verir: dönüş ünvanı stekdən çıxarılır və idarəetmə dönüş ünvanı üzrə ötürülür. *f* funksiyasının nəticəsi sıfır registrlə ötürülür. Çağıran proqramın özü *f* funksiyasının çağırılmasından əvvəl stekə yerləşdirilmiş *x* və *y* arqumentlərinin faktiki qiymətlərini stekdən uzaqlaşdırır.

STEK KALKULYATORU VƏ DÜSTURUN TƏRS POLYAK YAZILIŞI

1920-ci ildə Polşa riyaziyyatçısı Yan Lukaseviç riyazi düsturların mötərizələrdən istifadə etmədən yazılış üsulunu təklif etmişdir. Adət etdiyimiz yazılışda əməl işarəsi arqumentlər arasında yazılır: məsələn, 2 və 3 ədədlərinin cəmi 2+3 kimi yazılır. Yan Lukaseviç yazılışın iki başqa formasını təklif etmişdir: əməl işarəsinin arqumentlərdən əvvəl yazıldığı prefiks forması və əməl işarəsinin arqumentlərdən sonra yazıldığı postfiks forması. Riyaziyyatçının vətəninin şərəfinə bu yazılış formaları düz və tərs polyak yazılışı adlandırılmışdır. Adi yazılış formasına isə infiks forması deyilir.

Polyak yazılışında mötərizələrə gərək yoxdur. Məsələn,

$$(2 + 3) \times (15 - 7)$$

ifadəsi düz polyak yazılışında

$$\times + 2\ 3 - 15\ 7$$

kimi, tərs polyak yazılışında isə

$$2\ 3 + 15\ 7 - \times$$

kimi yazılır.

Düz polyak yazılışı geniş yayılmasa da, tərs polyak yazılışı olduqca faydalı oldu. Onun üstünlüyü onunla izah edilə bilər ki, üzərində əməliyyatlar aparılacaq obyektlər artıq verilmişsə, həmin əməliyyatı yerinə yetirmək olduqca əlverişli olur. Fərz edək ki, kimdənsə bir stəkan çay süzmək xahiş olunur. Tərs polyak yazılışında əvvəlcə stəkan, sonra çaydan verilir, ancaq bundan sonra çay süzmək xahiş olunur. İnfiks formasında əvvəlcə stəkan verilir, sonra çay süzmək xahiş olunur. Kimsə çaydan dalınca mətbəxə getməli və ancaq bundan sonra çay süzülməlidir. Düz polyak yazılışında əvvəlcə çay süzmək xahiş edilir, kimsə stəkan, sonra çaydan tapmalı və yalnız bundan sonra tələb olunan əməliyyatı yerinə yetirməlidir.



Yan Lukaseviç.

Düsturların tərs polyak yazılışı üxtiyari çətinlikli hesablamaları aparmağa imkan verir. Bu zaman aralıq nəticələrin saxlanılması üçün stekdən yaddaş qurğusu kimi istifadə edilir. Belə stek kalkulyatoru ilk dəfə Hewlett Packard (ABŞ) firması tərəfindən buraxılmışdır. Kalkulyatorların adi modelləri aralıq nəticələrin yazılması üçün kağız və qələmdən istifadə etmədən mürəkkəb düsturlara görə hesablamaların aparılmasına imkan vermir. Bəzi modellərdə bir, yaxud iki səviyyəli iç-içə mötərizələrdən istifadə nəzərdə tutulub, ancaq onlarda da daha mürəkkəb hesablamalar aparmaq mümkün deyil. Bundan başqa, adi kalkulyatorlarda giriş və hesablama proseslərində arqumentlərin və nəticənin kalkulyatorun registrləri arasında necə yerdəyişməsini başa düşmək çətindir. Əksinə, stek kalkulyatorunun quruluşu tamamilə aydındır və asan yadda qalır. Kalkulyatorun stek şəklində yaddaşı vardır. Ədəd daxil edilərkə o, sadəcə, stekə əlavə olunur. Əməl klavişi, məsələn, "+" klavişi basıldıqda əməliyyatın arqumentləri əvvəlcə stekdən çıxarılır, sonra onlar üzərində əməliyyat aparılır, nəhayət,





Hewlett Packard firmasının kalkulyatoru.

nəticə yenə də stekə yazılır. Stekin yuxarısı həmişə axırıncı əməliyyatın nəticəsini özündə saxlayır və o, kalkulyatorun displeyində işıqlanır.

Hesablama üçün əvvəlcə ifadəni tərs polyak yazılışına çevirmək lazımdır (müəyyən vərdişlərlə bunu beyində də etmək olar). Sonra yazıya ardıcıl olaraq soldan sağa baxılır. Ədəd görünürsə, onu kalkulyatora daxil edirik və beləliklə, onu stekə əlavə edirik. Əməl işarəsi görünürsə, kalkulyatorun uyğun klavişini basırıq. Bununla isə stekin yuxarısında olan ədədlərlə lazım olan əməliyyatı yerinə yetiririk.

Düsturların tərs polyak yazılışı kompüterlə işləmək üçün son dərəcə əlverişli idi. Hesablamalar üçün stekdən istifadə olunur ki, bu da istənilən mürəkkəb ifadələrlə işləməyə imkan verir. Stek hesablayıcısının reallaşdırılması heç bir çətinlik törətmir. Bundan başqa, ifadəni adi yazılışdan tərs polyak yazılışına çevirən sadə alqoritm yaradılmışdır. Bütün bunlar ona aparıb çıxardı ki, yüksək səviyyəli dillərin əksər kompilyatorları proqramların təqdim olunmasının daxili forması kimi tərs polyak yazılışından istifadə edir.

Java dilində proqramdan fraqment.

```
public class JEncoder extends Frame
{
    Image image;
    int outputArray[][] = new int[320][200];   // This is the
image, 0-255

    GrafixTools GT;
    DCT dctTrans;

    public JEncoder()
    {
        move(100,00);
        resize(320,240);
        repaint();

        // Load in an image
        image = Toolkit.getDefaultToolkit().getImage("dct.jpg");

        // Make sure it gets loaded
        MediaTracker tracker = new MediaTracker(this);
        tracker.addImage(image, 0);
        System.out.println("Initialized tracker.");
        try
        {
            System.out.println("Waiting for tracker...");
            tracker.waitForID(0);
        }
        catch (Exception e) {return;}
        System.out.println("Loaded image.");

        GT = new GrafixTools(image);
        System.out.println("Dimensions: " + GT.imageHeight + ","+
```

Məsələn, Java proqramlaşdırma dili kompilyasiya olunan deyil, interpretasiya olunan dildir. Bu o deməkdir ki, Java kompilyatoru ilkin Java-proqramını maşın koduna deyil, xüsusi Java-maşınında yerinə yetirilmək (interpretasiya olunmaq) üçün nəzərdə tutulmuş hər hansı aralıq dilə çevirir. Java halında bu aralıq dili bayt-kod adlandırırlar. Bayt-kodun Java-proqramının tərs polyak yazılışı, Java-maşının isə stek hesablayıcısı olduğunu hesab etmək olar.

POSTSCRIPT DİLİ

Tərs polyak yazılışından istifadəyə başqa bir parlaq misal, mətnlərin lazer printerlərində yüksək keyfiyyətli çapı üçün nəzərdə tutulmuş qrafik PostScript dilidir. O, printerin konkret modelindən asılı olmayaraq, mətnlərin mətbəə keyfiyyətində verilməsinin standartıdır.

"PostScript proqramlaşdırma dilidir" fikri mətbəə işilə tanış olan, amma proqramlaşdırmadan uzaq olan insanların çoxu üçün qeyri-adi səslənir. Ümumi qəbul edilmiş "kompüter ədədlərlə işləyir və əsasən nə isə hesablayır" fikri heç də həmişə doğru deyildir. Tez-tez kompüter proqramları mətnlər və görüntülərlə də işləyir. Mətn faylında yerləşən mətnə iki nöqteyi-nəzərdən baxmaq olar. Məsələn, onu sadəcə məqalə, yaxud kitab mətni kimi şərh etmək olar. Lakin mətnin çapı prosesinə qrafik deyil, adi printerdə baxaq. Printer kompüterlə kabel vasitəsilə birləşmişdir və

kompüter bu kabellə bir-birinin ardınca mətni təşkil edən simvolların kodlarını printerə göndərir.

Bu halda mətndə "A" adlanan hərfə komanda kimi baxmaq lazımdır. Bu komanda printerə əvvəlcədən verilmiş şriftdən istifadə edərək "A" simvolunu səhifənin cari nöqtəsindən çap etmək göstərişini verir. Sonra cari nöqtənin X koordinatını "A" hərfinin eni qədər artırmaq lazımdır. Bu mövqedən bütün mətni məxsusi olaraq özünü çap edən proqram kimi şərh etmək olar. İxtiyari mətn faylı halında bu proqram çox primitivdir. Məsələn, onda şriftlərin dəyişdirilməsi, cari mövqeyin dəyişdirilməsi, xətlərin çəkilməsi və sair komandalar yoxdur. Aydındır ki, mətbəə keyfiyyətli mətn mətn faylı ilə təqdim oluna bilməz.

PostScript dili halında PostScript-printerə göndərilən fayl mətni çap edən proqramdır. PostScript dili olduqca zəngindir və çətin ki, onu bilən çoxlu insan tapılsın. Çox zaman PostScript-proqram mətnləri emal edən digər proqram vasitəsilə yaradılır. Məsələn, PostScript-fayl printerdə çap üçün TeX dilinin köməyilə yaradılır (TeX riyazi düsturlardan ibarət mətnlərin yazılması üçün, məşhur riyaziyyatçı və proqramlaşdırma nəzəriyyəçisi Donald Knut tərəfindən yaradılmış dildir). Bir çox mətn redaktorları, məsələn, Adobe Acrobat, yaxud MS Word peşəkar printerə çap edəndə mətni PostScript-proqrama çevirirlər. PostScript-fayllar adi mətn formatında yazıldıqlarına görə yayılmaq üçün çox əlverişlidir və milli kodlaşdırmadan, əməliyyat sisteminin versiyasından, lazımi şriftlərin mövcudluğundan və asılı olmayaraq istənilən ölkədə eyni cür çap olunacaqdır.

PostScript-proqram tərs polyak yazılışıdır, yəni ixtiyari komanda öz arqumentlərindən sonra yazılır. PostScript-proqramın yerinə yetirilməsi zamanı stekdən istifadə olunur. Nümunə üçün səhifədə xaç çəkən sadə proqrama baxaq. PostScript dilində % simvolu şərhlərin yazılışı üçün istifadə olunur.

```
% Xaç proqramı
2.83 2.83 scale % millimetrlər şkalasını
qurmaq
0.2 setlinewidth % xəttin qalınlığını
0.2 mm qoymaq
100 0 moveto % (100, 0) nöqtəsinə
yerdəyişmə
100 350 lineto % (100, 350) xətti
nöqtəsinə aparmaq
stroke % və onu çəkmək
0 150 moveto % (0, 150) nöqtəsinə
yerdəyişmə
250 150 lineto (250, 150) xətti nöqtəsinə
aparmaq
stroke % və onu çəkmək
showpage % səhifəni göstərmək
```



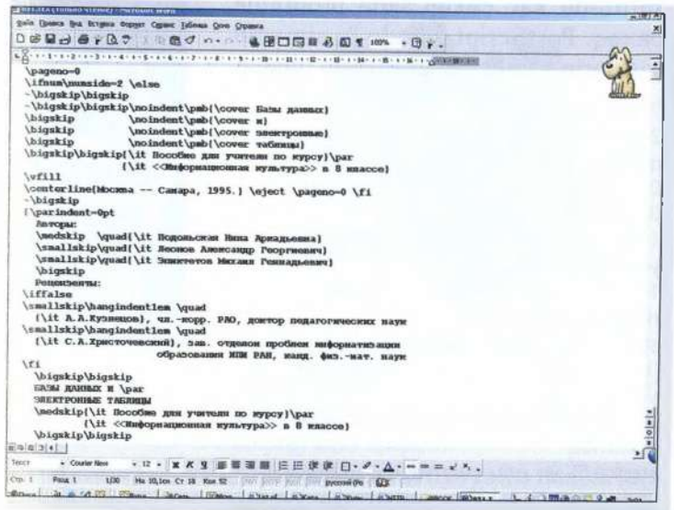
Donald Knut.

Proqramın birinci sətrində millimetrlər şkalası qurulur. İş ondadır ki, printer üçün ölçü vahidi susqunluqla 1 punkt, yaxud 1/72 düym qəbul edilmişdir. İstənilən proqramçı bilir ki, 1 düym 25,4 millimetrə bərabərdir. Bir millimetr və bir punkt arasındakı əlaqəni hesablamaq lazımdır:

1 mm / 1 pt = 1 mm / (1/72) =
1 mm / (25,4/72) mm = 2,834645.

Əgər miqyas 2,83 dəfə artırılsa, punktlardan millimetrlərə keçid baş verir. Miqyasın dəyişdirilməsi üçün əvvəlcə stekə miqyasın *X* üzrə artırılmasına uyğun 2,83 ədədi yerləşdirilir. Sonra miqyasın *Y* üzrə artırılmasına uyğun ikinci 2,83 ədədi stekə yerləşdirilir. Bundan sonra scale (miqyasın dəyişdirilməsi) komandası yerinə yetirilir. Bu zaman stekdən iki ədəd çıxarılır, *X* və *Y* üzrə miqyas dəyişdirilir.

Müxtəlif komandalar müxtəlif sayda arqumentlərə malik ola bilər. Məsələn, proqramın ikinci sətri xəttin qalınlığını quraşdırır. Setlinewidth komandasının bir arqumenti vardır. Komandanın çağırılmasından qabaq həmin arqument stekə yerləşdirilir. Komanda yerinə yetirilən zaman o, stekdən çıxarılır və xəttin qalın-



TeX dilində proqramdan fraqment.

lığı bu çıxarılmış ədədə bərabər götürülür.

Üçüncü sətir kursorun cari mövqeyini koordinatları $X = 100$, $Y = 0$ olan nöqtəyə dəyişdirir (vahid olaraq millimetr götürülür ki, birinci sətirdə bu nəzərə alınmışdır; koordinat başlanğıcı səhifənin aşağı sol küncündə yerləşir).

Əvvəlcə stekə X və Y koordonatlarına uyğun iki ədəd əlavə olunur, sonra moveto komandası yerinə yetirilir. Bu komanda iki ədədi stekdən çıxararaq kursorun yerini, koordinatları həmin ədədlər olan nöqtəyə dəyişdirir.

PostScript kifayət qədər güclü proqramlaşdırma dilidir. Onda parametrli funksiyalardan, dövrlərdən, rekursiya və başqa vasitələrdən istifadə olunmuşdur. Onda istifadə olunan tərs polyak yazılışı kompüterdə yerinə yetirilmək üçün çox rahatdır. Ona görə də PostScript dilinin interpretatoru asanlıqla, printerin dəyərini bir qədər artıraraq onun konstruksiyasına quraşdırıla bilir.

Bayt-kodu tərs polyak yazılışı olan Java dili də ilkin olaraq ucuz məişət cihazlarının (televizor, paltaryuyan maşın və sair) proqramlaşdırılması üçün yaradılmışdır. Stek hesablayıcısı sadə qurulmuşdur və sürətin xüsusi rol oynamadığı, əvəzində proqramların yaradılma sürətinin və ucuzluğun vacib olduğu yerlərdə tətbiq oluna bilər.

VERİLƏNLƏRİN DİNAMİK STRUKTURLARI VƏ İSTİNADLI REALLAŞDIRMALAR

Bütün verilənlər strukturlarını şərti olaraq iki sinfə ayırmaq olar: ardıcıl və ixtiyari irişimli strukturlar. Birinci strukturlarda bir və ya bir neçə elementə irişmək mümkündür. Adətən bu elementlər strukturun əvvəlində, yaxud axırında olur, bəzən isə xüsusi "göstərici" ilə qeyd olunmuş "ayrılmış" yerin yaxınlığında olur. Növbələrdə ancaq onun əvvəlində və axırındakı elementlər, stekdə isə ancaq onun təpəsindəki element irişiləbiləndir. İxtiyari, yaxud "birbaşa" irişimli strukturlarda bütün elementlərlə istənilən hərəkəti yerinə yetirmək olar. Bu zaman bütün elementlərin, yaxud elementlərin müəyyən hissəsinin seçilməsinə heç bir ehtiyac yoxdur.

Verilənlər strukturlarının əksəriyyəti massiv bazasında reallaşdırılır. Bütün reallaşdırmaları da iki sinfə ayırmaq olar: kəsilməz və istinadlı. Kəsilməz reallaşdırmalarda verilənlər strukturunun elementləri ardıcıl olaraq, bir-birinin ardınca massivin kəsilməz hissəsində yerləşirlər. Özü də onların yerləşmə ardıcıllığı reallaşdırılan strukturdakı ardıcıllığına uyğun gəlir. İstinadlı reallaşdırmalarda isə verilənlər strukturunun elementləri tamamilə nizamsız, ixtiyari şəkildə saxlanılırlar. Bu zaman hər bir elementlə yanaşı onun bir və ya bir neçə "qonşu" elementlərinə istinadlar da saxlanılır. İstinad rolunda massivin oyuqlarının indeksləri, yaxud yaddaşın ünvan-

ları çıxış edə bilər. Casuslar şəbəkəsini təsəvvür edək, belə ki, hər bir iştirakçı-casus ancaq bir, yaxud iki həmkarının koordinatlarını bilir. Şəbəkəni zərərsizləşdirmək istəyən əkskəşfiyyatçılar aşkar olunmuş casusdan başlayaraq, ardıcıl olaraq bütün zənciri keçməlidirlər.

İstinadlı reallaşdırmalar aşkar ifadə edilmiş iki çatışmazlığa malikdir:

1) istinadların yadda saxlanılması üçün əlavə yaddaş tələb olunur;

2) strukturun müəyyən elementinə çatmaq üçün əvvəlcə ardıcıl olaraq digər elementlər zəncirini keçmək lazımdır.

Elə görünə bilər ki, belə reallaşdırmalar nəyə lazımdır? İstinadlı reallaşdırmaların bütün çatışmazlıqları onların olduqca vacib bir yaxşı cəhətilə kompensasiya olunur: onlarda verilənlər strukturunun başqa elementlərini sürüşdürmədən, onun ortasına elementlər artırmaq və uzaqlaşdırmaq olar.

KÜTLƏVİ ƏMƏLİYYATLAR

Verilənlər strukturunun bütün elementlərinin önəmli bir hissəsinə toxunan əməliyyatlara kütləvi əməliyyatlar deyilir. Fərz edək ki, bir elementi əlavə etmək və ya uzaqlaşdırmaq lazımdır. Əgər bu zaman qalan elementlərin xeyli hissəsini bir yerdən başqa yerə yazmaq lazım gəlirsə, deyirlər ki, artırma, yaxud uzaqlaşdırma kütləvi əməliyyatlara səbəb olur. Verilənlər strukturu o zaman yaxşı reallaşdırılmış olur ki, belə əməliyyatlar ya tamamilə olmur, ya da çox nadir hallarda baş verir. Kütləvi əməliyyatlar proqramçının həmişə qaçmağa çalışdığı lənətdir. Kəsilməz reallaşdırmalarda strukturun ortasına elementlərin artırılması, yaxud uzaqlaşdırılmanı hökmən kütləvi əməliyyatlara gətirir. Ona görə də, ortasına elementlərin artırılması və ya uzaqlaşdırılması mümkün olan strukturlar mütləq istinadlı formada reallaşdırılmalıdır.

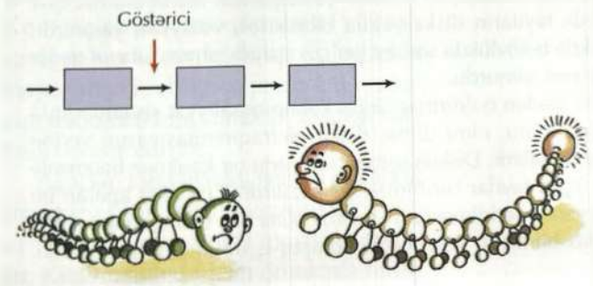


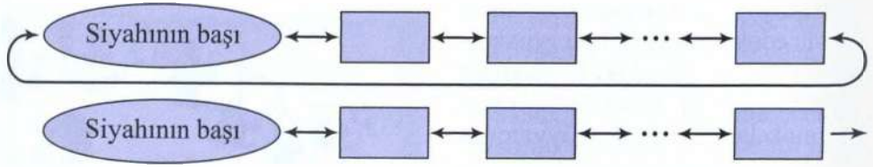
SİYAHI

Ortasına elementlərin artırılması və uzaqlaşdırılması mümkün olan ardıcıl irişimli verilənlər strukturuna klassik nümunə xətti siyahıdır. Siyahılar biristiqamətli və ikiistiqamətli olurlar (bəzən "birrabitəli" və "ikirabitəli" deyirlər).

Siyahının elementləri elə bil ki, birbirinin ardınca zəncir kimi düzülmüşlər. Siyahının başlanğıcı və sonu olur. Həmçinin, elementlər arasında yerləşən siyahı göstəricisi olur.

Təsəvvür etsək ki, siyahının qonşu elementləri kəndirlə bir-birilə əlaqələndirilmişdir, onda kəndirə asılan qırmızı lent göstərici olacaqdır. İstənilən zaman siyahıda ancaq iki elementə giriş vardır: göstəriciyə qədərki və göstəricidən sonrakı elementə. Biristiqamətli siyahıda göstəricini ancaq irəliyə doğru, başlanğıcdan sona doğru hərəkət etdirmək olar. Göstəricini siyahının başlanğıcında, birinci elementdən qabaq da qoymaq olar.





Biristiqamətli siyahıdan fərqli olaraq, ikiistiqamətli siyahı bu baxımdan simmetrikdir. Yəni göstərici həm irəli, həm də geri hərəkət edir və birinci elementdən qabaq, həm də axırıncı elementdən sonra qoyulur.

İkiistiqamətli siyahıda elementləri göstəricidən əvvəl, həm də sonra əlavə etmək, yaxud uzaqlaşdırmaq olar. Biristiqamətli siyahılarda da elementləri göstəricinin hər iki tərəfinə əlavə etmək olar, ancaq göstəricidən sonrakı elementləri uzaqlaşdırmaq olar.

Hesab etmək olar ki, siyahının birinci elementindən qabaq xüsusi "boş" element yerləşir. Bu elementi "siyahının başı" da adlandırırlar. Baş həmişə, hətta siyahı boş olduqda da olur. Buna görə də həmişə hesab etmək olar ki, göstəricidən əvvəl hansısa element var. Bu da elementlərin əlavə olunmasını və uzaqlaşdırılması prosedurunu sadələşdirir.

İkiistiqamətli siyahıda hesab edilir ki, axırıncı elementdən sonra, yenə də siyahının başı gəlir, yəni siyahı qapanmış dairəyə qapanmışdır.

Eyni qayda ilə biristiqamətli siyahını da qapamaq olardı. Lakin çox vaxt hesab edilir ki, biristiqamətli siyahının axırıncı elementindən sonra heç nə gəlmir. Beləliklə, bu siyahı "heç bir yerə" istinadı ilə qurtaran zəncirdir.

SİYAHININ İSTİNADLI REALLAŞDIRILMASI

Proqramlaşdırmada tez-tez siyahı anlayışını onun massiv, yaxud bilavasitə operativ yaddaş bazasında istinadlı reallaşdırılması ilə eyniləşdirirlər.

İkiistiqamətli siyahının reallaşdırılmasının əsas ideyası ondan ibarətdir ki, hər bir elementlə birlikdə ondan əvvəlki və sonrakı elementlərə istinadlar da saxlanılır. Massiv bazasında reallaşdırma zamanı istinadlar massivin oyuqlarının indeksləri olur. Lakin tez-tez siyahının elementlərini hər hansı massivdə deyil, verilmiş məsələ üçün ayrılmış *dinamik yaddaşda* ayrı-ayrılıqda yerləşdirirlər. Bu halda istinad kimi operativ yaddaşda elementlərin ünvanlarından istifadə olunur.

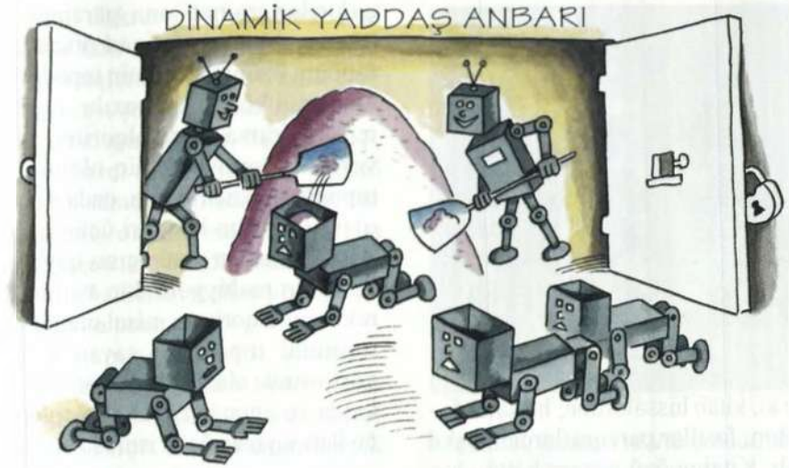
Siyahının başı birinci və axırıncı elementlərə istinadları saxlayır, amma heç bir element saxlamır. Siyahının başından sonra onun birinci elementi, əvvəl isə axırıncı elementi olacaqdır. Siyahı boş olduqda siyahının başı özü özünə dövrlənmiş olur.

Siyahının göstəricisi əvvəlki və sonrakı elementlərə istinadlar cütü şəklində

KƏSİLMƏZLİK PROBLEMLƏRİ

Kəsilməz reallaşdırmalardan uğursuz istifadəyə misal olaraq, bəzi köhnə əməliyyat sistemlərində (OC EC, CM ƏBM mini kompüterlərində tətbiq olunan RAFOS) fayl sistemlərini göstərmək olar. Müasir fayl sistemlərində fayllar fraqmentləşdirilir (böyük faylların hissələri bütün disk boyu "paylana" bilir), amma köhnə əməliyyat sistemlərində belə deyildi. Fayllar diskdə mütləq kəsilməz sahələrdə yerləşirdilər. Daimi iş zamanı onlar məhv edilərək yenidən yeni yerdə yaradılırdılar. Mətn faylları üzərində redaktə əməliyyatlarının aparılması zamanı onlar beləcə təzələnirdi. Nəticədə diskdə olan boş yerlər fraqmentləşdirilmiş olurdu, yəni böyük olmayan hissələr çoxluğundan ibarət olurdu. Çox tez-tez böyük faylların diskə yazıla bilməməsi vəziyyəti yaranırdı, halbuki, diskdə bütövlükdə sərbəst yer çox qalırdı, amma kifayət qədər böyük fraqment olmurdu.

Problemi aradan qaldırmaq üçün köhnə əməliyyat sistemlərində sıxma prosedurunu, elmi dildə, diskin defraqmentasiyasını yerinə yetirmək lazım gəlirdi. Diskdə sərbəst sahələrin bir kəsilməz fraqmentə çevrilməsi üçün fayllar bir-birinə sıx yazılırdı. Çox vaxt aparan bu prosedurun yerinə yetirilməsi zamanı kompüter heç bir başqa əməliyyat yerinə yetirə bilmirdi, yəni faktiki olaraq iş üçün yararsız olurdu.



reallaşdırılır. O yalnız elementlər zəncirində müəyyən yeri qeyd edir.

Biristiqamətli siyahılarda hər bir elementlə birlikdə yalnız növbəti elementə istinad saxlanılır, beləliklə, yaddaşa qənaət olunur. Siyahının başı birinci elementə istinadı, axırıncı element isə sıfır istinadını ("heç bir yerə" istinadı) saxlayır, çünki proqramlarda sıfır ünvan heç vaxt istifadə olunmur.

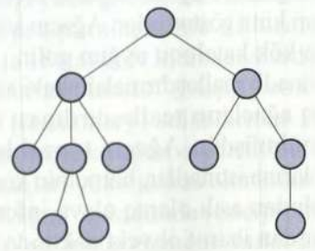
Siyahının elementlərinin əlavə olunması və uzaqlaşdırılması proseduru kütləvi əməliyyatlara səbəb olmur, çünki onların yerinə yetirilməsi üçün təkcə iki qonşu elementdə istinadları dəyişdirmək lazımdır.

AĞACLAR VƏ QRAFLAR

Qraf təpələrdən və təpələri birləşdirən tillərdən ibarət olan fiqurdur. Qrafa nümunə olaraq metro xətlərinin sxemini göstərmək olar. Tillər istiqamətə malik ola bilərlər, başqa sözlə oxlarla təsvir oluna bilərlər; belə qraflar yönəldilmiş qraflar adlanır.

Ağac, dövrlərsiz əlaqəli qrafdır. Bundan başqa, onda ağacın kökü adlanan bir təpə seçdirilmişdir. Qalan təpələr ağacın kökündən verilmiş təpəyə aparan yol boyunca nizamlanır.

Fərz edək ki, hər hansı X təpəsi seçilmişdir. Tillər vasitəsilə X təpəsilə birləşmiş və ağacın kökündən sonra yerləşmiş təpələri X təpəsinin "oğlanları" adlandırırlar. Onlar, adətən, soldan sağa nizamlanırlar. Oğlanları olmayan təpələr terminal təpələr adlanır. Ağacı çevrilmiş, kökü yuxarı təsvir edirlər.



Ağaclar proqramlaşdırmada qrafın daha tez-tez istifadə olunan növüdür. Çoxsaylı çeşidləmə və axtarış alqoritmləri onların qurulmasına əsaslanmışdır. Kompilyatorlar proqramların yüksək səviyyəli proqramlaşdırma dillərindən maşın dilinə keçirilməsi prosesində proqram fraqmentlərini sintaktik ağaclar şəklində təsvir edirlər.

Ağaclar, təbii olaraq, harada iyerarxik strukturlar (bir-birinə yerləşə də bilən strukturlar) varsa, orada meydana çıxırlar. Kitabın mündəricatı örnək ola bilər.

Dinamik yaddaş yaxud, "qalaq" operativ yaddaşın elə sahəsidir ki, lazım olduqda onun lazımi ölçüdə hissələrini tutmaq olar və istifadə etdikdən sonra azad etmək, yəni geriyə "qalağa" qaytarmaq olar.



Tutaq ki, kitab hissələrdən, hissələr fəsillərdən, fəsillər paraqraflardan təşkil olunub. Kitabın özü ağacın kökü olur, kökdən kitabın hissələrinə uyğun gələn təpələrə tillər çıxır. Kitabın hissələrinə uyğun hər bir təpədən bu hissənin fəsillərinə uyğun təpələrə tillər çıxır və sair.

Kompüterin fayl sistemini də ağac şəklində təsvir etmək olar. Təpələrə kataloqlar (onları direktoriyalar, yaxud qovluqlar da adlandırırlar) və fayllar uyğun gəlir. Fayllar ağacın terminal təpələri kimi göstərilirlər. Ağacın kökünə diskin kök kataloqu uyğun gəlir.

İstinadlı reallaşdırmalar sanki xüsusi olaraq ağacların reallaşdırılması üçün düşünülmüşdür. Ağacın təpəsi bütün oğlanlarına istinadlar, həmçinin konkret məsələdən asılı olaraq əlavə informasiyalardan ibarət obyekt şəklində təqdim olunur. Belə obyekt, adətən, dinamik yaddaşda yerləşən, qeyd olunmuş ölçüdə yaddaş sahəsi tutur. Oğlanların sayı, bir qayda olaraq, həll olunan məsələnin mənasından asılıdır. Çox tez-tez binar ağaclara baxılır. Belə ağacların ixtiyari təpəsində oğulların sayı ikini aşmır. Əgər təpədə bir, yaxud bir neçə oğul yoxdursa, onda uyğun istinadlar sıfır qiymət alırlar. Terminal təpələrdə bütün istinadlar sıfırdır.

Ağaclarla işləyən zaman rekursiv alqoritmlərdən (özü özünü çağıra bilən alqoritmlər) istifadə olunur. Alqoritm çağırılan zaman ona parametr kimi ağacın təpəsinə istinad ötürülür. Bu təpənin özünə isə həmin təpədə bitmiş altağacın kökü kimi baxılır. Əgər təpə terminaldırsa, onda alqoritm, sadəcə, verilmiş təpəyə tətbiq olunur. Əgər təpənin oğlanları varsa, onda o rekursiv olaraq onların hər biri üçün çağırılır. Altağaclardan yankeçmə qaydası alqoritmin mahiyyətindən asılıdır. Sadə rekursiv alqoritmə misal olaraq ağacın terminal təpələrini sayan alqoritmi göstərmək olar. Ondan verilmiş kataloqda və onun bütün altkataloqlarındakı faylları saymaq üçün istifadə etmək olar.

```
alq tam terminal təpələrin sayı
        (arg V təpəsi)
verilib V – ağacın kökünə istinad
gərəkdir Ağacın terminal təpələrinin
        sayını hesablamaq
baş
| əgər V təpəsinin oğulları yoxdursa
| | onda qiym := 1
| | əks halda
| | qiym := 0
| | db V təpəsinin hər bir X oğlu üçün
| | | qiym := qiym + terminal təpələrin
| | | | sayı (X)
| | ds
| tamam
son
```

ÇOXLUQ

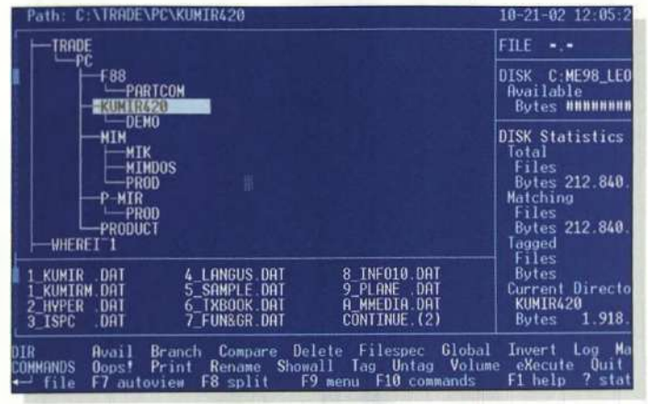
Çoxluq, hər hansı tip sonlu elementlər yığınından ibarət verilənlər strukturudur. Çoxluğun elementləri qətiyyən nizamlanmamışdır və öz aralarında bərabər ola bilməzlər. *M* çoxluğuna *x* elementini əlavə etmək və onu oradan uzaqlaşdırmaq da olar. Əgər elementi əlavə edən zaman, o orada varsa, heç nə baş vermir. Çoxluq potensial olaraq qeyri-məhdud strukturdur və ixtiyari sonlu sayda elementi özündə saxlaya bilər.

Bəzi proqramlaşdırma dillərində çoxluğun elementlərinin tipinə və maksimal sayına məhdudiyyətlər qoyulur. Bəzən

elementləri diskret tipli (bütün mövcud qiymətlər tam ədədlərlə nömrələnə bilən) çoxluqlara baxırlar. Orada elementlərin sayı çoxluq yaradılan zaman verilmiş sabiti aşmamalıdır. Belə çoxluqlar üçün Bitset (bitlər yığını), yaxud sadəcə, Set adından istifadə edirlər. Bir qayda olaraq, onların reallaşdırılması üçün çoxluğun tam ədədlər massivi bazasında "bit" adlanan reallaşdırılmasından istifadə olunur.

Hər bir tam ədədə ikilik göstərilmədə 32 elementdən ibarət bitlər yığını kimi baxılır. Bir ədəd daxilində bitlər sağdan sola (kiçik mərtəbədən böyük mərtəbəyə) nömrələnir; biz massivin elementlərini seçib ayırdıqca, bitlərin nömrələnməsi bir ədəddən digərinə davam edir. Məsələn, 10 tam ədəddən ibarət massiv nömrələri 0-dan 319-dək dəyişən 320 bitdən ibarətdir. Verilmiş reallaşdırmada çoxluq 0-dan 319-dək aralığında ixtiyari tam ədədlər yığınını özündə saxlaya bilər. *N* ədədi çoxluqda o zaman və ancaq o zaman yerləşir ki, *N* nömrəli bit vahidə bərabərdir. Uyğun olaraq, əgər *N* çoxluqda yerləşməmişdirsə, *N* nömrəli bit sıfra bərabərdir.

Baxmayaraq ki, Pascal proqramlaşdırma dilində Set sözü (tərcüməsi "çoxluq") elementləri diskret tipli məhdud çoxluqlara bağlanmışdır, proqramlaşdırmanın tələbatları belə çoxluqlarla bitmir.



Məsələn, ümumi növlü çoxluqlar üçün bit reallaşdırması yaramır.

Proqramlaşdırmada çox tez-tez sadə çoxluqlara nisbətən daha mürəkkəb strukturlara, "yüklənmiş çoxluqlara" baxırlar. Tutaq ki, çoxluğun hər bir elementi əlavə informasiya ilə birlikdə ona daxildir. Belə əlavə informasiya elementin "yükü" adlanır. Elementin çoxluğa əlavə olunması zamanı hökmən onun daşıdığı yükü göstərmək lazımdır. Müxtəlif proqramlaşdırma dillərində və müxtəlif standart kitabxanalarda belə strukturları "İnikas" (Map), yaxud "Lüğət" (Dictionary) adlandırırlar. Həqiqətən də, çoxluğun elementləri sanki daşıdıqları yükə inikas olunurlar.

"Lüğət"in izahında (interpretasiyasında) çoxluğun elementi xarici sözdür,

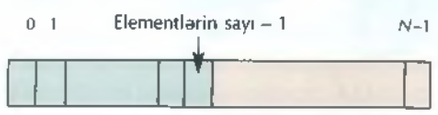




elementin yükü isə həmin sözün Azərbaycan dilinə tərcüməsidir. Bütün elementlər yüklənmiş çoxluqda bir nüsxədə saxlanılır, yəni çoxluğun ayrı-ayrı elementləri bir-birinə bərabər ola bilməzlər. Elementlərin özlərindən fərqli olaraq, onların yükləri üst-üstə düşə bilər (müxtəlif xarici sözlər eyni tərcüməyə malik ola bilər). Ona görə də bəzən yüklənmiş çoxluğun elementlərini açarlar, onların yüklərini isə açarların qiymətləri adlandırırlar. Hər bir açar bənzərsizdir. Açarlar onların qiymətlərinə "inikas olunurlar" demək qəbul olunmuşdur.

Yüklənmiş çoxluğa nümunə kimi "Lüğət"lə yanaşı bank hesabları çoxluğuna da baxmaq olar. Bank hesabı 14 ədəd onluq rəqəmdən ibarət olan təkrarolunmaz identifikatordur. Hesabın yükü hesaba uyğun bütün informasiyalardır (hesab sahibinin adı və ünvanı, valyutanın kodu, qalığın cəmi və sair).

Yüklənmiş çoxluqlarda daha tez-tez tətbiq olunan əməliyyat verilmiş x elementi üçün onun yükünün (x açarının qiymətinin) təyin olunmasıdır. Verilmiş



əməliyyatın reallaşdırılması x elementinin çoxluqda axtarışını da nəzərdə tutur. Ona görə də çoxluğun ixtiyari reallaşdırılmasının effektivliyi hər şeydən qabaq onun elementinin axtarışının sürətilə təyin olunur.

ÇOXLUĞUN REALLAŞDIRILMALARI: ARDICIL VƏ BİNAR AXTARIŞ, DOĞRAMA

Sadəlik üçün yüklənməmiş, adi çoxluğun reallaşdırılmasına baxmaq olar. Onun elementləri birinci oyuqdan başlayaraq massivdə ixtiyari ardıcıllıqla yerləşir. Xüsusi dəyişəndə isə çoxluğun cari elementlərinin sayı, yəni massivin verilmiş anda istifadə olunan oyuqlarının sayı yerləşir. x elementinin axtarışı zamanı ardıcıl olaraq bütün elementlərə baxılacaqdır: ya o tapılacaq, ya da onun çoxluqda olmaması aydınlaşacaqdır.

Çoxluğa x elementinin əlavə edilməsi zamanı biz əvvəlcə onun artıq orada olub-olmamasını müəyyənləşdirməliyik. Bunun üçün axtarış alqoritmindən istifadə olunur. Əgər x elementi çoxluqda varsa, onda onun yerləşdiyi massiv oyuğunun indeksi çıxışa verilir, əks halda o sadəcə sona yazılır (massivin "elementlərin sayı" indeksli oyuğuna) və "elementlərin sayı" dəyişəni bir vahid artırılır. Elementin uzaqlaşdırılması üçün çoxluğun axırıncı elementini uzaqlaşdıran elementin yerinə yazmaq və "elementlərin sayı" dəyişənini bir vahid azaltmaq kifayətdir.

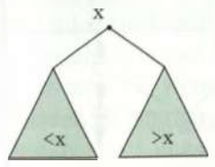
Belə reallaşdırma ancaq böyük olmayan çoxluqlar üçün yararlıdır.

BİNAR AXTARIŞ

Fərz edək ki, çoxluğun elementlərini bir-birilə müqayisə etməklə hansının boyük olduğunu müəyyənləşdirmək olar (məsələn, mətn sətirləri üçün leksikoqrafik müqayisə tətbiq olunur: birinci hərflər əlifba üzrə müqayisə olunur: əgər onlar bərabərdirsə, onda ikinci hərflər müqayisə olunur və sair). Onda binar axtarış alqoritmini tətbiq etməklə axtarışı hiss olunan dərəcədə sürətləndirmək olar. Bunun üçün çoxluğun elementlərini massivdə artan sıra ilə saxlayırlar.

Axtarışın hər addımında massivdə axtarılan x elementinin yerləşə biləcəyi parça yarıya bölünür. Parçanın ortasındakı y elementinə baxılır. Əgər x elementi y-dən kiçikdirsə, parçanın sol yarısı, əks halda isə sağ yarısı götürülür. Axtarış yeni parçada davam etdirilir.

Proqram baxımından binar axtarışın reallaşdırılması heç bir çətinlik törətmir. Ona görə də o, hər cəhətdən "sadəlövh" reallaşdırmadan üstündür. Bununla birlikdə onun çatışmazlıqları da var: 1) elementlərin massivin ortasına əlavə olunması və uzaqlaşdırılması zamanı onun sonundakı elementləri yeni yerə "sürüşdürmək" lazım gəlir; 2) axtarış lazım olduğu qədər tez yerinə yetirilir, amma hər halda ani olmur. Çatışmazlıqların birincisindən kəsilməz reallaşdırma əvəzinə massiv bazasında istinadlı reallaşdırmadan istifadə etməklə yaxa qurtarmaq olar. Bu zaman çoxluğun elementləri binar ağacın buğumlarında yerləşir. Təpələrdəki elementlər aşağıdakı formada nizamlanıblar: ixtiyari təpədə x elementini qeyd edək və bu təpədən çıxan iki altağaca baxaq. Sol altağacın bütün elementləri x-dən kiçik, sağ altağacın elementləri isə x-dən böyük olmalıdır. Axtarışın sürətlə yerinə yetirilməsi üçün ağac "balanslaşdırılmış" olmalıdır, yəni onun bütün budaqları





təxminən eyni uzunluğa malik olmalıdır. Elementlərin əlavə olunması və uzaqlaşdırılması balanslaşdırılma xassəsisini saxlamalıdır.

DOĞRAMA

Doğrama ideyası ondan ibarətdir ki, bir böyük çoxluqla olan iş böyük olmayan altçoxluqlar massivindən ibarət iş halına salınır. Məsələn, qeyd dəftərçəsində insanların soyadlarının siyahısı onların telefonları ilə birlikdə olur (telefonlar çoxluğun elementlərinin "yükü"dür). Qeyd dəftərçəsinin səhifələri əlifbanın hərflərilə nişanlanmışdır. Bütün soyadlar çoxluğu Azərbaycan dili əlifbasına uyğun olaraq 30 altçoxluğa bölünmüşdür ("ğ" və "ı" hərfləri nəzərə alınmamışdır). Soyadlarının axtarışı zamanı o dərhal qeyd dəftərçəsinin lazımi səhifəsi açılır və axtarış hiss olunan dərəcədə sürətlənir.

Çoxluğun altçoxluqlara bölünməsi doğrama-funksiya vasitəsilə həyata keçirilir. Doğrama-funksiyası çoxluğun elementlərində təyin olunan və mənfi olmayan tam qiymətlər olan funksiyadır.



Binar axtarış ideyası "Şiri səhrada necə tutmalı?" əyləncəli məsələsilə əyani göstərilə bilər. Əvvəlcə hasarla səhranı iki yerə bölmək lazımdır. Sonra şirin qaldığı hissənin yenidən hasarla iki yerə bölünməsi və bu qayda ilə şir tutulana qədər prosesin davam etdirilməsi lazımdır.

Doğrama-funksiyası elə seçilməlidir ki: 1) onu asan hesablamaq olsun; 2) o, təxminən bərabər ehtimallı və müxtəlif olan bütün mümkün qiymətləri alsın. Məsələn, verilmiş soyad üçün doğrama-funksiyası birinci hərfin Azərbaycan əlifbasındakı nömrəsini qiymət olaraq alır (proqramlaşdırmada həmişə nömrə sıfırdan başlayır).

Hər şeydən öncə doğrama-cədvəli adlanan altçoxluqlar massivini qurmaq lazımdır. Bu zaman altçoxluqların sayının, başqa sözlə doğrama-cədvəlinin ölçülərinin təyin olunması vacibdir. Effektiv reallaşdırma üçün hər bir altçoxluqda imkan dairəsində birdən çox element olmamalıdır. Deməli, doğrama-cədvəlinin ölçüsü (tutaq ki, o N-ə bərabərdir) çoxluğun elementlərinin orta sayından böyük olmalıdır. Adətən N-i çoxluğun elementlərinin sayından üç dəfə artıq seçirlər. Qeyd dəftərçəsi misalında bu onu göstərir ki, 10-a yaxın soyad yazılmalıdır. Bu halda səhifələrin əksəriyyəti ya boşdur, ya da bir soyaddan ibarətdir. Bir səhifədə bir neçə soyadın yazılması halı da istisna deyil.

Beləliklə, M çoxluğu 0-dan N-1-dək indekslərlə nömrələnmiş N sayda kəsişməyən altçoxluğa bölünür. İndeksi i olan altçoxluq M çoxluğunun doğrama-funksiyasının qiyməti i-yə bərabər olan bütün x elementlərini özündə saxlayır.

İndi doğrama-funksiyasını seçmək lazımdır. Bu funksiya 0, 1, 2, ..., N-1 qiymətlərini almalıdır. Əgər başlanğıcdan ixtiyari tam qiymətlər alan hər hansı $H(x)$ funksiyası varsa, onda dəhrələmə-funksiyası olaraq $H(x)$-in N-ə bölünməsindən alınan qalığa bərabər olan $b(x)$ funksiyasından istifadə etmək lazımdır.

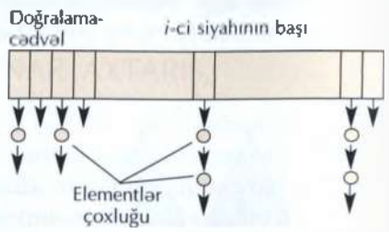


x elementinin axtarışı zamanı əvvəlcə onun doğrama-funksiyasının qiyməti hesablanır. Tutaq ki, o, b-yə bərabərdir. Sonra x indeksi b olan altçoxluqda axtarılır. Bu altçoxluq böyük olmadığından axtarış sürətlə həyata keçir.

Doğrama-reallaşdırmasının müxtəlif konkret sxemləri altçoxluqlar massivini müxtəlif cür təşkil edirlər və toqquşmalarla mübarizə aparırlar. İstinadlı reallaşdırmadan istifadə edən, daha çox universal sxem belə görünür: hər bir altçoxluq bu çoxluğun elementlərini saxlayan, xətti biristiqamətli siyahı şəklində təqdim olunur.

Başqa sözlə, N siyahı vardır. Bütün siyahıların başı doğrama-cədvəlində saxlanılır. Doğrama-cədvəlinin i indeksli elementi i-ci siyahının başıdır. Siyahının başı siyahının birinci elementinə istinad, birinci element ikinciyə istinad saxlayır və sair. Zəncirdə axırıncı element sıfır istinad saxlayır. Əgər siyahı boşdursa, doğrama-cədvəlinin i indeksli elementi sıfır istinad saxlayır.





"HƏR BİRİ ÜÇÜN DÖVRLƏR" VƏ İTERATORLAR

Tez-tez verilənlər strukturunun bütün elementlərinə bir-bir baxmaq və hər bir element üçün müəyyən hərəkəti yerinə yetirmək lazım gəlir. Qeyri-formal dildə alqoritmlərin yazılması zamanı "hər biri üçün dövr" konstruksiyasından istifadə olunur; məsələn, M çoxluğu halında aşağıdakı proqram fraqmentini yazmaq olar:

```
db hər bir X elementi üçün
| M çoxluğun dan
| əməl (X)
ds
```

Verilənlər strukturunun əksəriyyətində belə dövrü konkret tip üçün mövcud olan digər əməllərdən istifadə etməklə də reallaşdırmaq olar. Məsələn, siyahı üçün aşağıdakı proqram fraqmenti tətbiq oluna bilər:

```
göstəricini siyahının başlanğıcına qoymaq
db nə qədər ki deyil göstərici siyahının
| sonunda
| əməl (göstəricidən sonrakı element)
| göstəricini irəli sürüşdürmək
ds
```

Çoxluq digər verilənlər strukturundan onunla fərqlənir ki, "hər biri üçün dövr" konstruksiyasını çoxluğun digər əmrlərindən istifadə etməklə modelləşdirmək mümkün deyildir. Ona görə də, belə dövrün yerinə yetirilməsi çoxluğun reallaşdırılması ilə təmin olunmalıdır. "Hər biri üçün dövr"ün qurulması üsulu istifadə olunan proqramlaşdırma dilindən və konkret siniflər, yaxud altproqramlar kitabxanasından güclü şəkildə asılıdır. Adətən, strukturun elementlərinə ardıcıl olaraq baxmağa imkan verən obyekt yaradılır. Belə obyektin daxili quruluşu verilənlər strukturundan asılıdır və proqramçıdan gizlədilmişdir. Bu obyekti adətən iterator, yaxud enumerator adlandırırlar. İteratorla iki əməliyyat mümkündür:



- verilənlər strukturunun əvvəlki addımlarda baxılmamış elementlərinin olub-olmamasını yoxlamaq;
- verilənlər strukturunun növbəti elementini almaq.

M çoxluğu halında "hər biri üçün dövr" iteratorun köməyilə təxminən belə yazılır:

```
iterator := M çoxluğunun elementlərinin
   baxılmasına başlamaq
db nə qədər ki iterator. hələ elementlər var
| x := iterator. çoxluğun növbəti
|  elementini almaq
| əməl (x)
ds
```

* * *

Əlbəttə, verilənlər strukturunun bütün müxtəlifliklərı baxılan misallarla tükənmir. Lakin gerçək misalların əksəriyyəti ya onlara analojidir, ya da bir neçə elementar strukturun kombinasiyasından alınır. Məsələn, çoxluqların reallaşdırılması ağaclardan istifadə edə bilər. Proqramçının məharəti, həll olunan məsələyə daha çox uyğun gələn, kütləvi əməliyyatların olmadığı, yaddaşın qənaətlə işlədilməsinə və axtarışın sürətlə yerinə yetirilməsinə şərait yaradan verilənlər strukturunun seçilməsindən ibarətdir. Ancaq, ola bilsin ki, əsas meyar nəticədə alınan proqramın sadəliyi və gözəlliyidir. Qalan hər şey buradan alınır.

ÇEŞİDLƏMƏ



Herman Hollerit.



Con Mouçli.

Nizama, harmoniyaya can atmaq insana xasdır. Bu hər yerdə, telefon kitabçasından, yaxud kitabxana kataloqundan başlamış mağazanın rəflərində malların yerləşdirilməsinə qədər hiss olunur.

Ehtimal olunur ki, verilənlərin nizamlanmasının avtomatlaşdırılmasına ilk örnək Herman Holleritin 1890-cı ildə ABŞ-da əhalinin siyahıya alınmasının nəticələrinin işlənməsində tətbiq olunmuş elektromexaniki tabulyatorudur. Sonralar analoji qurğu geniş şəkildə yayıldı. Məlumdur ki, 1930-cu ildə İngiltərədə statistika və mühasibat firmaları çeşidləmə maşınlarını ayda cəmi 9 funt sterlinqə həvəslə icarəyə götürürdülər. Sonralar daha mürəkkəb, seçici maşınlar adlanan, bir keçidə iki nizamlanmış kart dəstini birləşdirən maşınlar icad edilmişdir.

Bununla belə, həqiqətən böyük həcmli verilənlər massivinin sürətli nizamlanması imkanı kompüterlərin kəşfindən sonra meydana gəldi. Məhz o zamandan çeşidləmə vacib praktiki əhəmiyyət kəsb etdi. Verilənlərin kompüterdə nizamlanmasını, öz üsulunu tətbiq etməklə, ilk dəfə 1946-cı ildə ENIAC kompüterinin yaradıcısı Con Mouçli yerinə yetirmişdir. Verilənlərin nizamlanmasında kompüterlərdən istifadənin zəruriliyi o zamanlar aydın görünmürdü. Təsadüfi deyildir ki, özünün çeşidləmə haqqında mühazirəsində Mouçli demişdir: "Bir maşında hesablama və çeşidləmə imkanlarının birləşdirilməsi tələbi bəzilərinə eyni bir cihazdan həm konserv bıçağı, həm də avtoqələm kimi istifadə etmək tələbi kimi görünə bilər"...

Burada ancaq *daxili çeşidləmə* adlanan üsullara baxılır (bu ad tarixən yaranmışdır, çünki onlar bütün elementləri kompüterin ixtiyari irişimli daxili, operativ yaddaşında yerləşən massivlərin nizamlanmasında tətbiq olunurdular. Bundan başqa, alqoritmi xarici daşıyıcılardakı köməkçi yaddaşdan istifadə edən *xarici çeşidləmə* üsulları da vardır). Çeşidləmə üsulları verilənlərin nizamlanması ideyasından asılı olaraq bir neçə qrupa bölünə bilər:

- daxiletmə vasitəsilə çeşidləmə;
- seçim vasitəsilə çeşidləmə;
- mübadilə vasitəsilə çeşidləmə.

BİRBAŞA DAXİLETMƏ VASİTƏSİLƏ ÇEŞİDLƏMƏ

Üsulun mahiyyəti ondan ibarətdir ki, giriş massivi şərti olaraq iki hissəyə ayrılır – artıq çeşidlənmiş və hələ çeşidlənməmiş (ilkin olaraq çeşidlənmiş hissə ancaq birinci elementdən ibarət olur). Sonra hər addımda çeşidlənməyən hissənin birinci elementi oradan çıxarılır və çeşidlənmə nəzərə alınmaq şərtilə çeşidlənmiş hissədə öz yerinə əlavə olunur.

i-ci elementin çeşidlənmiş hissədə öz yerinə qoyulması zamanı elementlərin müqayisəsinin sayı $i/2$-yə bərabərdir. Bu əməliyyat $n-1$ sayda elementin hər biri üçün yerinə yetirildiyindən, alqoritm n^2 tərtibində mürəkkəbliyə malikdir. Əgər giriş massivinin elementləri artıq nizamlanmışdırsa, onda müqayisələrin sayı minimal olacaqdır. Əgər giriş massivi əks sıra ilə nizamlanmışdırsa, müqayisələrin sayı maksimal olacaqdır.

İlk baxışdan elə görünür ki, alqoritmi yaxşılaşdırmaq olar. Həqiqətən, növbəti elementin daxil olunacağı ardıcıllıq nizamlanmışdır. Deməli, daxiletmə yerinin *ikilik axtarışı* mümkündür (daxil ediləcək element çeşidlənmiş ardıcıllığın ortasındakı elementlə müqayisə olunur və yarıya bölmə prosesi o vaxta qədər davam edir ki, elementin daxil olunacağı yer tapılmış olsun). Bu qaydada şəkli dəyişdirilmiş alqoritm *ikilik daxiletmə ilə çeşidləmə* alqoritmi adlanır. Elementin çeşidlənmiş hissəyə daxil edilməsi ondan sağda yerləşən bütün elementlərin yerinin dəyişdirilməsini tələb edir. Beləliklə, baxmayaraq müqayisələrin sayının azalması aşkardır, ancaq elementlərin yerdəyişmələrinin sayı o qədər qalır. Nəticədə, alqoritm yenə də n^2 tərtibində mürəkkəbliyə malik olur.

Bu nəticə onu göstərir ki, ilk baxışdan aydın görünən alqoritmin təkmilləşdirilməsi, heç də həmişə nəzərə çarpacaq effekt vermir. Ümumiyyətlə, birbaşa daxiletmə ilə çeşidləmə üsulu olduqca cəlbedici görünmür, axı yaddaşda massivin elementlərinin yerlərinin dəyişdirilməsi kifayət qədər uzunmüddətli əməliyyatdır.

Məsələnin mürəkkəbliyi onun həlli üçün məlum olan ən yaxşı alqoritmin mürəkkəbliyidir.





BİRBAŞA SEÇİM VASİTƏSİLƏ ÇEŞİDLƏMƏ

Alqoritmin ümumi sxemi belədir:

1) Massivin n elementindən ən kiçiyi seçilir.

2) Seçilmiş elementlə birinci elementin yerləri dəyişdirilir.

Birinci elementi massivin çeşidlənmiş hissəsi hesab edərək, bu addımları qalan $n-1$ sayda çeşidlənmiş elementlər üçün təkrarlayırıq, sonra $n-2$ element üçün və sair. Bu prosesi massivin çeşidlənməmiş hissəsində bir element – ilkin massivdə ən böyük element qalanadək davam etdiririk.

Bu üsul müəyyən mənada əvvəlkinin əksidir (antipodudur). Birbaşa daxiletmə ilə çeşidləmə üsulunun hər bir addımında giriş massivinin bir elementinə və çeşidlənmiş hissənin bütün elementlərinə baxılır. Burada isə giriş massivinin bütün çeşidlənməmiş elementlərinə baxılır və bunların içərisindən çeşidlənməmiş hissənin axırına yerləşdirilən bir element seçilir.



Tutaq ki, n elementdən ibarət $a_1, a_2, \dots, a_n$ ardıcıllığı verilmişdir; bu elementlərin yerdəyişmə əməliyyatı vasitəsilə $f(a_{i_k}) \le f(a_{i_{k+1}}), k \neq 1,\dots,n-1$ şərtini ödəyən $a_{i_1}, a_{i_2}, \dots, a_{i_n}$ ardıcıllığına çevrilməsi çeşidləmə adlanır.

Çox zaman f funksiyasının qiyməti hesablanmır, aşkar şəkildə verilir və ardıcıllığın müəyyən elementlərilə əlaqəli olur. Bu qiymətləri açarlar adlandırırlar; adətən açarlar tam, yaxud həqiqi ədədlər olur. Ümumiyyətlə, elementlərin özlərini deyil, ancaq onların açarlarını çeşidləmək olar, axı uyğun elementin açarla birlikdə yerinin dəyişdirilməsi alqoritmin mürəkkəbliyinə təsir etmir.

NİZAMLILIĞIN VACİBLİYİ

Obyektlərin əlverişli yerləşdirilməsi ideyası olduqca qədimdir. Ola bilsin ki, onun həyata keçirilməsinə ən qədim nümunə, 100-dən çox altmışlıq ədəd və onların tərs kəmiyyətindən ibarət qiymətlər cütünü özündə saxlayan Babil cədvəlidir (onun tarixi təxminən b.e.ə. 200-ci il götürülür). Axtarışı sadələşdirmək üçün cədvəlin elementləri artan sıra ilə yazılmışdır.

Ədədlərin müqayisəsi ta qədimdən təbii görünür. Ancaq qədim Yunanıstanda ədədləri işarə etmək üçün hərflərdən (və bəzi əlavə işarələrdən) istifadə olunmuşdur. Ona görə də sözlərin müəyyən ardıcıllıqla yerləşdirilməsi fikri də cəfəng görsənməməlidir. Arxeoloqlar dəfələrlə qədim yunan yaşayış məskənlərində yaşayan insanların əlifba sırasına görə (hətta birinci iki hərfə görə) nizamlanmış siyahısını tapmışlar. Antik və erkən Orta əsrlərin bir çox müəllifləri sözləri lüğətlərdə məsələn, məşhur romalı həkim və təbiətşünas Klavdiy Qalen (təxminən 130-200) "Hippokrat lüğətində" nizamlamışlar. Bu gün leksikoqrafik sıralanma adlandırılan proses ilk dəfə 1286-cı ildə Genuyalı Covanni adlı birisi tərəfindən təsvir olunmuşdur. Onun öz lüğətinin sözönündə gətirdiyi misallarda sözlərin sıralanması nəinki ikinci, hətta altıya qədər hərflərlə müəyyən olunur. Buna baxmayaraq, 300 il keçdikdən sonra bu üsul əlavə izahlar tələb etmişdir. Belə ki, 1604-cü ildə Londonda nəşr olunmuş "Table Alphabetical" adlı ingilis dili lüğətində izah olunur: *a* ilə başlayan sözləri kitabın əvvəlində, *v* ilə başlayanları isə sonunda axtarmaq lazımdır.



Klavdiy Qalen.

Aydındır ki, sözlərin lazım olan ardıcıllıqla düzülməsi lüğətlərin tərtibçiləri tərəfindən əllə yerinə yetirilmişdir. Bu isə kifayət qədər hövsələ və xeyli diqqət tələb etmişdir (adı çəkilən ingilis dili lüğətində səhvlər az deyildir).

Elementlərin müqayisəsinin sayı onların ilkin yerləşməsindən asılı deyildir və n^2 ilə mütənasibdir. Deməli, bu alqoritmin də kvadratik mürəkkəbliyi vardır. Amma yerdəyişmələrin orta sayı birbaşa daxiletməyə nisbətən azdır, ona görə də birbaşa seçim vasitəsilə çeşidləmə çox vaxt üstün tutulur. Ancaq, əgər giriş massivinin elementləri kifayət qədər yaxşı nizamlanmışdırsa, birbaşa daxiletmə bir qədər effektiv hesab olunur (bu halda bərabər sayda yerdəyişmələr olan zaman o az müqayisə tələb edir).



BİRBAŞA MÜBADİLƏ VASİTƏSİLƏ ÇEŞİDLƏMƏ

Baxmayaraq ki, təsvir olunmuş hər iki alqoritmdə elementlər yerlərini dəyişdirirlər, hər halda yerdəyişmələr onlarda köməkçi rol oynayır. Əksinə, indi veriləcək üsul məhz yerdəyişmələrə əsaslanır.

Alqoritm massivin nizamlanmamış hissəsi üzrə sağdan sola keçidlərdən ibarətdir; bu zaman qonşu elementlər cüt-cüt müqayisə edilir və əgər lazımdırsa yerləri dəyişdirilir. Hər bir keçidin nəticəsində massivin nizamlanmamış hissənin minimal elementi onun sol kənarına doğru yerini dəyişdirmiş olur. Demək olar ki, daha "yüngül" elementlər suda hava qovuqları kimi üzə çıxırlar. Ona görə də, bəzən bu üsul *qovuqlu çeşidləmə* üsulu adlanır. Keçidlər bütün elementlər nizamlanana qədər davam etdirilir. Əgər keçid zamanı heç bir yerdəyişmə baş verməmişdirsə alqoritmin yerinə yetirilməsini dayandırmaq olar.

Baxmayaraq ki, ən yaxşı halda (giriş massivi tamamilə nizamlanmışdır) elementlərin yerdəyişmələrinin sayı sıfra bərabərdir, hər halda o, orta hesabla n^2 ilə mütənasibdir. Amma müqayisələrin sayı sabitdir və $(n^2-n)/2$ kəmiyyətinə bərabərdir. Bu səbəbdən, alqoritm də kvadratik mürəkkəbliyə malikdir.

Minimal element bir keçidlə üzə çıxır, amma "ağır" elementlər hər bir keçid zamanı ancaq bir mövqe aşağı düşürlər. Ona görə də, giriş massivində ən böyük element solda yerləşmişdirsə, onda maksimal sayda keçidlər tələb olunur. Massivə baxışın istiqamətini dəyişdirməklə bundan qaçmaq olar (bu üsul *qarışıq çeşidləmə* adlanır). Təəssüf ki, bu zaman alqoritmin mürəkkəbliyi azalmır: yerdəyişmələrin sayı əvvəlkitək qalır.

YAXŞILAŞDIRILMIŞ ÇEŞİDLƏMƏ ALQORİTMLƏRİ

Beləliklə, bu vaxta qədər baxılmış çeşidləmə alqoritmlərinin hamısı (onları *birbaşa çeşidləmə* adlandırırlar) kvadratik mürəkkəbliyə malikdir.

Prinsipial sual meydana gəlir: hər hansı məsələnin həll alqoritmini yaxşılaşdırmaq olarmı və yaxşılaşdırmanı hansı həddə qədər etmək olar? Maraqlıdır ki, bu sualları Çarlz Bebic də vermişdir. O, hesab etmişdir ki, analitik maşının köməyilə hər hansı nəticəni aldıqdan sonra, mütləq aydınlaşdırmaq lazımdır ki, eyni nəticəni az vaxta verən hesablama üsulu mövcuddurmu.

Asanca görmək olur ki, bütün birbaşa çeşidləmə üsullarında bir elementar addımla massivin elementi ancaq bir mövqe yerini dəyişir. İsbat olunmuşdur ki, çeşidləmə vaxtı massivin n elementinin hər biri orta hesabla $n/3$ mövqe yerini dəyişdirməlidir. Bu onu göstərir ki, n elementin nizamlanması üçün n^2 tərtibdə elementar addım tələb olunur. Alqoritmin mürəkkəbliyini azaltmaq üçün, hər bir elementar addımda elementlərin daha böyük məsafələrə yerlərini dəyişdirməyə cəhd etmək lazımdır. Bütün yaxşılaşdırılmış çeşidləmə alqoritmlərinin qurulmasının əsasında məhz bu ideya dayanır.

ŞELL ÇEŞİDLƏMƏSİ

Bu üsul birbaşa daxiletmə ilə çeşidləmənin yaxşılaşdırılmasıdır. Onun mahiyyəti giriş massivinin dəfələrlə qruplara bölünməsindən və onların sərbəst çeşidlənməsindən ibarətdir.

Məsələn, əvvəlcə giriş massivini bir-birindən 4 mövqe məsafədə dayanan elementlərin 4 qrupuna ayırmaq və hər qrupu (o *dördqat* adlanır) sərbəst olaraq istənilən məlum üsulla çeşidləmək lazımdır. Sonra bir-birindən iki mövqe aralı dayanan elementlərdən ibarət iki qrup yaratmaq lazımdır. Yenə də, onların çeşidlənməsini ayrıca olaraq hər qrupun daxilində aparmalı (*ikiqat* çeşidləmə). Üçüncü mərhələdə (*birqat* çeşidləmə) bütün massiv çeşidlənir. Nəticədə çeşidlənmiş massiv alınır. Baxmayaraq ki, mərhələlərin sayı çox olur, onların hər birində ya sayı çox olmayan elementlər çeşidlənir, ya da artıq elementlər kifayət qədər nizamlanmışdır və az sayda yerdəyişmələr tələb olunur. Uduş onun hesabına əldə olunur ki, bir-birindən xeyli məsafədə yerləşən elementlər yerlərini dəyişdirirlər.

Ardıcıl çeşidləmələr üçün elementlər arasındakı məsafənin seçilməsi maraqlı riyazi problemdir. İsbat olunmuşdur ki, əgər əvvəlcə massiv i addımı ilə, sonra j addımı ilə çeşidlənirsə, onda onun nizamlığı saxlanılacaqdır. Ardıcıl çeşidləmələrdə istifadə olunan məsafənin 2-nin qüvvətləri olması məcburi deyildir (təkcə axırıncı mərhələdə məsafələr vahidə bərabər olmalıdır). Bundan başqa arzuolunandır ki, onlar bir-birlərinin vuruqları olmasınlar. Bu halda, hər bir növbəti keçiddə qrupların elementləri qarışacaqdır, bu isə önəmli effekt verir. Məsələn, işlərin birində məsafələrin belə ardıcıllığı təklif olunur: 1, 4, 13, 40, 121, ... (aydındır ki, əks ardıcıllıqda tətbiq olunan). Məsafələr ardıcıllığının 1, 3, 7, 15, 31, ... halında alqoritmin mürəkkəbliyi $n^{1,2}$ kimi qiymətləndirilir. Baxmayaraq ki, bu qiymət n^2-dən xeyli yaxşıdır, daha yaxşı çeşidləmə alqoritmləri də vardır.

Tapşırığın çətinliyi – onun həlli üçün məlum olan ən yaxşı alqoritmin çətinliyi deməkdir.





Donald Şell.

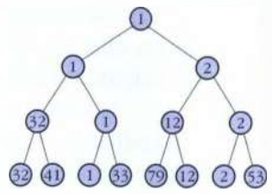


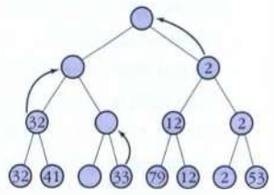
AĞACIN KÖMƏYİLƏ ÇEŞİDLƏMƏ

Bu üsul müəyyən mənada birbaşa seçim üsulunun analoqudur, çünki bu üsulda da minimumun tapılması, əvvəlcə n element, sonra $n-1$ element arasında və sair aparılır. Beləliklə, bütövlükdə $(n^2-n)/2$ sayda müqayisə aparmaq tələb olunur. Ancaq bu qiyməti yaxşılaşdırmaq olar. Məsələ ondadır ki, ən kiçik elementin axtarışı zamanı sonradan itirilən çoxlu sayda informasiya alınır. Ancaq, əgər massivi xüsusi qurulmuş *minimallaşdırıcı ağacın* köməyilə təqdim etsək, onu saxlamaq olar. Massivin elementlərini cüt-cüt müqayisə etmək və hər cütdə ən kiçik elementi seçmək, sonra onları da cüt-cüt müqayisə etmək və sair. Nəticədə, $n-1$ müqayisə yerinə yetirərək ən kiçik elementi tapırıq (o, ağacın kökündə yerləşir). Bu, çeşidləmənin başlanğıc mərhələsidir. Şəkildə {32, 41, 1, 33, 79, 12, 2, 53} massivi üçün minimallaşdırıcı ağac göstərilmişdir.



İndi ağacın kökündən minimal elementin olduğu yol boyunca geriyə qayıtmaq lazımdır. Bu zaman minimal elementi ağacdan çıxararaq onu boş elementlə əvəz etmək lazımdır. Bundan sonra müqayisə olunan elementlərdən kiçiyini boş ara təpələrinə yerləşdirərək, yenə yuxarı qalxmaq lazımdır. Bu hərəkətlər yerinə yetirildikdən sonra, ağacın kökünə yerini dəyişdirmiş element yenə də qalanların ən kiçiyi olacaqdır.





Yenə onu xaric etməli n belə addımdan sonra ağac boş olacaq və çeşidləmə prosesi bununla qurtaracaqdır. Minimal elementin seçilməsinin hər bir n addımında yalnız $\log_2 n$ sayda müqayisə (ağacın hər bir səviyyəsində bəzi təpələrin qiymətinin dəqiqləşdirilməsi üçün bir müqayisə) tələb olunur. Ona görə də, bütün prosesin mürəkkəbliyi $n\log_2 n$ təşkil edir. Beləliklə, hətta Şell alqoritmindən də yaxşı alqoritm alındı!

SÜRƏTLİ ÇEŞİDLƏMƏ

Çeşidləmənin bu üsulu əsasən mübadilədən istifadə edir. Onun əsas ideyası ondan ibarətdir ki, elementlərin yerdəyişməsini mümkün olan qədər daha böyük məsafələrlə həyata keçirmək lazımdır. Bu zaman hərəkətlər ardıcıllığı belədir:

1) Bəxtə-bəxt (məsələn, massivin ortasında) x elementini seçməli.

2) Massivə x-dən sola, ondan böyük a elementi tapılana qədər baxmaq. Əgər belə element yoxdursa, onda $a = x$ (başqa sözlə, axtarılan a olaraq x qəbul olunur).

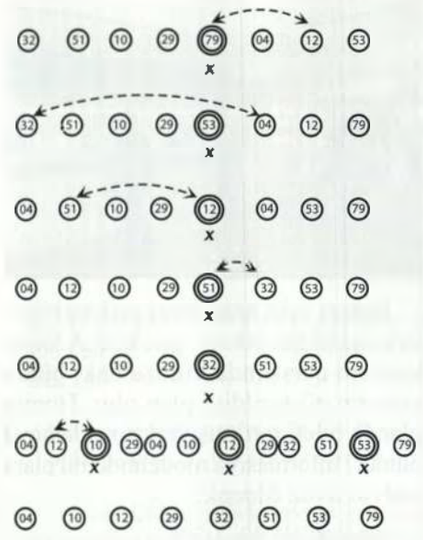
3) Massivə x-dən sağa, ondan kiçik b elementi tapılana qədər baxmaq. Əgər belə element yoxdursa, onda $b = x$ (başqa sözlə, axtarılan b olaraq x qəbul olunur).

4) Əgər $a > b$ isə, a və b elementlərinin yerini dəyişdirmək.

5) Baxış və mübadilə prosesini o vaxta qədər davam etdirməli ki, giriş massivi iki hissəyə bölünsün. Sol tərəf, sağ tərəfi təşkil edən elementlərin ixtiyari birini aşmayan elementlərdən ibarətdir. İndi, təsvir olunmuş proseduru (1-4 addımları) massivin hər iki hissəsi üçün tətbiq etmək.

İsbat olunmuşdur ki, əgər x-in sərhəddi qismində çeşidlənən massivin *mediani* (elementlərin yarısından böyük olmayan, digər yarısından isə kiçik olmayan element belə adlanır) seçilmişdirsə, onda müqayisələrin ümumi sayı $n\log_2 n$, yerdəyişmələrin sayı isə $n\log_2 n/6$ olacaqdır. Ümumiyyətlə, sərhədin seçilməsi maraqlı riyazi məsələdir. Müəyyən olunmuşdur ki, massivin ixtiyari elementi, o cümlədən birinci və axırıncı elementləri də *median* ola bilər; hətta belə mülahizə də var ki, onu təsadüfi olaraq seçmək lazımdır. Bütövlükdə alqoritmin mürəkkəbliyi $n\log_2 n$ olur.

Eyni bir məsələnin həlli üçün müxtəlif yanaşmalar mümkündür. Çeşidləmə alqoritmlərinin analizi alqoritmin mürəkkəbliyinin mahiyyətini açıb göstərir, onların müqayisəsi isə təkmilləşdirmə yollarının axtarışı üçün geniş material verir (ola bilsin ki, trivial alqoritmlərin kvadratik mürəkkəbliyindən effektiv alqoritmlərin $n\log_2 n$ tərtibli mürəkkəbliyinə keçid zamanı əhəmiyyətli uduş alınsın).



Demək olar ki, bütün çeşidləmə alqoritmləri şübhəsiz praktiki maraq doğurur. Hətta, ilk baxışdan effektiv görünməyən birbaşa üsullar da bir sıra hallarda uğurla tətbiq oluna bilərlər.

İNFORMASİYA MODELLƏRİNİN QURULMASI

Kompüterdə emal olunan informasiya istifadə olunan proqramlaşdırma dilindəki kəmiyyətlərin qiymətləri – massivlər, cədvəllər, dəyişənlər şəklində göstərilməlidir. Tədqiq olunan obyektlər və proseslər haqqında bütün vacib informasiyanı özündə saxlayan kəmiyyətlər çoxluğuna *informasiya modeli* deyilir.

İnformasiya modeli modelləşdirilən obyektlər (hadisələr) barədə hər şeyi deyil, yalnız baxılan məsələ üçün zəruri olan hissəni özündə saxlayır, yerdə qalanlar isə modelləşdirmə zamanı nəzərə alınmır. Həll olunan məsələlərin dairəsi genişlənərsə, modeli də genişləndirmək, ona daha çox informasiya daxil etmək lazım gəlir. Həll olunan məsələlər məhdud bir sahəni əhatə etdikdə belə, informasiya modelini müxtəlif üsullarla qurmaq olar.

Tutaq ki, hər birində 50 yeri olan 40 sıralı kinoteatrda bir seansa qabaqcadan bilet satışı təşkil edilməlidir.



Kinoteatrda bilet kassası.

Bunun üçün artıq hansı yerlərə biletlərin satıldığını bilmək gərəkdir. Adətən, kassirin qarşısında kinozaldakı bütün yerlərin göstərildiyi plan olur. Həmin planda bileti satılmış yerlər xaçla qeyd olunur. İnformasiya modelində bu planı cədvəl əvəz edəcək:

```
tam cəd a[1:40, 1:50],
```

yerin bileti satılıbsa a[i, j] = 1, satılmayıbsa a[i, j] = 0.

Satışdan əvvəl, hələ heç bir bilet satılmayanda, cədvəl sıfırlarla doldurulmalıdır:

```
db i üçün 1 dən 40 dək
| db j üçün 1 dən 50 dək
| | a[i, j] := 0
| ds
ds
```

20-ci sıradakı 1-ci yerə bilet satdıqda

```
a[20, 1] := 1
```

komandasını yerinə yetirmək lazımdır.

Bu cədvəl vasitəsilə satışın gedişi haqqında məlumat almaq və kassada qalan biletləri (tam qalıq) saymaq olar:



Kinoteatr zalı.

```
qalıq := 0
db i üçün 1 dən 40 dək
| db j üçün 1 dən 50 dək
| | əgər a[i, j] = 0
| | | onda qalıq := qalıq + 1
| | tamam
| ds
ds
```

Satılmamış biletlərin ümumi qiymətini təyin etmək üçün hər bir biletin qiymətini bilmək lazımdır, ancaq kinoteatrda sıradan asılı olaraq biletlərin qiyməti müxtəlifdir. Deməli, modelə biletlərin qiymətləri barədə informasiya daxil etmək məntiqli olardı və bunu xətti cədvəl, yaxud köməkçi alqoritm şəklində etmək olar:

```
alq tam qiymət (tam i)
```

Alqoritm *i*-ci sıra üçün biletin qiymətini rublla (qəpiksiz) geri qaytarır, ona görə də qiymət həqiqi yox, məhz tam olmalıdır, məsələn: qiymət (1) 100, qiymət (20) isə 250 qaytaracaq, çünki 20-ci sıra birincidən bahadır.

Satışdan əvvəl bu köməkçi alqoritm artıq təyin edilmiş olmalıdır, yəni biletlərin "qiyməti" verilməlidir. Biletlərin qiymətini dəyişmək üçün yardımçı qiymət alqoritmini dəyişmək və ya təzəsini yaratmaq kifayətdir, məsələn, bazar və bayram günləri üçün, yaxud da, əksinə, adi günlər üçün xüsusi təzə alqoritm yaratmaq olar.

İndi ixtiyari anda satışdan toplanmış pulu hesablamaq üçün modeldə kifayət qədər informasiya var:

```
s:=0
db i üçün 1 dən 40 dək
| db j üçün 1 dən 50 dək
| | əgər a[i, j]:=1
| | | onda s := s + qiymət (i)
| | tamam
| ds
ds
```

Kinozal modelində bütün sıralardakı yerləri (40×50=2000) dalbadal nömrələyərək düzbucaqlı cədvəl əvəzinə a[1:2000] xətti cədvəli daxil etmək olardı.

Lakin bu düzgün olmazdı, çünki real kinozalda yerlərin ucdantutma ardıcıl nömrələnməsi yoxdur, yerlər sıra nömrəsilə və oturacağın sıradakı nömrəsilə verilir. Digər tərəfdən isə informasiyanı daim bir formadan başqasına çevirərkən səhvlərdən qaçmaq praktiki olaraq mümkün deyil.

Artıq model seçildikdən sonra alqoritmi başqa cür də yazmaq olar. Əgər a[1:40, 1:50] cədvəlində satılmış yerlərə vahid, satılmamış yerlərə isə sıfır uyğundursa, onda satılmış yerlərin sayı, sadəcə, cədvəlin bütün elementlərinin cəmi kimi hesablanır, eynilə də satışdan toplanmış pul hesablanır:

```
n := 0; s := 0
db i üçün 1 dən 40 dək
| db j üçün 1 dən 50 dək
| | n := n + a[i, j]
| | s := s + a[i, j]*qiymət (i)
| ds
ds
```

Təklif olunan model yerlərin bronlanması imkanını, abonement satışını, pensiyaçılara güzəştləri və s. nəzərə almır. Bu modeldə həmçinin kinozalın giriş və çıxışlarının sayı haqqında, zalda keçidlərin mövcudluğu, bufetlərin sayı və s. barədə informasiya yoxdur. Satışdan toplanmış pulu və ya satılmış biletlərin sayını hesablamaq üçün, əlbəttə, bu informasiya lazım deyil, lakin yanaşı iki yer tapmaq lazım gələrsə, bu informasiya vacib olur.

Deməli, modeli genişləndirməli olacağıq. Tutaq ki, kinozalda birinci sıradan axırıncıyadək keçid var. Onda informasiyanı əvvəlcədən hazırlanmış *p* [1:49] (49=50-1) xətti cədvəlilə vermək olar. Sıralardakı yerlərin sayı 50 olduğundan



NƏQLİYYAT ŞƏBƏKƏSİNİN İNFORMASİYA MODELİ

Hansısa bir ölkədə aralarında aviaxətlər olan 100 şəhər var. Bir şəhərdən ixtiyari başqa şəhərə (ola bilsin, təyyarə dəyişməklə) uçmaq mümkündür. Hər şəhərdə yalnız bir aeroport var.

Elə informasiya modeli qurmaq tələb olunur ki, onun vasitəsilə bir şəhərdən digərinə minimal sayda təyyarə dəyişməklə necə uçmaq və bu zaman yolun uzunluğunun nə qədər olacağı suallarına cavab vermək mümkün olsun.

```
tam n; n:=100                 | şəhərlərin sayı
sim cəd ad[1:n]               | şəhərlərin adı
həq cəd məsafə[1:n, 1:n]      | i və j şəhərləri arasındakı məsafə

| bütün i-lər üçün məsafə [i, i]=0 və
| birbaşa aviaxətt yoxdursa, məsafə [i, j]=0
```

Bu modeldən istifadə edərək, məsələn, i1 şəhərindən i2 şəhərinə bir dəfədən çox təyyarə dəyişmədən necə uçmağı təhlil etmək olar:

```
çapet ad[i1], "şəhərindən", ad[i2], "şəhərinə"
əgər məsafə [i1, i2] > 0
| onda çapet "reys var, məsafə = ", məsafə [i1, i2]
| əks halda p:=0; tapılıb:="yox"
| db nə qədər ki p < n və tapılıb = "yox"
| | p := p + 1
| | əgər məsafə [i1, p] > 0 və məsafə [p, i2] > 0
| | | onda tapılıb := "hə"
| | tamam
| ds
| əgər tapılıb = "hə"
| | onda çapet ad [p] "-də təyyarə dəyişməklə marşrut var"
| | çapet "məsafə =", məsafə [i1, p] + məsafə [p, i2]
| tamam
tamam
```



Kompüter sisteminin hazırlanmasında adətən sifarişçi və icraçı iştirak edir. Sonuncunun vəzifəsi sifarişçinin tələblərini yerinə yetirməkdir. Lakin praktikada sifarişçi heç də həmişə kompüter sisteminə olan bütün tələblərini bir dəfəyə tam ifadə edə bilmir. Çox zaman o, şərtləri sistemin hazırlanması və hətta istismarı prosesində əlavə edir. Ona görə də təcrübəli icraçı sifarişçinin ehtiyaclarını qabaqcadan duyur və informasiya modelini elə qurmağa çalışır ki, sistemin inkişafı zamanı model yalnız genişlənsin, onun hazır hissələri və bu hissələrlə işləyən alqoritmlərsə mümkün qədər nadir hallarda dəyişilsin.

keçidlərin sayı 49-dan çox ola bilməz. $p[i] = 1$ olması göstərir ki, i-ci yerlə $i+1$-ci yer arasında keçid var. İki yanaşı yer tapılması məsələsini aşağıdakı alqoritm həll edir.

```
tapdıq := "yox"
i := 1; j := 1
db nə qədər ki tapdıq = "yox" və i <= 40
  əgər a[i, j] = 0 və a[i, j+1]=0 və
                              p [j] = 0
    onda tapdıq := "hə"
    əks halda j := j + 1
    əgər j = 50
      onda i := i + 1; j := 1
    tamam
  tamam
ds
```

Tutaq ki, kinoteatrda pensiyaçılara bilet növbəsiz və 50% güzəştlə satılır. Növbəsiz bilet almağın nə kompüter modelinə, nə də alqoritmə heç bir dəxli yoxdur, ancaq 50% güzəştlə satış məsələsinin həm modelə, həm də alqoritmlərə aidiyyatı var. Satışdan toplanmış pulu hesablayarkən güzəştlər nəzərə alınmalıdır, ona görə də bu informasiya modelə əlavə edilməlidir. Bu zaman a[1:40, 1:50] cədvəlində satılmamış yerlərə yenə də sıfır qiyməti, satılmış yerlərə isə biletin tam qiymətinin ödənilmiş faizi uyğun gəlir: bilet 100% ödənilibsə, $a[i, j]=100$, biletin dəyərinin 50%-i ödənibsə, $a[i, j]=50$.

Satılmış yerlərin sayını və satışdan toplanmış pulu hesablayan alqoritm indi belə şəkil alır:

```
n := 0
s := 0
db i üçün 1 dən 40 dək
  db j üçün 1 dən 50 dək
    əgər a[i, j] <> 0
      onda n := n + 1
        s := s + a[i, j]/100*qiymət (i)
    tamam
  ds
ds
```

Bu modelin üstünlüyü ondadır ki, o, lazım gələrsə, güzəştin ölçüsünü istənilən qaydada dəyişməyə və ya biletləri müxtəlif güzəştləri nəzərə almaqla satmağa (məsələn, 12 yaşına qədər uşaqlara 70% güzəştlə) imkan verir. Lakin bu modellə biletləri tamam pulsuz vermək mümkün deyil, çünki 100%-li güzəşt zamanı a[1:40,1:50] cədvəlinə biletin qiymətinin 0%-ni daxil etməliyik, belə yeri isə adi boş yerdən fərqləndirmək mümkün olmayacaq.

PARALEL PROQRAMLAŞDIRMA

Kompüterə qədərki dövrdə eyni bir seansa bilet satılarkən, adətən, kassirlər biletləri öz aralarında bölürdülər və hərə öz hissəsini satırdı. Bəzən bir kassanın qarşısında növbə düzülür, o birinin qarşısında isə heç kəs olmurdu, çünki ya bu kassanın bütün biletləri satılıb qurtarırdı, ya da ancaq pis yerlərə biletlər qalmış olurdu. Belə vəziyyət uzaq məsafəyə gedən qatarlara bilet satarkən də yaranırdı. Əgər dəmir yol kassası vağzalda deyil, başqa yerdə yerləşirdisə, onda kassir əvvəlcədən bilet blanklarını özü ilə götürür və qatarlarda yer olub-olmamasını öyrənmək və ya hansı biletin alındığını xəbər vermək üçün vağzala zəng edirdi. Təbii ki, iş çox ləng gedirdi, çoxlu növbələr yaranırdı, hər cür dolaşıqlıqlar meydana çıxırdı, məsələn, müxtəlif sərnişinlərə eyni yerə biletlər satılırdı. Kassaların işi paralel təşkil olunmuşdu (müasir kompüterlərdə eyni vaxtda bir neçə proqram işlətmək mümkün olduğu kimi), bu zaman hər bir kassirin işinə müstəqil *proses*, bilet siyahısına isə *ümumi resurs* kimi baxılırdı.

HƏNDƏSİ İNFORMASİYANIN İNFORMASİYA MODELLƏRİ

Həndəsi informasiyanın ədədi kodlaşdırılmasının əsasını XVII əsrin böyük fransız alimi Rene Dekart qoymuşdur: müstəvinin ixtiyari nöqtəsini bir cüt ədədlə vermək mümkündür (Dekart koordinat sistemi). Nöqtələrin ədədi kodlaşdırılması planimetriyanın daha mürəkkəb həndəsi obyektlərini kodlaşdırmağa imkan verir. Həndəsi əməliyyatları yerinə yetirə bilən ixtiyari kompüter sistemi buna bənzər modellərə əsaslanır.

Hər bir həndəsi obyekt üçün çoxlu sayda model fikirləşib tapmaq olar. Obyektlər üzərində bir qisim əməliyyatlar üçün bir model, başqa qisim əməliyyatlar üçünsə başqa modellər daha münasib ola bilər. Odur ki, hər bir obyekt üçün bir neçə informasiya modeli yaratmaq və ehtiyac olduqda birindən o birinə keçmək lazımdır. Yaxşı məlum olan bir neçə həndəsi obyektin modellərinə baxaq:

```
Nöqtə
həq x, y          | nöqtənin koordinatları

Çevrə
həq r             | çevrənin radiusu
həq a, b          | mərkəzin koordinatları

Düz xətt
həq  x1, y1       | düz xətt üzərindəki
həq  x2, y2       | iki nöqtənin koordinatları

Düz xətt parçası
həq x1, y1        | parçanın başlanğıc nöqtəsinin koordinatları
həq x2, y2        | parçanın son nöqtəsinin koordinatları

Üçbucaq
həq x1, y1        | üçbucağın
həq x2, y2        | təpə nöqtələrinin
həq x3, y3        | koordinatları

n-bucaqlı
həq cəd x[1:n]    | i-ci təpəsinin (x[i], y[i]) koordinatları
həq cəd y[1:n]
```



Rene Dekart.

Bu modellərdən istifadə edərək həndəsi anlayışları alqoritmik dilə çevirmək olar:

```
Parçanın uzunluğu:      (x1-x2)**2 + (y1-y2)**2
Çevrə daxilindəki nöqtə: (x-a)**2 + (y-b)**2 < r**2

n-bucaqlının perimetri:
p:=0
db i üçün 1 dən n - 1 dək
  p := p + uzunluq (i, i + 1)
ds
p := p + uzunluq (n, 1)
```

Eyni bir həndəsi obyekti müxtəlif üsullarla kodlaşdırmaq olar, məsələn, düz xətt parçası belə də verilir:

```
həq a, b  | parçanın orta nöqtəsinin koordinatları
həq l     | parçanın uzunluğu
həq φ     | parçanın absis oxu ilə əmələ gətirdiyi bucaq
```

QARŞILIQLI KƏNARLAŞDIRMA PROBLEMİ

Ümumi resursa müraciət zamanı icra olunan iki prosesin sinxronlaşdırılması problemini həll etmək tələb olunur. Bunun üçün hər bir prosesdə ümumi resursa müraciətin baş verdiyi *kritik bölmə*, yaxud *kritik interval* adlanan xüsusi hissə ayrılmalıdır. Məsələn, proses alqoritmi dövri qaydada yerinə yetirilir: kritik bölmə köməkçi alqoritminə müraciətdən başlanılır və son köməkçi alqoritminə müraciətlə sona çatır.

Proseslərin sayı çox olmadıqda və onlar əvvəlcədən qəti müəyyən olunduqda onların ümumi resursunu bölmək o qədər də çətin deyil. Belə ki, klaviaturanın drayveri (klavişləri basdıqda simvolları daxil edən proqram) və mətn redaktoru (növbəti simvolu soruşan proqram) kimi iki prosesi asanlıqla sinxronlaşdırmaq olar. Əvvəldən sayı məlum olmayan proseslərin sinxronlaşdırılmasını təmin etmək daha çətindir.

```
alq proses
baş
  | db
  |   | kritik bölmə
  |   | son
  | ds
son
```

```
alq paralel proseslər
baş tam çevirici
  | çevirici := 1
  | proses1 : proses2 | proseslərin paralel icrası
son
```

```
alq proses 1
baş
  | db
  |   | db nə qədər ki çevirici = 2
  |   | ds
  |   | kritik bölmə1
  |   | çevirici := 2
  |   | son 1
  | ds
son
```

```
alq proses 2
baş
  | db
  |   | db nə qədər ki çevirici = 1
  |   | ds
  |   | kritik bölmə2
  |   | çevirici := 1
  |   | son 2
  | ds
son
```

```
alq paralel proseslər
baş mənt bayraq1, bayraq2
  | bayraq1 := "yox"
  | bayraq2 := "yox"
  | proses1 : proses2 | proseslərin paralel icrası
son
```

```
alq proses 1
baş
  | db
  |   | bayraq1 := "hə"
  |   | db nə qədər ki bayraq2
  |   | ds
  |   | kritik bölmə1
  |   | bayraq1 := "yox"
  |   | son 1
  | ds
son
```

```
alq proses 2
baş
  | db
  |   | bayraq2 := "hə"
  |   | db nə qədər ki bayraq1
  |   | ds
  |   | kritik bölmə2
  |   | bayraq2 := "yox"
  |   | son 2
  | ds
son
```

Əgər proseslərin alqoritmlərini elə modifikasiya etsək ki, yalnız bir proses öz kritik bölməsini çağıra bilsin, onda bu, baxılan problemi həll edər. Proseslərin sinxronlaşdırılmasının ilk icrası tam qiymətli çevirici dəyişəninin daxil edilməsidir. Bu dəyişən proseslərdən hansının öz kritik bölməsini işə sala bilməsindən asılı olaraq iki qiymətdən birini – ya "1", ya da "2" qiyməti ala bilər.

Əgər çevirici 1-ə bərabərdirsə, proses 2 çevirici 2-yə bərabər olana qədər gözləmə vəziyyətindədir. Çevirici 2-yə bərabər qiyməti o vaxt ala biləcək ki, proses 1 öz kritik bölməsini keçmiş olsun. Proses 2 də öz növbəsində proses 1-i gözləməyə məcbur edərək analoji hərəkət edəcək. Beləliklə, hər bir konkret zaman anında yalnız bir proses öz kritik intervalında ola bilər. Lakin prosesin təkrarən öz kritik bölməsinə daxil olması üçün o, digər prosesi gözləməlidir, yəni proseslər öz kritik bölmələrinə ardıcıl olaraq daxil olurlar: birinci, ikinci, birinci, ikinci... Bu, iki paralel prosesin yerinə yetirilməsini çox sərt əlaqələndirir. Ona görə də sinxronlaşdırma alqoritminə zəruri şərt əlavə etmək tələb olunur: ixtiyari prosesin onun kritik intervalından kənarda hər hansı ləngiməsi qalan proseslərin gedişinə təsir etməməlidir. Sinxronlaşdırma alqoritmi də müvafiq surətdə dəyişəcəkdir: adi çevirici əvəzinə proses öz kritik bölməsinə daxil olduqda, yəni ümumi resursu tutduqda "hə" qiyməti alan bayraq1 və bayraq2 məntiqi dəyişənləri daxil ediləcək.

Bu alqoritm daha uğurludur: o, kritik bölmələrin dəqiq ardıcıl icrasını tələb etmir. Əgər proseslərdən biri o birindən qısadırsa, onda o, daha tez-tez kritik bölməyə düşə bilər. Lakin bu alqoritmin də çatışmazlıqları var. Hər iki proses öz kritik bölmələrinə daxil olmağa hazır olan halda, yəni bayraq1 və bayraq2 "hə" qiyməti alıbsa, mənimsəmə komandasından sonrakı gözləmə dövrləri sonsuz



icra olunacaqlar və proseslərdən heç biri öz kritik bölməsinə daxil ola bilməyəcək. Bu, N.V.Qoqolun "Ölü canlar" əsərindəki Çiçikov və Manilovun nəzakətliliyini xatırladır, onlar açıq qapı qarşısında duraraq bir-birinə birinci keçməyi təklif edirlər: "...Yalnız sizdən sonra". Hər iki halda daha bir şərtin ödənilməsinə zərurət yaranır: hansı prosesin öz kritik intervalına daxil olması haqqında qərar sonlu vaxtda qəbul olunmalıdır. Belə alqoritmdə sanki birinci və ikinci alqoritmlər birləşirlər. Sinxronlaşdırma üçün həm çevirici kəmiyyəti, həm də bayraq1 və bayraq2 kəmiyyətləri tələb olunur. Çeviricinin qiyməti əvvəlki kimi kritik bölmədə yerinə yetirilən prosesin nömrəsinə bərabərdir.

Qeyd etmək olar ki, çevirici kəmiyyəti ümumiyyətlə inisiallaşdırma (başlanğıc qiymətin verilməsi) tələb etmir, çünki gözləmə dövrünə qədər onun qiyməti "1" və ya "2" təyin olunur. Çeviricidən fərqli olaraq bayraqlar salına bilər ("yox" qiyməti). Onları inisiallaşdırmamaq olmaz, çünki proseslərdən biri "irəli çıxa" bilər və o biri öz bayrağını mənimsəməyə macal tapmaz. Proseslərin əvvəlində bayraq qaldırılır ("hə" qiyməti), proses kritik bölməyə daxil olmağa hazırdır. Çevirici bir növ qonşusunu "irəli buraxaraq" digər prosesin nömrəsini özünə götürür. Əgər o biri proses öz kritik bölməsini tərk edərsə, gözləmə dövrü

```
alq paralel proseslər
baş mənt bayraq1, bayraq2, tam çevirici
  | bayraq1 := "yox"
  | bayraq2 := "yox"
  | proses1 : proses2 | proseslərin paralel icrası
son
```

```
alq proses 1
baş
  | db
  |   | bayraq1 := "hə"
  |   | çevirici := 2
  |   | db nə qədər ki bayraq2 və
  |   | çevirici = 2
  |   | ds
  |   | kritik bölmə1
  |   | bayraq1 := "yox"
  |   | son 1
  | ds
son
```

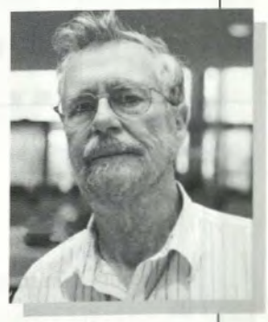
```
alq proses 2
baş
  | db
  |   | bayraq2 := "hə"
  |   | çevirici := 1
  |   | db nə qədər ki bayraq1 və
  |   | çevirici = 1
  |   | ds
  |   | kritik bölmə2
  |   | bayraq2 := "yox"
  |   | son 2
  | ds
son
```

kəsiləcək (öz bayrağını salaraq). Çevrici yalnız onun üçün lazımdır ki, Çiçikovla Manilovun düşdükləri vəziyyət baş verməsin.

Qarşılıqlı kənarlaşdırma problemi XX əsrin 60-cı illərinin əvvəlində meydana çıxmışdır. Alqoritmin ilk yaradıcılarından biri Donald Knut hesab olunur. Tanınmış amerika proqramçısı Edsger Deykstra 1965-ci ildə paralel proseslərin sinxronlaşdırılması probleminin tam və orijinal həllini verdi. Lakin proseslərdən biri öz "sonluğunu" icra edərkən sınırdısa (dayanırdısa), onda bu alqoritmlər işləmirdi. Sadəliyinə və gözəlliyinə görə ən yüksəkdə olan alqoritmi 1981-ci ildə amerikalı Harri Petersen təklif etmişdir (bu alqoritmin versiyası yuxarıdakı misalda kritik bölmə probleminin həlli kimi verilmişdir). Petersenin alqoritmi o qədər sadədir ki, deyilənə görə, ümumiyyətlə, izahata ehtiyacı yoxdur.

SEMAFOR TEXNİKASI

Edsger Deykstra təkcə alqoritm təklif etməyib, həm də paralel proseslərin sinxronlaşdırılması üçün orijinal vasitə fikirləşib tapmışdır. Sinxronlaşdırma



Edsger Deykstra.

SEMAFORLAR

Semaforlar paralel proseslərin sinxronlaşdırılması üçün ən sadə üsullardan biridir. Bundan heç də az olmayan gözəl və effektiv alqoritmlər şərti kritik intervallardan istifadə etməklə alınır (Ümumi resurslar xüsusi tip və ya ümumi dəyişənlərlə təmsil olunur. Bu dəyişənlərdən proses yalnız xüsusi operatorların – şərti kritik intervalların tərkibində istifadə edə bilər). Oxşar operatorlar Pascal dilinin xüsusi versiyasına daxil edilmişdir. Paralel proseslərin sinxronlaşdırılması texnikasının inkişafı monitor-proqramların meydana çıxmasına gətirib çıxarmışdır. Bu proqramlarda isə həm sinxronlaşdırılmanı idarə edən dəyişənlər, həm də resursların tutulması və azad edilməsi proqramları var.

Qarşılıqlı kənarlaşdırma problemini həll edən alqoritm asanlıqla N sayda paralel proses üçün genişləndirilə bilər. Sinxronlaşdırma üçün əvvəlki kimi çevrici və hər prosesə bir bayraq lazım gələcək.

zamanı əsas çətinlik ondan ibarətdir ki, kəmiyyətlərin qiymətinin yoxlanması və bu kəmiyyətlərin dəyişilməsi birbirindən asılı deyillər. Proseslərin birinin, onun kritik intervala daxil olmasına icazə verən şərti yoxladığı müddətdə digər prosesin həmin qiyməti dəyişməyə başladığı haqqında heç bir məlumatı yoxdur (yəni yoxlanılan müddət ərzində şərtin dəyişildiyi bilinmir).

Deykstra xüsusi *semafor* tipli dəyişənlər təyin etmiş və onlar üzərində apanılan iki əməliyyat daxil etmişdir. Semaforlar tam mənfi olmayan qiymətlər alırlar. Yalnız "0" və "1" qiymətlərini alan *binar semaforlar* da mövcuddur.

V əməliyyatı semaforun qiymətini bir vahid artırır. **P** əməliyyatı isə semaforun qiyməti sıfırdan böyükdürsə, onu bir vahid azaldır və ya dayanır və hər hansı bir proses semaforun qiymətini artırana qədər gözləyir (sem "semafor" tipini göstərir):

```
alq V (sem s)
baş
| s := s + 1
son
```

```
alq P (sem s)
baş
|db nə qədər ki s = 0
| ds
| s := s - 1
son
```

P və **V** əməliyyatlarının bölünməz olması çox vacibdir. Bu o deməkdir ki, **V** əməliyyatı yerinə yetirilməyə başlayandan qurtarana kimi semafora müraciət qadağandır. Bir neçə prosesin **P** əməliyyatında gözlədiyi halda (semafor sıfıra bərabər olduqda) və o başqa proses vasitəsilə dəyişdikdə (ona **V** əməliyyatının tətbiqi) yalnız bir **P** əməliyyatı sona çata bilər. Qalanları isə növbəti dəfə **V** əməliyyatının icra olunmasını gözləyəcəklər. Öz adına uyğun olaraq bir semafor bir bölünməz resurs üzrə hərəkəti təşkil edir. Başqa sözlə, proseslərin sinxronlaşdırılması probleminin həlli üçün proseslərin sayından asılı olmayaraq cəmisi bir semafor tələb olunur.

```
alq paralel proseslər
baş sem semafor
 | semafor := 1
 | proses1 : proses2 | proseslərin paralel icrası
son
```

```
alq proses 1
baş
 | db
 | | P (semafor)
 | | kritik bölmə1
 | | V (semafor)
 | | alqoritm 1-in sonu
 | ds
son
```

```
alq proses 2
baş
 | db
 | | P (semafor)
 | | kritik bölmə2
 | | V (semafor)
 | | alqoritm 2-nin sonu
 | ds
son
```

OXUCULAR – YAZIÇILAR

Semaforların köməyilə bir çox sinxronlaşdırma problemləri həll edilir. Maraqlı misallardan biri real əməliyyat sistemləri üçün tipik olan oxucular-yazıçılar məsələsidir. Bir neçə yazıçı-proses ümumi resurs olan yaddaş üzərində yazı alqoritmini yerinə yetirirlər. Oxucu-proses adlanan digər proseslər yaddaşdan oxuyurlar. Məsələnin şərtinə görə eyni vaxtda yalnız bir yazıçı öz kritik intervalında ola bilər və bu zaman yaddaşdan oxumaq olmaz. Əksinə, eyni vaxtda bir nesə oxucu ümumi yaddaşa müraciət edə bilər.

Məsələnin iki versiyası var. Birincidə yazıçının buferə yazmaq üçün gözləməsi növbəti oxucunun artıq oxumaqla məşğul olanlara qoşulmasına mane olmur. İkinci halda yazıçılar ümumi yaddaşa müraciətdə oxuculara nisbətən prioritetə malikdirlər: hansisa yazıçı yazmağa hazırdırsa, onda yeni oxuculara imkan verilmir. Məsələnin axırıncı versiyası, şübhəsiz ki, birincidən mürəkkəbdir. Sadəlik üçün birinci versiyanın həlli veriləcək.

Məsələnin həlli üçün iki semafor tətbiq olunur. Birinci semafor – yaddaş – yazıçıların yaddaşa müraciətinin sinxronlaşdırılması üçündür. Əgər heç olmasa bir oxucu oxumaqla məşğuldursa, semafor yazıçıya yazmağa icazə verməməlidir. Bunun üçün hal-hazırda oxumaqla məşğul olan proseslərin sayını göstərən oxucuların sayı tam qiymətli kəmiyyət daxil edilir. Birinci proses oxumağa başlamaq istəyəndə, o, oxucuların sayını artırır – qiymət "1" olur. Bu zaman növbəti şərt doğru olur və yaddaş semaforu ilə *P* əməliyyatı yerinə yetirilir. Əgər resurs azaddırsa, onda o, oxucular tərəfindən "tutulur", yaxud da yazıçının öz kritik bölməsini – yaddaşa yazmanı haçan qurtaracağını gözləyir. Analoji hərəkətlər sonuncu kritik bölmədən (yaddaşdan oxuma) sonra oxucu resursu azad edib ora yazmağa və ya oradan oxumağa icazə verdikdə yerinə yetirilir.

Oxucu semaforu oxucuların sayı kəmiyyətinə yalnız bir prosesin müraciətini




```
alq oxucular-yazıçılar
baş sem oxucu, yaddaş, tam oxucuların sayı
 | oxucuların sayı := 0
 | yaddaş := 1
 | oxucu := 0
 | oxucu : ... |oxucu-proseslərin paralel icrası
 | ... :yazıçı  |yazıçı-proseslərin paralel icrası
son
```

```
alq oxucu
baş
 | P (oxucu)
 | oxucuların sayı := oxucuların sayı + 1
 | əgər oxucuların sayı = 1
 | | onda P (yaddaş)
 | tamam
 | V (oxucu)
 | yaddaşdan oxuma
 | P (oxucu)
 | oxucuların sayı := oxucuların sayı - 1
 | əgər oxucuların sayı = 0
 | | onda V (yaddaş)
 | tamam
 | V (oxucu)
son
```

```
alq yazıçı
baş
 | P(yaddaş)
 | yaddaşa yazma
 | V(yaddaş)
son
```

məhdudlaşdırmağa xidmət edir. Bu kəmiyyətin artması (və ya azalması) onun şərtdə yoxlanması ilə birlikdə baş verir. Belə üsulu proseslərin hər hansı bir kəmiyyətə müraciətini məhdudlaşdırmaq lazım gəldikdə tətbiq etmək əlverişlidir. Diqqət yetirmək vacibdir ki, baxılan məsələnin həlli üçün binar semafordan istifadə olunub. Deykstra öz işlərində göstərib ki, semaforların vasitəsilə həll olunan bütün məsələlər yalnız binar semaforlar tətbiq etməklə asanlıqla icra oluna bilər. Ümumiyyətlə, bütün semaforları tam kəmiyyət və binar semafor kimi göstərmək olar. Bunun belə olduğu oxucuların sayı kəmiyyətinin və oxucu semaforunun timsalında nümayiş etdirildi.

NAHAR EDƏN FİLOSOFLAR HAQQINDA MƏSƏLƏ

Gündəlik həyatda paralel müraciətli məsələlərə tez-tez rast gəlmək olur. Ən sadə misal: əgər vanna otağında artıq kimsə yuyunursa və başqa kiməsə nə

Bir çox proqramlaşdırma dillərində də paralel proqramlaşdırma imkanı mövcuddur. Belə ki, SmallTalk-da xüsusi Process sinfi var, C++ və Java-da isə sinxronlaşdırma və paralellizm xüsusi siniflər kitabxanası vasitəsilə yerinə yetirilir.

```
alq nahar edən filosoflar
baş semcəd çəngəl [1:5], həq k
 | db k üçün 1 dən 5 dək
 |  | çəngəl[k] := 1
 | ds
 | filosof (1) : ... filosof (5) | filosof-proseslərin paralel başladılması
son
```

```
alq filosof (həq n)
  verilib n
  gərəkdir
baş
 | db
 |  | P(çəngəl [n])
 |  | P(çəngəl [növbəti (n)])
 |  | NAHAR
 |  | V(çəngəl [n])
 |  | V(çəngəl [növbəti (n)])
 |  | DÜŞÜNCƏLƏR
 | ds
son
```

```
alq tam iz (tam n)
verilib n
gərəkdir
baş
 | n := n + 1
 | əgər n > 5
 |  | onda n := 1
 | tamam
 | qiym := n
son
```

üçünsə vanna otağı lazım olarsa, onda ikinci adam vanna otağına girib birincilə bir yerdə yuyunmaqdansa, onu tələsdirməyə çalışar. Paralel proseslərə aid daha bir məsələ bütün sürücülərə yaxşı məlum olan tənzimlənməyən dörd yol ayrıcını keçmək problemidir. Yol hərəkəti qaydalarına görə sürücü sağındakı maşına yol verməlidir. Bəs əgər yol ayrıcına müxtəlif istiqamətlərdən eyni zamanda dörd avtomobil yaxınlaşarsa, onda nə etməli?



Sinxronlaşdırmaya aid klassik məsələ – nahar edən filosoflar haqqında məsələdir. Qəsrdə beş filosof yaşayırdı. Onlar bütün vaxtlarını varlığın mahiyyəti haqqında düşünməklə keçirirdilər. Nadir hallarda onların nahar masası arxasında görüşləri olurdu. Filosoflar ancaq çəngəllə yeyirdilər, özü də eyni zamanda ikisilə. Qəsrdə cəmi beş çəngəl var idi və onlar da masanın üstündə olurdu.

Alimlərdən kimsə yemək otağına düşəndə beşbucaqlı masanın arxasında boş yerə əyləşir, sağdakı və soldakı çəngəllər boşdursa, onları götürüb yeməyə başlayır. Sonra çəngəlləri yerinə qoyur və öz otağına gedir. Nahar və düşüncələr, nahar və düşüncələr – filosoflar qəsrdə öz vaxtlarını belə keçirirdilər.

Alqoritmin yazılışını sadələşdirmək üçün hər çəngəlin öz semafor resursu olmaqla beş semafordan ibarət semcəd cədvəlindən istifadə olunur. Alqoritm ixtiyari sayda filosof olan hal üçün yazılıb. İz köməkçi alqoritmi $(n+1) \bmod 5$ əməliyyatını həyata keçirir (mod 5 – ədədin 5-ə bölünməsindən alınan qalıqdır).

Lakin nəzəri olaraq elə hal mümkündür ki, bütün filosoflar eyni vaxtda nahara gəlsin, eyni zamanda hərəsi sağ əlinə bir çəngəl alsın və soldakı çəngəlin nə vaxt boşalacağını gözləməyə başlasınlar. Belə vəziyyət deadlock (*ing.* "ölümcül sarmaşma") adlanır, o, proseslərin qarşılıqlı bloklaşmasını yaradır. Bloklaşma zamanı filosoflar sadəcə olaraq acından öləcəklər. Aşağıdakı alqoritm bu problemdən can qurtarmağa kömək edir. İndi çəngəl – onu heç kim götürməyən halda hə qiyməti alan məntiqi dəyişəndir. Bu kəmiyyətin dəyişməsini adi semafor (sem) "qoruyur". Filosof yeyə bilməyəndə nə etməli? O, masanın arxa-



sındaca yatmalıdır. Yuxu semaforu müvafiq prosesi bloklayır: əgər filosof yeməyə hazırlaşıbsa, çəngəl isə başqasındadırsa, onda o, gözləyə-gözləyə yatır. Lazım olan çəngəli azad edən başqa filosof onu oyadır. Ümumiyyətlə, artıq nahar etmiş filosof hər ehtimala qarşı qonşularını oyatmağa başlayır. Bu zaman yenə də binar semafor tələb olunur ki, semaforun qiyməti heç vaxt biri aşmasın. Göstərilən alqoritm proseslərin bloklanmasından can qurtarmağa imkan verir, lakin hansısa bir filosofun gah sol, gah da sağ çəngəlin olmaması səbəbindən daim yatması halı da mümkündür. Belə vəziyyəti starvation (*ing.* "acqalma") adlandırırlar. Doğrudur, bu problemin də həlli var, lakin bu həll uşaq ensiklopediyası üçün yetərincə mürəkkəbdir.

```
alq nahar edən filosoflar
baş semcəd yuxu[1:5], sem tam k, mənt cəd çəngəl[1:5]
 | db k üçün 1 dən 5 dək
 |  | yuxu [k] := 1
 |  | çəngəl [k] := "hə"
 | ds
 | filosof (1) : ... filosof (5) | filosof-proseslərin paralel başladılması
son
```

```
alq filosof (tam n)
 verilib n
 gərəkdir
 baş
 | db
 |  | P(sem)
 |  | db nə qədər ki deyil (çəngəl [n] və çəngəl [növbəti (n)])
 |  |  | V(sem)
 |  |  | P(yuxu [n])
 |  |  | P(sem)
 |  | ds
 |  | çəngəl [n] := "yox"
 |  | çəngəl [növbəti (n)] := "yox"
 |  | V(sem)
 |  | NAHAR
 |  | P(sem)
 |  | çəngəl [n] := "hə" | n – sağdakı çəngəl
 |  | çəngəl [növbəti (n)] := "hə" | n+1 - soldakı çəngəl
 |  | V(yuxu [növbəti (n) ] ) | n+1 - soldakı qonşu
 |  | V(yuxu [növbəti(növbəti(növbəti(növbəti (n))))]) | n+4 sağdakı qonşu
 |  | V(sem)
 |  | DÜŞÜNCƏLƏR
 |  | ds
son
```





PROQRAMLAŞDIRMA NƏZƏRİYYƏSİ

Məntiq leyni zamanda "söz" və "məna" anlamını verən *yun.* "loqos" sözündən) – düzgün təfəkkürün qanunları, formaları və əməliyyatları haqqında elmdir. Onun əsas vəzifəsi düzgün düşünmə üsullarının tapılması və sistemləşdirilməsindən ibarətdir.

RİYAZİ MƏNTİQ

KLASSİK MƏNTİQ

Məntiq elmi çox qədimlərdən məlumdur, qədim yunan filosofu Aristotel (Ərəstun, b.e.ə. 384-322-ci illər) onun atası hesab olunur.

Əgər qar yağırsa, onda indi qışdır; qar yağır; deməli, indi qışdır.

Bu mühakimə üç müddəadan ibarətdir. Birinci iki müddəa *hökm*, üçüncüsü isə *nəticə* adlanır.



Hələ antik filosoflar qeyd etmişdilər ki, çox zaman mühakimələr onların tərkibinə daxil olan müddəaların məzmunundan asılı olmayaraq quruluşca xeyli oxşarlığa malik olurlar.

Başqa misala baxaq:

Əgər suyu 100°C-yə kimi qızdırsaq, onda o qaynayar; suyu 100°C-yə qədər qızdırdılar; deməli, su qaynayır.

Məzmun fərqinə baxmayaraq hər iki mühakimənin forması (sxemi) eynidir:

Əgər birinci varsa, onda ikinci də var; birinci var; deməli, ikinci də var.

Mühakimənin düzgünlüyünü onun gətirib çıxardığı nəticələrə əsasən qiymətləndirmək ağlabatan olardı. Bu sxemlə aparılan mühakimə hansı obyektdən söhbət getməsindən asılı olmayaraq düzgün olacaq. Əgər mühakimə düzdürsə, onda onun hökmlərindən nəticə çıxır. Belə hallarda deyirlər ki,

mühakimənin sxemi *məntiqi qanun* təşkil edir.

Düzgün mühakimənin bir çox başqa sxemləri də mövcuddur. Məsələn:

Ya birinci, ya da ikinci var; birinci yoxdur; deməli, ikinci var.

Məntiqin əhəmiyyəti ondadır ki, o, təcrübəyə və ya intuisiyaya müraciət etmədən formal mühakimələrin köməyilə yeni biliklər yaratmağa imkan verir.

Aristotelin dövründən keçmiş iki minillik ərzində ənənəvi məntiq o qədər də irəli getməmişdir. Böyük alman filosofu İmmanuil Kant (1724-1804) hətta hesab edirdi ki, bu elm artıq özünün inkişafını başa çatdırmışdır. Lakin tədricən məntiqdə inqilab yetişirdi və onu başqa alman filosofu, riyaziyyatçı, fizik, ixtiraçı, hüquqşünas, tarixçi, linqvist Qotfrid Vilhelm Leybnis (1646-1716) hazırladı.

Hələ 1672-ci ildə Parisdə işləyən holland ixtiraçısı və fiziki Xristian Hüygensə (1629-1695) baş çəkərkən onun zəruri riyazi hesablamalara nə qədər çox qüvvə və vaxt sərf etdiyi Leybnisi sarsıtmışdı. Bəlkə də, elə bu təəssüratlar Leybnisin hesablama problemi üzərində düşüncələrinin başlanğıc nöqtəsi olmuşdur. O, sonralar yazırdı: "Belə gözəl insanlar, şübhəsiz ki, maşından istifadə etməklə ixtiyari bir şəxsə tapşırıla bilən hesablama işlərinə qullar kimi vaxt itirməyə layiq deyillər". Leybnis belə bir maşını bir çox illərdən sonra, 1694-cü ildə yarada bildi.

Leybnis hesablama problemini həm də nəzəri cəhətdən öyrənirdi. 1703-cü ildə o, ilk dəfə dünya elminin diqqətini ikilik say sisteminə cəlb etdi. O, ikilik sistemin onluq sistemə nisbətən üstünlüyünü isbat edərək göstərirdi ki, sıfır və birlərin ardıcıllığında onların davranış qanunauyğunluqlarını (onları "möcüzəli nizam" adlandırırdı) aşkar etmək daha asandır.

Leybnis birinci olaraq fikir vermişdir ki, hesabi əməliyyatların yerinə yetirilməsi sadə qaydaların köməyilə aparılır və bu qaydalar üçün ədədlərin mənası deyil, onların necə yazılması əhəmiyyətlidir. Bu qaydalar kifayət qədər aşkardır, onların tətbiqi birqiymətli nəticə verir və deməli, bu nəticələri şübhə altına almaq və ya qəbul etməmək olmaz. Bundan sonra Leybnisdə belə bir ideya yarandı: insanın çıxardığı nəticələri hesablamalar şəklində göstərməklə, onları da belə danılmaz etmək olmazmı? Doğrudur, "Hesablayacağıq" devizini təklif edərək o, bu problemi daha çox filosof kimi tədqiq etmişdir.



Leybnis hesab edirdi ki, vahid ilahi başlanğıcı, sıfır isə qeyri-mövcudluğu (yoxluğu) təcəssüm etdirir. İxtiyari ədədi ifadə etməyə imkan verən sıfıra vahidlərin əlavə edilməsini o, ilahi qüvvə tərəfindən bütün mövcudiyyətin heçdən yaradılması ilə eyniləşdirirdi.



İmmanuel Kant.



Qotfrid Vilhelm Leybnis.

Hökmlərin riyaziləşdirilməsi haqqında Leybnisin ideyaları öz dövrünü xeyli qabaqlasa da, onun müasirlərinin elmi baxışlarına hiss olunacaq təsir etməmişdi. Daha yüz əlli il lazım gəldi ki, ingilis riyaziyyatçısı Corc Bulun (1815-1864) əsərlərində məntiqə riyazi yanaşmanın əsasları qoyulsun. Maraqlıdır ki, Bulun riyazi təhsili olmamışdır, lakin bu, ona 1849-cu ildə Kork (İrlandiya) Kuins-kollecinin riyaziyyat kafedrasının professoru olmağa mane olmamışdır. Onun "Məntiqin riyazi analizi" (1847), "Məntiq hesabı" (1848), "Təfəkkür qanunlarının tədqiqi" (1854) əsərləri müasir riyazi məntiqin təməlində dayanır.



Corc Bul.

MƏNTİQ CƏBRİ

Məktəb cəbri hərflərdən, hesab əməliyyatlarının işarələrindən və mötərizələrdən təşkil olunmuş *düsturları* öyrənir.

Düsturları çevirmək olar. Cəbri düsturların hər hansı mümkün (ekvivalent) çevrilmələri aşağıdakı qanunlarla verilir:

• assosiativlik –

$$(a + b) + c = a + (b + c),$$

$$(a \cdot b) \cdot c = a \cdot (b \cdot c);$$

• kommutativlik –

$$a + b = b + a,$$

$$a \cdot b = b \cdot a;$$

• distributivlik –

$$a\,(b + c) = ab + ac;$$

həmçinin onların köməyilə alınan aşağıdakı kimi vacib nəticələr (məsələn, müxtəsər vurma düsturları) vasitəsilə verilir:

$$a^2 - b^2 = (a + b)\,(a - b).$$

Bundan başqa, düsturlar hesablamaq üçün də istifadə oluna bilər; bu halda hərflərin yerinə onların ədədi qiymətləri qoyulur və uyğun hesab əməlləri yerinə yetirilir. Hər bir düstur, arqumentlərin sayı düsturdakı fərqli hərflərin sayına bərabər olan hesabi funksiya verir.

Məntiq cəbri də düsturları öyrənir. Lakin buradakı dəyişənlər yalan və doğru kimi iki mümkün qiymətdən yalnız birini ala bilər. Bu qiymətlər *məntiqi* qiymətlər adlanır. Onlar həmçinin 0 və 1, Y və D, false və true kimi də işarə olunurlar. Xüsusi məntiqi əməliyyat işarələri daxil edilir.

Bu halda da hər bir düstur funksiya təyin edir. Bunlar *məntiqi funksiyalar* adlanırlar. Onların arqumentləri məntiqi dəyişənlərdir, funksiyalar özləri isə yalnız məntiqi qiymətlər ala bilər.

İxtiyari belə funksiyanı arqumentlərin qiymətlərinin bütün mümkün yığını və funksiyanın müvafiq qiymətləri göstərilmiş cədvəl şəklində vermək olar. Bir arqumentli $\varphi(x)$ funksiyası halında belə cədvəl çox sadə olacaq:

x	φ_1	φ_2	φ_3	φ_4
0	0	0	1	1
1	0	1	0	1

Birinci funksiya, φ_1, arqumentlərin bütün mümkün qiymətlərində 0 qiyməti alır, yəni φ_1 funksiyası 0 *sabitidir*, analoji olaraq φ_4 funksiyası 1 *sabitidir*. Bu funksiyaların qiyməti x-dən asılı deyil, ona görə də deyirlər ki, onlar üçün x dəyişəni *əhəmiyyətsizdir*. Öz arqumentinin qiymətini tərsinə dəyişən φ_3 funksiyası *inkar* adlanır. İnkar müxtəlif çür işarə olunur: $\varphi_3(x) = \neg x$ və ya daha tez-tez $\varphi_3(x) = \bar{x}$. Nəhayət, $\varphi_2(x) = x$ funksiyası *təkrar* adlanır.

Bir məntiqi dəyişənin cəmi dörd funksiyası var. Ümumi halda isə n dəyişənin məntiqi funksiyalarının sayı 2^{2^n} (2^n – 0 və 1-dən ibarət olan n uzunluqlu bütün mümkün yığınların sayıdır, yəni cədvəldəki sətirlərin sayıdır. Onda 2^{2^n} 0 və 1-lərin bu qədər sayda sətirdə sütun üzrə yerdəyişmələrinin sayına bərabərdir). Deməli, iki dəyişənin məntiqi funksiyalarının sayı $2^{2^n} = 16$.

Bir dəyişənli funksiyası halında olduğu kimi, φ_1 və φ_{16} müvafiq olaraq 0 və 1 sabitləridir. Onlar üçün hər iki dəyişən əhəmiyyətsizdir. $\varphi_4(x_1, x_2)$ və $\varphi_{13}(x_1, x_2)$ funksiyaları üçün x_2 dəyişəni əhəmiyyətsizdir, çünki $\varphi_4(x_1, x_2) = x_1$ və $\varphi_{13}(x_1, x_2) = \bar{x}_1$.

x_1	x_2	φ_1	φ_2	φ_3	φ_4	φ_5	φ_6	φ_7	φ_8	φ_9	φ_{10}	φ_{11}	φ_{12}	φ_{13}	φ_{14}	φ_{15}	φ_{16}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

$\varphi_6(x_1, x_2)$ və $\varphi_{11}(x_1, x_2)$ funksiyaları üçün isə x_1 dəyişəni əhəmiyyətsizdir, çünki $\varphi_6(x_1, x_2) = x_2$ və $\varphi_{11}(x_1, x_2) = \bar{x}_2$.

$\varphi_2(x_1, x_2)$ funksiyasına x_1 və x_2-nin *konyunksiyası* deyilir. Bu funksiyanı $x_1 \& x_2$ və ya $x_1 \wedge x_2$ kimi işarə etmək qəbul olunmuşdur. Konyunksiya yalnız və yalnız o zaman 1-ə bərabərdir ki, həm x_1, həm də x_2 1-ə bərabər olsun. Ona görə də konyunksiyanı tez-tez VƏ funksiyası da adlandırırlar. Ona hərdən *məntiqi vurma* da deyirlər, çünki funksiyanın qiymətləri arqumentlərin hesab qaydalarına vurulmasına bərabərdir.

$\varphi_8(x_1, x_2)$ funksiyasına x_1 və x_2-nin *dizyunksiyası* deyilir. Dizyunksiyanı $x_1 \vee x_2$ kimi işarə edirlər. Onu həm də VƏ YA funksiyası adlandırırlar, çünki o, ya x_1, ya da x_2 vahidə bərabər olduqda 1 qiyməti alır.

Daha bir funksiya, $\varphi_7(x_1, x_2)$, *2 moduluna görə toplama* adlanır. Onun üçün $x_1 \oplus x_2$ işarələməsi tətbiq olunur. Bu funksiya, arqumentlərinin qiymətləri müxtəlif olduqda 1-ə və üst-üstə düşdükdə 0-a bərabərdir. Buradan da bu funksiyanın *müxtəlif qiymətlilik* kimi digər adı meydana çıxır.

Əksinə, $\varphi_{10}(x_1, x_2)$ funksiyası arqumentlər bərabər olduqda 1, əks halda isə 0 qiyməti alır. Ona görə də onu *ekvivalentlik* və ya *eyniqiymətlilik* adlandırırlar. Bu funksiya $x_1 \sim x_2$ və ya $x_1 \equiv x_2$ kimi işarə olunur.

Daha üç funksiyanın konkret adı var.

İmplikasiya – $\varphi_{14}(x_1, x_2)$, $x_1 \rightarrow x_2$ və ya $x_1 \supset x_2$ kimi işarə olunur; "əgər x_1, onda x_2" kimi oxunur.

Pirs oxu (başqa adı İNKAR-VƏ YA) – $\varphi_9(x_1, x_2)$. $x_1 \downarrow x_2$ kimi işarə olunur.

Şeffer ştrixi (başqa adı İNKAR-VƏ) – $\varphi_{15}(x_1, x_2)$, $x_1 | x_2$ kimi işarə olunur. Digər funksiyalar nadir hallarda istifadə olunur.

Əgər düsturda yalnız daxil etdiyimiz əməliyyatların işarələri varsa, ixtiyari dəyişənlər yığını üçün onun qiymətini asanlıqla hesablamaq olar. Məsələn, $x_1 \wedge x_2 \mid (x_1 \vee x_3)$ düsturu ilə verilən $\varphi(x_1, x_2, x_3)$ funksiyasını aşağıdakı cədvəllə vermək olar:

x_1	x_2	x_3	$\varphi(x_1, x_2, x_3)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Ümumiyyətlə, düsturda dəyişənlərin bütün dəstlərini yerinə qoymaqla funksiyaya uyğun cədvəli hər bir düstur üçün qurmaq olar. Funksiyanın (deməli, həm də düsturun) cədvəl təsviri yeganədir. Lakin eyni bir funksiya müxtəlif düsturların köməyilə yazıla bilər. Məsələn, Şeffer ştrixi funksiyasını həm $\varphi_{15}(x_1, x_2) = \bar{x}_1 \vee \bar{x}_2$, həm də $\varphi_{15}(x_1, x_2) = \overline{x_1 \wedge x_2}$ kimi yazmaq olar. Eyni funksiyanı təmsil edən belə düsturlar *ekvivalent* və ya *eynigüclü* adlanırlar.

Düsturları yazmaq üçün bütün funksiyalardan istifadə etməyə ehtiyac yoxdur. Məsələn, düsturlarda Şeffer ştrixindən yaxa qurtarmaq üçün onu konyunksiya, dizyunksiya və inkar vasitəsilə ifadə etmək olar. Ona görə də $\varphi(x_1, x_2, x_3)$ funksiyası $\overline{x_1 \wedge x_2} \vee \bar{x}_1 \vee x_3$ şəklində yazıla bilər. Məntiqi funksiyaların elə dəstləri var ki, onlar vasitəsilə bütün digər məntiqi funksiyaları ifadə etmək olar. Belə məntiqi funksiya dəstlərinə *funksional tam* və ya *bazis dəstləri* deyilir.



Əgər balina küçə ilə gedirsə, onda gecə yarısıdır!



Şeffer ştrixi kimi digər iki dəyişənli məntiqi funksiyaları da konyunksiya, dizyunksiya və inkar vasitəsilə ifadə etmək olar:

$$x_1 \to x_2 = \bar{x}_1 \vee x_2,$$

$$x_1 \sim x_2 = x_1 x_2 \vee \bar{x}_1 \bar{x}_2,$$

$$x_1 \oplus x_2 = x_1 \bar{x}_2 \vee \bar{x}_1 x_2,$$

$$x_1 \downarrow x_2 = \overline{x_1 \vee x_2} = \bar{x}_1 \bar{x}_2.$$

Yalnız bu üç əməliyyat (həmçinin dəyişənlər və mötərizələr) vasitəsilə qurulan düsturlara *Bul düsturları* deyilir (C.Bulun şərəfinə). Belə bir teorem doğrudur ki, ixtiyari məntiqi funksiya Bul düsturu şəklində göstərilə bilər. Odur ki, konyunksiya, dizyunksiya və inkardan ibarət dəst bazis əmələ gətirir. Bu üç əməliyyatın təyin olduğu məntiqi funksiyalar çoxluğu *Bul cəbri* adlanır.

Bul cəbrinin əmælləri adətən *Bul əmælləri* adlandırılır.

Bul əmælləri üçün ümumi olan aşağıdakı əsas xassələr xarakterikdir:

• Assosiativlik qanunları:

$$(x_1 \vee x_2) \vee x_3 = x_1 \vee (x_2 \vee x_3),$$

$$(x_1 \wedge x_2) \wedge x_3 = x_1 \wedge (x_2 \wedge x_3).$$

• Kommutativlik qanunları:

$$x_1 \vee x_2 = x_2 \vee x_1,$$

$$x_1 \wedge x_2 = x_2 \wedge x_1.$$

• Distributivlik qanunları:

$$x_1 \wedge (x_2 \vee x_3) = (x_1 \wedge x_2) \vee (x_1 \wedge x_3),$$

$$x_1 \vee (x_2 \wedge x_3) = (x_1 \vee x_2) \wedge (x_1 \vee x_3).$$

• De Morqan qaydaları:

$$\overline{x_1 \wedge x_2} = \bar{x}_1 \vee \bar{x}_2,$$

$$\overline{x_1 \vee x_2} = \bar{x}_1 \wedge \bar{x}_2.$$

• İdempotentlik:

$$x \wedge x = x,$$
$$x \vee x = x.$$

(Bu o deməkdir ki, cəbri düsturlardan fərqli olaraq məntiq düsturlarında qüvvət və əmsallar olmur).

• İkiqat inkar:

$$\bar{\bar{x}} = x.$$

• Ziddiyyət qanunu:

$$x \vee \bar{x} = 0.$$

• Üçüncünün qeyri-mümkünlüyü qanunu:

$$x \vee \bar{x} = 1.$$

• Sabitlərin xüsusiyyətləri:

$$x \wedge 1 = x, \quad x \wedge 0 = 0,$$

$$x \vee 1 = 1, \quad x \vee 0 = x,$$

$$\bar{0} = 1, \ \bar{1} = 0.$$

Bu *ekvivalentlik münasibətlərindən* hər birinin doğruluğu asanlıqla yoxlana bilər, sağ və sol tərəfləri orada iştirak edən dəyişənlərin bütün mümkün qiymətləri üçün hesablayaraq müqayisə etmək kifayətdir.

Bul düsturlarının yazılışı zamanı konyunksiya işarəsini cəbrdəki vurma işarəsi kimi çox vaxt yazmırlar. Analoji olaraq, bir neçə konyunksiya və ya dizyunksiya ardıcıl yerinə yetirilərkən mötərizələr yazılmır, konyunksiyada xarici mötərizələr də yazılmır:

$$(x_1 \wedge (x_2 \wedge x_3)) \vee ((x_4 \vee x_5) \wedge x_6) =$$
$$= x_1 x_2 x_3 \vee (x_4 \vee x_5) x_6.$$

Əsas ekvivalentlik çevirmələri iki cür olur. Birincisi – düsturların sadələşdirilməsi əvvəlkinə ekvivalent daha qısa düstur almaq imkanı verir.

• Udma:

$$x \vee xy = x; \ x(x \vee y) = x.$$

• Yapışdırma:

$$xy \vee x\bar{y} = x,$$

$$(x \vee y)(x \vee \bar{y}) = x.$$

• Ümumiləşdirilmiş yapışdırma:

$$xz \vee y\bar{z} \vee xy = xz \vee y\bar{z}.$$

İkinci tip isə dizyunktiv normal formaya (DNF) gətirilmədir.

Dəyişənlər və onların inkarlarından ibarət olan və hər bir dəyişənin bircə dəfə işlənməsi şərtilə konyunksiyası *elementar konyunksiya* adlanır. DNF elementar konyunksiyaların dizyunksiyası şəklində düsturdur. Ümumiləşdirilmiş yapışdırma münasibətindən göründüyü kimi hər hansı düsturun DNF-i ümumiyyətlə yeganə deyil.

Əgər n dəyişənli $\varphi(x_1, x_2, \ldots, x_n)$ funksiyası cədvəl vasitəsilə verilmişsə, onun Bul düsturunu aşağıdakı şəkildə qurmaq olar. Funksiyanın 1-ə bərabər olduğu bütün dəyişən yığınlarının konyunksiyaları yazılır. Bu zaman 0-a bərabər olan dəyişənlərin inkarı götürülür. Sonra isə bütün konyunksiyalar dizyunksiya işarələrilə birləşdirilir. Alınan düstur *mükəmməl dizyunktiv normal forma* (MDNF) adlanır. Hər bir funksiya üçün MDNF yeganədir (bu, elə onun qurulması qaydasından çıxır), onlar ancaq konyunksiyaların daxilində dəyişənlərin ardıcıllıq sırası və konyunksiyaların sırası ilə fərqlənə bilər. Fəqət 0 sabit funksiyasının MDNF-i yoxdur.

Cədvəldən istifadə edərək $\varphi(x_1, x_2, x_3)$ funksiyasının MDNF-ini quraq:

$$\bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_2 \bar{x}_3 \vee x_1 x_2 \bar{x} \vee x_1 x_2 x_3.$$

Alınan düstur kifayət qədər uzundur, lakin onu ekvivalentlik çevirmələrindən istifadə edərək sadələşdirmək olar. Bu halda iki dəfə yapışdırmanı tətbiq etsək (uyğun olaraq birinci və ikinci, həmçinin üçüncü və dördüncü konyunksiyalara):

$$\bar{x}_1 \bar{x}_3 \vee x_1 x_2.$$

Alınan DNF xeyli qısadır.

Konyunksiya və inkar, dizyunksiya və inkar dəstləri də bazisdir. İnkar vasitəsilə konyunksiyanı dizyunksiya ilə və tərsinə ifadə etmək olar:

$$x_1 \vee x_2 = \overline{\bar{x}_1 \vee x_2} = \bar{x}_1 \bar{x}_2$$

$$x_1 x_2 = \overline{\bar{x}_1 x_2} = \bar{x}_1 \vee \bar{x}_2$$

Başqa bazislər də mövcuddur. Bunların arasında konyunksiya, modul 2-yə görə toplama və 1 sabitindən ibarət Jeqalkin bazisi daha çox tanınır. Doğrudan da,

$$\bar{x} = x \oplus 1,$$
$$x_1 \vee x_2 = x_1 x_2 \oplus x_1 \oplus x_2.$$

Bazis hətta bir funksiyadan ibarət ola bilər. Məsələn, həm Şeffer ştrixi, həm də Pirs oxu funksiyaları bazis əmələ gətirir, çünki inkar, konyunksiya və dizyunksiya funksiyalarından istənilən biri onların vasitəsilə ifadə oluna bilər:

$$\bar{x} = x \mid x = x \downarrow x,$$

$$x_1 x_2 = \overline{\bar{x}_1 \mid \bar{x}_2} = (x_1 \mid x_2) \mid (x_1 \mid x_2)$$

$$x_1 \vee x_2 = \overline{x_1 \downarrow x_2} = (x_1 \downarrow x_2) \mid (x_1 \downarrow x_2)$$



BUL CƏBRİ VƏ KOMPÜTER



M.A.Qavrilov.

Məhz proqramlarda şərtin yoxlanması, şərti keçidlərin olması hesablayıcı qurğularla (arifmometr və ya kalkulyator kimi) kompüter arasında dəqiq sərhədi təyin edir. Bu, proqramçıya yazdığı proqramın bütün icra variantlarını nəzərə almaq imkanı verir.

Bir şərtin doğruluğunu yoxlamaq üçün məntiqi düsturların tərtib olunması və məntiq cəbrinin qaydaları əsasında məntiqi ifadələrin hesablanması lazım gəlir. Buna görə də FORTRAN və Algol-60 kimi ilk dillərdən başlayaraq bütün proqramlaşdırma dillərində məntiqi verilənlər və onlar üzərində əməllər təyin olunub. Proqramlaşdırma dillərində məntiqi qiymətlər ənənəvi olaraq "false" və "true" ilə işarə olunur ki, bu da proqramın mətnində onları 0 və 1 ədədlərilə qarışıq salmamağa imkan verir (hərçənd ki, məntiqi dəyişənin istənilən sıfırdan fərqli qiyməti faktiki olaraq "doğru" hesab olunur).

Proqramlaşdırmada məntiq cəbrinin maraqlı bir tətbiqi *predikat hesablamaları* adlanır. Kompilyator bəzi maşın komandalarını predikatlarla (məntiqi şərtlərlə) təmin edir. Predikatların qiymətləri xüsusi registrlərdə saxlanır, kompüterin komandalar sisteminə isə predikatlar üzərində komandalar daxil olunur.

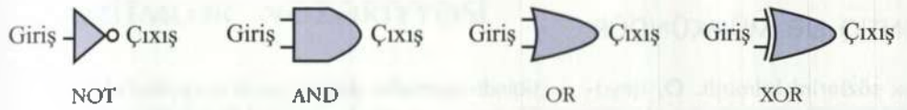


Ventillər təkcə sxemlərdə olmur.

Komandalar qiymətindən asılı olmadan predikatların idarəsi altında icra olunur. Amma bu zaman əgər predikatın qiyməti "*true*"dırsa, komanda adi qaydada yerinə yetirilir, predikatın qiyməti "*false*" olduqda isə o, prosessorun vəziyyətini dəyişdirmir. Predikat mexanizminin istifadəsi kompilyatora proqramın effektiv icra olunması üçün onun komandalarını yenidən nizamlamaq imkanı verir.

İnformatikada məntiq cəbrinin daha bir olduqca vacib tətbiqi *məntiqi sxemlərin* qurulmasıdır. Texnikada məntiqin tətbiq olunmasının mümkünlüyü alim və mühəndisləri hələ çoxdan düşündürürdü. Məsələn, bir neçə il Rusiyada işləmiş holland fiziki Paul Erenfest (1880-1933) hələ 1910-cu ildə yazırdı: "...tutaq ki, avtomatik telefon stansiyasının naqillər sxemi verilib. Təyin etmək lazımdır ki: 1) o, stansiyanın işi zamanı rast gələ biləcəyimiz istənilən kombinasiyalar üçün düzgün işləyəcəkmi; 2) bəlkə o, zəruri olmayan izafi mürəkkəbliyə malikdir. Hər bir belə kombinasiya məntiqi mühakimə, hər kiçik kommutator isə ebonit və latunda təcəssüm olunmuş məntiqi "ya, ya da"dır; bunların hamısı birlikdə mürəkkəblik və qarışıqlıq cəhətdən geniş imkanları olan sırf keyfiyyət "mühakimələri" sistemidir... Doğrudanmı məntiq cəbrinin mövcud olmasına baxmayaraq özünəməxsus "paylayıcı sistemlər cəbri" utopiya hesab olunmalıdır?". Daha sonra Mixail Aleksandroviç Qavrilov (1903-1979) tərəfindən yaradılmış rele-kontakt sxemləri nəzəriyyəsi bunun heç də utopiya olmadığını göstərdi.

Mürəkkəb məntiqi sxemlərin qurulmasının əsasında *ventillər* dayanır. Girişinə ilkin verilənlər daxil olan, çıxışında isə hər hansı Bul əməlinin nəticəsi alınan ən sadə qurğular belə adlandırılır. Texnikanın inkişafının müxtəlif mərhələlərində ventillər mövcud olan texnologiyadan istifadə edilməklə, məsələn, dişli

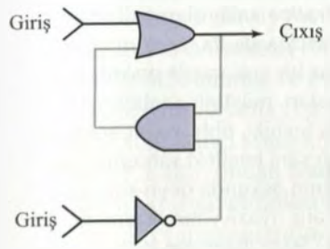


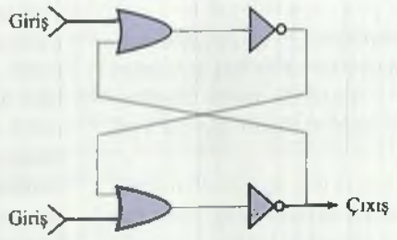
çarxlar və ya elektromexaniki relelər şəklində qurulurdu. Müasir kompüterdə isə ventil kiçik elektron dövrədir ki, burada 0 və 1 qiymətləri elektrik gərginliyinin müxtlif səviyyələrinə uyğun gəlir.

Şəkildə NOT (inkar), AND (məntiqi VƏ), OR (məntiqi VƏ YA) və XOR (müxtəlif qiymətlilik, istisnalı VƏ YA) funksiyalarına uyğun olaraq məntiqi sxemlərdə istifadə olunan ventil işarələri təsvir olunub.

Ventillərdən daha mürəkkəb sxemlər, məsələn *triggerlər* təşkil olunur. Digər sxemdən gələn tək impuls onun vəziyyətini əksinə dəyişənə qədər 0 və ya 1 qiymətini sabit olaraq çıxışa verən sxem trigger adlanır. Deməli, trigger kompüterdə bir bit informasiya saxlamaq imkanı verir.

Triggerin qurulması üçün müxtəlif variantlar mümkündür (şəkildə onlardan ikisi göstərilmişdir).





Strukturdakı fərqə baxmayaraq, bu triggerlərin hər ikisi tamamilə eyni xassələrə malikdir. Buradan görünür ki, məntiqi sxemin formal təsviri nə qədər əhəmiyyətlidir. Bu ona görə lazımdır ki, layihələşdiricinin ventillərdən ibarət sxemin qurulması üçün ən uyğun variantı seçmək imkanı olsun. Sadəcə, triggerlər deyil, digər əsas məntiqi sxemlər də (məsələn, bir girişi və girişə verilən siqnalın müəyyən yubanma ilə verildiyi bir çıxışı olan *yubanma elementi*) getdikcə daha mürəkkəb sxemlərin qurulmasında istifadə olunur. Beləliklə, hər-hansı qurğunun ümumi məntiqi sxeminin layihələşdirilməsi prosesi iyerarxik struktura malik olur və bu zaman hər növbəti səviyyədə "kərpiclər" kimi əvvəlki mərhələdə qurulmuş məntiqi sxemlər istifadə olunur.

Məntiq cəbri konstruktorlara məntiqi sxemlərin güclü bir layihələşdirmə, analiz və təkmilləşdirmə vasitəsini vermiş oldu. Həqiqətən də bir sxemi ifadə edən düstur vasitəsilə onun xüsusiyyətlərini öyrənmək və düzgün işləməsini isbat etmək real texniki qurğunun qurulmasından xeyli asan, tez və ucuz başa gəlir. İstənilən riyazi modelləşdirmənin məğzi məhz bundadır.

Məntiqi elementlərin yerinə yetirdiyi əməliyyatlar çoxluğu funksional tam olmalıdır (bazis anlayışını yada salaq). Bu halda əminliklə demək olar ki, istənilən məntiqi sxem realizə oluna bilər. Düsturların ekvivalent çevirmələri isə onları hiss olunacaq dərəcədə sadələşdirməyə imkan verir. Nəticədə minimal mümkün sayda elementdən məntiqi sxemlər qurmaq mümkün olur, bu isə öz növbəsində qurğunun yüksək işləmə sürəti və etibarlılığını təmin edir.

BAŞQA BİR MƏNTİQ DƏ MÜMKÜNDÜR

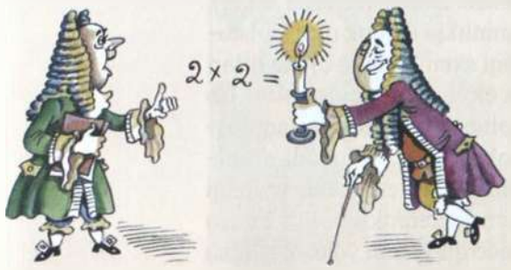
"İnsan – hə və yox sözlərini bilmirdi. O, qeyri-müəyyən cavab verirdi: Ola bilər, bəlkə də, biz fikirləşərik...". Bu yazını görkəmli yazıçı İlya İlfin ("On iki stul" və "Qızıl dana" romanlarının müəlliflərindən biri) məşhur "Qeyd kitabçaları" əsərinin səhifələrində tapırıq.

Doğrudan da, çox vaxt bizə bu iki söz, daha doğrusu iki məntiqi kəmiyyət bəs etmir. Biz tez-tez doğru və ya yalan olması haqqında fikir yürütmək çətin olan anlayışlar eşidirik. "Ola bilsin ki, mən imtahanda əla qiymət alacağam". Yaxud, məsələn, nəyi isə edib-etməmək haqqında qərar qəbul etmək tələb olunarkən, bu barədə lazımı miqdarda informasiyanın olmaması və ya bu informasiyanın dəqiqlik dərəcəsi haqqında məlumatımızın olmaması kimi hallar adi hallardır.

Alimlər çoxdan klassik Aristotel məntiqinin məhdudiyyətlərini aradan qaldırmağa çalışırdılar. Məsələn, rus məntiqçi alim N.A.Vasilyev 1910-cu ildə "xəyali məntiq" adlandırdığı orijinal sistem yaratmışdı. Vasilyevə əsasən hər bir hökm *təsdiqlənən, inkar edilən* və *aksidental* ola bilər. Əgər aksidental hökm doğrudursa, onda təsdiqlənən və inkar edilən hökmlər yalandırlar. Lakin eyni bir hökm eyni zamanda həm doğru, həm də yalan ola bilməz. Vasilyevin məntiqi çox da tanınmış deyildi və yalnız axırıncı illərdə alimlər yenidən onun ideyalarına müraciət etməyə başlamışlar.

Lakin *çoxqiymətli məntiqlər* çox geniş yayılmışdı. Onlarda dəyişən və funksiyaların doğruluq qiymətləri 0-dan k–1-ə qədər olan diapazonda dəyişir (onda 0-ı mütləq yalan, k–1-i isə mütləq doğru qiymət kimi qəbul etmək olar).

Yeni elmin əsasını 1920-ci ildə üçqiymətli məntiqi təklif etmiş polşalı məntiqçi və riyaziyyatçı alim Yan Lukaseviç (1878-1956) qoymuşdur. Lukaseviçin məntiqində qiymətlər *doğru, yalan* və *neytral* ola bilirdi. Bir il sonra amerikalı alim Emil Post (1897-1954) onun ümumiləşmiş modeli olan *k-qiymətli məntiqi* yaratmışdı. Daha sonra 1930-cu ildə Yan Lukaseviç və Alfred Tarskiy (1902-1983) *sonsuzqiymətli məntiqi* yaratdılar.



Çoxqiymətli məntiqlər üçün də Bul cəbrinə oxşar cəbrlər təyin etmək olar. k-qiymətli məntiq üçün inkar, konyunksiya və dizyunksiya əməliyyatlarını aşağıdakı şəkildə ifadə etmək olar:

$$\bar{x} = (k-1)-x,$$

$$x_1 \wedge x_2 = \min(x_1, x_2),$$

$$x_1 \vee x_2 = \max(x_1, x_2).$$

İkiqiymətli məntiq üçün, yəni $k=2$ halı üçün bu tərif artıq məlum Bul əməliyyatlarına gətirib çıxarır.

Məntiqdə linqvistik yanaşmadan da istifadə olunur. Bu zaman dəyişənlərin doğruluğu ədədi qiymətlərlə deyil, şifahi xarakteristikaların nizamlı yığını ilə verilir. Məsələn, "İnsan boyu" linqvistik dəyişənin qiymətlər yığınını (artma sırası ilə) aşağıdakı kimi nizamlamaq olar: liliput, çox qısa, qısa, ortadan qısa, orta, ortadan uzun, uzun, çox uzun, nəhəng.

Linqvistik dəyişənlər, birbaşa olaraq *qeyri-səlis çoxluq* və *qeyri-səlis nəticə* anlayışlarının bazasında qurulmuş *qeyri-səlis riyaziyyata* (*bulanıq riyaziyyata*) aiddir. Onun köməyilə bir çox vacib praktik məsələlər həll olunur. Məsələn, müxtəlif analitik və diaqnostik məqsədlər üçün texniki, tibbi və digər *ekspert sistemlərin* hazırlandığı süni intellekt sahəsində qərar qəbuletmə mexanizminin əsasında qeyri-səlis nəticə anlayışı durur. Qeyri-səlis riyaziyyatın əsasını qoyan, bu nəzəriyyəni XX əsrin 60-cı illərindən inkişaf etdirən amerikalı alim Lütfi Zadə (Lütfi Ələsgərzadə) olmuşdur.

Bundan əlavə, tamamilə məxsusi qadın məntiqi də mövcüddür! Böyük rus yazıçısı İvan Serqeyeviç Turqenev bu məntiqi Piqasovun ("Rudin" romanının personajı) dililə aşağıdakı kimi xarakterizə etmişdi: "...məsələn, kişi deyə bilər ki, iki vur iki dörd eləmir, beş, yaxud üç yarım edir, qadın isə deyəcək ki, iki vur iki stearin şamına bərabərdir".

Burada artıq riyaziyyat da, informatika da öz gücsüzlüyünü boynuna alır...

ALQORİTMLƏR NƏZƏRİYYƏSİ

ALQORİTM ANLAYIŞININ FORMALLAŞDIRILMASI

Alqoritmlər nəzəriyyəsi alqoritmlərin konkret modellərini qurur və öyrənir. Bu modellər çevrilən obyektlərin və elementar addımların toplusu, növbəti addımın seçilməsi üsulları və s. ilə seçilir.

Alqoritmlərin nəzəri modellərinin üç əsas qurulma xəttini ayırmaq olar.

Birinci xətt ixtiyari verilənlərin ədədlərlə kodlaşdırılmasının mümkünlüyünün aşkar faktı ilə bağlıdır. Verilənlərin hər bir çevrilməsi hesabi (arifmetik) hesablamalara gətirib çıxarır, hər bir alqoritm isə müəyyən ədədi funksiyanın qiymətini tapır. Alqoritmin elementar addımları isə yerinə yetirilən hesab əməlləri olur. Bu cür modellərdə elementar addımların ardıcıllığı ya *funksiyaların superpozisiyası*, ya da *rekursiyanın* köməyilə təyin edilir.

İkinci xətt, ixtiyari alqoritmin, sonlu əlifbanın simvollarından qurulmuş giriş sözlərinin çevrilməsi ardıcıllığı kimi ifadə edilə bilməsi fərziyyəsilə bağlıdır. Beləliklə, A.A.Markov (1903–1979) tərəfindən təklif olunmuş və öyrənilmiş *normal alqoritmlər* anlayışı yarandı.

Üçüncü xətt, ikinci xəttə kifayət qədər yaxındır. Lakin ondan fərqli olaraq burada abstrakt əvəzetmələrdən deyil, hansısa konkret mexanizmlərdən istifadə olunur. Bu zaman fərz olunur: alqoritmin hər bir addımı elə təyin olunur ki, onu kifayət qədər sadə qurğu (maşın) yerinə yetirə bilər. Arzu olunardı ki o, *universal* olsun, yəni onda ixtiyari alqoritmi yerinə yetirmək mümkün olsun.

Deməli, maşının iş mexanizmini məntiqi struktur etibarı ilə maksimal sadə, lakin riyazi araşdırmalar predmeti ola biləcək dərəcədə dəqiq standart sxem formasında ifadə etmək lazımdır. İlk

Superpozisiya – bir funksiyanın başqa funksiyaya daxil edilməsidir. O, riyazi düsturlar əmələ gətirir. Məsələn, əgər $xy+z$ funksiyasında y əvəzinə $t-u$ funksiyasını, z əvəzinə isə vw funksiyasını qoysaq, onda $x(t-u)+vw$ düsturunu alarıq.

MARKOVUN NORMAL ALQORİTMLƏRİ



A.A.Markov.

Normal alqoritmlər üçün sonlu əlifba və yerdəyişmələrin (sözün bir hissəsinin digərlərilə əvəz edilməsi qaydaları) sonlu çoxluğu verilir. Onun işi müəyyən qaydada iki operatorun: *tanıdıcı operatorların* və *əvəzetmə operatorlarının* yerinə yetirilməsindən ibarətdir. *Tanıdıcı* operator əvəzetmənin sol tərəfinin sözə daxil olmasını tapır, əvəzetmə operatoru isə bu daxil olmanı uyğun sağ tərəflə əvəz edir.

Tutaq ki, $A=\{x, y\}$ əlifbası və $yyx \to y$; $xx \to y$; $yyy \to x$ (bu sonuncu əvəzetmədir, o yerinə yetirildikdən sonra proses dayandırılır) əvəzetmələr çoxluğu verilmişdir!

Giriş sözü $p=xyxxxyy$.

Əgər *tanıdıcı* operatorun köməyilə müəyyən olunubsa ki, birinci əvəzetmə sözə uyğun gəlmir, onda ikinci əvəzetmənin analizinə keçmək lazımdır. *Tanıdıcı* operator, onun sol hissəsinin (xx) p-yə birinci daxil olmasını ayırd edir: $p=xy(xx) \to xyy$. Əvəzetmə operatoru əvəzetməni yerinə yetirir, nəticədə yeni $p_1=xyyxyy$ sözü əmələ gəlir. Bu sözə birinci əvəzetmə tətbiq oluna bilər: $p_1=x(yyx)yy=xyyy=p_2$. Alınmış sözə yalnız üçüncü əvəzetmə tətbiq oluna bilər: $p_2=x(yyy) \to xx=p_3$. Bu axırıncı əvəzetmə idi, buna görə də p_3 sözü alqoritmin işinin son nəticəsidir.

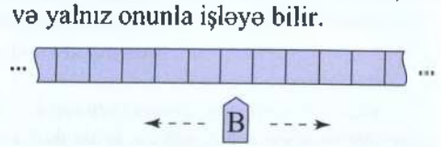


Funksiyanın növbəti qiymətinin onun əvvəl alınmış qiymətlərindən istifadə etməklə hesablanmasına rekursiya deyilir. Funksiyanın rekursiv verilməsinə misal olaraq faktorialın təyin edilməsini göstərmək olar: $f(0)=1$; $f(n+1)=f(n)(n+1)$. Burada funksiyanın qiyməti arqumentin başlanğıc qiyməti üçün verilir, hər bir sonrakı qiymət isə əvvəlkinin köməyilə təyin edilir.

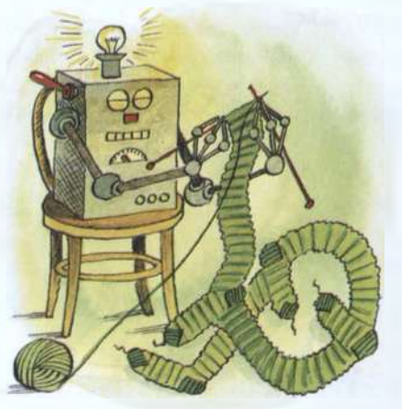
dəfə bu amerikalı alim Emil Post tərəfindən 1936-ci ildə müasir hesablama maşınları yaradılana qədər, ingilis riyaziyyatçısı Alan Türinq ilə praktik olaraq eyni zamanda yerinə yetirilmişdir.

POST MAŞINI

Post maşını lentdən və başcıqdan ibarətdir. Lent damalara bölünüb və *şərti sonsuz* hesab edilir (bu o deməkdir ki, ehtiyac olduqda damaların sayı artırıla bilər). Hər bir damada qeyd olunmuş *əlifbadan* bir *simvol* yazıla bilər. İxtiyari konkret anda başcıq bir damanı nəzərdən keçirir və yalnız onunla işləyə bilir.



Post maşını *proqramların* idarəsilə işləyir. Proqramlar komandanın nömrəsi, əməliyyat və göndəriş yeri yazılan üç sahədən ibarət komandalardan təşkil olunub. Komandalar 1-dən başlayaraq nömrələnir və proqramın yerinə yetirilməsi 1 nömrəli komandadan başlanır. Göndəriş yeri, verilmiş komandadan sonra hansı komandanın yerinə yetiriləcəyini göstərir. Post maşını üçün aşağıdakı beş əməliyyat tipi müəyyən olunub:



• Başcığın bir dama sağa hərəkəti;

• Başcığın bir dama sola hərəkəti;

• S simvolunun nəzərdən keçirilən damaya yazılışı, bu zaman damada olan əvvəlki verilənlər silinir;

• STOP – Post maşınının işinin sona çatdırılması, bu zaman göndəriş yeri üçün sahəyə ehtiyac qalmır;

• Aşağıdakı kimi yazılan budaqlanma komandası: $?\begin{cases} S_1, & M_1 \\ S_2, & M_2 \\ \dots & \\ S_n, & M_n \end{cases}$

Burada $S_1, S_2, \dots$ – Post maşınının əlifbasından olan hər hansı simvollar, $M_1, M_2, \dots$ – isə göndəriş yerləridir. Bu komanda əsasında nəzərdən keçirilən damadakı simvol analiz olunur. Əgər o, S_i ilə üst-üstə düşürsə, onda M_i nömrəli komanda yerinə yetiriləcək. Bu zaman başcıq heç bir yerə hərəkət etmir və lentə heç bir şey yazılmır. Əgər damada budaqlanma komandasında göstərilməmiş simvol dursa, onda Post maşınının nəticəsiz dayanması baş verir (başqa sözlə desək, maşın sınır).

Proqramın Post maşınında yerinə yetirilməsinin digər mümkün nəticəsi sonsuz iş və STOP komandasının yerinə yetirilməsilə baş verən nəticəli dayanmadır.

Bir misala baxaq. Tutaq ki, Post maşınının əlifbası {–, 0, 1} simvollarından ibarətdir (burada "–" simvolu ilə boşluğu işarə edirik). Lentdə isə 1 və 0 simvollarının ardıcıllığı yazılıb. Başcıq bu ardıcıllığın ən soldakı elementinə "baxır". Bütün 1 simvollarını 0-la, 0 simvollarını isə 1-lə əvəz etmək tələb olunur. Proqram (% ... % simvolları arasında yerləşdirilən şərhlərlə birlikdə) aşağıdakı kimi olacaq:


```
1. ?  1 2 %   əgər 1 simvoludursa, onda 2 komandasını yerinə yetirməli %;
      0 3 %   əgər 0 simvoludursa, onda 3 komandasını yerinə yetirməli %;
     – 5 %   əgər "–" simvoludursa, onda 5 komandasını yerinə yetirməli %;
2. 0 4 %     nəzərdən keçirilən damaya 0 yazmalı, 4 əmrini yerinə yetirməli %;
3. 1 4 %     nəzərdən keçirilən damaya 1 yazmalı, 4 komandasını yerinə yetirməli %;
4. →1 %      başcığı sağa hərəkət etdirməli, 1 komandasını yerinə yetirmək %;
5. STOP %    "–" simvoluna rast gəldikdə maşını dayandırmalı %.
```

Misalda ilkin verilənlər, maşının giriş əlifbasından götürülən boş olmayan simvolların sonlu ardıcıllığı şəklində yazılır. Belə ardıcıllıqlar bu əlifbada *sözlər* adlanır. Belə bir *Post təklifi* məlumdur: "İlkin verilənləri və nəticələri sözlər olan ixtiyari alqoritm üçün, ona ekvivalent olan Post maşını üçün proqram mövcüddür". Buna görə də sözlərlə işləyən alqoritmləri çox vaxt *Post alqoritmləri* adlandırırlar.

TÜRİNQ MAŞINI

Türinq maşını aşağıdakı üç hissədən ibarətdir:

• xanalara bölünmüş və hər iki tərəfdən məhdudiyyəti olmayan lent;

• idarəetmə qurğusu (İQ);

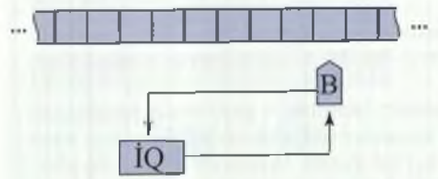
• başcıq (B).

Hər bir Türinq maşını ilə iki sonlu əlifba əlaqədardır. Onlardan birini $S=\{s_0, s_1, s_2, \dots, s_k\}$ xarici simvollar əlifbası, digərini isə $Q=\{q_0, q_1, q_2, \dots, q_n\}$ daxili vəziyyət əlifbası adlandırırlar. Burada xüsusi olaraq başlanğıc q_0 və son q_n vəziyyyətlərini qeyd edirlər. Zamanın hər anında lentin hər bir xanası S əlifbasından yalnız bir hərfi özündə saxlaya bilir. Xanada yazının olmaması s_0 hərfinin yazılışı kimi qəbul olunur. İdarəetmə qurğusu $q_i \in Q$ vəziyyətlərindən birində ola bilər. Başcıq xanalardan birinə "baxır".



Alan Türinq.

Hər bir anda Türinq maşını idarəedici qurğunun vəziyyəti haqqındakı məlumatı, lentdəki yazılımı və nəzərdən keçirilən xanaya göstərişləri əhatə edən öz *konfiqurasiyası* ilə təsvir olunur. Türinq maşınının işi taktlardan ibarətdir; $(i+1)$-ci takt zamanı maşının konfiqurasiyası cari K_i-dən yeni K_{i+1}-ə çevrilir. Çevrilmə yalnız İQ-nun vəziyyətindən və nəzərdən keçirilən xananın tərkibindən asılıdır. O, aşağıdakı əməliyyatlardan ibarətdir:



• q_i vəziyyətini q_j vəziyyətinə dəyişmək;

• nəzərdən keçirilən xanada yazılmış s_i hərfini s_p hərfilə əvəz etmək;

• başcığın vəziyyətini dəyişmək – onu bir xana sola və ya sağa hərəkət etdirmək (və ya heç yerindən tərpətməmək).

Bu cür çevirməni Türinq maşınının *komandası* adlandırırlar. Komandalar $q_i s_l \to q_j s_p R$ şəklində yazılır. R burada L, C və ya G hərflərindən biri olacaq. L – soldakı qonşu xananı nəzərdən keçirmək; C – cari xananın nəzərdən keçirilməsini davam etdirmək; G – sağdakı qonşu xananı nəzərdən keçirmək deməkdir. Türinq maşınının yerinə yetirdiyi bütün komandalar külliyyatı *proqram* adlanır.

Hər bir $s_i \in S$ hərfi və $q_i \in Q$ vəziyyəti üçün proqram $q_i s_i$ sol tərəfli cəmi bir komandaya malikdir. Buna görə də maşının işi başladığı K_1 konfiqurasiyasını qeyd etsək, onda Türinq maşınının işi

ALQORİTMİK HƏLL OLUNA BİLMƏYƏN PROBLEMLƏR

Məsələni o zaman alqoritmik həll oluna bilməyən adlandırırlar ki, onu həll edə bilən Türinq maşını (və ya Markovun normal alqoritmi və yaxud Post maşını) mövcud olmasın.

Aşağıda alqoritmlər nəzəriyyəsindən bir neçə misal göstərilmişdir.

Fərz edək ki, A alqoritminin təsvirinə və x arqumentinə görə müəyyən etmək lazımdır ki, x alqoritmin girişinə verilərsə, A alqoritmi dayanacaqmı? Bu məsələ alqoritmik həll oluna bilməyən məsələdir (onu *dayanma problemi* adlandırırlar). Xüsusi halda belə bir teorem isbat olunmuşdur ki, istənilən T Türinq maşını üçün dayanma probleminin həll edən T_0 Türinq maşınını qurmaq olmaz.

Bu faktın vacib praktik nəticəsi odur ki, sazlama proqramları üçün universal (istənilən proqram üçün yararlı olan) alqoritm yaratmaq mümkün deyil. Doğrudan da, proqramçının səhvinin ən çox rast gəlinən nəticəsi o olur ki, proqram dövrə düşür, bu zaman proqramın işi yekunlaşa bilmir və o, nəzəri olaraq "sonsuz çox" işləyə bilər. Dayanma məsələsinin alqoritmik həll oluna bilməməsi o deməkdir ki, ixtiyari alqoritmin nəticə verə biləcək qədər işləməsini təyin etmək mümkün deyil.

Alqoritmlər nəzəriyyəsində məlum müddəa – *Rays teoremi*: "Hesablanabilən funksiyaların heç bir trivial olmayan xassəsi alqoritmik həll olunan deyil".

Bu teoremin mənası belədir: İstənilən A alqoritmilə hesablanan funksiyanın bu və ya digər xassəyə malik olub olmadığını təyin edən alqoritm (yəni Türinq maşını, Post maşını, Markovun normal alqoritmi və s.) qurmaq mümkün deyil. Rays teoreminə görə elə bir alqoritm qurmaq olmaz ki, o, funksiyanın sabit, periodik, yaxud məhdud olduğunu, onun qiymətləri arasında verilmiş ədədin varlığını, bu funksiyanın digər bir funksiyaya bərabərliyini və s. müəyyən edə bilsin. Hətta düşünmək olar ki, ümumiyyətlə, dünyada hər şey həll edilə bilən deyil!

Amma təsadüfi deyil ki, biz funksiya anlayışı ilə onu reallaşdıran alqoritm anlayışını fərqləndirdik, axı Rays teoremində onların hər ikisi var. Onda Rays teoremini belə başa düşmək olar ki, alqoritmin təsvirindən onun hesabladığı funksiyanın xassələri haqda nə isə öyrənmək olmaz.

Bu və ya digər məsələnin alqoritmik həll oluna bilməməsi uğursuzluq kimi qiymətləndirilməməlidir. Əksinə, bu istənilən təbiət elmlərində olduğu kimi, yeni bir elmi nəticədir. Məsələlərin təbiətini bilmək həddən ziyadə qlobal məsələ qoyuluşlarından, xəyalpərəstlikdən və gerçəkləşdirilə bilməyən layihələrə uymaqdan yaxa qurtarmağa imkan verir. Hətta məsələ alqoritmik həll oluna bilməyəndirsə belə, onun xüsusi hallarının həlləri ola bilər. Beləliklə, istənilən proqramın ümumi sazlama alqoritminin mövcud olmamasını qəbul edirik. Bununla belə, axı hər bir proqramçı konkret proqramı dəfələrlə sazlayır!

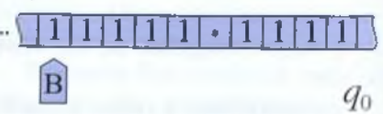


birqiymətli təyin edilir. Birinci taktda K_1 konfiqurasiyası (onda yeganə komanda tətbiq edilə bilər) K_2 konfiqurasiyasına çevrilir. Analoji olaraq ikinci taktda K_2 yeganə üsulla K_3 konfiqurasiyasına çevrilir və s.

Türinq maşınının işi ixtiyari konfiqurasiyadan başlayaraq məhdudiyyət olmadan uzun müddət davam edə bilər, buna görə də prosesin sona çatdırılması qaydalarını müəyyən etmək tələb olunur. Əgər hər hansı bir taktda və uyğun olaraq ondan sonrakı bütün taktlarda konfiqurasiya dəyişmirsə, onda maşının işinin bu taktda qurtardığını hesab etmək olar. Bu cür konfiqurasiyanı – *sonuncu* adlandırırlar.

Proqramı, nəzərdən keçirilən xananın yeni tərkibi, başcığın sonrakı hərəkəti və yeni vəziyyət arasındakı əlaqəni göstərən cədvəl şəklində vermək olar.

Tutaq ki, Türinq maşınının xarici simvollarının əlifbası iki simvoldan {*, 1} ibarətdir. Bundan başqa, tutaq ki, "0" simvolu boş damaya uyğun gəlir. Lentə "*" simvolu ilə ayrılmış iki söz (hər biri yalnız vahidlərdən ibarət olan) verilir. Bu ayırıcı işarəni ləğv edib, sözləri bir sözdə birləşdirmək lazımdır. Deməli, Türinq maşınının başlanğıc konfiqurasiyasını belə təsvir etmək olar:



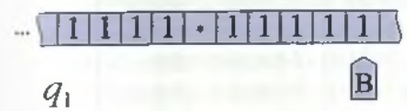
Lentin son vəziyyəti isə belə olmalıdır:



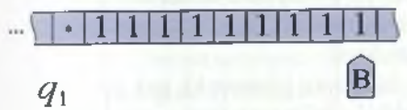
Müvafiq proqram aşağıdakı şəkildədir:

	q_0	q_1	q_2
1	0 L q_2	L	G
0	G	G q_1	1 q_1
*	0!	L	G

İşin əvvəlində başcıq sol qıraqdakı 1 ədədini görür, maşın isə q_0 vəziyyətində olur. Gözdən keçirilən xanaya "0" yazılır, başcıq bir xana sağa sürüşdürülür və q_2 vəziyyətinə keçid baş verir. Bu vəziyyətdə başcıq ardıcıl olaraq sağa, "•" və 1 simvolunu buraxaraq sözün sonuna qədər sürüşür. "0" simvolunu aşkar edən başlıq onun yerinə 1 yazır, maşın isə q_1 vəziyyətinə keçir:



Başcıq yenidən birinci 1 simvoluna qayıdana qədər, maşın bu vəziyyətdə qalır (o sola sürüşür, "boş" xanaya çatıb bir xana sağa sürüşür). Bundan sonra konfiqurasiya q_0 vəziyyətinə gəlir. Birinci 1 ədədinin əvəzinə 0 yazılır və maşın q_2 vəziyyətinə keçir. Bundan sonra əvvəlki dövr dörd dəfə təkrar olunacaq və nəhayət maşın

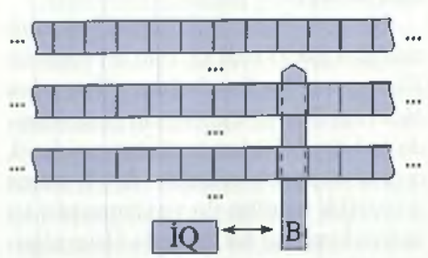


şəklini alacaq.

1 və "*" simvolları q_1 vəziyyətində başcığı sola elə sürüşdürürlər ki, birinci "0" simvoluna çatdıqda maşın G q_0 komandasını icra edir və onun vəziyyəti q_0 olur. Ancaq bu vəziyyətdə "*" simvolunun əvəzinə "0" simvolu yazılır və maşın dayanır.

Türinq maşını ilə hesablama maşınları arasında olan müəyyən oxşarlığı görmək asandır, çünki onların hər ikisi məsələnin ilkin verilənləri üzərində onun həlli olan müəyyən proqram ilə işləyir. Hətta demək olar ki, müəyyən mənada Türinq maşını kompüterin ideal modelidir. Fəqət o, ilk, müasirlik baxımından həddən ziyadə primitiv hesablama maşınları ilə müqayisədə də çox sadədir. Aydındır ki, Türinq maşını geniş praktik tətbiq üçün nəzərdə tutulmamışdır. İriliyinə görə o, hətta sırf nəzəri mühakimələr üçün də bir o qədər də rahat deyil.

Buna görə də, onun müxtəlif təkmilləşdirilmiş nümunələri yaradılmışdır ki, bunlardan biri də *Türinqin çoxlentli maşınıdır*. Onun bir neçə giriş lentləri və bir çıxış lenti var ki, bura da nəticə simvolları yazılır. Çıxış lentinin tərkibini nə oxumaq, nə də ona düzəliş etmək olar. Bu maşında rahatlıq üçün fərz etmək olar ki, başcıq hərəkətsizdir, lentlər isə hərəkət edir, özü də, ola bilər ki, müxtəlif istiqamətlərə.



Bu maşının işləmə taktı aşağıdakı işlərdən ibarətdir: bütün lentlərindən (çıxış lentindən başqa) eyni zamanda cari simvolun oxunması, oxunmuş simvollar yığımına asılı olaraq maşının vəziyyətinin dəyişməsi, yeni simvolların yazılması, lentin sürüşdürülməsi.

HESABLANABİLƏN VƏ HESABLANABİLMƏYƏN FUNKSİYALAR

Təbii sual meydana çıxır: prinsipcə alqoritmin ümumi xassələrindən danışmaq mümkündürmü və belə konkretləşmədə bu ümumilik itmirmi?

Sən demə, alqoritmlər nəzəriyyəsində isbat olunmuşdur ki, bir modeldə təsvir olunmuş alqoritm başqa modeldə də təsvir oluna bilər. Alqoritmik modellərin bir-birinə bu cür qarşılıqlı gətirilə bilməsi konkret formallaşdırma üsulunun





Alonzo Çerç.



Emil Post.

seçilməsindən asılı olmayan anlayışlar sistemi qurmağa imkan vermişdir. Və bunun əsasında *hesablanabilən* funksiya anlayışı durur, yəni elə bir funksiya ki, onun hesablanması üçün alqoritm mövcuddur.

Alqoritmlərin köməyilə hansı funksiyalar hesablana bilər? Riyaziyyat tarixində ilk dəfə bu suala cavabı Alonzo Çerç (1903-1995) vermişdir. 1930-cu ildə o, belə bir müddəa irəli sürmüşdür: "Müəyyən bir alqoritm ilə hesablanabilən istənilən funksiya rekursiv funksiyadır". İndi bu müddəa *Çerç tezisi* kimi tanınır.

Bir neçə il sonra A.Türinq başqa bir müddəa təklif etdi ki, onu da sonralar *Türinq tezisi* adlandırdılar: "İstənilən alqoritm Türinq maşınında gerçəkləşdirilə bilər". Türinq tezisi teorem deyil, onu isbat etmək mümkün deyil, amma o çoxillik təcrübə ilə yoxlanmışdır və indiyə kimi heç bir dənə də olsun alqoritm nümunəsi tapılmayıb ki, onu Türinq maşınında gerçəkləşdirmək mümkün olmasın.

Bu iki tezisin müqayisəsi göstərir ki, onları belə ümumiləşdirilmiş formada yazmaq olar: "Funksiya yalnız və yalnız o zaman Türinq maşınında hesablana bilər ki, o rekursiv olsun" (*Türinq-Çerç tezisi*). Beləliklə, Türinq-Çerç tezisi alqoritmin verilmiş hər iki modelinin ekvivalent olmasını göstərir. Bununla belə bu ciddi isbat edilə bilən bir mühakimədir. Alqoritmin tərifinin digərinə gətirilməsi teoremləri də isbat olunur. Məsələn, Türinq maşınlarının Markovun normal alqoritmlərinə gətirilməsi, Sözlər üzərində alqoritmlər və Post maşınlarının ekvivalentliyi haqqında Post təklifi isbat edilmişdir. Başqa sözlə, doğrudan da, istifadə olunan modeli konkretləşdirmədən ən ümumi mənada alqoritmlər haqqında danışmaq olar.

Faktiki olaraq alqoritm – funksiyanın verilməsi üsuludur. Amma funksiya müxtəif üsullarla verilə biər, məsələn, cədvəl, yaxud düstur ilə. Buna baxmayaraq elə asılılıqlar var ki, onu nə bu, nə də digər üsulla təsvir etmək mümkün deyil. Məsələn, məlum

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$$

düsturu yalnız bu onun qiyməti bizi qane edəcək dəqiqliklə hesablamağa imkan verir, amma e^x funksiyasının dəqiq təsviri deyil.

Bu misal göstərir ki, çox güman, *hesablana bilməyən* funksiyalar da mövcuddur. Elə məsələlər də var ki, giriş və çıxış kəmiyyətləri arasındakı əlaqə mürəkkəb olduğundan, ilkin verilənlərin nəticəyə çevrilməsi ciddi, addımba-addım təyin olunmuş proses kimi verilə bilmir. Başqa sözlə, bu məsələlərin həll alqoritmini qurmaq olmur. Belə məsələlər *alqoritmik həll oluna bilməyən* məsələlər adlanır.

ALQORİTMLƏRİN MÜRƏKKƏBLİYİ

MÜRƏKKƏBLİK ANLAYIŞI

Bu və ya digər bilik sahəsində çalışan insanın təcrübəsi artdıqca intuitiv hiss etməyə başlayır ki, bir məsələ digər məsələdən daha çətindir. Aydındır ki, "asandır", "çətindir" kimi subyektiv, ciddi kəmiyyət göstəricisi olmayan anlayışlar yetərli deyil. Bəs məsələnin həqiqi mürəkkəbliyini necə təyin etmək olar (və olarmı?)? Bunun üçün müxtəlif yanaşmalar mövcuddur.

Tez-tez məsələni ona görə mürəkkəb adlandırırlar ki, o şaxələnmiş məntiqi struktura malik olur, çoxsaylı şərtlərin yoxlanmasından, keçidlərdən və s. ibarət olur. Proqram, onun həllini reallaşdıran alqoritm bizə ona görə mürəkkəb görünür ki, onun nə etdiyini heç cür anlamaq olmur. Kompüter üçün isə belə proqramın icrası heç bir çətinlik törətmir! Ona görə ki, o, sadəcə olaraq, bir-birinin ardınca proqramda yazılmış komandaları icra edir. Kompüter üçün bu komandaların necə olmasının heç bir fərqi yoxdur – toplama, yaxud şərtin yoxlanması, ya da ki, keçid olsun.

Çoxsaylı operatorlardan ibarət olan proqram mətni daha mürəkkəb görünə bilər. Fəqət mürəkkəbliyə bu cür baxış da özünü o qədər də doğrultmur. Doğrudan da, kompüter üçün iki ədədin toplanmasını 100 dəfə təkrar edən bir dövr operatorlu proqram ilə 100 ardıcıl yazılmış toplama opetatorlarından ibarət proqram praktiki olaraq eyni şeydir.

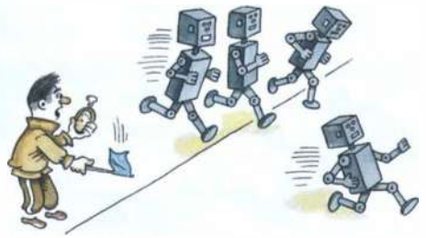
Alqoritmlər nəzəriyyəsinin qurulmağa başladığı dövrdə insanın hesablama imkanları çox cüzi idi. Buna görə də uzun müddət praktikada alqoritmlərin işlənib hazırlanması mümkün hesab edilmirdi. Yalnız ilk kompüterlərin meydana gəlməsilə alimlər alqoritmlərin yaradılması məsələlərinə xüsusi diqqət yetirməyə başladılar. Onların praktiki istifadə edilməsilə məlum oldu ki, daha bir xassə – *effektivlik* xassəsi böyük əhəmiyyətə malikdir. Yalnız alqoritm deyil, həm də onun hər bir addımı elə olmalıdır ki, icraçı onu müəyyən ağlabatan hüdudlarla məhdudlaşan sonlu vaxtda icra edə bilsin. Əgər söhbət hesablama maşınından gedirsə, onda həmçinin konkret yaddaş həcmilə şərtlənmiş olan hüdudlar çərçivəsində icra başa düşülür. Alqoritmin effektivliyilə intuitiv olaraq *dəyər* anlayışı bağlıdır ki, onu da alqoritmi reallaşdırmaq üçün tələb olunan resursların kəmiyyət ifadəsi kimi başa



düşmək olar. Dəyər anlayışına daha dəqiq mənanı *alqoritmlərin mürəkkəbliyi* anlayışı verir. Bu anlayış dəyəri zaman və fəzada maddiləşdirir, icra müddətini və tələb olunan yaddaşı qiymətləndirməyə imkan verir.

Alqoritmin zamana görə mürəkkəbliyi – onun icrası üçün zəruri olan T zamanıdır. O, alqoritmin elementar addımlarının sayı (k) ilə bir addımın orta icra müddətinin (t) hasilinə bərabərdir, $T = kt$. t parametrinin alqoritmi reallaşdıran icraçıdan asılı olduğunu nəzərə alsaq, təbii olaraq fərz edə bilərik ki, alqoritmin mürəkkəbliyi ilk növbədə məhz k parametrindən asılıdır. *Alqoritmin fəza mürəkkəbliyi* onu reallaşdıran alqoritmə tələb olunan yaddaş vahidlərinin miqdarı ilə ifadə olunur. Fəza mürəkkəbliyi zaman mürəkkəbliyindən daha sürətlə arta bilməz, çünki hər bir əlavə yaddaş vahidinin istifadəsi üçün əlavə də zaman tələb olunacaq. Buna görə də sadəlik üçün, adətən, zaman mürəkkəbliyi anlayışı ilə kifayətlənirlər.





Praktikada alqoritmin mürəkkəbliyini adətən onun əsas parametri – icra edilən əməliyyatların miqdarına ən çox təsir edən parametr ilə əlaqələndirirlər. Bir qayda olaraq, bu emal edilən verilənlər massivinin ölçüsüdür (məsələnin ölçüsü). Məsələn, iki kvadrat matrisin vurulması məsələsində bu onların *sətirlərinin sayı*, mətndəki sözlərin əlifba sırası ilə nizamlanması məsələsində isə bu göstərici *sözlərin sayıdır* və s. Alqoritmin mürəkkəbliyini (onu reallaşdırmaq üçün zəruri olan əməliyyatların sayını) onun giriş verilənlərinin həcminin funksiyası şəklində ifadə etmək olar.

Məsələn, yalnız oxuma və n sayda verilənin yaddaşa daxil edilməsi əməllərindən ibarət olan alqoritmin icra vaxtı aşağıdakı düstur ilə təyin olunur: $T = an + b$, burada a – yazının oxunma vaxtı, b – köməkçi əməliyyatların yekun icra vaxtıdır. a və b əmsalları naməlumdur, ancaq onlar kompüterin, translyatorun və s. xarakteristikalarından asılıdır. Amma ən vacib olan əsas parametrdən asılılığın xarakteridir, çünki məhz o, mürəkkəbliyi müəyyən edir. Burada n parametrindən xətti asılılıq mövcuddur və buna görə də alqoritmin mürəkkəbliyi *xətti* adlanır. O, $O(n)$ kimi yazılır və "O böyük n" kimi oxunur.

Massivin n elementinin artma sırasına görə nizamlanması prosesinə birbaşa seçim adlanan üsul ilə baxmaq olar. O, belə icra olunur: massivin ən kiçik elementi müəyyən edilir və bu ədəd ilə massivin birinci elementinin yerləri dəyişdirilir; sonra massivin yerdə qalan hissəsində olan ən kiçik element tapılır və onunla massivin ikinci elementinin yerləri dəyişilir və s. Birinci dəfə minimal element axtarılarkən $(n-1)$ müqayisə aparılır, sonra $(n-2)$ müqayisə və s. Cəmi $(n-1) + (n-2) + ... + 1 = (n^2 - n)/2$ müqayisə etmək lazımdır. Böyük qiymətli n ədədi üçün hesab etmək olar ki, mürəkkəblik yalnız polinomun yüksək dərəcəli həddilə müəyyən olunur. Ona görə də, onu $O(n^2)$ kimi yazırlar. Deyirlər ki, bu alqoritm *kvadratik mürəkkəbliyə* malikdir.

nxn ölçülü matrislərin (cədvəllərin) vurulması alqoritminin mürəkkəbliyi isə artıq kubik, yəni $O(n^3)$ olacaq. Axı nəticə matrisinin hər bir elementinin hesablanması üçün n sayda vurma və $(n-1)$ sayda toplama əməli tələb olunur, belə elementlərin sayı isə n^2-dır.

Praktikada alqoritmin icra vaxtı onun tətbiq olunduğu verilənlər yığınından asılı ola bilər. Məsələn, əgər bu verilənlər əvvəlcə qismən nizamlanıblarsa, onda bəzi nizamlama alqoritmləri üçün bu vaxt xeyli ixtisar olunur. Alqoritmi verilənlərdən asılı olmayaraq analiz etmək imkanını saxlayaraq, bu məqamı nəzərə almaq üçün aşağıdakıları fərqləndirirlər:

– *maksimal mürəkkəblik* $T_{max}(n)$ – seçilmiş n verilənlər yığınının alqoritmin icra vaxtını çox artırdığı hal;

– *orta mürəkkəblik* $T_{orta}(n)$ – n ixtiyari verilənlər yığınına tətbiq olunmuş alqoritmin orta icra vaxtı.

Məsələnin mürəkkəbliyi – məsələnin həlli üçün məlum olan ən yaxşı alqoritmin mürəkkəbliyidir ki, o da həll metodlarının inkişaf səviyyəsindən asılıdır. Məsələn, qrafların planarlığının (yəni qrafı müstəvi üzərində elə təsvir etmək olarmı ki, onun tilləri kəsişməsinlər) yoxlanması məsələsi məlumdur. Bu məsələnin 1930-cu ildə təklif olunmuş birinci alqoritminin mürəkkəbliyi $O(n^6)$ idi. Sonra onu dəfələrlə yaxşılaşdırmışlar, 1974-cü ildə isə mürəkkəbliyi $O(n)$ olan alqoritm qurmaq mümkün oldu! Məsələnin həlli üçün müxtəlif mürəkkəbliyə malik olan alqoritmlər qurmaq olar. Məntiqi olardı ki, bunlar arasından ən az mürəkkəbliyə malik olanını seçəydik, ancaq bilmək istəyirik ki, onu daha da yaxşılaşdırmaq olarmı? Bunun üçün məsələləri (və alqoritmləri) qruplara – *mürəkkəblik siniflərinə* bölmək lazımdır.

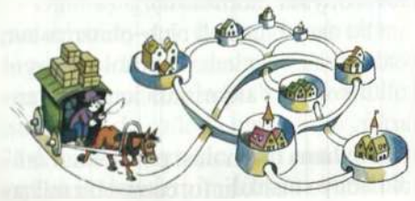
Aydındır ki, ən yaxşı alqoritmlər mürəkkəblikləri $O(n)$ olan xətti alqoritmlərdir, burada n – giriş verilənlərinin ölçüsüdür. Belə alqoritmlər doğrudan da mövcuddur. Məsələn, biri m, digəri n rəqəmdən ibarət olan ədədlərin sütunlu (alt-alta) toplanması əməliyyatı $max(n, m)$-dən çox olmayan toplama və $max(n, m)$-dən çox olmayan yadda saxlama tələb edir, yəni bu alqoritm $O(n+m)$ mürəkkəbliyə malikdir.

Bu ifadə yalnız kəmiyyətin tərtibini göstərir – daimi faktorlar burada nəzərə alınmır. Təəssüf ki, xətti mürəkkəbliyə malik alqoritmlər çox azdır (ələlxüsus da mürəkkəbliyi giriş verilənlərinin ölçüsündən asılı olmayan).

NP-MÜRƏKKƏB MƏSƏLƏLƏR

Kommivoyajör məsələsi adlanan məlum məsələyə baxaq. Kommivoyajör ticarət agenti, arşınmalçı mənalarını verir. Kommivoyajör evindən çıxır, o, özünün başqa-başqa məntəqələrdə yaşayan bütün müştərilərinə baş çəkməlidir. Məntəqələr arasındakı məsafələr məlumdur. Kommivoyajör elə marşrut seçməlidir ki, hər bir müştəriyə yalnız bir dəfə baş çəksin və keçdiyi yolun ümumi uzunluğu ən az olsun (və ya verilmiş məsafədən artıq olmayan bir məsafə keçsin). Bu məhdudiyyəti anlamaq asandır. Doğrudan da, kommivoyajör benzinə çəkilən məsrəfləri mümkün qədər azaltmaq istəyir (və yaxud buna verilmiş məbləğdən artıq xərcləmək istəmir).

Həlli aşağıdakı kimi axtarmaq olar. Növbə ilə bütün ehtimal olunan marşrutlara baxırıq və onlar arasında elәsini



Fərz edək ki, A alqoritmi mövcuddur. Demək olar ki, həmişə alqoritmin emal etdiyi verilənlərinin həcmini xarakterizə edən n parametri mövcuddur. Fərz edək ki, $T(n)$ – A alqoritminin icra müddəti, f – isə n dəyişəninin funksiyasıdır.

$$\frac{T(n)}{f(n)} \xrightarrow[n\to\infty]{} k$$

şərti ödənirsə. Deyirlər ki, A alqoritmi $O(f(n))$ *nəzəri mürəkkəbliyinə* malikdir. Burada k – sabitdir. Əgər alqoritm məsələnin ölçüsündən asılı olmayan, qeyd olunmuş vaxt ərzində icra olunursa, onda deyirlər ki, onun mürəkkəbliyi $O(1)$-ə bərabərdir.

POLİNOMİAL VƏ EKSPONENSİAL ALQORİTMLƏR

Elə alqoritmi *polinomial* (*və ya polinomial zaman mürəkkəblikli alqoritm*) adlandırırlar ki, onun zaman mürəkkəbliyi $O(p(n))$ olsun, burada $p(n)$ – n dəyişənindən asılı olan polinomdur (çoxhədlidir). Mürəkkəbliyi sabit, n dəyişəninin qiymətindən asılı olmayan, müəyyən bir polinomla ifadə olunan həll alqoritminə malik məsələni P sinfinə aid edirlər. Bu faktiki olaraq onu göstərir ki, onu kompüterdən istifadə edərək həll etmək olar. Axı n dəyişəninin hətta çox böyük qiymətlərində belə polinomial alqoritmin reallaşdırılması üçün tələb olunan əməliyyatların sayı elə səviyyədə qalır ki, müasir kompüter onu məqbul zaman kəsiyində çox da çətinlik çəkmədən həll edə bilir.

Çoxsaylı məsələlər var ki, onların mürəkkəbliyi $O(x^n)$ kəmiyyətindən az deyil (burada x – sabitdir). Belə məsələləri E sinfinə aid edirlər və onları *eksponensial* adlandırırlar. Bunlara misal olaraq, verilmiş çoxluğun bütün alt çoxluqlarının qurulması və yaxud verilmiş qrafın bütün alt ağaclarının tapılması məsələsini göstərmək olar. Doğrudan da, əgər M çoxluğu n elementdən ibarədirsə, onda onun bütün alt çoxluqlarının sayı 2^n olacaq. n ədədinin böyük qiymətlərində bu məsələ praktiki olaraq həll edilə bilmir, çünki onun həll alqoritmi 2^n addımdan ibarət olmalıdır. Buna görə də n dəyişəninin qüvvət kimi mürəkkəblik qiymətləndirməsinə daxil olan alqoritmlər də *eksponensial* alqoritmlər sinfinə aid edilir.

Hesab olunur ki, polinomial həll alqoritmi tapılmayana qədər məsələ *yaxşı həll olunan* deyil. Əgər onun həlli üçün belə alqoritm yoxdursa, onda məsələni *çətin həll olunan* adlandırırlar. (Doğrudur, n ədədinin böyük olmayan qiymətlərində eksponensial alqoritm polinomial alqoritmdən, hətta, daha az mürəkkəb ola bilər. Buna baxmayaraq bu iki tip alqoritm arasında böyük fərq var və bu fərq n ədədinin böyük qiymətlərində özünü daha qabarıq göstərir).

Məsələlərin polinomial gətirilməsi anlayışı məlumdur. Əgər A məsələsinin həllini B məsələsinin həllinə çevirən polinomial mürəkkəblikli funksiya mövcüddursa, onda deyirlər ki, A məsələsi polinomial olaraq B məsələsinə gətirilir. Müəyyən edilib ki, NP sinfindən olan ixtiyari digər məsələyə polinomial olaraq gətirilməsi mümkün olan NP-mürəkkəbli məsələlər mövcüddur. Onları *NP-tam* məsələlər (onlara xüsusi halda kommivoyajör haqqındakı məsələ də aiddir) adlandırmışlar. Əgər heç olmasa bir NP-tam məsələsinin həlli üçün polinomial mürəkkəbliyin determinə olunmuş alqoritmini tapmaq mümkün olsa idi, bu NP sinfindən olan ixtiyari məsələnin həlli üçün polinomial mürəkkəblikli alqoritmlərin olması demək olardı.



seçir ki, onun uzunluğu minimum olsun. Marşrutların variantların sayını asanlıqla saymaq olar: bu ədəd $n!$-dır (əlbəttə ki, ! – faktorial işarəsidir, nida işarəsi deyildir). Deməli, ən qısa marşrutu tapmaq üçün biz $n!$ həll variantına baxmalıyıq. Ancaq $n!$ ifadəsi 2^n ifadəsindən də sürətlə artır. Buna görə də, əgər kommivoyajora çox sayda müştərilərə baş çəkmək lazımdırsa, onda bu məsələnin həlli çox zəhmət tələb edəcək.

Eyni zamanda, marşrutları saf-çürük edərək, bizi qane edən variantı bəzən çox tez, bəxtimiz gətirərsə, praktiki olaraq o saat alacağıq. Beləliklə, həllərin axtarılması alqoritmi iki mərhələdən ibarət olur:

• müəyyən bir marşrutun irəli sürülməsi (duyma mərhələsi);

• bu marşrutun həll olub-olmamasının yoxlanması (yoxlama mərhələsi); əgər həlldirsə, onda alqoritmin icrası dayandırılır.

Yoxlama mərhələsi polinomial mürəkkəbliyə malikdir (o, cəmisi bir müqayisənin yerinə yetirilməsindən ibarət ola bilər). Duyma mərhələsi növbəti marşrutun tapılmasından və onun uzunluğunun hesablanmasından ibarətdir; aydındır ki, onun icra vaxtı yaşayış məntəqələrinin sayı olan n ilə mütənasibdir. Başqa sözlə, bu mərhələ də polinomial mürəkkəbliyə malikdir.

Bu cür alqoritmlər *determinə olunmayan* adlanır.

Bəxt gətirsə, kommivoyajör məsələsini polinomial zaman müddətində həll etmək olar. Determinə olunmayan həlləri olan və həmin həlləri polinomial zaman ərzində tapmaq mümkün olan məsələləri *NP-mürəkkəb* (*nondeterministic* – "*determinə olunmayan*" və *polinomial* sözlərindən) adlandırırlar.

Bu cür məsələlər çoxdur və onlar böyük praktik əhəmiyyətə malikdir. Buna görə də, aşağıdakı sualın cavabını bilmək çox vacibdir: onların həlləri üçün polinomial mürəkkəblikli determinə olunan alqoritmlər mövcuddurmu? Bu məsələ üzrə alimlərin fikirləri diametral əksdir. Onlardan çoxu hesab edir ki, bu məsələlərin determinə olunan alqoritmlərini tapmaq heç vaxt mümkün olmayacaq.

Bir çox NP-mürəkkəblikli məsələlərin həlli üçün alqoritmlər mövcuddur. Onlar məsələni başlanğıc verilənlərin geniş diapozonu üçün qənaətbəxş zamana başa gələn həll ilə təmin edirlər. Bu cür alqoritmlərin əsasında çox vaxt *budaqlar* və *sərhədlər* üsulu durur. Bu üsul, həll variantlarının perspektivli olmayan araşdırma istiqamətlərini dinamik olaraq kəsib atır. Buna görə də NP-mürəkkəblikli məsələlərin praktik həllində əsas məqam müəyyən gümanlara (*evristikalara*) əsaslanan təqribi və ya evristik alqoritmlərin qurulması bacarığındadır. Təqribi alqoritmlərin qurulması zamanı adətən, tələb olunur ki, onlar polinomial mürəkkəbliyinə malik olsunlar. Məlumdur ki, bəzən optimal həldən yalnız 10–15% geri qalan həllər tapmaq olur.

Lakin bu cür yaxşı statistikaya baxmayaraq, məsələlər yenə də NP-mürəkkəbli olaraq qalırlar və təəssüf ki, onlar təminatlı zaman qiymətləndirməsinə malik deyillər. Başlanğıc verilənlərin hətta kiçik dəyişiklikləri onun kəskin dəyişmələrinə gətirib çıxara bilər. Bəzi hallarda isə bu zaman imkansız dərəcədə böyük ola bilər.



FORMAL DİLLƏR NƏZƏRİYYƏSİ. TƏRCÜMƏ VƏ KOMPİLYASİYA

İlk kompüterlər üçün proqramlar ikilik, yaxud səkkizlik kodlarda təsvir olunmuş maşın komandaları ardıcıllığı şəklində yazılırdı. Əlbəttə, belə şəkildə böyük proqramların tərtib olunması və sazlanması çox çətin idi.

Ona görə də, proqramçılar həmişə kompüteri insan dilinə yaxın dili başa düşməyə öyrətməyə cəhd etmişlər. Assembler, FORTRAN, Algol-60, C++ və digər proqramlaşdırma dilləri belə əmələ gəlmişlər. Lakin onların hamısı bir cəhətdən təbii dillərdən əsaslı şəkildə fərqlənirlər.

Şifahi və yazılı nitqdə müəyyən qeyri-dəqiqliklər və fikrini axıra qədər deməmək halları həmişə yol veriləndir – dinləyici, yaxud oxucu kontekstdən mənanı bərpa edir (ümumi halda bu kontekst bəşəriyyətin bütün mədəni bilik yükünü birləşdirir). Kompüter isə dəqiq və birmənalı təlimatlara möhtacdır. Deməli, kompüter proqramlarının yazılışı heç bir ikimənalılığa imkan verməyən, ciddi qaydalara tabe olmalıdır. Kompilyator isə onları səhvsiz olaraq maşın komandaları ardıcıllığına çevirməlidir. Burada elmlərin ən dəqiqi olan riyaziyyatın köməyi olmadan keçinmək olmaz. Təəccüblü deyildir ki, "maşın" dillərinin analizi üçün riyazi üsullar yaradılmışdır və onlar kompilyatorların yaradılmasında da geniş şəkildə tətbiq olunurlar.

FORMAL DİLİN TƏRİFİ

İstənilən dildə cümlə elementar simvollardan: hərflər, rəqəmlər, boşluq simvolları və durğu işarələrindən təşkil olunur. Ona görə də, cümləni formal olaraq simvollar ardıcıllığı, yaxud zənciri kimi təyin etmək olardı. Ancaq belə təyinetmə həddindən artıq genişdir. Simvolların bütün mümkün zəncirləri çoxluğunda, *düzgün qurulmuş zəncirlərin* ("düzgün ardıcıllıqla düzülmüş", "düzgün sözlərdən" ibarət olan) alt çoxluğunun ayrılmasının mənası vardır. Məsələn, "rehklou aszpöu mutvfdas mnqasdl" və "Araba üzür boyunca yol" zəncirləri Azərbaycan dilinin düzgün tərtib olunmuş cümlələri deyil. "Araba yol boyunca üzür" cümləsi isə düzgün tərtib olunmuşdur, ancaq məna baxımından şübhəlidir.

Proqramlaşdırma dillərinin də aid olduğu süni dillərdə düzgün qurulmuş cümlələr çoxluğu formal riyazi konstruksiyaların köməyilə əvvəlcədən təsvir oluna bilər. Ona görə də, belə dilləri *formal dillər* adlandırırlar.

Əgər aşağıdakılar verilmişdirsə formal dil müəyyən olunmuş hesab olunur:
1) təkcə hərfləri deyil, verilmiş dilin bütün simvollarını, o cümlədən boşluq və durğu işarələrini də özündə birləşdirən *əlifba;*
2) dilin əlifbasından olan simvolların verilmiş zəncirinin düzgün qurulub-



qurulmamasını müəyyən etməyə imkan verən *meyarlar*. Bu meyarların necə verilməsinin mənası yoxdur. Ən sadə halda sadəcə olaraq bütün mümkün zəncirləri sadalamaq olar. Yaxud oxşar zəncirlərin "yaradılması" qaydaları toplusunu tərtib etmək olar. Belə qaydalar toplusu *qrammatika*, dilin zəncirlərinin quruluşu isə *sintaksis* adlanır.

Qrammatika kontekstdən-azad adlanır, çünki terminal olmayanları sağ hissəyə dəyişərkən, qaydalar nəzərə alınmır. Anlayışın interpretasiyası kontekstdən asılı deyildir.

KONTEKSTDƏN-AZAD QRAMMATİKALAR

Kontekstdən-azad qrammatikalar əsla riyaziyyatçılar, yaxud proqramçılar tərəfindən düşünülüb tapılmayıb. İlk dəfə onlardan təbii dilləri təsvir etmək üçün dilçilər istifadə etmişlər.

Zəncirlərin tərkib hissələrini ya dilin əlifbasına daxil olmayan hərflərlə, ya da anlayışların adlarıyla – künc mötərizələrinə alınmış *metasimvollarla*, yaxud *qeyri-terminallarla* işarə edirlər. Azərbaycan dilinin qeyri-terminallarına: cümlə, mübtəda, mübtəda qrupu, xəbər, xəbər qrupu, təyin, zərfliyin istifadə olunduğu daha sadə cümlələrinə baxmaq olar. Qrammatika *qaydalardan* ibarətdir.

Qaydanın sol tərəfində qeyri-terminal, sağ tərəfində terminal və qeyri-terminallardan ibarət ixtiyari zəncir yazılır, sanki sol tərəfdə göstərilmiş anlayışın mənası açıqlanır. Qaydanın sol və sağ tərəfləri "bu ibarətdir" kimi oxunan oxla ayrılır.



Bu qrammatikanın birinci qaydası

⟨cümlə⟩→
⟨mübtəda qrupu⟩⟨xəbər qrupu⟩

təyin edir ki, cümlə iki hissədən ibarətdir – mübtəda qrupu və onun ardınca gələn xəbər qrupu.

Qrammatikada dilin bütün anlayışları üçün qaydalar yazılmalıdır. Örnək olaraq mübtəda qrupu üçün iki qayda daxil etmək olar:

⟨mübtəda qrupu⟩→
⟨mübtəda⟩
⟨mübtəda qrupu⟩→
⟨təyin⟩⟨mübtəda qrupu⟩.

İkinci qaydada mübtəda qrupu qeyri-terminalına qaydanın həm sol, həm də sağ tərəflərində rast gəlinir. Bu, kontekstdən-azad qrammatikaların istənilən qədər uzunluqda cümlələrin qurulmasına imkan verən vacib əlamətləridir. Bu halda ikinci qayda sol tərəfdə mübtədaya istənilən sayda təyin əlavə etməyə imkan verir.

Əgər qrammatikada sol tərəfləri eyni olan bir neçə qayda varsa, onda bütün sağ tərəfləri şaquli xətlərin köməyilə birləşdirərərək (o "yaxud" kimi oxunur), bir yazıya gətirmək olar:

⟨mübtəda qrupu⟩→
⟨mübtəda⟩|⟨təyin⟩|⟨mübtəda qrupu⟩.

Qrammatikanın bütün qaydaları bunlardır:

⟨cümlə⟩→
⟨mübtəda qrupu⟩⟨xəbər qrupu⟩;
⟨mübtəda qrupu⟩→
⟨mübtəda⟩|⟨təyin⟩⟨mübtəda qrupu⟩;
⟨mübtəda⟩→qayıq|qazalaq;
⟨təyin⟩→yaşıl|ağ|böyük;
⟨xəbər qrupu⟩→
⟨xəbər⟩⟨zərflik⟩;
⟨xəbər⟩→hərəkət edir|üzür;
⟨zərflik⟩→yolda|çayda.

Qrammatikada başlanğıc qeyri-terminal seçdirilir, dilin zəncirləri ondan çıxarılır. Misalda bu cümlə qeyri-terminaldır. Birinci addımda ona sol tərəfi verilmiş qeyri-terminalla üst-üstə düşən qaydalardan birini tətbiq edirlər. Simvolun qaydanın sağ tərəfilə əvəzlənməsi baş verir. Belə əvəzlənmələr ardıcıllığı *nəticə* adlanır. Nəticə o zaman qurtarır ki, zəncirdə bir dənə də qeyri-terminal qalmasın.

Nəticəyə aid misala baxaq: ⟨cümlə⟩ qeyri-terminalı üçün yeganə qaydadan istifadə edək:

⟨cümlə⟩→
⟨mübtəda qrupu⟩⟨xəbər qrupu⟩.

Sonra ⟨mübtəda qrupu⟩ qeyri-terminalına ikinci qayda tətbiq olunur:

⟨mübtəda qrupu⟩⟨xəbər qrupu⟩→
⟨təyin⟩⟨mübtəda qrupu⟩⟨xəbər qrupu⟩.

Nəticəni davam etdiririk:

⟨təyin⟩⟨mübtəda qrupu⟩⟨xəbər qrupu⟩→
yaşıl⟨mübtəda qrupu⟩⟨xəbər qrupu⟩→
yaşıl⟨mübtəda⟩⟨xəbər qrupu⟩→
yaşıl qazalaq⟨xəbər qrupu⟩→
yaşıl qazalaq⟨zərflik⟩⟨xəbər⟩→
yaşıl qazalaq yolda⟨xəbər⟩→
yaşıl qazalaq yolda hərəkət edir.

Axırıncı addımda bitkin nəticə alındı. Buna analoji olaraq ⟨cümlə⟩-nin başqa variantlarını da qurmaq olar:

böyük ağ qayıq çayda üzür;
qazalaq yolda üzür;
ağ yaşıl qazalaq çayda gedir və s.

İki axırıncı nümunə göstərir ki, qrammatika cümlələrin mənalarını nəzərə almır.

HESAB DÜSTURLARI DİLİ

Dəyişənlərin adlarını, tam sabit kəmiyyətləri, hesab əməllərinin işarələrini, həmçinin mötərizələri özündə birləşdirən *hesabi ifadə* anlayışını təyin edərək proqramlaşdırma baxımından daha vacib



olan misala baxmaq olar. Dəyişənlərin adları bir latın hərfindən ibarətdir. Aşağıdakı qrammatika verilmişdir:

⟨düstur⟩→⟨elementar düstur⟩|⟨düstur⟩+
+⟨düstur⟩|⟨düstur⟩–⟨düstur⟩|⟨düstur⟩·
·⟨düstur⟩|⟨düstur⟩/⟨düstur⟩;
⟨elementar düstur⟩→⟨dəyişən⟩|⟨sabit⟩|
|(⟨düstur⟩);
⟨dəyişən⟩→a|b|c|...z;
⟨sabit⟩→⟨rəqəm⟩|⟨sabit⟩⟨rəqəm⟩;
⟨rəqəm⟩→0|1|2|...|9;

Bu qrammatika ilə verilmiş dil, adları bir hərfdən ibarət dəyişənləri, tam sabitləri özündə birləşdirən bütün düzgün hesabi ifadələrdən ibarətdir. $x+7$ zəncirinin çıxarılması misalı:

⟨düstur⟩→⟨düstur⟩+⟨düstur⟩→
⟨düstur⟩+⟨elementar düstur⟩→
⟨düstur⟩+⟨sabit⟩→⟨düstur⟩+⟨rəqəm⟩→
⟨düstur⟩+7→
⟨elementar düstur⟩+7→
⟨dəyişən⟩+7→x+7.

Nəticənin hər addımında cari zəncirin ən sağ qeyri-terminalına qaydalardan biri tətbiq olunur. Belə nəticə *sağ nəticə* adlandırılır. Həmçinin, sol nəticə də vardır. Hər bir zənciri onun köməyilə də çıxarmaq olar. Doğrudur, bir çox qrammatikalarda nəticələr yeganə olmur. *Determinə olunmamaq* qeyri-formal olaraq dildə cümlələrin yozumunun birqiymətli olmamasını göstərir. Hesab düsturları dilinin yuxarıda göstərilən qrammatikası determinə olunmamışdır: 1-2-3 düsturu iki müxtəlif sağ nəticəyə malikdir:

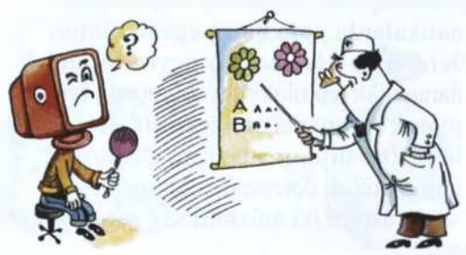
1) ⟨düstur⟩ → ⟨düstur⟩ - ⟨düstur⟩ →
   ⟨düstur⟩ - ⟨elementar düstur⟩ → ...
   ⟨düstur⟩ - 3 → ... → 1-2-3;
2) ⟨düstur⟩ → ⟨düstur⟩ - ⟨düstur⟩ →
   ⟨düstur⟩ - ⟨düstur⟩ - ⟨düstur⟩ → ...
   ⟨düstur⟩ - 2-3 → ... → 1-2-3;

Birinci nəticə ona uyğun gəlir ki, əməliyyatlar soldan sağa yerinə yetirilir, yəni əvvəlcə 1-2 əməliyyatı altdüstur şəklində təsvir olunur, sonra ondan 3-ü çıxırlar. İkinci nəticə: əməliyyatları sağdan sola həyata keçirirlər, yəni 2-3 – altdüsturdur, onu 1-dən çıxırlar. Real hesablamalar zamanı, birinci halda nəticə (1-2)-3=-4, ikinci halda 1-(2-3)=2 olacaqdır. Düstur birqiymətli olmayaraq yozulur, yəni qrammatika uğursuzdur.

Əgər ixtiyari zəncirin sağ nəticəsi yeganədirsə, onda qrammatika *determinə olunmuş* adlanır. Bu misalda əlavə term və vuruq anlayışlarını daxil etməklə qrammatikanı determinə olunmuş etmək olar:

⟨düstur⟩ → ⟨düstur⟩ + ⟨term⟩ | ⟨düstur⟩ - ⟨term⟩ | ⟨term⟩;
⟨term⟩ → ⟨term⟩ · ⟨vuruq⟩ | ⟨term⟩ / ⟨vuruq⟩ | ⟨vuruq⟩;
⟨vuruq⟩ → ⟨elementar düstur⟩.

Qalan qaydalar birinci halda olduğu kimidir. Burada, düstur termlərin cəmi, yaxud fərqi şəklində göstərilir; hər term – ya hasil, ya qismətdir, vuruq – ya dəyişən, ya sabit, ya da mötərizəyə alınmış ixtiyari düsturdur. Verilmiş qrammatika artıq determinə olumuşdur, o, riyaziyyatda qəbul olunduğu kimi, mötərizələr olmadıqda əməliyyatların soldan sağa yerinə yetirilməsinə uyğun gəlir. Bundan başqa, o, vurma və bölmə əməliyyatlarının toplama və çıxmaya nəzərən birinciliyini də nəzərə alır.



Müasir alqoritmik dillər kontekstdən-azad qrammatikalar vasitəsilə təsvir olunurlar. Hətta, alimlər hesab edirlər ki, təbii dilləri də bu qaydayla vermək olar. Maraqlıdır ki, C dilinin qrammatikası ilə verilmiş kontekstdən-azad qrammatika özünə düzgün C-proqramlardan əlavə bəzi səhv proqramlar da daxil edir. Ancaq bu sintaksislə deyil, təsvir olunmamış dəyişənlər və ifadələrin düzgün olmayan tipləriylə bağlıdır. Əgər onu kimsə yazıb təsvir edə bilsəydi, elə bunun kimi Azərbaycan dilinin kontekstdən azad-qrammatikası da, normal cümlələri qurmaqla yanaşı, mənasız cümlələr qurmağa da imkan verərdi.

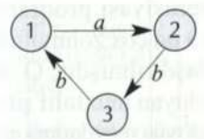
AVTOMAT VƏ REQULYAR DİLLƏR

Adətən, kontekstdən-azad qrammatika dilin sintaksisini cümlə quruluşu səviyyəsində təsvir edir. Məsələn, Azərbaycan dilinin sintaksisi baxımından "yaşıl" və "sarı" sifətləri arasında heç bir fərq yoxdur. Eynilə bunun kimi, ixtiyari alqoritmik dildə müxtəlif tam sabitlər arasında qrammatik fərq yoxdur. Ona görə də, alqoritmik dilin qrammatikasını vermək üçün elementar anlayışlar olaraq "dəyişən", "sabit" anlayışlarını götürmək daha əlverişlidir (sözləri təşkil edən hərf və rəqəmləri götürməyə gərək yoxdur).

"Sözlər dili" "cümlələr dili"ndən fərqlənir, onun quruluşu xeyli sadədir və kontekstdən-azad qrammatikanın tətbiqini tələb etmir. Alqoritmik dillərin əksəriyyətində ayrı-ayrı sözlər *sonlu avtomatların*, yaxud *requlyar ifadələrin* köməyilə verilir.

Sonlu avtomat – istiqamətlənmiş qrafdır, başqa sözlə öz aralarında oxlar-*tillər* (təpələri özü-özüylə birləşdirən til-ilgəklər də mümkündür) vasitəsilə birləşdirilmiş təpələrdən ibarət fiqurdur. Tillər giriş əlifbasının hərflərilə qeyd olunur. Təpələrdən biri başlanğıc, bir neçəsi isə son təpələr kimi işarə olunur. Qrafın təpələrini sonlu avtomatın *vəziyyətləri*, tilləri isə *keçidlər* də adlandırırlar.

Sonlu avtomatla verilmiş dil bütün mümkün zəncirləri doğurur. Onları başlanğıc təpədən son təpələrdən birinə qrafın tilləri boyunca hərəkət etməklə oxumaq olar. Məsələn,



sonlu avtomatında 1 – başlanğıc, 2 – son təpədir. Bu avtomatla verilmiş dil 1 təpəsindən başlayan və 2 təpəsində qurtaran yollara uyğun *a, abba, abbabba, abbabbabba, ...*, zəncirlərindən ibarətdir.

İstənilən *avtomat dili* kontekstdən-azaddır. Əksi doğru deyildir, kontekstdən-azad dillər sinfi aftomat dillərdən genişdir. Məsələn, sıfır və birlərin bütün mümkün zəncirlərindən ibarət olan dil, zəncirlərdə sıfır və birlərin sayı bərabər olmaq şərtilə, kontekstdən-azad dildir, lakin avtomat dil deyildir.



Avtomat dillərin başqa üsulla – *requlyar ifadələrin* köməyilə verilməsi də vardır. Requlyar ifadələr, artıq qurulmuş requlyar dillərdən yeni requlyar dillər qurmağa imkan verən bir neçə sadə əməliyyatı özlərində birləşdirir. Bu, *konkatenasiya* (iki zəncirin birləşdirilməsi), iki variantdan birinin *seçilməsi*, həmçinin *iterasiya* (ixtiyari ədəd dəfə zəncirin

LEX və GREP PROQRAMLARI

UNIX əməliyyat sistemində bir neçə standart proqram var ki, onların işi requlyar dillər nəzəriyyəsinə əsaslanır. Hər şeydən əvvəl bu, mətnin leksik analizini həyata keçirən, yəni requlyar ifadələrin müəyyən yığınına uyğun olaraq mətni sözlərə bölən LEX proqramıdır. Həmçinin, bir və ya bir neçə faylda kontekst axtarışı həyata keçirən GREP (*ing.* "Get Regular Expression Pattern" – "requlyar ifadəyə görə nümunənin ayrılması") proqramıdır. Axtarışa nümunə kimi ya ixtiyari mətn fraqmenti, ya da requlyar ifadə vermək olar. Özü də axtarış çox sürətlə baş verir, o, giriş faylının uzunluğundan xətti asılı olan və axtarılan fraqmentin uzunluğundan asılı olmayan vaxt aparır. İşin yüksək sürəti onunla izah olunur ki, GREP proqramı axtarış üçün sonlu avtomat qurur. Avtomata simvollar ardıcıl olaraq giriş axınından verilir.

```
[root@linus root]# cat aze.hasp | tr '\000' '\n' | grep -i '^[FT]..R.[AI]$'
FİRNİ
FƏRDA
FİRMA
FERMA
TURBA
TORDA
TƏRBİ
TERMİ
[root@linus root] #
```

GREP proqramının işinin nəticəsi. Lüğətdə 5 simvoldan ibarət olan "F" yaxud "T" ilə başlayan, 3-cü simvolu "R" olan və "A", yaxud "I" ilə qurtaran sözlərin tapılması tələb olunmuşdur.

Ümumiyyətlə, təbii dilin cümləsinin araşdırılması məsələsi birqiymətli həllə malik deyildir. Məsələn, "ərəb atları sürətlə qaçırdı" cümləsində söhbətin atların qaçırılmasından, yoxsa atların qaçmasından getdiyi aydın deyildir. Bəzən cümlənin mənasını kontekstdən başa düşmək olur, amma bu kompüter üçün irişilməzdir.

təkrarlanması) əməliyyatlardır. Qurma prosesi bir hərfli yeganə zəncirdən ibarət elementar dillərdən başlayır. Klini teoreminə görə, avtomat və requlyar dillər sinfi üst-üstə düşür, yəni hər bir avtomat dili requlyar ifadələr vasitəsilə və əksinə, hər bir requlyar dili sonlu avtomatlar vasitəsilə vermək olar. Misalda requlyar ifadə aşağıdakıdır:

$$a(bba)^*,$$

burada * ilə iterasiya işarə olunmuşdur, mötərizələr isə qruplaşdırma üçün istifadə olunurlar.

Dilin requlyar ifadələrin köməyilə verilməsi insan üçün, dilin sonlu avtomat vasitəsilə göstərilməsi kompüter üçün daha asandır.

KOMPİLYATORUN İŞİNİN ƏSAS MƏRHƏLƏLƏRİ

İxtiyari kompilyatorun işi bir neçə addımdan ibarətdir:

Addım 1: giriş proqramının təhlil edilməsi və onun sonrakı iş üçün əlverişli olan daxili təsvir formasına çevrilməsi. Bu mərhələdə həmçinin giriş proqramındakı səhvlər də aşkarlanır.



Addım 2: proqramın konkret kompüterin komandalar sistemindən asılı olmayan ara dilə çevrilməsi.

Addım 3: proqramın kodunun optimallaşdırılması.

Addım 4: çıxış kodunun Assemblerdə, yaxud bilavasitə maşın komandaları dilində generasiya olunması.

Proqramçılar 1, 2 və 4 addımlarını sürətlə və effektiv yerinə yetirməyi öyrənmişlər. Əsas nailiyyət formal dillər nəzəriyyəsindən istifadə və onlar əsasında yaradılan kompilyatorların avtomatik generasiyası proqramları oldu.

Addım 3 ən çox zəhmətlidir və daha az formallaşdırılmışdır. O, kompilyatorlar hazırlayan müxtəlif şirkətlər arasında əsas döyüş meydanına çevrilmişdir. Kompilyatorların yaradıcılarının gücünün böyük hissəsi məhz kodun optimallaşdırılmasına sərf olunur.

ARAŞDIRMA MƏSƏLƏSİ

Kompilyatorun işinin birinci addımında (giriş proqramının araşdırılması) verilmiş simvollar zəncirinin düzgün qurulmuş cümlə olub-olmamasının təyin olunması tələb edilir və əgər zəncir düzgün cümlədirsə onun nəticəsi bərpa olunur. Nəticəsinin bərpa olunması prosesi *araşdırma*, yaxud *sintaktik analiz*, araşdırmanı həyata keçirən proqram isə *parser* (*ing.* "to parse" – "ayırd etmək", "təhlil etmək") adlanır.

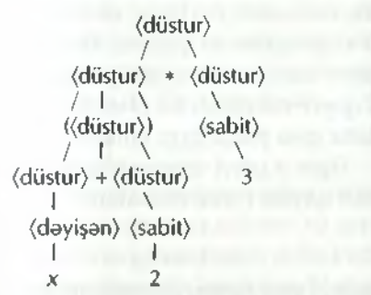
Məsələn, ingilis dilindən Azərbaycan dilinə avtomatik tərcümə sisteminə baxmaq olar. Ayrı-ayrı sözlərin tərcüməsi qaneedici nəticələr vermir, çünki ingilis və Azərbaycan dillərinin sintaksisi üst-üstə düşmür. Ona görə də, əvvəlcə ingilis dili cümləsinin araşdırılmasını həyata keçirmək lazımdır, yəni nəyin mübtəda, xəbər, tamamlıq və sair olduğunu başa düşmək, ancaq sonra Azərbaycan dilinin qaydalarına əsasən Azərbaycan cümləsini tərtib etmək lazımdır.

Təbii dillərdən fərqli olaraq, alqoritmik dillər üçün araşdırma məsələsinin həmişə birqiymətli həlli var. Proqramı yüksək səviyyəli alqoritmik dildən Assemblerə, yaxud maşın komandaları dilinə çevirən translyator işin birinci addımında araşdırma məsələsini həll edir. Bu zaman o, proqramı müəyyən daxili formaya çevirir; adətən bu, *sintaktik ağacların* köməyilə verilir.

Sintaktik ağac anlayışını misalla nümayiş etdirmək olar. Tutaq ki, aşağıdakı qrammatika verilmişdir:

⟨düstur⟩ → ⟨düstur⟩+⟨düstur⟩ |
⟨düstur⟩*⟨düstur⟩ |(⟨düstur⟩) |
⟨dəyişən⟩ | ⟨sabit⟩

(yığcamlıq üçün ⟨dəyişən⟩ və ⟨sabit⟩ qeyri-terminalları üçün qaydalar buraxılmışdır). Bax $(x+2)*3$ ifadəsi üçün sintaktik ağac belədir:



Sintaktik ağacın hər bir qeyri-terminal təpəsi zəncirin nəticəsinin bir addımına uyğun gəlir. Təpələr qeyri-terminallarla nömrələnir, həmin qeyri-terminallar nəticə zamanı uyğun qaydanın sağ tərəfinə əvəz olunurlar. Təpələrdən qaydaların sağ tərəfini təşkil edən bütün simvollara budaqlar çıxır.

Zəncirin nəticəsinə görə birqiymətli olaraq sintaktik ağac bərpa olunur və əksinə, sintaktik ağaca əsasən birqiymətli olaraq sağ, yaxud sol nəticələr yazılır.

Nəzəri olaraq bütün kontekstdənazad qrammatikalar üçün araşdırma məsələsinin alqoritmik həlli var, lakin ümumi alqoritm çox yavaş, təxminən N^3



vaxta (N – giriş zəncirinin uzunluğudur) işləyir. Bu translyatorlar üçün qəbuledilməzdir: giriş proqramının uzunluğu 10 dəfə artdıqda translyasiya vaxtı 1000 dəfə artmış olur! Ona görə də, praktikada daha sürətli araşdırmaya imkan verən, xüsusi sinif qrammatikalarla işləyirlər. İki araşdırma alqoritmi daha əhəmiyyətlidir:

- *enən* (*rekursiyalı*) araşdırma. Enən araşdırmalara icazə verən qrammatikaları LL(1)-qrammatikalar adlandırırlar;
- *yüksələn* araşdırma, yaxud stekli sonlu avtomatın (bəzən "sandıq yaddaşlı avtomat" deyirlər) köməyilə araşdırma. Uyğun qrammatikalar sinfini LR(1)-qrammatikalar adlandırırlar.

Hər iki halda giriş zəncirinə soldan sağa baxılır. Sinfin adındakı birinci hərf – L baxışın istiqamətini bildirir (*ing.* "left" – "sol"). Birinci halda sol nəticələr, ikincidə isə sağ nəticələr qurulur. Bu da addakı ikinci hərflərlə işarələnmişdir (*ing.* "right" – "sağ"). Addakı 1 aydınlaşdırır ki, alqoritm giriş zəncirinə bir simvol irəli baxır.



Hesab düsturları qrammatikasında belə qayda var idi: ⟨düstur⟩→ →⟨düstur⟩+⟨term⟩. Bu qrammatikada enən araşdırmaya cəhd sonsuz rekursiyaya gətirər. Ona görə ki, bu halda *bu* ⟨düstur⟩ alt proqramı giriş zəncirindən heç bir simvol oxumadan daim özü-özünə müraciət edəcəkdir. Deməli, bu qrammatika enən araşdırmaya icazə vermir (amma yüksələn araşdırmaya icazə verir!).

Enən araşdırma zamanı sintaktik ağac kök təpədən başlamaqla, yuxarıdan aşağıya qurulur, yüksələn araşdırma zamanı isə aşağıdan yuxarıya bərpa olunur.

Bir qayda olaraq, LL(1) xassəsinə malik hər hansı qrammatika LR(1) xassəsinə də malik olur. Başqa sözlə, LR(1)-qrammatikalar sinfi LL(1)-qrammatikaları da özündə birləşdirir. Bununla belə, LL(1) olmayan LR(1)-qrammatikalar da vardır: məsələn, hesabi düsturlar dilinin determinə olunmuş qrammatikası və C dilinin qrammatikası. Ümumiyyətlə, sol-rekursiyalı qrammatika qurmaq həmişə mürəkkəbdir: hətta bunu etmək mümkün olsa belə, nəticə olduqca qeyri-təbii görünəcəkdir. Əksinə, LR(1)-qrammatika, adətən, çox asanlıqla əksər alqoritmik dillər üçün yazılır.

Ümumiyyətlə, LL(1)-qrammatikalara və rekursiyalı alqoritmlərə baxmaq nəyə lazımdır? Yeganə səbəb ondan ibarətdir ki, əksər hallarda istifadə olunan yüksələn araşdırma alqoritminin başa düşülməsi bir qədər proqramçı mədəniyyəti tələb edir. Heç olmasa, stekin nə olduğunu bilmək lazımdır. Ona görə də, proqramlaşdırmaya yeni başlayanlar üçün olan ədəbiyyatların çoxu rekursiyalı araşdırma alqoritmlərilə kifayətlənirlər.



ENƏN (REKURSIYALI) ARAŞDIRMA

Tutaq ki, qrammatikada aşağıdakı qayda var

$$A \to BtD,$$

burada A, B, D – qeyri-terminallar, t – terminaldır (yəni, adi simvoldur).

Giriş zəncirindən başlanğıc parçanı ayırmağa cəhd edən, A qeyri-terminalından çıxan "A" adlı altproqramı yaratmaq lazımdır. Göstərilən qaydaya əsasən bu parça terminal olmayan B-dən çıxan parçadan başlamalıdır; sonra adi t simvolu və ardınca D qeyri-terminalından çıxan parça gəlir. A altproqramı əvvəlcə B qeyri-terminalına uyğun gələn, zəncirin başlanğıc parçasını oxuyan və araşdıran B altproqramını çağırır. Sonra A altproqramı zəncirin növbəti simvolunun t simvolu ilə üst-üstə düşüb-düşməməsini yoxlayır, onu buraxır və D altproqramını çağırır. Beləliklə, A qeyri-terminalına uyğun parçanın emalı B qeyri-terminalı və D-ni təmsil edən daha qısa parçaların emalına gətirilir.

Əgər A qeyri-terminalı üçün iki müxtəlif qayda varsa, məsələn,

$$A \to BtD,$$
$$A \to Efg,$$

onda A altproqramı giriş zəncirinin birinci simvoluna görə bu qaydaların hansının tətbiq olunmasının lazım olduğunu araşdırmalıdır. Bunu ancaq o zaman etmək olar ki, B qeyri-terminalı və E-dən çıxan ixtiyari zəncirlər müxtəlif simvollarla başlasınlar.

Alqoritmin adında "rekursiyalı" sözü ona görə işlədilir ki, B qeyri-terminalının təyinində yenidən A qeyri-terminalına rast gəlinə bilər (ya da qrammatikada elə qayda olacaq ki, solunda yaxud da sağında A qeyri-terminalı olacaqdır). Beləliklə, B altproqramı öz növbəsində A-nı çağıracaqdır (ya da A altproqramı özü-özünü çağıracaqdır). Ona görə də enən araşdırma zamanı rekursiya mümkündür.

YÜKSƏLƏN ARAŞDIRMA

Yüksələn araşdırma alqoritmində stekdən istifadə olunur. Onu terminal və qeyri-terminallardan ibarət, yalnız sağ tərəfdən uzana, yaxud dəyişdirilə bilən söz kimi təsəvvür etmək olar. Misal olaraq hesabi düsturlar qrammatikasına baxmaq olar:

$$F \to F+F \mid V \mid C$$
$$V \to x \mid y$$
$$C \to 1 \mid 2 \mid \ldots \mid 9,$$

burada, F – qeyri-terminalı düsturu, V – dəyişəni, C – sabiti işarə edir. Burada ancaq toplama əməliyyatı, iki x, y dəyişənləri və sabit kimi rəqəmlərlə kifayətlənmişlər.

$x+y+1$ zəncirini araşdıraq. Alqoritm LR-proses adlanan prosesi modelləşdirir. İxtiyari addımda LR-prosesin vəziyyəti steklə və giriş zəncirinin oxunmamış hissəsiylə müəyyənləşir. Stek, elə bil, giriş zəncirinə qarşı döndərilmişdir. Başlanğıcda stek boşdur, giriş zənciri isə ilkin zəncirlə üst-üstə düşür:

$$\boxed{} \Leftarrow x+y+1$$

Alqoritm iki hərəkəti yerinə yetirə bilər. Birinci *sürüşmə* adlanır və giriş zəncirinin birinci simvolunun stekə keçirilməsindən ibarətdir (ərəb hesablamalarında simvolların sağdan sola "atılması" kimi):

$$\boxed{\Leftarrow x+y+1} \to \boxed{x \Leftarrow +y+1}$$

Tutaq ki, stekin təpəsi, yəni steki təmsil edən sözün sonu hər hansı qaydanın, indiki halda $V \to x$ qaydasının sağ tərəfilə üst-üstə düşür. Onda stekin təpəsini qaydanın sol tərəfilə əvəz etmək olar. Bu əməliyyat uyğun qayda üzrə *burma* adlanır:

$$\boxed{x \Leftarrow +y+1} \to \boxed{V \Leftarrow +y+1}$$

İndi $F \to V$ qaydası üzrə burma yerinə yetirilir, yəni stekin təpəsində V simvolu F ilə əvəz olunur:



$$\boxed{V \Leftarrow +y+1} \to \boxed{F \Leftarrow +y+1}$$

Sonra iki dəfə sürüşmə yerinə yetirilir:

$$\boxed{F \Leftarrow +y+1} \to \boxed{F+ \Leftarrow +y+1} \to \boxed{F+y \Leftarrow +1}$$

İndi $V \to y$ qaydası sonra $F \to V$ qaydası üzrə burma yerinə yetirilir:

$$\boxed{F+y \Leftarrow +1} \to \boxed{F+V \Leftarrow +1} \to \boxed{F+F \Leftarrow +1}$$

və nəhayət, $F \to F+F$ qaydasına əsasən burma yerinə yetirilir, yəni stekin təpəsində $F+F$ zəncirinin qaydanın sol tərəfilə – F hərfilə əvəz olunması baş verir:

$$\boxed{F+F \Leftarrow +1} \to \boxed{F \Leftarrow +1}$$

Sonra iki dəfə sürüşmə yerinə yetirilir, ardınca $C \to 1$ qaydasına, sonra isə $F \to C$ qaydasına əsasən burmalar yerinə yetirilir. Axırıncı addımda isə $F \to F+F$ qaydasına əsasən burma baş verir:

$$\boxed{F \Leftarrow +1} \to \boxed{F+ \Leftarrow 1} \to \boxed{F+1 \Leftarrow} \to$$
$$\boxed{F+C \Leftarrow} \to \boxed{F+F \Leftarrow} \to \boxed{F \Leftarrow}$$

Axırda giriş zənciri tam oxunmuş olur, LR-prosesin steki isə qrammatikanın başlanğıc qeyri-terminallarını özündə saxlayır. Beləliklə, araşdırma uğurlu nəticə ilə sona çatdırıldı, giriş zənciri isə dilə məxsusdur.



Yüksələn araşdırma zamanı çoxqiymətlilik əmələ gələ bilər. Belə ki, bəzi anlarda sürüşmə, yaxud burma yerinə yetirmək olar, ya da ki, burmanı müxtəlif qaydalar əsasında yerinə yetirmək olar. Araşdırma alqoritmində stekin təpəsinə və giriş zəncirinin hansı hərəkətin yerinə yetirilməsini təyin edən birinci simvoluna əsasən sonlu avtomat qurulur. Bu avtomatın qurulmasının dəqiq proseduru vardır. Lakin real alqoritmik dillər üçün onun vəziyyətlərinin sayı minlərlə ölçülə bilər. İnsan üçün bu olduqca çoxdur. Kimsə bir dəfə əllə bütün vəziyyətləri yazmağı sınamışdırsa və araşdırma cədvəllərini doldurmuşdursa, yəqin ki, bənzər işi yenidən yerinə yetirməyə razılaşmaz, proqram yazar ki, bunları avtomatik yerinə yetirsin. Lakin belə proqram yazmaq lazım deyildir, o, artıq çoxdan yazılmışdır və YACC adlanır.

KOMPİLYATORLARIN YARADILMASININ LEX VƏ YACC İNSTRUMENTAL VASİTƏLƏRİ

Mətnin sintaktik analizi (araşdırılması) məsələsinin həlli üçün LEX və YACC (*ing.* Yet Another Compiler Compiler – daha bir "kompilyatorlar kompilyatoru") proqramları nəzərdə tutulub. Sintaktik analiz iki mərhələdə yerinə yetirilir. Birincisində simvollar skaner vasitəsilə sözlərə (leksemlərə) ayrılır (*ing.* "scaner". Bu, mətni avtomatik olaraq LEX proqramıyla hasil edilən, "C" dilində olan proqramdır). Giriş faylı giriş dilinin müxtəlif tipli sözlərini təqdim edən requlyar ifadələrin siyahısından ibarətdir.

İkinci mərhələdə skaner tipini (dəyişən, sabit, açar söz) göstərməklə növbəti sözü və məlumatları parserə bildirir – məsələn, sabitlər üçün onların ədədi qiymətləri ötürülür. Parser yüksələn araşdırmanın (LR-araşdırmanın) alqoritmini reallaşdırır. LR prosesi stekinə paralel olaraq parser proqramçıya araşdırılan dilin anlayışlarıyla bağlı olan istənilən əlavə məlumatı saxlamaq və bu yolla ona vacib olan məsələni həll etmək imkanı verən semantik steki dəstəkləyir. Məsələn, ingilis dilindən Azərbaycan dilinə tərcümə zamanı sözlər üçün onların mənasını, təyinlər üçün təyin olunan ismə istinadları saxlamaq lazımdır ki, cinsi və s. uzlaşdırmaq olsun.

C dilində olan parser proqramının mətni YACC-ın işinin nəticəsində yaranır. C proqramıyla yaradılmış YACC faylı daha sonra kompilyatoru reallaşdıran böyük layihənin hissəsi kimi istifadə edilir.

Kompilyatorların hazırlanmasının belə texnologiyası "əllə" primitiv proqramlaşdırmayla müqayisədə çox böyük üstünlüklərə malikdir. Birincisi, o, qüvvələrə və vaxta qənaət edir, beləliklə də, səhvlərin olması ehtimalını azaldır. İkincisi, giriş dilinin kontekstdən-azad qrammatikası, sadəcə, uyğun gələn standartdan köçürülür. Bu zəmanət verir ki, bu yolla yaradılmış bütün kompilyatorlar standarta uyğun olacaq və hətta müxtəlif kompüterlərdə eyni cür (ən azından sintaktik araşdırma mərhələsində) işləyəcək. Proqramların bir platformadan digərinə köçürülə bilməsi isə proqramlaşdırmanın ən ağrılı nöqtələrindən biridir. Nəhayət, parserin mətni olduqca gözəldir və ümumiyyətlə, elmi nailiyyətlərdən praktikada istifadə etmək həmişə xoşdur.

PROQRAMLARIN DÜZGÜNLÜYÜNÜN İSBATI

Proqramların düzgünlüyünün isbatı riyazi teoremlərin isbatını xatırladır. Hər bir proqrama belə formada olan teorem kimi baxılır: "Əgər giriş verilənləri bir neçə şərti ödəyirsə, onda verilmiş proqram yerinə yetirildikdən sonra onun nəticələri üçün başqa bir neçə şərt doğru olacaqdır".

İsbat, adətən, aşağıdakı mərhələləri özündə birləşdirir:

1. Proqramda bir neçə nəzarət nöqtəsi ayrılır, onların hər biri üçün verilmiş nöqtədə proqramın dəyişənlərinin vəziyyətini təsvir edən mühakimə formallaşdırılır. Bu mühakimələr isbat zamanı aralıq dayaq nöqtələri kimi istifadə olunurlar (riyaziyyatda analoji rolu *lemmalar* oynayır).

2. Proqramın məntiqinə əsaslanaraq, onun hansı nöqtədən hansı nöqtəyə keçidinin mümkünlüyünü, bu zaman proqramın dəyişənlərinin qiymətlərinin hansı şərt daxilində və necə dəyişdiyini göstərən blok-sxemi tərtib olunur.

3. Hər bir keçid üçün aşağıdakı müddəa isbat olunur: əgər giriş nöqtəsində yazılmış mühakimə yerinə yetirilirsə, onda keçidin son nöqtəsində yazılmış mühakimə də yerinə yetiriləcəkdir. Əgər bu bütün keçidlər üçün doğrudursa, proqramın yerinə yetirilməsi son nöqtəyə çatır və son mühakimə da doğrudur.

Sadalanan bəndlər göstərir ki, əgər proqram sona çatırsa, onun nəticələri doğru olacaqdır, lakin onlar zəmanət vermirlər ki, o, sona çatacaqdır. Ona görə də tam isbat üçün mütləq bir bənd də olmalıdır:

4. İsbat olur ki, gec-tez proqramın yekun nöqtəsinə nail olunacaq. Bu, həm də proqramın bitməsidir.

Misal 1. Qarpız kisələri vardır. Tərəzinin gözlərinin köməyilə (çəki daşlarısız) ən ağır qarpızı tapmaq lazımdır.

Bu məsələnin mümkün həlli şəkildə blok-sxem şəklində verilmişdir. Onu alqoritmik olaraq belə yazmaq olar:



(1) *ən ağır qarpız kisədədir*
db nə qədər ki kisədə birdən çox qarpız var
kisədən ixtiyari iki qarpızı tərəzinin gözlərinə qoymaq
(2) *ən ağır qarpız kisədə, yaxud tərəzidədir*
daha yüngül qarpızı kənara qoymaq
daha ağır qarpızı kisəyə qaytarmaq
ds
(3) *kisədə düz bir qarpız var – ən ağırı* kisədən ən ağır qarpızı götürmək.

Qeyd: fərz olunur ki, kisədə eyni çəkidə qarpızlar yoxdur.

Blok-sxemdə nəzarət nöqtələri balaca dairələrlə qeyd olunmuşdur, uyğun mühakimələr alqoritmdə şərhlər formasında yazılmışdır.

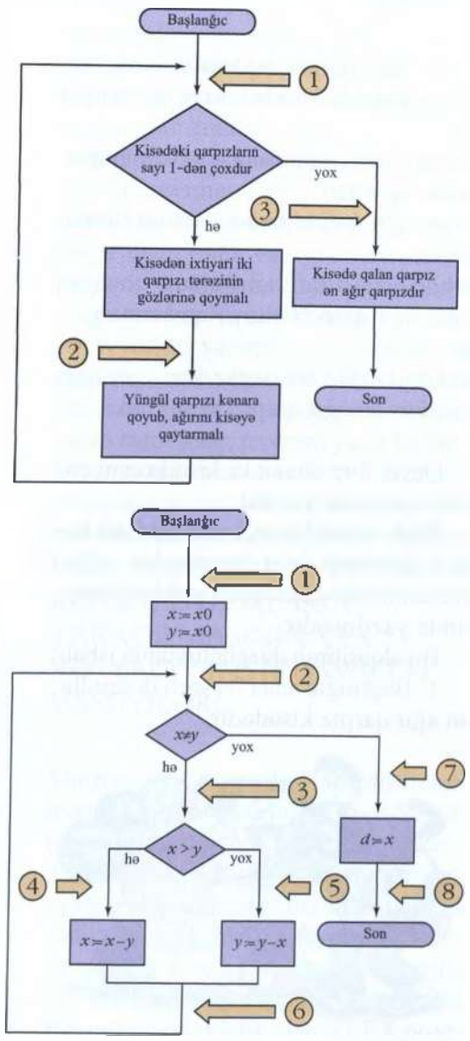
Bu alqoritmin düzgünlüyünün isbatı:

1. Başlanğıc anda (1) şərti doğrudur: ən ağır qarpız kisədədir.



2. Əgər (1) şərti doğrudursa, onda dövrə daxil olarkən (2) şərti doğrudur. Bu aydındır: bu nöqtələr arasında qarpızlar kisədən ancaq tərəziyə "çıxmışlar".

3. Əgər (2) şərti doğrudursa, onda dövrün növbəti keçidində (1) şərti doğrudur. Başqa sözlə, əgər (2) nöqtəsində ən ağır qarpız kisədə olmuşdursa, onda o, orada qalmışdır; (2) və (1) nöqtələri arasında kisədən heç nə çıxarmamışlar. Əgər (2) nöqtəsində ən ağır qarpız tərəzidə olsaydı, onda növbəti hərəkət yerinə yetirilərkən o, kisəyə qaytarılardı.



4. Əgər (1) şərti doğrudursa, onda dövrdən çıxdıqda (3) şərti doğru olacaq. (1) şərtinin yerinə yetirilməsi onu göstərir ki, kisə boş deyil. (3)-ə keçid ancaq o zaman mümkündür ki, dövrün şərti ödənməsin. Başqa sözlə, kisədə birdən çox qarpız yoxdur. Deməli, kisədə bir qarpız qalmışdır və (1) şərtinə uyğun olaraq o, ən ağır qarpızdır.

5. Alqoritmin yerinə yetirilməsi hökmən sona çatacaqdır, çünki dövrün hər bir icrası zamanı kisədə bir qarpız azalır.

Misal 2. Şəkildə iki tam ədədin ən böyük ortaq böləninin (ƏBOB) tapılması alqoritminin (Evklid alqoritmi) bloksxemi və Pascal dilində prosedur təsvir olunmuşdur.

Bu alqoritmin düzgünlüyünü aşağıdakı kimi isbat etmək olar:

1. Nəzarət nöqtələrini ayırır və onlara uyğun mühakimələri yazırıq.

2. Mümkün keçidlər, şərtlər və dəyişənlərin dəyişilməsi şəkildən görünür.

3. Nəzarət nöqtələrində bütün mühakimələr doğrudur.

$$(1)\quad x0 > 0,\ y0 > 0$$

Bu hal tərifə görə doğrudur.

$$(2)\quad x > 0,\ y > 0,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

Bu nöqtəyə keçid iki yerdən mümkündür: (1) və (6) nöqtələrindən. (1)-dən keçid proqramın işinin başlanğıcında baş verir. (6)-dan isə dövrün hər bir növbəti dönəmində. (2) mühakiməsinin doğruluğunu ayrılıqda hər bir keçid üçün isbat etmək lazımdır.

(1) və (2) nöqtələri arasında mənimsəmələr yerinə yetirilir: $x := x0$; $y := y0$. Bu mənimsəmələrdən sonra $x = x0$, $y=y0$. Əgər (1) ödənərsə, onda (2)-nin doğruluğu aşkardır.

(6) və (2) arasında ümumilikdə heç bir hərəkət baş vermir, mühakimələr isə bu nöqtələrdə üst-üstə düşür. Beləliklə, əgər (6) ödənirsə, onda (2) də ödənəcəkdir.

(2) mühakiməsi həmişə *dövrün invariantı* adlanan şərtinin yoxlanılmasından əvvəl yerinə yetirilir.

$$(3)\quad x>0,\ y>0,\ x<>y,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

(2)-dən (3)-ə keçərkən dəyişənlərin qiymətlərində heç bir dəyişiklik yoxdur, deməli (2) mühakiməsinə daxil olan bütün şərtlər doğru qalırlar. (3) nöqtəsində onlara dövrün başlığından şərt əlavə olunmuşdur – $x <> y$. Burada o, mütləq doğrudur, çünki, əks halda dövrün gövdəsi yerinə yetirilməzdi.

$$(4)\quad x > y > 0,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

(4) nöqtəsində $x > y$ şərti daxilində (3)-dən keçid baş verir. Əgər bu şərti (3) mühakiməsinə əlavə etsək, (4) mühakiməsi alınacaqdır.

$$(5)\quad y > x > 0,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

Əgər əvvəlki, $x > y$ şərti ödənmirsə, yəni, $y >= x$ olduqda (5) nöqtəsinə (3)-dən keçirik. $y = x$ halı istisna olunur, çünki, (3)-də iddia olunur ki, $x<>y$. Əgər $y = x$ olarsa, keçid mümkün olar.

$$(6)\quad x>0,\ y>0,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

(6) nöqtəsinə (4) və (5)-dən keçirik. (4)-dən keçid zamanı $x := x - y$ mənimsəməsi yerinə yetirilir. Fərz edək ki, x-in köhnə qiyməti (mənimsəməyə qədər) x_k, mənimsəmədən sonra isə x_y olmuşdur.

(4)-ə uyğun olaraq $x\_k > y$, ona görə də $x\_y = x\_k - y > 0$.

Tutaq ki, d x_k və y kəmiyyətlərinin ixtiyari ortaq bölənidir. Aydındır ki, $x\_k - y$ d-yə bölünəcək. Deməli d x_y və y-in ortaq bölənidir.

İndi isə tutaq ki, d x_y və y-in ixtiyari ortaq bölənidir. Aydındır ki, $x\_y + y$ cəmi d-yə bölünər. Deməli d x_k və y-in ortaq bölənidir.

Buradan o nəticə alınır ki, x_k ilə y-in ortaq bölənləri çoxluğu, x_y ilə y-in ortaq bölənləri çoxluğuyla üst-üstə düşür. Deməli, bu cütlərin ən böyük ortaq bölənləri də üst-üstə düşür, yəni:

$$ƏBOB(x\_y, y) = ƏBOB(x\_k, y).$$

(6) mühakiməsi isbat olundu.

$$(7)\quad x = y > 0,\ ƏBOB(x, y) = ƏBOB(x0, y0)$$

$x<>y$ şərti pozulduqda, başqa sözlə $x = y$ olduqda, (2)-dən (7)-yə keçid baş verir. (7) iddiası elə (2) iddiasına $x = y$ şərti əlavə olunduqdan sonra alınan iddiadır.

$$(8)\quad d = ƏBOB(x0, y0)$$

(7)-də iddia olunur ki, $x = y$. $ƏBOB(x, x) = x$, deməli $d := x$ mənimsəməsindən sonra (8) şərti ödənir.

4. Proqram mütləq sona çatacaqdır, çünki hər bir icra zamanı x ədədlərindən böyüyü azalır, ədədlər isə bu zaman müsbət qalmalıdırlar.



Evklid.

Həmişə dövrün şərtinin yoxlanılmasından əvvəl yerinə yetirilən mühakimə *dövrün invariantı* adlanır.

EVKLİD ALQORİTMİ PASCAL DİLİNDƏ

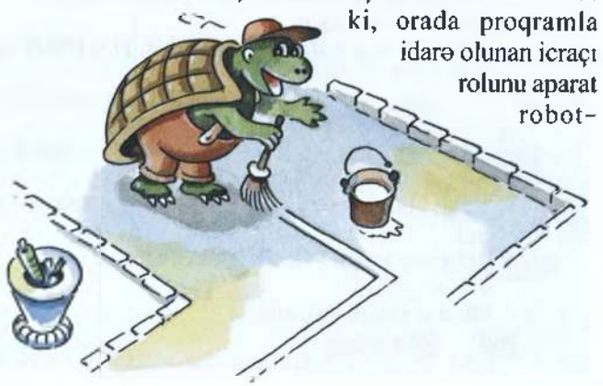
```
procedure Euclid (x0, y0: integer; var d: integer)
  var x, y: integer
begin
  {(1) x0 > 0, y0 > 0}
  x := x0; y := y0;
  {(2) x > 0, y > 0, ƏBOB (x, y) = ƏBOB(x0, y0)}
  while x<>y do begin
    {(3) x > 0, y > 0, x<>y, ƏBOB (x, y) = ƏBOB (x0, y0)}
    if x > y
      then {(4) x > y > 0, ƏBOB (x, y) = ƏBOB (x0, y0)}
        x := x - y
      else {(5) y > x > 0, ƏBOB (x, y) = ƏBOB (x0, y0)}
        y := y - x
      {(6) x > 0, y > 0, ƏBOB (x, y) = ƏBOB (x0, y0)}
  end
  {(7) x = y > 0, ƏBOB (x, y) = ƏBOB (x0, y0)}
  d := x;
  {(8) d = ƏBOB (x0, y0)}
end
```



PROQRAMLAŞDIRMA DİLLƏRİ

LOGO PROQRAMLAŞDIRMA DİLİ

LOGO proqramlaşdırma dilini pedaqogika, riyaziyyat, psixologiya və informatika sahələrində öz işlərilə məşhur olan, görkəmli amerika alimi Seymur Paypert 1969-cu ildə yaratmışdır. Dilin əsas ideyası icraçının kompüterin köməyilə idarə olunan fəaliyyətinin proqramlaşdırılmasından ibarətdir. Əslində, LOGO icraçının idarəetmə dilidir. LOGO proqramlaşdırma sisteminin çoxsaylı reallaşdırmaları içərisində elə də var ki, orada proqramla idarə olunan icraçı rolunu aparat robot-



ları oynayırlar. Lakin proqram-reallaşdırılmış icraçı Tısbağa daha çox şöhrət qazanmışdır. Tısbağa qaldırılmış, yaxud endirilmiş qələmlə müstəvi üzərində hərəkət edir və bu ona müstəvi üzərində iz, başqa sözlə hərəkətin çəkilmiş trayektoriyasını qoymağa imkan verir.

Tısbağa tərəfindən qəbul edilən və yerinəyetirilən komandalar sistemi, bir qayda olaraq, sadədir və çox deyildir. Bu komandalar sisteminə, xüsusən, istifadəçi tərəfindən verilmiş uzunluqda irəliyə və geriyə yerdəyişmə, göstərilmiş bucaq qədər dönmə, həmçinin çəkən alətlə əməliyyatlar – "qaldırmaq" və "endirmək" daxil edilmişdir.

LOGO balaca məktəblilərə proqramlaşdırmanın öyrədilməsi üçün nəzərdə tutulmuşdur. Bu dil ölməmişdir (bir çox süni dillər belə olmuşdur), artıq 40 ilə yaxındır ki, yaşayır və inkişaf edir. Bu faktın özü sübut edir ki, seçilmiş predmet sahəsi (uşaqlara erkən yaşda informatika-

nın öyrədilməsi) və dilin yaradıcılarının təklif etdikləri vasitələr bir-birlərilə yaxşı uzlaşırlar.

Tısbağa ilə idarəetmə dilində təkrarlanmalar və budaqlanmaların köməyilə yaradılmış idarəedici strukturlar qurulmuşdur. Tipik nümunə – LOGO dövrü:

```
TƏKRARLA 360 [İRƏLİ 1 SAĞA 1]
```

Bu dövr 360 parçadan ibarət çevrə çəkir,

```
ƏGƏR :A [SAĞA 90] [SOLA 90]
```

şərti komanda isə Tısbağanı düz bucaq altında cari :A dəyişəninin qiymətindən asılı olaraq sağa, yaxud sola dönməyə məcbur edir.

LOGO-da prosedurlar sadə, təbii şəkildə yazılır və istifadə olunurlar. Ona görə də, informatikanın fundamental anlayışlarından biri kimi prosedur ən birinci dərslərdə öyrədilir. Həqiqətən,

```
TƏKRARLA 4 [İRƏLİ 10 SAĞA 90]
```

dövrü asanlıqla KVADRAT prosedurunu təsvir etməyə imkan verir:

```
BU KVADRAT
  TƏKRARLA 4 [İRƏLİ 10 SAĞA 90]
SON
```

Burada, KVADRAT adlanan prosedurun birinci sətri yadda saxlama (yerinə yetirmə yox!) rejimini qoşur, başlığı təsvir edir. SON sözü isə tamamlayan təsvirdir və proqramı yadda saxlama rejimindən yerinə yetirmə rejiminə qaytarır. Bundan sonra prosedurun, sadəcə, çağırılması ilə, yəni onun adının verilməsilə kvadratı çəkmək olar.

KVADRAT



Məsələn, aşağıdakı prosedurun köməyilə kvadrat kərpicləri üst-üstə döşəməklə divarın qurulması prosedurunu (DİVAR) yazmaq olar (bu prosedur KVADRAT prosedurunu çağırır):

DİVAR



```
BU DİVAR
TƏKRARLA 4
 [
  KVADRAT
  QƏLƏMİ QALDIR
  İRƏLİ 10
  QƏLƏMİ ENDİR
 ]
SON
```

Prosedurların təsvirində bu qədər təbii formada da parametrlər daxil edilir: tərəflərinin uzunluğu müxtəlif olan kvadratlar çəkmək lazım gələrsə, onda bu məqsədlə proseduru parametrlə təsvir etmək faydalı olar (*:TƏRƏF* kəmiyyəti KVADRATCIQ prosedurunun parametridir):

```
BU KVADRATCIQ :TƏRƏF
 TƏKRARLA 4
  [
   İRƏLİ :TƏRƏF SAĞA 90
  ]
SON
```

İndi faktiki parametrin müxtəlif qiymətlərilə eyni KVADRATCIQ proseduru çağırmaqla müxtəlif ölçülü kvadratlar çəkmək olar:

```
KVADRATCIQ 10
KVADRATCIQ 20
KVADRATCIQ 30
```



Bir halda ki, bir prosedur digərini çağırmağa qadirdirsə və prosedurlar bir-birlərinə daxil olaraq zəncir əmələ gəti-



Seymur Paypert.

rirlərsə, onda belə bir xüsusi halı da təsəvvür etmək olar.

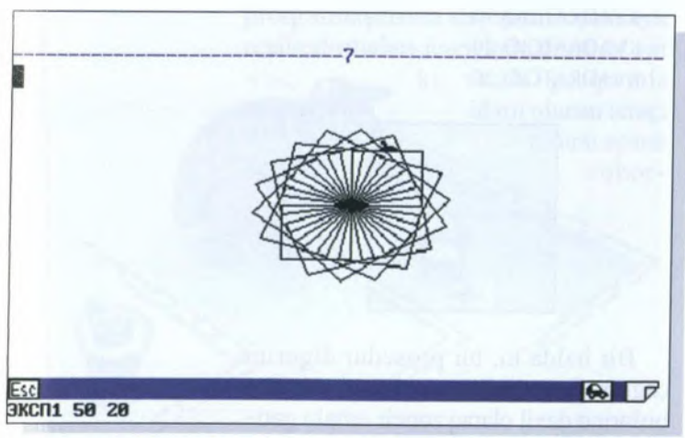
Çağıran və çağırılan prosedurun adları üst-üstə düşür, başqa sözlə prosedur özü-özünü çağırır. Proseduru, eyni adla çağıran (bilavasitə, yaxud bir-birinə daxil olan çağırışlar zəncirilə) proseduru *rekursiv*, uyğun çağırışı *rekursiv çağırış*, yaxud qısaca olaraq *rekursiya* adlandırırlar. LOGO-da belə çağırışlara icazə verilir (elə dillər vardır ki, onlarda *rekursiv çağırışlara* prinsipcə icazə yoxdur). Bu *rekursiv* prosedur Tısbağanı eyni bir yolla dayanmadan hərəkət etməyə (kvadrat çəkməyə) məcbur edəcəkdir:

```
BU KVADRATCIQ :TƏRƏF
  TƏKRARLA 4
    [
     İRƏLİ :TƏRƏF SAĞA 90
    ]
  KVADRATCIQ :TƏRƏF
SON
```

Çağırış parametrinin qiymətinin (faktiki parametrin) bir rekursiv çağırışdan digərinə keçdikcə dəyişməsi daha maraqlı rekursiv çağırışlardır. Belə prosedura misal:

```
BU NAXIŞ :A
  TƏKRARLA 8
    [İRƏLİ :A SAĞA 45]
  NAXIŞ :A + 2
SON
```

50 addım və
20 bucaqlı spiral.



və onunla çəkilmiş şəkil:

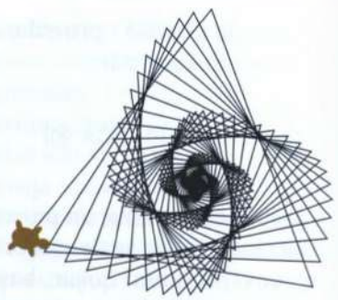
NAXIŞ 10

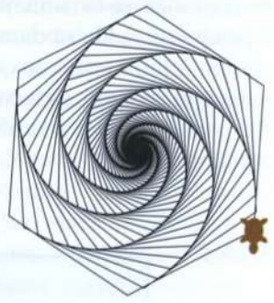


Spiral çəkən rekursiv prosedurun iki parametri vardır – *ADDIM* və *BUCAQ*:

```
BU SPİRAL :ADDIM :BUCAQ
  İRƏLİ :ADDIM SAĞA  :BUCAQ
  SPİRAL (:ADDIM + 5) :BUCAQ
SON
```

Bucağın müxtəlif qiymətlərinə uyğun spirallara bir neçə nümunə:

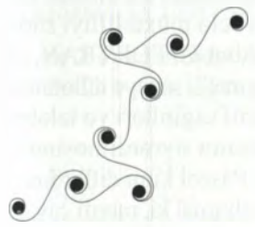




Mürəkkəb olmayan, ancaq daha çox cəlbedici başqa bir misal. Burada hər bir rekursiv çağırışda yerdəyişmənin uzunluğu deyil, dönmə bucağı dəyişir:


```
BU SPİRAL :ADDIM :BUCAQ
  İRƏLİ :ADDIM SAĞA  :BUCAQ
  SPİRAL :ADDIM ( :BUCAQ + 7)
SON
```

Bu prosedurun yerinə yetirilməsinin nəticəsi:



İllər ərzində inkişaf edən Paypertin Tısbağası və onu idarə edən LOGO dili çoxlu yeni imkanlar əldə etmişdir. Əvvəlcə rənglərin idarə olunması vasitəsi meydana gəlmişdir. Bir qədər sonra Tısbağa melodiyalar səsləndiməyi "öyrənmişdir". Bir proqramda eyni zamanda bir neçə icraçının idarə olunması imkanı yarandıqdan sonra LOGO-da işləmək daha maraqlı olmuşdur.

LOGO dili öz alətlər dəsti içərisində təkcə Tısbağanın yeni imkanlarına xidmət üçün vasitələri əks etdirməmişdir. O, proqramlaşdırılan icraçıya yeni "bacarıqlar" öyrətməyə imkan verərək avtonom inkişaf etmişdir.

Ədədi informasiya ilə işləyən zaman hesablama imkanlarının və mətn tipli informasiyanın emalının geniş vasitələrinin varlığı sayəsində LOGO dili təkcə qrafika ilə işləməyə deyil, başqa vasitələrlə də informasiyanın təqdimatına imkan verir.

LOGO siyahılar kimi verilənlər strukturunun emalı üçün də effektiv üsullar təklif edir. Çox önəmlidir ki, rekursivlik ideyası idareedici strukturlarda olduğu kimi ardıcıl olaraq verilənlər strukturunda da tətbiq edilir. Bax, nümunə üçün "DAYAN" sözünün ardıcıl olaraq KƏSILMƏ-si LOGO-da necə proqramlaşdırılmışdır:

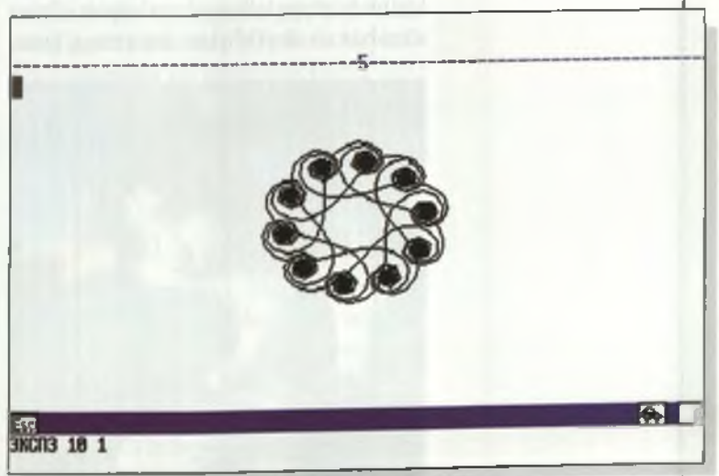
```
BU KƏSİLMƏ :SÖZ
  GÖSTƏR BİRİNCİSİZ :SÖZ
  KƏSİLMƏ BİRİNCİSİZ :SÖZ
SON
```

Ekrana belə sözlər sütunu çıxarılacaqdır:

```
AYAN
YAN
AN
N
```

LOGO-da "tədris" və istehsalat dillərinin imkanları uğurla uzlaşır. Bu dilin müxtəlif tətbiq sahələrinin maraqlı inteqrasiyasını fransız şəhəri Le-Manedə professor M.Vive tərəfindən yaradılmış elmi mərkəz nümayiş etdirmişdir. Burada eyni zamanda ibtidai məktəbdə proqramlaşdırmanın öyrədilməsi; informatika üzrə ixtisaslaşan universitet tələbələrinin hazırlanması; regionun ədədi proqramlarla idarə olunan dəzgahlardan geniş istifadə olunan sənaye müəssisələrində ixtisaslaşmış fəhlələrin təkmilləşdirilməsi; informasiya texnologiyalarının insanların düşüncə və yaradıcılığına təsirini öyrənən psixoloqların tədqiqatları keçirilib. LOGO müxtəlif səviyyəli və müxtəlif yaradıcılıq sahəsindən olan istifadəçilər üçün effektiv alət rolunu oynayır.

10 addım və
1 bucaqlı spiral.



LOGO təkcə dilin və Tısbağanın funksiyalarının artması hesabına inkişaf etməmişdir. Onun dünyada geniş yayılması birinci növbədə müxtəlif çeşiddə leksik versiyalarının olması ilə əlaqədardır. Praktiki olaraq hər bir inkişaf etmiş ölkədə bu dilin milli versiyası istifadə olunur. Fransız dilinin əsasında, həmçinin elə ingilis (LOGO-nun birinci versiyası leksikada yaradılmışdır) dilinin əsasında da latın dilindən istifadə olunmuşdur. Buna baxmayaraq, onlar gənc fransız məktəblilərinə LOGO-nun fransız açar sözlərini başa düşən Tısbağa haqqında danışmağa üstünlük verirlər.

Milli dillərində kiril əlifbasından istifadə olunan ölkələrdə də oxşar vəziyyətdir. Slavyan dillərinin yaxınlığına baxmayaraq, LOGO rus, ukrayna və bolqar versiyalarında mövcuddur. Bundan əlavə rus məktəblərində LOGO dilinin öyrədilməsində tez-tez belə tapşırıqlar qoyulur: ingilis dilindəki komandaların köməyilə Tısbağanın hərəkətlərini proqramlaşdırmalı. Uşaqlar da bunu asanlıqla edirlər:

BU FORWARD :*ADDIM*	BU LEFT :*BUCAQ*
İRƏLİ :*ADDIM*	SOLA :*BUCAQ*
SON	SON

Proqramlaşdırma dilində leksik versiyaların belə müxtəlifliyi zəruridirmi? Əgər söhbət C, FORTRAN, Ada kimi "professional", sənaye dillərindən, yaxud yuxarı sinif şagirdləri və tələbələrin informatikanın öyrənilməsində istifadə etdikləri Pascal kimi dillərdən getsəydi, bu suala ehtimal ki, mənfi cavab alınardı. Lakin, heç də sadə olmayan fundamental anlayışlarla dolu olan proqramlaşdırma fənninin əlifbasını, ilk anlayışlarını mənimsəmək istəyən azyaşlı məktəblilərin əlavə metodiki maneələrdən azad olunması çox vacibdir. Şübhəsiz, özgə, xarici dilin leksikası da maneə hesab olunmalıdır. Bu baxımdan LOGO dilinin müxtəlif versiyalarına ehtiyac var.

PROQRAMLAŞDIRMA DİLLƏRİNİN TARİXİ

Ev heyvanlarının öyrədilməsinin (əhliləşdirilməsinin) ən geniş yayılmış üsulu onlara "oturmaq", "uzanmaq" və sair kimi məhdud komandalar yığınının öyrədilməsidir. Lakin bu üsul pişiklər fəsiləsinə həmişə tətbiq oluna bilmir. Onlar sözəbaxan deyildirlər, bu azmış kimi, komandaları da yaxşı başa düşmürlər. Bəzi gələcəyi görən sahiblər inadcıl yetişdirmələrilə onların öz dilində danışmaqla (fışıldamaqla, miyoldamaqla və s.) praktiki olaraq tam şəkildə qarşılıqlı anlaşma əldə edirlər.



Kompüterin mərkəzi qurğusu olan prosessorun da özünün xüsusi dili vardır və kompüterin proqramçını "başa düşməsini" təmin etməklə olduqca mürəkkəb məsələdir.

Prosessor üçün komandaları necəsə kodlaşdırmaq lazımdır. Məsələn, ədədi formada təsvir etmək olar: 1 – toplamanı, 2 – vurmanı, 3 – bölməni və sair işarə edə bilər. Komandalardan əlavə prosessora bu və ya digər əməliyyatın yerinə yetirilməsi üçün zəruri olan verilənlər təqdim olunur. Tutaq ki, hər hansı xəyali prosessorun komandalarının yazılışı sadəlik üçün belə formadadır:

komandanın kodu	birinci operand	ikinci operand	nəticə üçün oyuq

Komandalar üçün verilənlər, yəni operandlar isə belə yazılır:

00 xx - xx ədədi;
01 xx - yaddaşın xx nömrəli oyuğu.

Onda 1 və 2 oyuqlarında yerləşmiş iki ədədin ədədi ortasını tapan proqram belə olar:

01 03	0101 0103	0102 0002	0103 0103

Yaxud, əgər onu bir sətirdə yazsaq belə olar:

```
01 01 01 01 02 01 03 03 01 03 00
02 01 03
```

Hətta belə sadə proqram üçün *maşın kodu* müəmmalı şifrlərdən ibarətdir. Bu kodları başa düşmək üçün Şerlok Holmsın bacarıqlarına malik olmaq lazımdır. Müasir, real kompüterlərdə maşın kodu xeyli mürəkkəbdir. Belə kodlarla proqramlaşdırmaq və proqramı yoxlamaq çox çətin idi və əlverişli deyildi, çünki bütün əməliyyatlar üçün ya onların kodlarını və formatlarını yadda saxlamaq, ya da həmişə xüsusi cədvəllərdən istifadə etmək lazım gəlirdi. Kodlaşdırma zamanı xırdaca diqqətsizlik, yazılarda səhv, qarışıq salınmış ədədlər qabaqcadan bilinməyən nəticələrə səbəb olurdu. Belə səhvlərin tapılması isə çox çətin olurdu, çünki proqramçının qarşısında bu və ya digər dərəcədə aydın dildə təsvir olunmuş alqoritm deyil, rəqəmlər yığını dayanırdı.

Proqramçılar öz yaşamlarını yüngülləşdirmək üçün, xırda detallara vaxt itirmədən və kodlaşdırmanın özünə fikir vermədən proqramlar yazmağa imkan verən alətlər yaratmaq üsullarını axtarırdılar. Onlar yorucu və ağır işləri, belə işləri hamıdan yaxşı bacaran kompüterə həvalə etmək, özləri isə insanın daha



yaxşı bacardığı yaradıcı işlərlə, alqoritmlərin tərtibilə məşğul olmaq istəyirdilər.

PROQRAMLARIN MNEMONİK YAZILIŞINA KEÇİD. ASSEMBLER DİLİ

İnkişafın norması sadədən mürəkkəbdir. Proqramlaşdırmada komandaların simvollarla işarə olunması sisteminin yaradılması birinci addım olmuşdur. Komandalar əməliyyatları göstərən adi sözlərin qısaltmaları ilə təsvir olunurdular. İdeyanın sadəliyinə baxmayaraq, onun reallaşdırılması mövcud proqramların qavranmasının güclü şəkildə yaxşılaşdırılmasına, səhvlərin sayının azaldılmasına və proqramçıların işini yüngülləşdirməyə imkan vermişdir.

Simvollarla işarələmənin köməyilə yazılmış proqramlara nümunə:

```
TOP     YN№1, YN№2   > YN№3
BÖL     YN№3, 2      > YN№3
```

Bu onu göstərir ki, Yaddaşın №1 və №2 oyuqlarını TOPla, nəticəni Yaddaşın №3 oyuğuna yerləşdir. Sonra Yaddaşın №3 oyuğunu 2-yə BÖLərək nəticəni yenə Yaddaşın №3 oyuğuna yerləşdir.

Komandaların yazılışının belə simvollar sistemi *mnemonik* (*yun.* "mneme"





– "yaddaş"), yəni "yadda saxlamaq üçün asan" adlanır.

Lakin təkcə mnemonik yazılış proqramların effektiv yaradılması üçün azdır. Proqramçılar həmişə "velosiped ixtira etmək" – eyni bir hərəkət üçün kod yazmalı olurdular. Bu və ya digər alt məsələni həll edən proqram kodu hissələrini, əlbəttə, sonradan istifadə üçün saxlamaq olar. Lakin belə saxlanılmış kodun qiyməti böyük deyildir, çünki yeni proqramların yaradılması zamanı dərhal onlardan istifadə etmək olmaz. Maşın kodunun hissələri təkcə konkret prosessora, kompüterə deyil, eləcə də onların yazılmış olduğu, başqa sözlə hissəsi olduğu proqrama da "bağlanmış" olurdu. Maşın kodunda hesablamaların nəticələrini saxlamaq üçün yaddaşın oyuqlarının ünvanları və hər hansı hadisənin baş verməsi zamanı idarəetmənin ötürüldüyü ünvan aşkar şəkildə verilir. Yeni proqramda isə yaddaşın bu oyuqları tamamilə başqa cür istifadə oluna bilər. Qəflətən bütün küçələrinin adları dəyişdirilmiş şəhərdə sürücü özünü necə hiss edərsə, köhnə maşın kodu da özünü yeni proqramda elə "hiss edərdi". Yol vərəqi sürücünü şəhərin tamamilə başqa bir hissəsinə gətirərdi.

İdeyanı, alqoritmi və ümumi həll prinsipini saxlamaq olardı. Maşın kodunun özünü isə hər dəfə yenidən yazmaq (yaxud düzəlişlər aparmaq) lazım gələrdi.

Məsələnin həlli üçün yaddaşın oyuqlarının ünvanlarını və kodun hissələrini adlandırma sistemi fikirləşdilər. Yaddaşın hər bir oyuğuna özünün bənzərsiz adı mənimsədilirdi, sonra isə proqramda artıq yaddaşın oyuqlarının nömrəsi deyil, uyğun adlardan istifadə olunurdu:

```
AD  "X"              YN№1
AD  "Y"              YN№2
AD  "NƏTİCƏ"         YN№3

TOP   X, Y       >   NƏTİCƏ
BÖL   NƏTİCƏ, 2  >   NƏTİCƏ
```

Beləliklə, ayrı-ayrı alt məsələlərin həlli üçün, kodun hissəsinin bir tam kimi ayrılması mümkün oldu. Bu balaca proqramlar *altproqramlar*, yaxud *prosedurlar* adlanırlar. Prosedurda onun işi üçün zəruri olan, ciddi təyin olunmuş başlanğıc verilənlər yığını və onun işinin nəticələri olan çıxış verilənləri yığını vardır.

Rahatlıq üçün prosedurları *prosedurlar kitabxanasında* birləşdirirlər. Adətən, eyni bir kitabxana çərçivəsində olan prosedurlar eyni, yaxud oxşar məsələlərin həlli üçün tətbiq olunur, oxşar qaydalara malik olur və birgə istifadə üçün təyin edilirlər. Cəbri tənliklərin həlli üçün, səs və videonun emalı üçün və hətta digər kitabxanalarla işləmək üçün prosedurlar kitabxanası mövcud ola bilər.

Ancaq təkcə proqramların gözəl yazılışı sistemini fikirləşmək yetərli deyil. Belə proqramları kompüter yerinə yetirə bilməz, ona maşın kodları lazımdır. Mnemonik yazılışı koda çevirmə prosedurunu sadə, addım-addım təlimatlar şəklində təsvir etmək asandır. Lakin, əgər onu insan yerinə yetirəcəksə, onda yeni sistemin üstünlükləri əvəzinə böyük sayda səhvlər əmələ gələcək və bu çox vaxt aparacaqdır. Bunun kimi darıxdırıcı, sırf mexaniki, yorucu və sadəcə maraqsız, eyni zamanda məsuliyyətli işi (cədvələ görə bir işarələr – operatorlar yığınının əvəzinə digər işarələr – əməliyyatların yığınını qoymaq) kompüterə həvalə etmək lazımdır.

1952-ci ildə amerikalı qadın Qreys M. Hopper dünyada ilk mnemonik proqramlaşdırma dili olan Assembler dilini yaratdı. Onun adı "assemble" ingilis sözündən götürülmüşdür, "yığmaq", "quraşdırmaq" mənasını verir. O, özündə mnemonik komandalar sistemini (komandaların siyahısı), prosedurlar kitabxanasını və proqram mətnlərini maşın koduna çevirmək üçün xüsusi proqramı birləşdirirdi. Maşın kodunun alınmasının belə proseduru *kompilyasiya* (*ing.* "compile" – "tərtib etmək", "yığmaq"), onu həyata keçirən proqram isə *kompilyator* adlanır. Bunları da Qreys M.Hopper düşünüb tapmışdır.

Bu, proqramlaşdırma dilinin həm insana, həm də kompüterə aydın olan xüsusi dil kimi yaradılması sahəsində birinci addım olmuşdur. O vaxta qədər ancaq bir dil – kompüterlərin dili mövcud olmuşdur, insan isə daha ağıllı varlıq kimi, sadəcə, həmin dildə "danışmışdır".

İkinci addım – prosedurlar kitabxanasının yaradılması, yəni kodun təkrarən istifadə olunmasıdır. Buna qədər, bir az şişirtsək, hər bir proqramçı hər dəfə hazır dərsliklərdən istifadə etməklə, təməl biliklərə əsaslanaraq yeni nə isə ixtira etmək əvəzinə öz cəbrini, həndəsəsini, kimyasını "fikirləşməli" olurdu. Prosedurlar kitabxanası proqramçılara yeni mürəkkəb proqramları sürətlə və keyfiyyətlə yaratmağa imkan verdi.

Bu iki istiqamət yeni dillərin və proqramlaşdırma sistemlərinin inkişafı üçün əsas rol oynadı.

DÜSTURLAR, DÜSTURLAR, DÜSTURLAR... FORTRAN

Assembler dili tətbiqi proqramlar və əməliyyat sistemləri yaradan proqramçılar üçün yaxşı dil idi. Lakin onun mnemonikliyinə baxmayaraq, o, dövrün əsas kompüter istifadəçiləri olan, mürəkkəb riyazi hesablamalar aparan alimlər üçün yaramırdı.

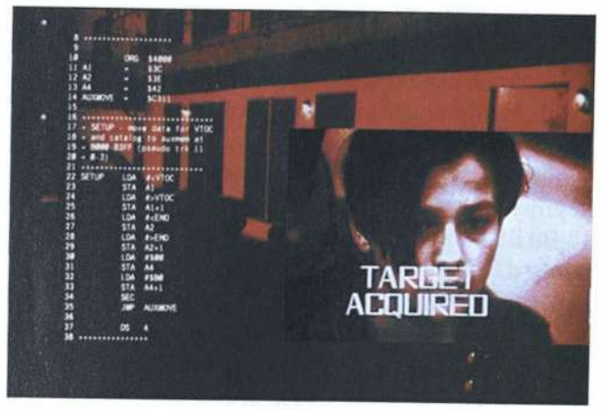
Assembler dili maşın koduna yaxın olduğuna görə onu öyrənmək çətindir. Bundan əlavə, o, müəyyən prosessor üçün nəzərdə tutulduğundan, bir neçə kompüterdə işləyən insan bəzən çox fərqli olan müxtəlif dilləri (müxtəlif Assemblerlər) bilmək məcburiyyətində qalırdı. Nəhayət, Assembler dili maşın kodunun bir qədər "insanlaşdırılmış" yazısı idi. Bu dildə proqramlaşdırmaq üçün bütün hərəkətləri sadə kərpiciklər şəklində təqdim etmək, mürəkkəb düsturları ayrı-ayrı əməliyyatlar ardıcıllığına parçalamaq lazımdır. Proqramçı beynində daha ümumi kateqoriyalarla əməliyyat aparır: düsturu hesablamaq, ədədi ekrana çıxarmaq, əməliyyatı on dəfə təkrar etmək və sair.

Bu problemlərin həlli üçün, ixtira olunması zamanı yeni üsul (dilin qabaqcadan hazırlanmış spesifikasiyalara və təsvirlərə görə yaradılması) tətbiq olunmuş proqramlaşdırma dili təklif olundu.

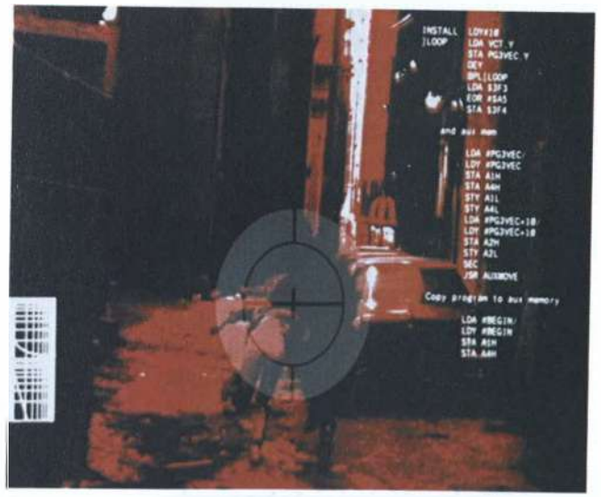
IBM firmasının əməkdaşları yeni proqramlaşdırma dilini ətraflı təsvir edən sənədlər və ona tələblər hazırladılar.



Qreys Hopper.



"Terminator" və "Terminator-2" filmlərində Assemblerlə proqramlaşdırılmış robot-qatil.



Assembler dilindən fərqli olaraq, o, maşın kodu əsasında deyil, kompüterə aydın olan dil kimi deyil, insana əlverişli dil kimi yaradılmışdır.

1956-cı ildə Con Bəkusun rəhbərlik etdiyi proqramçılar qrupu tərəfindən yeni, insana daha yaxın proqramlaşdırma dili üçün ilk kompilyator yaradıldı. O, FORTRAN adlandırıldı (*ing.* "formula translation" – "düsturun çevrilməsi").



Con Bəkus.

Yeni proqramlaşdırma dili universal oldu, yəni eyni bir proqramı müxtəlif kompüterlərdə işlətmək reallığa çevrildi (təbii ki, əvvəlcədən kompilyasiya etməklə). O, düsturları insana aydın şəkildə yazmağa imkan verdi. Məsələn, ədədi ortanın hesablanması aşağıdakı kimi görünür:

```
RES = (A + B) / 2
```

Artıq belə yazılış adi riyazi yazılışa yaxındır və kompüter üçün proqram yazan insan heç olmazsa, düsturları yazmağı öyrənməli olmur. Hesablamaların vahid düsturlar şəklində məntiqi göstərilməsindən əlavə FORTRAN zəngin riyazi funksiyalar kitabxanası da təqdim etdi. Məsələn, kvadrat kökün hesablanması üçün xüsusi SQRT prosedurunu çağırmaq kifayətdir:

```
RES = SQRT (169)
```

Bu prosedur mürəkkəb düstura operandlardan biri kimi də daxil ola bilər:

```
X1 = (-B - SQRT(B*B - 4*A*C)) / (2*A)
```

Assembler dilində kvadrat tənliyin köklərinin hesablanması çoxlu maşın kodu sətirləri tutardı. Kvadrat kökün hesablanması əməliyyatını da proqramlaşdırmaq, yaxud heç olmasa, prosedurlar kitabxanasından götürmək lazım gələrdi. Yalnız dörd hesab əməlindən istifadə etməklə kvadrat kökün hesablanması üçün alqoritm fikirləşmək çətindir.

PROSEDURLU PROQRAMLAŞDIRMA

Altproqramlar, yaxud prosedurlar "kitabxanası" termini hələ Assembler dilinin yaradılması zamanı meydana gəlmişdir. Kodun tam vahid kimi tərtib olunmuş, ayrılmış hissələri altproqramlardır. Proqramçı verilənlərin altproqrama necə ötürüldüyünü (məsələn, operativ yaddaşın oyuqları vasitəsilə), altproqramın kimin onu çağırdığını necə yadda saxladığını və altproqramın idarəetməni geriyə necə ötürdüyünü təyin müəyyənləşdirməli idi.

Altproqramların çağırılması haqqında olan bu razılaşmalar müxtəlif kitabxanalarda müxtəlif idi. Bundan başqa, bu *razılaşmalar* eyni kitabxana daxilindəki müxtəlif altproqramlar üçün də müxtəlif ola bilərdi.

Proqramçıların effektiv işinin təşkili üçün altproqramların yazılması və istifadəsi prosesinin standartlaşdırılması və avtomatlaşdırılması tələb olunurdu. Bu, FORTRAN dilində həyata keçirildi. Başqa sözlə, altproqramlar, altproqram kitabxanaları artıq nə isə xarici, dilə "əlavə tikili" deyildi və birbaşa onun özündə yaradılmışdı.

Prosedurun yerinə yetirilməsindən qabaq onun işləyəcəyi verilənləri, yəni *giriş verilənlərini* ona ötürmək lazımdır. Prosedurun nəticəsi *çıxış verilənləri* adlanır. Əgər bu kvadrat tənliyin köklərini hesablayan prosedurdursa, onda onun üçün giriş verilənləri məchulun qüvvətləri qarşısındakı üç əmsal, çıxış verilənləri isə axtarılan iki kök olacaqdır (ola bilər ki, kök bir dənə olsun, köklər olmasın, yaxud istənilən ədəd tənliyi ödəmiş olsun. Onda prosedurun çıxışında bir parametrin – axtarılan köklərin *sayının* da olması lazımdır).

Prosedur giriş, yaxud çıxış parametrlərinə, bəzən isə onların heç birinə malik olmaya bilər. Cari zamanın alınması prosedurunun giriş verilənləri yoxdur, o, vaxtı kompüterin "içərisindən" götürür. Ədədi çap edən prosedurda aşkar çıxış verilənləri yoxdur, onun işinin nəticəsi ekranda, yaxud printerdən çıxan kağızda olan təsvirdir. Sistemin bütün sərt disklərini formatlaşdıran prosedur nə giriş, nə də çıxış parametrlərinə malik deyildir. Onlar ona lazım deyildir, prosedur onsuz da nə etmək lazım olduğunu bilir və bayıra verməli verilənləri də yoxdur, hər şey göz qabağındadır.

FORTRAN dilində prosedur təsvir olunan zaman ona ad verilir. Prosedur məhz həmin ad vasitəsilə çağırılacaqdır. Elə bu sətirdə də giriş və çıxış parametrləri təsvir olunurlar. Hər bir verilənlər elementinə ad mənimsədilir ki, prosedurun daxilində bu addan istifadə olunacaqdır:



```
SUBROUTINE SQ (A, B, C, X1, X2)
```

Verilmiş misalda SQ adlı prosedur təsvir olunmuşdur. Onun üç giriş parametri – A, B, C və iki çıxış parametri – X1, X2 vardır. Nələrin giriş, nələrin isə çıxış parametrləri olduğunu proseduru yazan proqramçı müəyyən edir və onları proqramın kodunda olan, lakin kompilyator tərəfindən fikir verilməyən şərhlərdə təsvir edir.

```
SUBROUTINE SQ (A, B, C, X1, X2)
C  Kvadrat tənliyin köklərini
C  hesablayan prosedur
C  Giriş verilənləri: A, B, C –
C  məchullar qarşısındakı əmsallar
C  Çıxış verilənləri: X1, X2 – kvadrat
C  tənliyin kökləri
C  Diqqət: yalnız
C  tənliyin həlli olduqda işləyir!
D = SQRT (B*B - 4. *A*C)
X1 = (-B-D) / (2.*A)
X2 = (-B+D) / (2.*A)
RETURN
END
```

Prosedurun çağırılması zamanı ancaq parametrlərin düzgün göstərilməsi lazımdır, tənliyin həlli alınacaqdır:

```
CALL SQ (1, -3, 2, RES1, RES2)
PRINT RES1, RES2
```

Maşın $x^2 - 3x + 2 = 0$ tənliyinin gözlənilən 1 və 2 həllərini çap edir. Pro-



sedurların dəqiq ayrılması proqramlaşdırmanın yeni modelinin meydana gəlməsinə səbəb oldu. Böyük məsələ çoxlu sayda daha kiçik sadə altməsələlərə bölünür.

Proqramın ümumi kodu detallarla yüklənməmişdir. Müəllif həmişə, bütövlükdə proqramın (yəqin ki, bu nəzəriyyə raketin uçuşu üçün tətbiq oluna bilməz?) yoxsa ayrıca fraqmentin alqoritmini araşdırmaq barədə qərar qəbul edə bilər (nəyə görə o partlamışdır: mən yoxlamadım, kvadrat tənliyi həll edərkən diskriminant müsbətdirmi!).

Proqramlaşdırmanın belə üsulu *prosedurlu proqramlaşdırma* adlanır. Böyük proqramı kiçik kərpicciklərə – prosedurlara bölmə onun əsasıdır. Prosedurlu proqramlaşdırma texnologiyasından istifadə etməklə proqram iki üsulla yaradıla bilər: yuxarıdan aşağıya və aşağıdan yuxarıya.

Birinci halda əvvəlcə proqramın ümumi həll alqoritmi yaradılır; hansı köməkçi funksiyaların (prosedurların) lazım gələcəyi müəyyən olunur; həqiqi prosedurların işini imitasiya edən, xüsusi boş "boğucu-prosedurlar" yazılır; məsələnin ümumi alqoritmi sazlanır, sonra köməkçi funksiyalar yazılır.

İkinci halda əvvəlcə köməkçi prosedurlar yığını yaradılır və onlar böyük olmayan, xüsusi proqramlar vasitəsilə sazlanır. Məsələn, kvadrat tənliyin həll proseduru üçün bu klaviaturadan əmsalları daxil edən və həlli ekrana çıxaran proqram ola bilər. Proqramçı prosedurlarının düzgün işlədiyinə arxayın olduqdan sonra onlardan əsas proqramı yığmağa başlayır.

STRUKTURLU PROQRAMLAŞDIRMA. ALGOL DİLİ

Növbəti proqramlaşdırma dilinin yaradılmasına elm sahələrinin (kibernetika, alqoritmlər nəzəriyyəsi və s.) alimləri girişmək qərarına gəldilər. İngilis dilli ədəbiyyatlarda bu elm sahələrini, adətən, "kompüterlər haqqında elm" ("Computer Science") adı altında birləşdirirlər.

1960-cı ildə elmi konfransların hesabatları əsasında yeni Algol-60 dilini təsvir edən ümumiləşmiş sənəd yaradıldı. Onun adı ingilis *algorithmic language* ("alqoritmik dil") söz birləşməsindən əmələ gəlmişdir.

Dəyişənlər üçün, həmin dəyişənlərin istifadə oluna biləcəyi proqram hissələri dəqiq müəyyənləşdirildi. Proqramın belə hissələri *görünmə sahələri* adlanır. Dəyişənlər haqqında isə deyirlər ki, o, verilmiş proqram sahəsində *görünür*, yaxud *görünmür*. Funksiyanın, prosedurun, yaxud xüsusi məntiqi blokun daxilində elan olunmuş dəyişən ancaq verilmiş konstruksiyanın daxilində görünəcəkdir.

Məsələn, əgər A funksiyasında *tmp* dəyişəni varsa və bu funksiya onda da *tmp* dəyişəni olan B funksiyasını çağırırsa, bu zaman B funksiyasında *tmp* üzərində aparılan əməliyyat A funksiyasında *tmp* dəyişəninin qiymətini dəyişdirməyəcəkdir. A funksiyasındakı *tmp* dəyişəni B funksiyasında görünmür.

Dəyişənlərin istifadəsindən qabaq onların tipinin göstərilməsi, başqa sözlə *elan edilməsi* məcburi oldu. Belə məhdudiyyətin daxil edilməsilə aşağıda göstərilən səhvə oxşar səhvlər qeyri-mümkün oldu:

```
REAL I1
.... (kodun bir neçə səhifəsi)
I1 = 3. /2.
```

Burada, I1 dəyişənini istifadə edən proqramçı güman edir ki, FORTRAN standartının təsvirinə görə I hərfilə başlayan dəyişən tam tipə (tam dəyişənər *I, J, K, L, M, N* hərflərilə başlayırlar, qalanları kəsrlərdir) malik olur. Lakin o, diqqət etməmişdir ki, tipin belə göstərilməsi məcburi deyildir və kodun əvvəlində bu dəyişən aşkar şəkildə elan olunmuşdur. Nəticədə I1 dəyişənin qiyməti gözlənilən 1 deyil, 1.5 olur.

Dəyişənlədən daha ağıllı istifadə ilə əlaqəli "ümumi sağlamlaşdırıcı effekt"-dən savayı, görünmə sahələrinin daxil edilməsi funksiyanın özündən özünü çağırması imkanını dilə əlavə etməyə şərait yaratmışdır. Məsələn, ədədin faktorialının hesablanmasını məhz funksiyanın iç-içə çağırışlarının köməyilə yerinə yetirmək olar:

$$N! = 1 \cdot 2 \cdot 3 \cdot \ldots \cdot (N-1) \cdot N$$

```
alq tam faktorial (tam N)
  verilib N
  gərəkdir
baş
  əgər N <= 1
    onda qiym := 1
    əks halda qiym := N* faktorial (N-1)
  tamam
son
```



İç-içə belə funksiyaların çağırılması *rekursiya*, özü özünü çağıran funksiya isə *rekursiv* adlanır.

Algol dilində strukturlu idarəedici operatorlar dəqiq təsvir olunmuşdur: seçim operatorları və təkrarlanma (dövr) operatorları. Seçim operatorları hesablanan şərtdən asılı olaraq kodun bu və ya digər hissəsini yerinə yetirməyə imkan verir. Dövr operatorları kodun müəyyən hissəsinin verilmiş sayda, yaxud müəyyən şərt yerinə yetirilənə qədər təkrarlanması üçün nəzərdə tutulmuşdur.

Əlbəttə ki, idarəedici operatorlar Assembler və FORTRAN dillərində də mövcud olmuşdur. Lakin proqramlar yazılarkən onlar əsas elementlər olmamış və istifadə üçün də rahat olmamışlar.

Səkkiz il sonra, 1968-ci ildə Algol-60 dilinin yenidən işlənmiş, müəyyən əlavələr edilmiş variantı hazırlandı və gözlənilən Algol-68 adını daşımağa başladı. 1969-cu ildə Edsger B.Deykstranın strukturlu proqramlaşdırma haqqında çap olunmuş məqaləsi də bu standartla əlaqəli olmuşdur.

O, isbat etmişdir ki, istənilən alqoritmin yazılması üçün struktur proqramlaşdırmanın aşağıdakı əsas konstruksiyaları kifayətdir:



Edsger Deykstra.



· operatorların ardıcıllığı;
· alternativ (seçim);
· təkrarlanma (dövr).

Bu məqalə strukturlu proqramlaşdırma üzrə təxminən on illik tədqiqatları yekunlaşdırmış oldu. Orada iddia olunur ki, proqramlaşdırarkən proqramda müəyyən nişana keçmə operatorundan (GOTO) istifadə etmədən də keçinmək olar. Bu operatordan aktiv istifadə zamanı proqramda lüzumsuz keçidlərin sayı çoxalır və proqram anlaşıqlı olmur. Bu operatorun məntiqi "spagetti xörəyi"ni xatırladır. "Strukturlu proqramlaşdırma" termini proqramların strukturunun dəqiq təsvir olunması ilə qurulması meylini əks etdirir.

Vaxt ötdükcə kompüterlər və onlar üçün proqramlar daha da mürəkkəb xarakter aldı. Proqram məhsulu anlayışı meydana gəldi, yəni böyük proqramçılar kollektivi tərəfindən yaradılan və FORTRAN dilində tənliyin dərhal həll edilməsindən fərqli olaraq, uzunmüddətli istifadə üçün nəzərdə tutulan proqram sistemləri hazırlandı. Proqram öz məxsusi həyatını yaşamağa başladı və tez-tez elə hallar baş verirdi ki, ona əlavələri və dəyişiklikləri onun yaradılmasına heç bir aidiyyatı olmayan insanlar yerinə yetirirdilər. Bununla əlaqədar olaraq proqramın etibarlılığı, onun kodunun aydınlığı, nizamlılığı və strukturlu olması birinci yerdə dayanmağa başladı.



Təəssüf ki, yeni Algol-68 versiyasının tətbiqi proqramçılar arasında şöhrət qazanmadı, hətta kommersiya məhsulları sırasına da daxil ola bilmədi. Algol dili elmi tədqiqat işi, "akademik" dil olaraq qaldı. Lakin ümumilikdə proqramlaşdırma dillərinin inkişafına həddsiz təsir göstərdi. O, faktiki olaraq elmi tədqiqatlarda kompüter alqoritmlərinin təsviri üçün standart və alqolabənzər dillər ailəsinin (demək olar ki, bütün müasir prosedur dillər: Pascal, Modula, C və s.) yaranmasına səbəb oldu.

VERİLƏNLƏRİN STRUKTURLARI. PASCAL DİLİ

1970-ci ildə məşhur İsveçrə alimi Niklaus Virt özünün proqramlaşdırma dilini yaratdı. Bu zaman o, Algol dilinin bütün çatışmazlıqlarından xilas olmağa cəhd etmişdi. Yeni dil Pascal adını aldı.

60-cı illərdə işlənmiş strukturlu proqramlaşdırma konsepsiyalarını özündə reallaşdıran Pascal ilk geniş yayılmış dil oldu. Nə üçün bir nəfər tərəfindən yaradılmış Pascal dili, üzərində görkəmli alimlər qrupunun illərlə zəhmət çəkdiyi Algol dilinin əldə edə bilmədiklərinə nail oldu?

Yeni olan Pascal dili Algol dilinin təsvirin ciddiliyi, idarəedici strukturların bolluğu kimi bir çox xüsusiyyətlərini

mənimsəmişdi, lakin öyrənmək üçün daha asan idi. Bu xüsusiyyətlər proqramlaşdırmanın universitet və kolleclərdə tədris olunmasında ondan istifadə imkanlarını müəyyənləşdirmiş oldu. Ancaq Pascal yalnız "tədris" dili olmaq taleyindən yaxa qurtara bildi. O, kifayət qədər böyük layihələrin yaradılmasında yararlı olan güclü proqramlaşdırma dilidir.

Dilin ümumi strukturu, dövrlər, "əgər-onda" seçim operatorları tipli idarəedici konstruksiyalar birbaşa Algol dilindən götürülmüşdür. Eyni zamanda, Pascal proqramlaşdırmanı öyrətmək üçün yaradıldığından, onun sintaksisi də xeyli sadələşdirilmişdir. Yeni, sadələşdirilmiş dil praktiki olaraq alqoritmlərin effektiv şəkildə reallaşdırılmasına imkan vermişdir, lakin, bununla yanaşı dilin öyrənilməsinə sərf olunan vaxt əhəmiyyətli dərəcədə azalmışdır. Onun əsas şüarı belədir: "Ölçülüb-biçilmə, sadəlik, yığcamlıq".

Ancaq təkcə sintaksisin sadələşdirilməsi yeni dilin uğurla təşəkkül tapmasına və inkişafına səbəb ola bilməzdi. Algol və Pascal dillərinin çox mühüm fərqləri hansılardır?

Birincinin yaradıcıları nəzərə almamışlar ki, kompüter təkcə mürəkkəb hesablamalar üçün deyil, verilənlərin emalı üçün də alətdir. Algol və FORTRAN dillərində verilənlərin ifadə tərzi: ədədlər, simvollar (simvollar sətri) və nömrələnmiş ədədlər, yaxud simvollar yığını – massivlərdir. Massivin elementinə giriş bu elementin xüsusi jetonunu – nömrəsini göstərdikdə baş verir.

Kompüterlərin inkişafının ilkin mərhələlərində bu üç növ verilənlər tipi yetərli idi. O zaman kompüterlər əsasən *hesablamalar* aparırdı, başlıcası düsturların və sadə hesablama alqoritmlərinin kodlaşdırılması idi. Mürəkkəb proqramların yazılması üçün bu kifayət etmirdi. Pascal dilininin yaradılması zamanı Virt ikinci inqilabi addım atdı: verilənlərin yeni tiplərini daxil etdi və hər bir proqramçının öz proqramında istifadə edə biləcəyi törəmə tiplərin yaradılmasının ümumi mexanizmini verdi.

Birincisi, bu verilənlərin (ədədlərin, simvolların) vahid tam kimi təsvir olunan nömrələnməmiş yığını idi. Massivdən fərqli olaraq belə yığının elementlərində məxsusi identifikator yoxdur, bütün elementlər bərabər hüquqludurlar. Bundan başqa, verilmiş qiymət ilə elementlər bir-bir onda yerləşirlər. Verilənlərin belə yığını *çoxluq* adlanır və doğrudan da, müasir riyaziyyatın əsas anlayışlarından biridir. Çoxluq o hallarda tətbiq olunur ki, elementlərin sırası, hətta elementlərin özləri də vacib deyildir, elementin verilənlər yığınında *varlığı* əlaməti vacibdir.



Niklaus Virt.

STRUKTURLAŞDIRILMAMIŞ KOD

```
alq kvadrat tənlik
  verilib   a, b, c
  gərəkdir
baş həq diskriminant
| diskriminant := b*b - 4*a*c
| əgər diskriminant < 0
| | onda çapet "köklər yoxdur"
| | əks halda çapet "köklər vardır"
| tamam
son
```

Kvadrat tənliyin köklərinin olub-olmamasını müəyyənləşdirən bu sadə əməliyyatlar FORTRAN dilində belə yazılacaqdır:

```
   DISCR = B*B - 4.*A*C
   IF (DISCR) 1,2,2
1  PRINT 11
11 FORMAT "(Köklər yoxdur)"
   GOTO 3
2  PRINT 22
22 FORMAT "(Köklər vardır)"
3  CONTINUE
```

Əgər diskriminantın qiyməti sıfırdan kiçikdirsə, 1 nişanlı operatora, əks halda isə 2 nişanlı operatora keçid baş verir. Çap operatorları da öz növbəsində başqa operatora (11 və 22 nişanlı) istinad edir. Bu operatorlarda çıxarılan məlumatların formatı təsvir olunmuşdur. Bizim misalda "Köklər yoxdur", yaxud "Köklər vardır". Diskriminantın mənfi olduğu halda, proqram uyğun məlumatı çap etdikdən sonra başqa variantın sahəsinə "girməsin" deyə GOTO operatorundan istifadə olunur. Bu operator sadəcə idarəetməni 3 nişanlı operatora ötürür.

STRUKTURLAŞDIRILMIŞ KOD

```
    DISCR = B*B-4.*A*C
    IF (DISCR) 1,3,2
1   PRINT 11
11 FORMAT (Köklər yoxdur)
    GOTO 4
2   X1 = (-B-D)/(2.*A)
    PRINT 22, X1
22 FORMAT (E8.3, 4X)
3   X2 = (-B+D)/(2.*A)
    PRINT 33, X2
33 FORMAT (E8.3, 4X)
4   CONTINUE
```

```
discr := b*b-4*a*c
if discr < 0 then
  begin
    Print "Köklər yoxdur"
  end
else if discr = 0 then
  begin
    x1 := (b) / (2*a)
    Print x1
  end
  else
  begin
    x1 := (b-d) / (2*a)
    x2 := (b+d) / (2*a)
    Print x1, x2
end
```

FORTRAN dilindəki proqramla müqayisədə müəyyən nişana keçid operatorları yox oldu. Bunun əvəzinə Algol tam bir vahid kimi yerinə yetirilən operator bloklarını təklif edir. Başqa sözlə, əgər diskriminant sıfırdan böyükdürsə, seçim operatorundan bilavasitə sonra gələn begin – end bloku yerinə yetirilir. Əgər diskriminant sıfırdan kiçikdirsə, növbəti blok yerinə yetirilir.

Belə yazılış, birincisi, kodun başa düşülməsini yaxşılaşdırır, çünki aydın olmayan nişanlar üzrə keçidlərin hamısını izləmək lazım deyildir, belə bloklara insan vahid bir tam kimi baxır. O, əvvəlcə hərəkəti bütövlükdə müəyyənləşdirir (əgər diskriminant sıfırdan böyük, yaxud bərabərdirsə, köklər var, əks halda köklər yoxdur), sonra onun içərisinə girərək həmin hərəkətin necə qurulduğunu aydınlaşdırır (mətnin ekrana çıxarılması necə icra edilir, hansı operatorla, məhz hansı mətn çap olunur).

İkincisi, keçid operatorlarından nəzarətsiz istifadə etdikdə, tez-tez idarəetmənin hansısa hərəkəti yerinə yetirən kod hissəsinin *daxilinə* ötürüldüyü hallara təsadüf edilir. Belə metodika hərdənbir daha effektiv proqramlar yaratmağa imkan versə də, həm proqramın yazılması zamanı, həm də sonradan dəyişikliklər apararkən (artıq yaddan çıxmışdır ki, orijinal versiyada nə və necə reallaşdırılmışdır) çoxlu sayda səhvlərin yaranmasına səbəb olur.

Məsələn, kolleksiyaçı üçün onun kolleksiyasındakı poçt markalarında hansı unikal nömrənin olması əhəmiyyətli deyildir, çətin ki, təkrarlanmalar da onu maraqlandırsın. Onun üçün vacibi verilmiş nadir nüsxənin onda olub-olmamasıdır. Həmçinin, ola bilsin, ona aşağıdakı əməliyyatlar lazım gələcəkdir: *birləşmə*, iki kolleksiyaçı qüvvələrini birləşdirmək istədikdə baş verir. Onlar öz kolleksiyalarını birləşdirəcək, təkrarları isə ya dostlarına bağışlayacaqlar, ya da başqa kolleksiyaçılardan onları maraqlandıran markalar ilə dəyişdirəcəklər. Çox ehtimal ki, əgər o, hansı markaların həm özündə, həm də dostunda olduğunu bilmək istəyirsə, ona *kəsişmə* daha maraqlı olacaqdır (sonra o, beynində öz kolleksiyasından kəsişməni çıxır və həmkarlarında olmayan, ona qürrələnmək imkanı verən markalar çoxluğunu alır).

İkincisi, Pascal dilində ilk dəfə verilənlərin yeni mürəkkəb struktur tiplərinin yaradılmasının universal mexanizmi meydana gəldi. Bu mexanizm bir obyektə aid olan verilənləri qruplaşdırmağa, onları yanaşı yazmağa və bu yazıları vahid tam kimi istifadə etməyə imkan verdi. Özü də təkcə ədədlər, yaxud mətn sətirləri deyil, istənilən digər tiplər də qruplaşa bilərlər. Məsələn, əgər şagird haqqında informasiyanın saxlanılması tələb olunursa, onun soyadını, adını, boyunu bir yazıda birləşdirmək lazımdır. Əvvəlcə adları və soyadları saxlamaq üçün verilənlər tipi yaratmaq lazımdır (name_t). Bu 32 simvoldan ibarət massiv olacaqdır:

```
type name_t = array [1..32] of char
```

Sonra şagirdin adından, soyadından və boyundan ibarət yazı təsvir olunur. Şagirdin boyunu adi ədədlə ifadə edək.

```
type pupil_t = record
  firstname : name_t;
  surname : name_t;
  height : real
end
```

Nəhayət, sinfin təsvir olunması üçün şagirdləri təsvir edən massivdən və sinfin nömrəsindən istifadə olunacaqdır (sinfin nömrəsi ancaq tam ədəd ola bilər, ona görə də səhv olma ehtimalını azaltmaq üçün elə indidən kompilyatora deyirik ki, bu tam ədəddir. İstənilən kəsr ədədin ona mənimsədilməsinə edilən cəhdə səhv kimi baxmaq lazımdır):



```
type class_t = record
  pupils : array [1..40] of pupil_t;
  number : integer
end
```

İndi sinifdə boyu orta boydan yuxarı olan bütün şagirdlərin siyahısını çap etmək üçün aşağıdakı koddan istifadə etmək olar:

```
var class : class_t;
var N : integer;
{Burada yazı verilənlərlə doldurulmalıdır}
...
{ Sinifdə orta boyun hesablanması }
for i := 1 to N do
begin
  sum := sum + class.pupils[i].height;
end
mean := sum / N;
{Boyu orta boydan yuxarı olan şagirdlərin
ad və soyadlarının çapı}
for i := 1 to N do
begin
  if (class.pupils[i].height > mean)
  then
    begin
      writeln (class.pupils[i].surname,
        class.pupils[i].firstname)
    end
end
```

Pascal dilinin yaradılması proqramlaşdırma dillərinin inkişafında mühüm mərhələ olmuş və XX əsrin 70-ci illərində onların inkişaf yolunu müəyyən etmişdir; növbəti onilliyin bütün dillərini Algol və Pascal dillərinin birbaşa törəmələri (gələcək nəsilləri) hesab etmək olar.

MASSİVLƏR

Sinfin bütün şagirdlərinin boyu haqqında informasiyanın saxlanılması və emalı üçün, uyğun ədədlərin saxlanılacağı Hegiht massivini yaratmaq olar. Jurnalda nömrəsi 13 olan şagirdin boyunu almaq lazımdırsa, proqramda bu Hegiht[13] kimi təsvir olunacaqdır. Pascal dilindəki növbəti proqram nümunəsi sinifdə şagirdlərin orta boyunun tapılmasını nümayiş etdirir.

```
{Tutaq ki, Height – şagirdlərin boyunu göstərən massiv
N – sinifdə şagirdlərin sayıdır}
for i := 1 to N do
begin
  sum := sum + Hegiht[i];
end
mean := sum / N;
```

Əgər bütün verilənləri massivdə birləşdirmək imkanı olmasaydı, onda proqramda hər bir şagirdin boyu üçün ayrıca dəyişəndən istifadə etmək lazım gələrdi. Kod təxminən belə görünərdi:

```
sum := sum + heightAlpay;
sum := sum + heightAnar;
sum := sum + heightAyxan;
...
```

Sinifdə şagirdlərin sayı dəyişdikdə proqramın da kodunu dəyişdirmək lazım gələrdi. Massivlə olan misalda isə təkcə N dəyişəninin qiymətini dəyişdirmək lazımdır. Əgər verilmiş hesablama prosedurun daxilində istifadə olunursa və N onun parametridirsə, onda yuxarıdakı massivlə olan kod universal olacaqdır. Sinifdəki şagirdlərin (ola bilər ki, komandada idmançıların, yaxud gəmidə kosmonavtların) orta boyunun tapılması lazım gələn istənilən proqramda ondan istifadə oluna bilər.

Bununla belə, Pascal dilinin bir sıra çatışmazlıqları da olmuşdur. O, massivlərdə yuxarı "açıq", kompilyasiya zamanı naməlum olan sərhəddən istifadə edə

PASCAL DİLİNİN BİRİNCİ KOMPİLYATORUNUN YAZILMA TARİXİ



1969-cu ildə Virt öz şagirdlərindən biri E.Marmyeyə Pascal üçün kompilyator yazmağı tapşırır. O, kompilyatoru yeganə bildiyi FORTRAN dilində yaratmağa başlayır. Bu, səhv idi, çünki sonralar Marmye özü də etiraf edirdi ki, o, mürəkkəb verilənlər strukturu ilə işləyən zaman dilin məhdudiyyətlərinə görə tamamilə dolaşığa düşüb qalmışdı.

1970-ci ildə Pascal dilinin sintaksisinin formal təsviri yaradıldıqdan sonra kompilyatoru bu dilin özündə yaratdılar. Sonra yaradıcılardan biri R.Şild iki həftəyə kompilyatorun proqramını maşın koduna çevirdi. Başqa sözlə, o, dilin hər bir konstruksiyasını götürmüş, "insan-kompilyator" kimi işləyərək onu maşın kodunda yazdı. Beləliklə, Pascal özü özündə reallaşdırılmış yüksək səviyyəli ilk dil oldu.

Digər maşınlara keçirmək üçün xüsusi olaraq yaradılmış P-kod adlanan kodu yerinə yetirən və interpretasiya üsulu ilə zamana görə sınaqdan çıxarılan P-maşından istifadə olunurdu. P-kod xəyali kompüter üçün maşın kodundan ibarətdir. Kompilyatoru əllə bu uydurma koda çevirdilər. Yeni maşında verilmiş P-kod üçün ancaq interpretator yaradıldı. Bu, Şildin qarşısında durmuş məsələdən daha sadə məsələ oldu. P-kod üçün interpretator yaradılan kimi, onun köməyilə P-kodda yazılmış kompilyatoru buraxmaq və bir daha, artıq verilmiş kompüter üçün maşın koduna kompilyasiya etmək mümkün oldu.



N.Virt və E.Deykstra.

Beləliklə, Niklaus Virti və onun Pascal dilini virtual maşınlardan geniş istifadə olunması ideyasının baniləri saymaq olar. Yaxşı tanınmış Java texnologiyası və çox şeylər vəd edən Microsoft. Net də virtual maşınlara əsaslanır.

bilməmişdir. Bu misaldan da görünür sinfin təsviri zamanı 40 elementli massivdən istifadə olunmuşdur. Ona ümid edilmişdir ki, şagirdlərin sayı 40-dan çox olmayacaqdır. Əgər çox böyük sinif tapılarsa, onda nə olacaqdır?

Geniş yayılmasına baxmayaraq, Pascal dilində simvol sətirlərilə işləmək üçün verilənlər tipi nəzərdə tutulmamışdır. Hər dəfə onları yenidən yaratmaq lazım gəlirdi (name_t tipinin təyin olunmasında olduğu kimi. Yenə də şərt qoyulurdu ki, məktəblinin ad və soyadının uzunluğu 32 simvoldan kiçikdir).

Pascal proqram kodunun müxtəlif fayllarda yazılmasını və kompilyasiya vaxtı onların birləşdirilməsini nəzərə almamışdır. Ona görə də, əgər böyük proqram yaradılırdısa, mətnlərilə birlikdə faylın həcmi yüzlərlə kilobaytdan ibarət ola bilirdi. Bu isə praktiki olaraq proqramçılar komandasının işini qeyrimümkün edirdi. Bir proqramçıya, müəllifin özünə də, ona lazım olan fraqmenti tapmaq çətinlik törədirdi.

Pascal dilində informasiyanın giriş və çıxışının imkanları da məhduddur. Köməkçi funksiyaların standart kitabxanası yaradılmamışdır (məsələn, sətirlərin müqayisəsi zamanı hər dəfə proqramçı bunu simvol-simvol edirdi).

C DİLİ

Bu gün ən məşhur dillərdən biri, çox sadə C adı ilə (ingilis əlifbasının "c" – "si" hərfi kimi oxunur) 1972-ci ildə Dennis Ritçi tərəfindən yaradılmışdır. C dili o zamanlar az tanınmış, yeni UNIX əməliyyat sistemində proqramlaşdırmaq üçün yaradılmışdır (bu şişirtmə deyil: o zaman UNIX-dən ancaq Bell Telephone Laboratories korporasiyasının daxilində bir neçə maşında istifadə olunurdu). O, praktiki olaraq naməlum B dilinin ardıcılı kimi ortaya çıxdı. C dili Algol və Pascal dillərinin varisi hesab olunur, çünki bir çox xüsusiyyətlərini onlardan götürmüşdür. Onlar kimi C də strukturlu proqramlaşdırma üçün prosedurlu dildir. C-nin bir fərqləndirici xüsusiyyəti ondan ibarətdir ki, onun yaradılması zamanı qarşıya mürəkkəb, konseptual məqsədlər qoyulmamışdı: hesablamaların aparılması üçün universal dilin alınması (FORTRAN), prosedurlu proqramlaşdırma üçün akademik dilin yaradılması (Algol), didaktik və tədris üçün nəzərdə tutulan dilin hazırlanması (Pascal). C peşəkar proqramçılar tərəfindən onlar üçün asan dil kimi yaradılmışdır.

C komitələr və alimlərin deyil, real proqramlar yazan proqramçıların əməyinin nəticəsi olduğundan, onda mürəkkəb sistemlərin yaradılması zamanı digər dillərdən normal istifadəyə mane olan bütün məqamlar nəzərə alınmışdır. FORTRAN dilindən sonra, yenidən faylların ayrı-ayrılıqda kompilyasiya olunması və kompilyasiyadan sonra onların yerinə yetirilə bilən vahid fayl şəklində quraşdırılması imkanı meydana gəldi (kompilyator "yarımfabrikatlar" hazırlayır, xüsusi proqram-toplayıcı isə "onları bir tavaya yığaraq, hazır olana qədər qızardırdı"). Dillə birlikdə proqramçının işini asanlaşdıran, trivial hərəkətlərin yazılması qayğısını onun üzərindən götürən və proqram üzərində fikrini cəmləşdirməyə imkan verən əsas funksiyaların standart kitabxanası yaradıldı. Özü də əsas funksiyalar kitabxanası, onun bölünməz bir hissəsi kimi dilin təsvirinə daxil oldu. Hər bir proqramçı arxayın ola bilər ki, C kompilyatorunun olduğu istənilən platformada bu kitabxana da vardır.

Pascal dililə müqayisədə C dili, proqram yaradıcıları tərəfindən proqram yaratma vasitələrinin seçilməsi zamanı böyük çevikliyə malik olur. Effektivlik naminə strukturlu proqramlaşdırmanın ciddi konsepsiyalarını qurban vermək, trivial olmayan və təhlükəli genişlənmələrdən istifadə etməklə proqram yazmaq mümkün oldu.



Dennis Ritçi.

C dilində təkcə yeni konstruksiyalar meydana gəlmədi, köhnələri də yenidən işləndi. İdarəedici strukturlar baxımından əsaslı heç bir şey daxil edilmədi, ancaq yeni operatorlar mövcud proqramları daha yığcam yazmağa imkan verdi. Proqramların "mənası" maşın koduna yaxın oldu.

Bir neçə alternativdən seçmə operatoru – switch meydana gəldi. Əgər qabaqlar dəyişənin qiymətindən asılı olaraq hərəkətin seçilməsi üçün "əgər – onda" zəncirindən istifadə etmək lazım gəlirdisə:

```
if mark = 3
then
  writeln ("Yarayar")
else
  if mark = 4
  then
    writeln ("Yaxşıdır")
  else
    if mark = 5
    then
      writeln ("Əladır!")
```

indi C-də bir dənə switch operatorundan istifadə ilə bu mümkün oldu:



MASSİVLƏR VƏ GÖSTƏRİCİLƏR

Əgər *a* massivi və *p* göstəricisi varsa,

```
int a[5]; /* 5 elementdən ibarət tam ədədlər
massivinin elan edilməsi */
int *p; /* tam ədədə göstəricinin elan edilməsi */
```

onda p=a ifadəsi *p*-yə *a* massivinin birinci elementinin ünvanını yerləşdirəcəkdir. Ona giriş əldə etmək üçün *p şəklində göstəricidən istifadə edirlər. Göstəricini 1 artırdıqda o, növbəti elementi göstərəcəkdir, yəni *(p+1) yazısı *a[1]* ilə ekvivalentdir. Göstəriciləri çıxmaq da olar. Əgər iki göstərici varsa: biri massivin birinci elementinə, digəri isə hansısa ixtiyari elementə, onda bu elementin indeksini onun göstəricisindən massivin başlanğıcının göstəricisini çıxmaqla alırlar. Başqa sözlə, əgər *pa* massivin başlanğıcına olan göstərici, *pe* isə ixtiyari elementə olan göstəricidirsə, onda ixtiyari elementin indeksi *i=pe–pa* olacaqdır. Bu aydındır, çünki massivin elementinə olan göstərici massivin başlanğıcına olan göstəricilə elementin indeksinin cəminə bərabərdir, yəni *pe=pa+i*.

Bəzən göstəricilərdən istifadə yaxşı tanınmış məsələni qəşəng və qeyri-adi həll etməyə imkan verir. Məsələn, sıfır kodlu simvolla qurtaran bir simvol sətrinin digərinə köçürülməsi məsələsinin Pascal dilində sadə həlli belə ola bilər:

```
var str : array [0...255] of char;
var copy : array [0...255] of char;
var i : integer;
for i := 0 to 255
begin
copy[i] := str[i];
if str[i] = "\0" then goto finish;
end;
finish: ;
```

For operatoru başlanğıcda *i* dəyişəninə 0 qiyməti mənimsədilən dövrü təmin edir, dövr dəyişəninin qiyməti hər iterasiyadan sonra 1 artırılır, dövrün davam etməsi isə *i<255* şərtidir. Dövrün daxilində *str* sətrinin *i-ci* simvolu *copy* sətrinə köçürülür. Sonra *str* sətrinin *i-ci* simvolu sıfır simvolla ("\0") müqayisə olunur və əgər bərabərlik yerinə yetirilirsə, *goto* operatorunun köməyilə dövrdən çıxış həyata keçirilir.

Əgər göstəricilərdən istifadə olunarsa və fərz etsək ki, sətir mütləq sıfır kodlu simvolla qurtaracaqdır, onda sətirlərin köçürülməsi C dilində daha yığcam olacaqdır:

```
char *pstr, *pcopy;
pstr = str; pcopy = copy;
...
while (*pcopy++ = *pstr++)
;
```

pstr və *pcopy* sətirlərinə iki göstəricidən istifadə olunur. Sonra, C dilində operatorların öncüllüyünə uyğun olaraq müəmmalı mənimsəmə operatoru bu cür emal olunur:

1. Mənimsəmənin sağ operandının hesablanması.

a) *pstr* göstəricisinin qiyməti götürülür, * operatorunu tətbiq etmək üçün istifadə olunur;

b) *pstr* göstəricisi 1 artırılır (yəni massivin növbəti elementinə keçid baş verir).

2. Mənimsəmənin sol operandının hesablanması.

a) *pcopy* göstəricisinin qiyməti götürülür, * operatorunu tətbiq etmək üçün istifadə olunur;

b) *pcopy* göstəricisi 1 artırılır (yəni, massivin növbəti elementinə keçid baş verir).

3. Mənimsəmənin özü yerinə yetirilir, yəni yeni sətrin 2-ci addımda alınmış elementinə 1-ci addımda alınmış elementin qiyməti mənimsədilir.

4. Bütün ifadənin qiyməti bütövlükdə massivin emal olunan elementinin qiymətidir.

Əməliyyatların aparılması nəticəsində bütün məqsədlər əldə olunmuşdur. Birincisi, yeni sətrin elementinə köçürülən sətrin elementinin qiyməti düşmüşdür. İkincisi, hər iki sətrin göstəriciləri növbəti elementlərə keçmişdir. Nəhayət, ifadənin qiyməti qismində, biz elementin qiymətini almışıq ki, sətrin sonunun çatdığını təyin etmək üçün onu sıfırla müqayisə etmək olar.




```
switch( mark)
{
  case 3:
    printf ("Yarayar");
    break;
  case 4:
    printf ("Yaxşıdır");
    break;
  case 5:
    printf ("Əladır!");
    break;
}
```

Yenə də proqramda qlobal dəyişənlərdən istifadə imkanı meydana gəlmişdir. Proqramın bütün funksiyalarında, hətta kitabxana funksiyalarında da, istifadə olunan dəyişənlər potensial olaraq təhlükəlidirlər, çünki çətin tutula bilən səhvlərə səbəb ola bilərlər. Belə dəyişənlər "düzgün" strukturlu proqramlaşdırmanın əsas akademik ideyasına ziddirlər. Digər tərəfdən, əgər proqramın bütün funksiyaları eyni bir dəyişənlər yığını ilə işləyirsə, onda onları qlobal dəyişənlər kimi elan etmək olar. O zaman bu dəyişənləri bütün funksiyalara parametrlər kimi ötürməyə ehtiyac olmayacaqdır ki, bu da proqramın işini xeyli sürətləndirəcəkdir.

C dilində yenidən yaddaşın oyuqlarına göstəricilərlə işləmək imkanı meydana çıxmışdır. Əlbəttə, elə görünür ki, bu çox təhlükəlidir. Əgər proqram ixtiyari verilənləri 21324 ünvanlı oyuğa yazarsa, nə olacağını heç kim bilmir. Orada əməliyyat sisteminin kodu, yaxud proqramın özünün verilənləri saxlanıla





bilər. Bu zaman sanki o işini davam etdirəcək, amma qabaqcadan nəticələrin necə olacağı bilinməyəcəkdir.

Onda yaddaşın ünvanına göstəricilərin daxil edilməsi nəyə lazımdır? Bütün iş ondadır ki, proqramçının ixtisas bacarığı lazımi səviyyədə olduqda proqramın verilənlərinin ünvanları ilə iş (*ünvan hesabı*) praktiki olaraq Assembler dilində olduğu kimi effektiv, strukturlu dillərin bütün rahatlıqları ilə birlikdə (funksiyalar, idarəedici konstruksiyalar, mürəkkəb verilənlər tipləri və s.) kod yaratmağa imkan verir.

Verilənlərin obyektlərinə göstəricilər – istinadlar hələ Pascal dilində meydana gəlmişdir, amma onları ancaq digər dəyişənlərdə saxlamaq və sonra lazım olan obyektlərə giriş üçün istifadə etmək

C DİLİ VƏ UNIX ƏMƏLİYYAT SİSTEMİ

1974-cü ildə UNIX əməliyyat sistemini tamamilə Assembler dilindən C dilinə köçürdülər. Beləliklə, o, yüksək səviyyəli proqramlaşdırma dilində yazılmış ilk ticari əməliyyat sistemi oldu. Bu, əməliyyat sistemilə proqramlaşdırma dilinin qəribə simbiozunun yaranması anı idi.

Əməliyyat sisteminin yüksək səviyyəli dildə yaradılması faktiki olaraq onu EHM-dan asılı olmayan etdi. Belə sistemi istənilən kompüterə sadəcə keçirmək kifayətdir. Bunun üçün əsas şərt həmin kompüterdə C dilinin kompilyatorunun olmasıdır. UNIX sisteminin yaradıcıları – Bell Telephone Laboratories – onu ilkin kodlarında öyrənmək və istifadə etmək məqsədilə ali təhsil müəssisələrinə pulsuz olaraq təqdim etdilər. Bu da öz növbəsində C dilinin, nəticə etibarilə elə UNIX sisteminin də geniş yayılmasına səbəb oldu.

PLANKALKUL – İLK PROQRAMLAŞDIRMA DİLİ

İkinci dünya müharibəsinin axır günlərində alman mühəndisi və ixtiraçısı Konrad Suze Alp dağlarındakı balaca Hinterşteyn kəndində idi. Son bir neçə ildə ilk dəfə sakit şəraitə düşdüyündən (o yazmışdı: "bombardmanlar, telefon zəngləri və görüşə gələnlər olmadan") və özünün Z4 kompüterində işini davam etdirmək imkanı olmadığından, 1943-cü ildə başlamış olduğu, müxtəlif hesablama məsələlərinin təsvirinin öyrənilməsini davam etdirmək qərarına gəlir. Alman sözü "rechnen" "hesablamaq" mənasını verir. Ona görə də, Suze bu təsvirlər üçün Rechnenplan – "hesablama planı" ("proqram" sözünü bir qədər gec fikirləşmişlər) terminindən istifadə etmişdir. "Hesablama planı"nın yazılması üçün yaradılmış sistemi Suze Plankalkul adlandırmışdır. Hesab olunur ki, Plankalkul on ildən az olmayaraq müddətə FORTRAN və Algol-60 dillərini qabaqlayan, tarixdə ilk proqramlaşdırma dili olmuşdur.

Lakin Suzenin özü Plankalkula proqramlaşdırma dili kimi baxmamışdır. O, informasiyanın ixtiyari çevrilməsini təsvir etməyə imkan verən universal işarələmələr sistemi yaratmaq istəyirdi. Plankalkul-dan alqoritmləri yazmaq üçün istifadə etmək olardı, elə əslində də, Suze ən müxtəlif məsələlərin həllini onda yazmışdır. O, dəyişməz uğurla kompüterin qurğularının işini təsvir etmək üçün də Plankalkul-u tətbiq etmişdir. Çox-çox sonralar meydana gələn proqramlaşdırma dillərinin bir çox xüsusiyyətlərini qabaqcadan bilməsinin Suzeyə bu dərəcədə nəsib olması, xüsusən təəccüblüdür. Onun bir çox fikirləri isə yaxınlarda reallaşdırılmışdır. Bəs Plankalkul dili nə idi?

Hər şeydən öncə, dildə verilənlər tipinin çox maraqlı sistemi qəbul edilmişdir. Suze hesab edirdi ki, "hesablama bitdən başlayır". Ona görə də, onda sadə tip *S0* ilə işarə olunan bitdir (hə/yox vəziyyəti). Bitlər *L* və *0* qiymətlərini alırlar, belə ki, bitlər ardıcıllığı *LOLO*, *OLLO* və s. kimi yazılır. Bitlərdən skalyar tiplər – qeyd olunmuş və sürüşkən vergüllü qurulur. Mürəkkəb obyektlər də (indi onları verilənlər strukturu adlandırardılar) bitlərdən təşkil edilir. Məsələn, ixtiyari ölçülü massiv təyin etmək olar: $n \times S0$ (n uzunluqlu birölçülü Bul massivi), $m \times n \times S0$ ($m \times n$ ikiölçülü Bul massivi) və s.

Massivlərdən əlavə Plankalkul dilində müxtəlif tipli verilənləri birləşdirən yazılar vardır (onların elementləri başqa yazılar ola bilər). Yazılar 1959-cu ildə xüsusiləşdirilmiş Cobol dilində meydana gəlsə də, sonralar PL/1 dilində istifadə olunsa da, universal proqramlaşdırma dillərində 10 il keçdikdən sonra təkcə Pascal dilində daxil edildi.

Əvəzində Suzenin bir ideyasının başqa dillərdə uzun müddət bənzəri olmamışdır. O, proqramlarının mətninə bu və ya digər operatorların yerinə yetirilməsi zamanı doğru olmalı olan şərtlərdən ibarət ifadə daxil edirdi. Məsələn, tutaq ki, aqobyekt (dörd bitdən ibarət massiv) onluq rəqəmlərin yazılması üçün istifadə olunur. Amma rəqəmlər cəmisi on dənə olduğundan, obyekt isə 16 müxtəlif qiymət ala bildiyindən onun üzərinə



B.Meyer həmkarları ilə A.Yerşovun qonağıdır. 1977-ci il.



Konrad Suze.

belə bir şərt qoymaq olar: 0.10-dan böyük olmamalıdır. Ancaq 50 il keçdikdən sonra analoji ideya Kaliforniyadan olan proqramçı, EIFFEL proqramlaşdırma dilinin müəllifi Bertran Meyer tərəfindən reallaşdırıldı.

Plankalkul dilinin, obyektlər çoxluğundan yalnız verilmiş xüsusiyyətə malik olanlarının seçilməsi təklifi də bir o qədər unikaldır. Əlbəttə, dövr operatoru, yaxud prosedur ilə bunu etmək çox sadədir, amma xüsusi konstruksiyalar müasir proqramlaşdırma dillərinin heç də hamısında yoxdur.

Plankalkul kifayət qədər mənalı imkanlara malikdir. Mənimsəmə operatorundan savayı, onda şərt operatoru, təkrarlanma operatoru (müasir dövrlərin analoqu), alt-proqramlara müraciət imkanı və b. vardır. Şübhəsiz, o, müasir proqramlaşdırma dilləri üçün adi olan çoxlu xüsusiyyətlərə də malikdir, ancaq onda bizim adət etdiyimiz imkanlar, məsələn, şərtsiz keçidlər, yaxud istinadlarla iş yoxdur. Bundan başqa, qəbul olunmuş yazı formasında artıq şeylər çoxdur. Çatışmazlıqların siyahısını davam etdirmək olar, ancaq onların heç biri belə bir başlıca cəhətə kölgə salmır: Plankalkul yüksək səviyyəli proqramlaşdırma dilləridir. O, kompüter proqramlarının yazılması üçün nəzərdə tutulmasa da, 25 ildən sonra Suze alqoritmlərini digər dillərə çevirdikdən sonra bəlli oldu ki, proqramlar işləyir! Ona görə də, mübaliğəsiz söyləmək olar ki, həqiqətən çoxlu mürəkkəb proqramlar (verilənlərin nizamlanması, məntiqi düsturların sintaktik düzgünlüyünün analizi, tam və sürüşkən vergüllü ədədlər üzərində hesab əməliyyatlarının, o cümlədən kvadrat kökün tapılmasının yerinə yetirilməsi və sair) yazmış Suze tarixdə ilk proqramçı olmuşdur. Onun şahmat oynamaq üçün olan proqramları xüsusilə maraqlıdır. Bu proqramlar faktiki olaraq süni intellekt sahəsində (bu elmin yaradılmasına xeyli qalmış) ilk proqramlaşdırma təcrübəsi olmuşdur. Yeri gəlmişkən, hələ 1938-ci ildə Suze söyləmişdir ki, 50 ildən sonra maşın şahmatda insanı məğlub etməyə başlayacaqdır. Təəccüblüdür, ancaq o cəmisi bir neçə il səhv etmişdir: dünya çempionu Harri Kasparov 1997-ci ildə Deep Blue superkompüterilə olan yarışda uduzdu. Adi həvəskarlar isə bundan 10 il əvvəl şahmat proqramları ilə yarışa bilmirdilər. Deməli, Suze öz proqnozunda doğru olmuşdur. Təəssüf ki, bu dil heç bir kompüterdə reallaşdırılmamışdır, onun tam təsvirini Konrad Suze ancaq 1972-ci ildə nəşr etdirmişdir. Ona görə də, çoxları hesab edir ki, Plankalkul müasir proqramlaşdırma dillərinin inkişafına təsir etməmişdir, lakin bu heç də belə deyildir.

olardı. C dilində obyektlərə, dəyişənlərə və funksiyalara göstəriciləri nəinki yadda saxlamaq, həmçinin toplamaq və çıxmaq olar.

C dili qoyulmuş məsələnin çox yığcam və effektiv həllinə imkan verir. Lakin bu zaman böyük ehtimalla çətin aşkarlana bilən səhvlər yarana bilər. Onların istisna olunması üçün proqramçının yüksək dərəcədə özünənəzarəti və ustalığı olmalıdır, çünki C-ni peşəkar proqramçılar özləri üçün yaratmışlar, onlar isə öz qüvvələrinə inanırdılar.

Proqramçı necə kodun ona lazım olmasını özü həll edir: (bir platformadan başqasına) keçirilə bilən, yaxud effektiv, aydın və etibarlı, yaxud daha məhsuldar? Proqramın ümumi məntiqi klassik strukturlu proqramlaşdırmanın bütün tələblərini nəzərə ala bilər, ayrı-ayrı funksiyalar isə göstəricilərdən və başqa maşın "fokuslarından" istifadə edə bilər. Proqramçı nəyi və necə etməyi müəyyən edir, vahid alət olan C isə ona bu məsələnin həllində, Pascal və Algol dillərində olduğu kimi ciddi məhdudiyyətlər qoymadan kömək edir.

OBYEKT-YÖNLÜ PROQRAMLAŞDIRMA DİLLƏRİ

OBYEKT-YÖNLÜ YANAŞMA

XX əsrin 50-ci illərində proqramların yazılması zamanı əsas diqqət məsələnin həlli üçün lazım olan alqoritmlərə, prosedurlara və hərəkətlərə verilirdi. Həmin illərdə proqram sistemlərinin mürəkkəbliyinin müəyyən səviyyəni aşmadığı bir şəraitdə prosedur yanaşma özünü doğruldurdu.

Kompüterlər inkişaf etdikcə daha tez-tez onları elə məsələlərin həlli üçün tətbiq etməyə başladılar ki, onların mürəkkəbliyi adətən, alqoritmlə deyil, prosesdə iştirak edən böyük sayda iştirakçılarla və onlar arasında çox mürəkkəb münasibətlərlə şərtlənir. Bankın fəaliyyətinin avtomatlaşdırılmasında istifadə olunan alqoritm elə də mürəkkəb deyildir. Onlar da dəqiq olaraq aerodinamika,



Alan Key.

yaxud nüvə fizikası tənliklərinin həllini özlərinə qoşmurlar. Amma bu onu göstərmir ki, alınacaq proqram, yaxud, daha doğrusu, proqramlar kompleksi sadə olacaqdır. Bankın gündəlik işində çoxlu sayda müxtəlif obyektlər iştirak edir: əməliyyatçılar, dəllallar (brokerlər), klerklər, müştərilər, bankomatlar, digər banklarla əlaqə üçün şəbəkələr, dövlət və digər auditor komissiyaları, birjalar və s.

XX əsrin 60-70-ci illərində meydana gəlmiş strukturlu dillər, onlarda reallaşdırılmış idarəetmə vasitələri və verilənlərin tiplərilə mürəkkəb proqram komplekslərinin yaradılmasını asanlaşdırdı. Ancaq, yenə də onlar problemin tam qaneedici həllini vermədilər. Çünki böyük məsələlər dairəsinin həllinə yararlı və yeni proqramlaşdırma modellərinin təqdim olunmasına deyil, artıq mövcud olan proqramlaşdırma modellərinə əlavələr etməyə və ya onların yaxşılaşdırılmasına cəhdlər olunurdu. Əksər proqramlaşdırma dillərinin göstərişləri *imperativdir,* yəni onlarda icra olunan vasitələr (EHM, yaxud mücərrəd proqram sistemi) ada malik olmur, kompüteri tam olaraq təqdim edirlər.

Həmin illərdə *obyekt-yönlü dillər* yaradılmışdır. Onlarda icra olunan vasitələr şərti olaraq, hər birinə ayrılıqda müraciət oluna bilən obyektlərə bölünürlər. Bu obyektlər öz aralarında məlumatlarla əlaqələndirilə bilərlər.



Proqram obyektləri anlayışı ilk dəfə Norveçin Oslo şəhərindəki hesablama mərkəzində, Ole-Coan Dal, Born Myurhoq və Kristen Nıqard tərəfindən Algol-60 əsasında yaradılmış Simula-67 dilində verilmişdir. Simula-67 mürəkkəb sistemlərin modelləşdirilməsi və təsviri dilidir.

Bununla belə, bu ideyanın əsl geniş tətbiqi 1970-ci ildə SmallTalk dilinin yaradılması zamanı Alan Key tərəfindən Palo-Altoda Xerox firmasının Tədqiqat mərkəzində baş verdi. SmallTalk ancaq obyekt-yönlü konstruksiyalardan ibarətdir. Bu yanaşma sistemlərin təsvirini daha yığcam və universal etməyə imkan verir, amma məsələlərin həllində təkcə belə mexanizmdən istifadə həmişə effektli alınmır.

Obyekt verilənlər strukturu və alqoritmlərdən ibarətdir. Hər bir obyektə öz verilənləri üzərində əməliyyatları necə yerinə yetirmək məlumdur, ancaq digər obyektlər üçün o, özünü "qara qutu" kimi göstərə bilər. Müxtəlif obyektlər əməliyyatları həyata keçirmək üçün, eyni bir açar sözlə təyin olunan müxtəlif alqoritmlərdən istifadə edəcəklər. Məsələn, verilənləri kompleks ədədlərdən ibarət olan obyektlər və verilənləri tam ədədlərdən ibarət olan obyektlər hesab əməliyyatlarının yerinə yetirilməsi üçün müxtəlif üsullardan istifadə edəcəklər.

Obyektin və onun xassələrinin formal təsvirini *sinif* adlandırırlar. Eyni bir sinfə aid olan bütün obyektlər eyni struktura və xassələrə malikdirlər, ancaq atributlarının qiymətlərinə görə fərqlənə bilərlər: Levi's 501 modelindən olan bütün cins şalvarlar, eyni materialdan tikilmişdir, iki balağı və beş cibi var, ancaq müxtəlif ölçülü, geyilmiş, yaxud tam təzə formada ola bilərlər.

Hal-hazırda aşağıdakı obyekt-yönlü proqramlaşdırma dilləri daha populyardırlar: C++, Python, Java, C#. Obyekt-yönümlü proqramlaşdırma modeli adi prosedurlu modeldən fərqlənir.

Abstraktlaşdırma. Mürəkkəb modellərin qurulması zamanı əşyaların və hadisələrin ən vacib detallarını (təfərüatlarını) ayırmaq, əhəmiyyətsiz detallardan isə uzaqlaşmaq lazımdır (bu əhəmiyyətsiz detallar, əksinə, digər modellərin qurulması zamanı əhəmiyyətli ola bilərlər). Belə proses abstraktlaşdırma adlanır. Nümunə üçün kitabxananın avtomatlaşdırılması sistemini götürsək, bu sistemin əsas şəxsləri (obyektləri) aşağıdakılar olacaqdır:



• kitab;

• kitabların saxlanıldığı qəfəslər;

• kitabları oxuculara verə bilən kitabxanaçı;

• oxucu.

Əhəmiyyətsiz detalları nəzərə almamaq və yalnız hər bir obyektin əhəmiyyətli xüsusiyyətlərini ayırmaq lazımdır. Oxucunun saçının və gödəkçəsinin rəngi, kitabxanaçının yaşı, qəfəslərin hansı materialdan hazırlanması, yəqin

Oxucu
– Soyadı – Adı – Pasport № – Əlaqə telefonu – Kitabların siyahısı

Kitab
– Müəllifi – Adı – Daxili № – ISBN – Səhifənin sayı – Üz qabığının rəngi – Nüsxələrin sayı – Yeri

Qəfəsə
– Mərtəbə – Otağın nömrəsi – Harada yerləşir



ki, heç bir əhəmiyyət kəsb etməyəcəkdir, amma kitabın üzlüyünün rəngi və qalınlığı əhəmiyyətli olacaqdır. Ona görə ki, oxucu "üç müəllifin yazmış olduğu qəhvəyi üzlüklü, qalın alman dili dərsliyini" soruşa bilər.

Kitabın xüsusiyyətlərinə onun müəllifləri, daxili kitabxana qeydiyyat nömrəsi və xüsusi beynəlxalq ISBN qeydiyyat nömrəsi, səhifələrin sayı, cildin tipi və rəngi aiddir. Oxucu isə adı, soyadı, pasport verilənləri, əlaqə telefonları ilə xarakterizə olunur.

İnkapsulyasiya. Baxmayaraq ki, "kitab" obyekti öz xassələrinin siyahısı

İNTERFEYS VƏ REALLAŞDIRMA

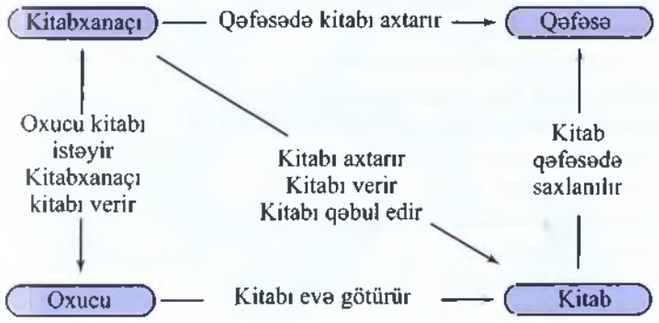
İnterfeys:

```
class Book
{
  String Author();          /* Müəllifin adını qaytaran
                               funksiya */
  String Name();            /* Kitabın adını qaytaran funksiya */
  String Code();            /* Kitabın kodunu qaytaran
                               funksiya */
  String ISBN();            /* Kitabın ISBN nömrəsini qaytaran
                               funksiya */
  Integer NoPages();        /* Səhifələrin sayını qaytaran
                               funksiya */
  Color CoverColor();       /* Üzlüyün rəngini qaytaran
                               funksiya */
  Integer NoCopies();       /* Kitabxanada kitabın verilmiş
                               növündən olan nüsxələrin sayını
                               qaytaran funksiya */
  Location Location();      /* Kitabxanada verilmiş kitabın
                               yerini qaytaran funksiya */
}
```

Reallaşdırma:

```
/* Müəllifin adını almaq üçün funksiya */
String Book: :Author()
{
  /* Verilənlər bazasına qoşulmaq */
  DB db = GetDatabaseConnection( );
  /* Verilənlər bazasına sorğu vermək */
  RS rs = db.Execute("Select Author FROM Books WHERE
        Book_Id = %1", m_Id);
  /* Müəllifin alınmış adını qaytarmaq */
  return rs("Author");
}
```

kimi təsvir olunur, bu onu göstərmir ki, o, sadəcə, hər hansı verilənlər strukturu, sətirlər yığınıdır (C dilinin struct { ... }, yaxud Pascal dilinin record konstruksiyalarına bənzər).

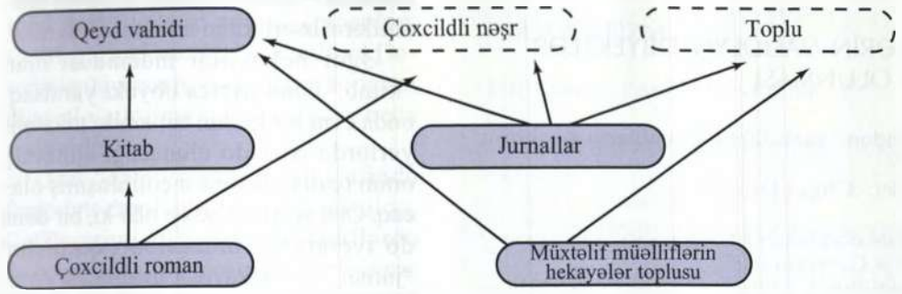


"Kitab" obyekti oxucuya sorğular siyahısı, yəni onun reallaşdırdığı xüsusiyyətlər və onların hansı formada reallaşdırılmasını verir. Məsələn, "müəllif" sorğusu müəllifin tam adı ilə simvollar sətri, "nüsxələrin sayı" sorğusu isə ədəd çıxarır. Bu obyektin *interfeysi* adlanır.

Obyektə müraciət etmək üçün, yalnız onun interfeysini bilmək lazımdır. Onun daxili quruluşu isə *inkapsulyasiya* vasitəsilə, yəni ayrı-ayrı detalların reallaşdırılmasının gizlədilməsilə ("kitab" ilə ünsiyyətdə olan digər obyektlər üçün onun daxili quruluşu tamamilə əhəmiyyətsizdir) "gizlədilmişdir". İnkapsulyasiya proqramın digər hissələrində heç bir dəyişiklik aparmadan obyektin reallaşdırılmasını tamamilə dəyişməyə imkan verir. Yeni imkanlar əlavə etməklə asan və sürətlə proqramın işini təkmilləşdirmək olar. Məsələn, kitabın təsvirini verilənlər bazasında saxlamaq üçün obyekt-yönlü yanaşmadan istifadə zamanı ancaq "kitab" obyektinin reallaşdırılmasını yenidən yazmaq lazımdır. Əgər adi, prosedurlu yanaşma tətbiq olunsaydı, onda yəqin ki, proqramın çox hissəsini yenidən yazmaq lazım gələcəkdi.

İyerarxiya. Abstraktlaşdırma və inkapsulyasiyadan istifadə hələ Pascal və C kimi, "yuxarı" nəsil strukturlu dillərdə də mümkün olmuşdur. Lakin ən sadələrdən başqa praktiki olaraq bütün hallarda sistemdə abstraktlaşmaların sayı çox böyük alınır.

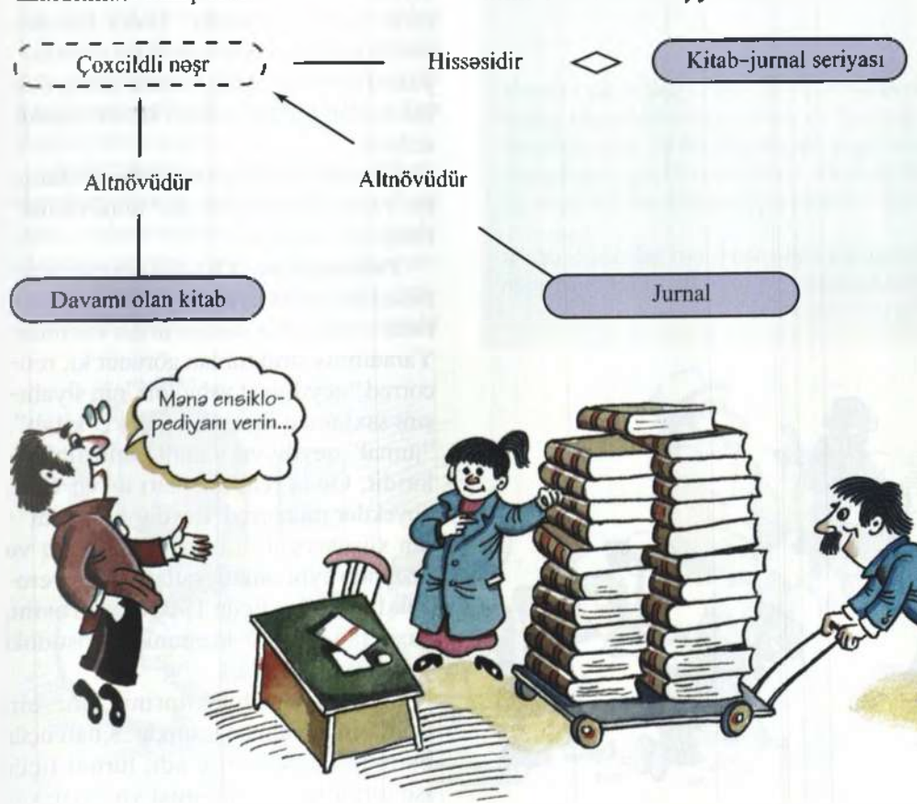
Doğrudan da, "kitab" kitabxanada əsas anlayışdır. Bu, bir müəllifin kitabı, hekayələr məcmuəsi, jurnal, tikmə qəzet də ola bilər. Kitablar bir neçə cilddən ibarət ola bilər, onların davamı ola bilər. Əgər belə obyektlərin hər biri ayrıca, müstəqil sinif kimi təsvir olunsa, onda sistemdə abstraktlaşmaların sayı çox



olacaqdır. Bundan savayı, oxuculara verilən jurnalları, kitabları və s. obyektləri ayrıca nəzərə almaq lazım gələcəkdir. Əgər cəmi bir dənə "hesabat vahidi" obyekti yaradılsa, onda bu sinif də həddən artıq mürəkkəb olacaqdır. Başqa sözlə, obyekt-yönlü yanaşmanın əsas göstəricilərindən biri: tamın fraqmentlərə bölünməsi yolu ilə mürəkkəbliyin azaldılması itmiş olur.

Bu halda abstraktlaşmanın *iyerarxiyalı* strukturundan – *altnövdən* istifadə olunur. Kitabxanada saxlanılan və oxuculara verilə bilən "hesabat vahidi" abstraktlaşmasını əsas sinif kimi təyin edək. "Kitab", yaxud "jurnal" hesabat vahidinin altnövü olacaqdır.

"Jurnal" sinfində çıxma (nəşr) ili və ayı, müəlliflərin və məqalələrin siyahısı kimi əlavə xüsusiyyətlər daxil etmək



OBYEKT-VARİSLƏRİN VALİDEYN OBYEKTLƏR KİMİ İSTİFADƏ OLUNMASI

```
/* Oxucular tərəfindən "vaxtı ötmüş" kitabların siyahısını
qaytaran funksiya */
BiblioItems Reader: :CheckDue()
{
  /* Oxucuların əlində olan bütün kitabların siyahısını almaq */
  BiblioItems bi = GetItems();
  /* Qaytarılma vaxtı ötmüş kitabların siyahısı yerləşdirilən
  obyektin yaradılması */
  BiblioItems due;

  /* Oxucuların əlində olan bütün kitablar üçün dövr yerinə
  yetirmək */
  for ( bi.First(); !bi.Last(); bi.GetNext())
  {
    /* Əgər kitabın qaytarılma tarixi bu günkü tarixdən kiçikdirsə, demək, oxucu bu kitabı vaxtında qaytarmamışdır və
    onu "vaxtı ötmüş" kitablar siyahısına əlavə etmək lazımdır */
    if ( bi.Current().GetReturnDate() < Today())
    due.Add( bi.Current());
  }

  return due;
}
```

Bu funksiyada oxucuların əlində olan kitablar siyahısında "kitab", "jurnal", "məcmuə" və digər tip obyektlərə rast gəlinə bilər. Madam ki, onlar üçün "qeydiyyat vahidi" baza sinfində qaytarma tarixini almaq üçün GetReturnDate() funksiyası elan olunmuşdur, onda bu funksiyanı obyekt-varislər üçün də çağırmaq olar.

Əks halda bir neçə siyahı saxlamaq lazım gələrdi: oxucular tərəfindən götürülmüş kitabların siyahısı, oxucular tərəfindən götürülmüş jurnalların siyahısı və s.



lazımdır. "Kitab" özü də müxtəlif müəlliflərin əsərlərinin məcmuəsi ola bilər.

Əgər hekayələr məcmuəsi olan "kitab" üçün ayrıca obyekt yaratsaq, onda eyni bir kodun tamamilə müxtəlif yerlərdə istifadə olunacağı alınacaq, onun reallaşdırılması çətinləşmiş olacaq. Ona görə də, yaxşı olar ki, bir dənə də iyerarxiya münasibəti qurulsun: "jurnal" və "hekayələr məcmuəsi" mücərrəd "məcmuə" elementinin altnövü olur.

Hesabatlar və axtarış üçün bəzən jurnalların illik tikmələrinə və çoxcildli romanlara vahid tam kimi baxmaq faydalı olur. O zaman iyerarxiyanın bir növü də: "hissəsidir" münasibəti meydana gəlir. Çoxcildli nəşr ümumi seriyanın hissəsidir. Bəzən deyirlər ki, "altnövüdür" əlaqəsi siniflərin iyerarxiyası, "hissəsidir" əlaqəsi isə obyektlərin iyerarxiyasıdır. Daha ümumi sinfin altnövü olan siniflər *varislər*, yaxud *nəsillər*, daha ümumi sinfin özü isə *əcdad*, yaxud sadəcə *valideyn* sinfi adlanır.

Obyekt-yönlü proqramlaşdırmanın bir vacib xüsusiyyəti də "altnövüdür" əlaqəsilə bağlıdır.

Polimorfizm. "Oxucu" obyekti üçün oxuculara verilmiş kitab, jurnal və qəzetlərin siyahısının saxlanılması vacibdir. Yaradılmış strukturdan görünür ki, mücərrəd "qeydiyyat vahidləri"nin siyahısını saxlamaq lazımdır. Çünki "kitab", "jurnal" qeydiyyat vahidlərinin altnövləridir. Onda real kitabları təsvir edən obyektlər mücərrəd "qeydiyyat vahidi"nin xüsusiyyətlərinə malik olacaq və üzərində eyni əməliyyatlara icazə verəcəkdir. Onlar üçün ISBN nömrəsini, daxili nömrəni təyin etmək, qəfəsədəki yerini tapmaq olar.

Qeydiyyat obyektlərinin hər biri üçün tam adı almaq lazımdır. Kitab üçün bu müəllif və əsərin adı, jurnal üçün isə jurnalın adı, nömrəsi və çıxma ili

olacaqdır. Müxtəlif siniflər üçün müxtəlif funksiyaların təyini (məsələn, GetMagazineDescription, GetBookDescription) iyerarxiyalı strukturu korlayar, diqqətlə qurulmuş iyerarxiyanı heçə endirərdi.

Əsas "qeydiyyat vahidi" sinfində funksiya daxil edildikdən (məsələn, GetDescription()) və qalan siniflərdə də o, yenidən təyin olunduqdan sonra, "kitab" sinfində təsvir müəllifin adından və əsərin adından ibarət olacaqdır, çoxcildli roman üçün yəqin ki, sıra nömrəsi, məcmuə üçün – tərtibatçının adı, jurnal üçün – nömrə və buraxılış ili əlavə olunacaqdır. Əgər belə xüsusiləşdirilmiş obyektlərin hər biri "qeydiyyat vahidi" valideyn sinfinin nümayəndəsi kimi istifadə olunacaqsa, onda bu funksiyaya (GetDescription) həmişə giriş olacaqdır. Özü də istifadə olunan obyektin hansı alt tipə aid olduğunu çağıran obyektin bilməsi vacib deyildir. Obyekt-yönlü proqramlaşdırma dilinin özü hər şeyin qayğısına qalacaq və alt sinfin uyğun funksiyasını çağıracaqdır.

Belə "davranışı" *polimorfizm* adlandırırlar: obyektlətin müxtəlif sinifləri üçün, onların valideyn siniflərində təyin olunmuş ümumi funksiyalar yığını vardır. Bu funksiyalar varis-siniflərdə yenidən təyin oluna bilərlər. Hər bir obyekt

POLİMORFİZM

```
BiblioItems Reader::RemindDue ( )
{
   /* Qaytarılma vaxtı çatmış bütün kitabların siyahısının tərtib
olunması */
   BiblioItems due = GetDue();
   /* Yeni elektron məktub yaratmaq və onu oxucuya ünvanlamaq
*/
   MailMessage mail ( Reader.EMailAddress ) ;
   /* Məktubun mətninə nəzaktəli xatırlatma əlavə etmək */
   mail.addText ( "Sizə xatırladırıq ki, aşağıdakı kitabların qaytarılma
vaxtı çatmışdır:" ) ;

   /* Qaytarılmalı olan bütün kitablar üçün dövrü yerinə yetirmək */
   for ( due.First(); ! due.Last (); due.GetNext () )
   {
   /* Məktubun mətninə kitabın geniş adını yazmaq */
   mail.addText( due.Current() .GetDescription() ) ;
   }

   /* Məktubu oxucuya göndərmək */
   mail.Send();
   }
```

Bu funksiyada məktubun mətni tərtib edilərkən qaytarılmalı olan bütün kitabların siyahısına baxılır. Siyahıdan hər bir obyekt üçün geniş izah (müəllifin adı, kitabın adı, ola bilsin ISBN) verən GetDescription() funksiyası çağrılır. Lakin əvvəlki nümunədən fərqli olaraq burada bu funksiya müxtəlif "qeydiyyat vahidi" altsiniflərində müxtəlif cür təyin olunmuşdur.

İstifadə olunan obyekt-yönlü proqramlaşdırma sistemi hər bir obyekt üçün düzgün funksiyanın çağırılmasının qayğısına qalacaq.





Börn Straustrup.

üçün, ona verilmiş sinfin, yaxud valideyn-sinfin obyekti kimi müraciət olunmasından asılı olmayaraq, uyğun funksiya çağırılır.

Tipləşdirmə. Effektiv və düzgün işləyən proqramın yaradılması üçün mümkün qədər tez bütün əməliyyatların yol verilənliyini yoxlamaq lazımdır. Bunu proqramın yerinə yetirilməsi zamanı deyil, onun yaradılması mərhələsində etmək məqsədəuyğundur.

Obyektlərin yaradılması zamanı onların strukturunu və yol verilən əməliyyatları təsvir edən yeni verilənlər tipi meydana gəlir. Proqramlaşdırma dillərində xüsusi mexanizm yerləşdirilmişdir. Bu mexanizm müxtəlif verilənlər tipindən olan obyektlərin istifadəsini izləyir və bir verilənlər tipi üçün nəzərdə tutulmuş əməliyyatların başqa tipin obyektləri üzərində yerinə yetirilməsinə imkan vermir.

Məsələn, əgər sistem elektron məktubu oxucuya deyil, "kitaba" göndərərsə, elə bir qorxulu şey baş verməyəcəkdir. Yəqin ki, məktub sadəcə olaraq kompüter mühitində itib-batacaqdır. Lakin əgər oxucunun götürdüyü kitablar siyahısına "qəfəsə", yaxud başqa bir "oxucu" düşərsə, artıq bu xoşagəlməz hal olacaqdır. Daha həssas oxucular təcili olaraq kitabxanaya bütöv kitab dolu qəfəsin qaytarılması xahişilə məktub aldıqda bayıla bilərlər.

Belə halların baş verməməsi üçün, obyekt-yönlü proqramlaşdırma dillərində *tipləşdirmə* mexanizmi, yaxud bir sinfin obyekti əvəzinə digərindən istifadədən daxili müdafiə üsulu vardır. Bu mexanizm ya proqramların yazılması mərhələsində (o zaman səhvi tapmağa cəhd edən proqramçı yorğunluqdan bayılacaqdır), ya da yerinə yetirilməsi zamanı (proqram dayanacaqdır, ancaq hər halda dəhşətli məktub oxucuya göndərilməyəcəkdir) işə düşəcəkdir.

OBYEKT-YÖNLÜ PROQRAMLAŞDIRMA DİLLƏRİ

SmallTalk. SmallTalk-72 və SmallTalk-74 versiyalarında obyekt-yönlü proqramlaşdırmanın əsasları qoyuldu, ancaq siniflərin varislik mexanizmi hələlik yox idi (sinifləri olan, lakin varisliyi olmayan belə dili obyekt-yönlünün əksinə olaraq obyektli proqramlaşdırma dili adlandırmaq olar).

SmallTalk təkcə proqramlaşdırma dili deyil, mətn redaktoru, sinfin ierarxiyasına baxmaq üçün vasitələri, qrafik istifadəçi interfeysi kitabxanaları olan bütöv bir hazırlanma (gəlişdirmə) mühiti idi. Onun SmallTalk-80 versiyası geniş tanındı və bir çox kompüter platformalarına keçirildi.

Object Pascal. Object Pascal proqramlaşdırma dili Apple Computers firmasının əməkdaşları tərəfindən Pascal prosedur dilinin müəllifi Niklaus Virtlə birlikdə yaradılmışdır.

Object Pascal dili 1986-cı ildə Macintosh kompüterləri üçün MPW (Macintosh Programmer's Workshop) hazırlanma mühitinə daxil edilmişdir. Bu dil fərdi kompüterlər üçün ilk obyekt-yönlü hazırlanma dili olmuşdur. O, proqramçıya rahat alətlər, zəruri olan bütün kitabxanaları verməklə yanaşı, proqramların məhsuldarlığını itirmədən, obyekt-yönlü yanaşmanın bütün üstünlüklərindən istifadə etməyə imkan vermişdir.

Eyni zamanda dilin sadələşdirilməsi üçün ondan coxqat varislik ("jurnal" sinfi bir çox siniflərin – "qeydiyyat vahidi", "çoxcildli nəşr" və "məcmuə" siniflərinin varisidir), qlobal fuksiyalar və sinfin dəyişənləri kimi bir çox imkanlar çıxarılmışdır. Məsələn, kitabxanada kitabların ümumi sayını "qeydiyyat vahidi" sinfinin qlobal dəyişəni kimi təyin etmək olar. Yeni obyektin yaradılması (kitabların

alışı) zamanı o artır, obyektin uzaqlaşdırılması (hesabdan silmə) zamanı isə azalır.

C++. AT&T Bell Laboratories firmasının əməkdaşı Börn Straustrup tərəfindən C və Simula dillərinin bazasında "C with Classes", yəni "Siniflərlə C" dili yaradıldı. Sonradan onun əsasında tək varisliliyi, metodların yüklənməsini, ciddi tipləşdirməni və polimorfizmi dəstəkləyən əsl obyekt-yönlü C++ dilini işləyib hazırladılar.

C++ dilinin bir neçə standartı çıxıb ki, onların hər birində geniş istifadə təcrübəsi əsasında dil yenidən qurulmuş, zəruri imkanlarla tamamlanmışdır. C++ dilinin ikinci versiyasına çoxqat varislik daxil edilmiş, üçüncü versiyada proqramın səhvlərə reaksiyası mexanizmi, qabaqcadan nəzərdə tutulmamış hadisələr – istisna mexanizmi yenidən işlənmişdir. Bu mexanizm də obyekt-yönlü əsasda qurulmuşdur, yalnız iyerarxiyaya düzülmüş, mümkün səhvlər tipini təsvir edən xüsusi siniflərin obyektləri istisna təşkil etmişdir.

C++ dili o dövr üçün bəlkə də ən effektiv və populyar proqramlaşdırma dili əsasında yaradıldığından, o özü də proqramların yerinə yetirilmə sürətinə görə effektiv alınmışdır (daxilən gözəl, lakin çox ləng SmallTalk dilindən fərqli olaraq). C və C++ dillərinin sintaksisindəki oxşarlıq proqramçılara yeni dilin yeni imkanlarını tədricən mənimsəyərək, ona asanlıqla keçməyə imkan vermişdir.

XXI əsrin əvvəllərində C++ dili ən məşhur proqramlaşdırma dillərindən biridir. Bu dilin kompilyatorları və proqramlaşdırma sistemləri (həm kommersiya, həm də açıq kodlu havayı) praktiki olaraq bütün kompüter platformalarında vardır.

Java. Java dili ("cava" kimi oxunur) qurğularda yerləşdirilmiş kompüterlər üçün proqram təminatının yaradılması üçün xüsusiləşdirilmiş dil kimi meydana gəlmişdir.

C# dilinin adı "si şarp" kimi tələffüz olunur. # – sharp, notu yarım ton qaldıran "diyez" musiqi simvolunun ingilis dilində adıdır. Adın özündə hiyləgər söz oyunu gizlənmişdir. Bir tərəfdən sharp "kəskin", "nüfuzedici", "çevik", "məharətlə" mənalarını, yəni yeni dilin müsbət xarakteristikalarını göstərir. Digər tərəfdən, "+" simvolu ilə "bemol", yəni notu yarım ton aşağı salan simvol işarə edilir. Bu o deməkdir ki, C# dili C++ dilindən üstündür, yaxşıdır.

XX əsrin 90-cı illərində Sun Microsystems firmasının əməkdaşı Patrik Noton müxtəlif qurğular üçün çoxlu sayda eyni tipli proqramlara dəstək işinin yerinə yetirilməsinin gedişində ümidsizliyə qapılmağa başladı. O, artıq şirkəti tərk edib başqa işə keçməyə hazırlaşdığı vaxtda ona meydana çıxmış problemin həlli layihəsini tərtib etməyi təklif etdilər. Həmin sənəd nəinki şirkətin digər mühəndisləri tərəfindən, hətta onun rəhbərliyi tərəfindən də dəstəkləndi. Dərhal yerləşik sistemlər üçün xüsusi obyekt-yönlü dilin yaradılması üzrə qrup yaradıldı.



Yeni proqramlaşdırma dilinin əsası olaraq həmin an üçün ən populyar obyekt-yönlü dil olan C++ götürüldü. Lakin yaddaş və prosessorun gücü kimi maşın resurslarına məhdudiyyətlərlə əlaqədar yerləşik sistemlərdə ondan tam şəkildə istifadə etmək mümkün olmadı. Ona görə də onun sintaksisi yenidən quruldu, sadələşdirildi və daha ciddi edildi.

Yeni dilin digər xüsusiyyəti ondan ibarət oldu ki, o, hər bir konkret qurğu üçün maşın koduna deyil, müəyyən bir maşın üçün xüsusi aralıq koda kompilyasiya olunurdu. Yerləşik sistemlərdə bu maşınların funksionallığını təmin edən və alınmış kodu yenidən kompilyasiya etmədən qəbul etməyə imkan verən xüsusi icra mühiti vardı.

Alınmış proqramın digər kompüter platformasına (əlbəttə, onun uyğun icra mühiti olan platformaya) keçirmək üçün onu yenidən kompilyasiya etmək lazım gəlmir, çünki ikilik kod prosessordan asılı deyil və xüsusi interpretator vasitəsilə yerinə yetiriləcək.

WWW kompüter şəbəkələrinin ildırım sürətilə məşhurlaşmasıyla əlaqədar olaraq bu dil yüksək şöhrət qazandı. WWW-səhifələrə baxış zamanı geniş şəkildə istifadə olunan çox da böyük olmayan proqramların – apletlərin yaradılmasına bu dil çox uyğun gəldi. Serverdə aralıq kod şəklində yeganə aplet saxlanılır ki, o, müştəri brauzeri tərəfindən yüklənir və ixtiyari sistemdə yerinə yetirilə bilir.

C#. C# dili 2000-2001-ci illərdə Microsoft şirkəti tərəfindən yeni Microsoft.NET platforması üçün əsas proqramlaşdırma dili kimi yaradılmışdır. Yeni dilə keçid prosessini asanlaşdırmaq üçün onun sintaksisi əsasən ən məşhur C++ və Java dillərindən götürülmüşdür. Bəziləri hətta onu Java dilinin kommersiya məqsədilə yaradılmış dəqiq surəti adlandırırlar, ancaq bu belə deyil. Java dili C++ dilinə nə qədər oxşayırsa, C# da Java dilinə bir o qədər oxşayır; sintaksisin aşkar üst-üstə düşməsiylə yanaşı yetərincə böyük fərqlər də vardır.

Məsələn, Java dilində, hətta ən sadə tip, ədəd, yaxud simvol da həmişə uyğun sinfin obyektləri olur ki, bu da proqramların məhsuldarlığını önəmli dərəcədə aşağı salır. C, yaxud C++ dillərində olduğu kimi C# dilində isə ədəd sinfin obyekti də ola bilər, adi dəyişən də.

OBYEKT-YÖNLÜ MICROSOFT.NET PLATFORMASI

Java kimi C# dili də müxtəlif maşınlar üçün eyni olan aralıq koduna kompilyasiya olunur. Ancaq Java dilindən fərqli olaraq, bu psevdo-Assembler aralıq kodunun özü də obyekt-yönlüdür. Başqa sözlə, kompilyasiya vaxtı proqramın obyekt strukturu haqqında informasiya itmir. Bu MSIL (MicroSoft Intermediate Language) aralıq kodu yeni Microsoft.NET platformasının əsas "tikinti bloklarından" biridir.

Belə qeyri-standart həll ayrı-ayrı hissələri müxtəlif dillərdə yazılmış və bu zaman hissələr arasında qarşılıqlı əlaqənin sadəliyinin və varislik imkanlarının təmin olunduğu obyekt-yönlü böyük sistemlərin yaradılmasına imkan verir. Başqa sözlə, sinif-valideyn C# dilində reallaşdırılmış modulda, törəmə sinif isə MSIL-də kompilyasiyanı dəstəkləyən Eiffel, Python, SmallTalk, Pascal və sair dillərdə reallaşdırılmış modulda yerləşə bilər. Proqram vasitələrinin müxtəlifliyinə baxmayaraq bir obyektin başqa obyektin xidmətlərinə müraciəti zamanı da heç bir problem yaranmayacaqdır.

Microsoft.NET platforması hələ çox gəncdir və buna görə də dəqiq proqnozlar vermək çətindir. Ancaq çox ehtimal ki, nəhayət məhz o, proqramlaşdırmanın adi problemini, yəni artıq yaradılmış

kodun təkrar istifadəsinin qeyri-mümkünlüyünü həll edəcəkdir.

Obyekt-yönlü proqramlaşdırmanın yarandığı andan əsas məsələlərdən biri artıq yazılmış kodun təkrar istifadəsi olmuşdur. Minlərlə proqramçı praktiki olaraq eyni bir kodu yenidən yazır, çünki artıq hazır olan koddan bu və ya digər texniki səbəblərə (məsələn, proqramlaşdırma dillərindəki, yaxud hətta eyni bir dilin dialektlərindəki fərqlərə) görə birbaşa istifadə etmək mümkün deyil.

Bəzən kodun hazır fraqmentləri xarici sistem kitabxanası kimi tərtib olunur ki, bu da eyni bir platformada, müxtəlif proqramlaşdırma dillərində yazılmış proqram hissələrindən istifadəyə imkan verir. Amma bu zaman obyekt-yönlülük itir, axı kitabxana öz daxili quruluşunu tamamilə gizlədərək "çıxışa" yalnız funksiyalar toplusunu verir.

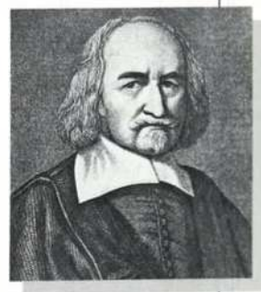
Microsoft.NET platformasında *yığmalar* adlanan xüsusi kodlar kitabxanası meydana çıxır. Bu yığmalarda proqram fraqmentləri, müxtəlif xidmətləri təmsil edən ayrı-ayrı modullar saxlanıla bilər. MSIL-dən istifadə sayəsində bu yığmalar sonrakı işlərdə proqram komponentlərini təkrarən tətbiq etməyə imkan verir, çünki bu aralıq dilin özünün obyekt-yönlülüyü həmin proqram komponentlərində təsvir olunmuş siniflərin varislik imkanını təmin edir.

SÜNİ İNTELLEKT DİLLƏRİ

Əsrlər boyunca insan ədədlər üzərində əməliyyatlar aparmış və eyni zamanda onların yazılış formasını, hesablama üsullarını təkmilləşdirmiş, bu əməliyyatları asanlaşdırmaq üçün daha mürəkkəb texniki qurğular fikirləşmişdir. Lakin çox az insan hesablamanı daha geniş başa düşmək haqqında fikirləşmişdir. Məsələn, böyük ingilis filosofu Tomas Qobbs (1588-1679) özünün "Elementa philosophiae" əsərində yazmışdır: "Fikirləşmək lazım deyildir ki, hesablama ancaq ədədlərlə aparıla bilər". XIX əsrdə Ada Lavleys demişdir: "Maşın ədədi qiymətlər kimi hərfləri və digər ümumi xarakterli simvolları da nizamlaya, kombinasiya edə bilər. Mahiyyətcə, uyğun şərtlər ödənildikdə o, nəticələri cəbri formada da verə bilər".

Lakin ilk kompüterlərin meydana gəlməsinə qədər belə ideyalar praktiki həyata keçməkdən yetərincə uzaq idi. Kompüterlər də ilk vaxtlar əsasən ədədlər üzərində əməliyyatlar üçün istifadə olunurdu, baxmayaraq ki, Konrad Süze artıq 1945-ci ildə şahmat oyunu üçün proqram yazmışdı. Alimlər çox tezliklə kompüterlərin ədədi olmayan verilənlərin emalı üçün tətbiq olunmasının mümkünlüyü haqqında ciddi şəkildə düşünməyə başladılar. Söhbət ilk növbədə *süni intellekt* sahəsindən olan məsələlər haqqında gedirdi; onlara insan şüurunda gedən prosesslərin modelləşdirilməsi, teoremlərin avtomatik isbatı, robotun hərəkətlərinin planlaşdırılması və digər proseslər aid edilirdi. Onların həlli üçün *simvollarlı hesablamalardan* (ədədlər deyil, simvollar emal olunan) istifadə olunması xarakterikdir.

FORTRAN, yaxud Algol-60 kimi erkən proqramlaşdırma dilləri simvolların emalı məsələlərinin həlli üçün uyğunlaşdırılmamışdı. Lazım olan xüsusiyyətlərin bu dillərə daxil edilməsi cəhdləri (məsələn, IBM korporasiyası 1958-ci ildə FORTRAN dilinin genişlənməsini – FLPL dilini yaratdı) isə uğurlu olmadı. Buna görə də süni intellekt məsələlərinin həlli üçün xüsusi dillərin yaradılmasına başlanıldı.



Tomas Qobbs.



Konrad Süze.



Con Mak-Karti.

LISP

İlk belə dil 1959-cu ildə meydana çıxan LISP oldu (*ing.* list prosessing – "siyahıların emalı"). Onun müəllifi informatikanın inkişafına böyük töhfələr vermiş görkəmli amerika alimi Con Mak-Karti idi. Yeri gəlmişkən, "süni intellekt" terminini də ilk dəfə o işlətmişdir. İlk öncə Mak-Kartinin proqramlaşdırma dili yaratmaq məqsədi yox idi. Onun məqsədi Türinq maşınlarından daha əlverişli yeni hesablama modelinin yaradılması idi.

İş prosesində aydın oldu ki, belə modelin əsası simvol siyahılarının rekursiv emalı ola bilər. Ondan istifadə etməklə Mak-Kartiyə cəbri ifadələrin simvollara ayrılması məsələsinin həlli üçün olduqca qəşəng alqoritm tapmaq müyəssər oldu. Lakin o, alqoritmi siyahıların dinamik şəkildə yaddaşa yerləşdirilməsi və yaddaşdan silinməsi operatorlarının aşkar daxil edilməsi kimi texniki elementlərlə son dərəcə ağırlaşdırmaq istəməsə də, FLPL dilindən istifadə zamanı bu qaçılmaz oldu. Üstəlik, FLPL rekursiyanı da dəstəkləmirdi. Onda Mak-Karti başa düşdü ki, ona yeni dil lazımdır. Beləliklə, simvolların emalının əlverişli vasitələrinə tələbat LISP dilinin (onu "təmiz LISP" də adlandırırlar) birinci versiyasının meydana gəlməsi üçün zəmin oldu.

Yeni hesablama modelinin sonrakı tədqiqatı Mak-Kartini LISP dilinin ixtiyari funksiyasını hesablaya bilən universal funksiyanın yaradılması fikrinə gətirdi (istənilən digər Türinq maşınlarının işini imitasiya edən universal Türinq maşınına bənzər). Belə funksiya (o, EVAL adını aldı) yaradıldı və təsviri 1965-ci ildə dərc olunmuş LISP 1.5 dilinin interpretatoru kimi istifadə olundu. Məhz bu andan dilin geniş yayılması başlanır. O, ildırım sürətilə şöhrət qazanır; demək olar ki, süni intellekt sahəsində işləyən bütün tədqiqatçılar öz proqramlarını bu yeni dildə yazırlar.

Dil təkmilləşir, onun çoxlu versiya və dialektləri meydana gəlir (onların içərisindən 1975-ci ildə yaradılmış, maraqlı Scheme dilini xüsusi qeyd etmək olar), ancaq eyni zamanda proqramların keçirilməsi (bir dialektdən, versiyadan başqasına) problemi meydana çıxır. Onun həlli üçün 1984-cü ildə dilin vahid standartı – Common LISP hazırlandı.

LISP *funksional* proqramlaşdırma dilləri sinfinə aiddir. Adi (*imperativ*) prosedur dillərdən fərqli olaraq, onlar nə dəyişəndən, nə də mənimsətmə buyruğundan istifadə edir. Bu dillərdə dövr buyruqları olmur, əməllərin təkrarlanması isə rekursiyanın köməyilə reallaşdırılır. Belə dillərdə proqram funksiyanın təsvirindən, proqramın yerinə yetirilməsi isə funksiyanın arqumentlərə tətbiqi prosesindən ibarətdir. Doğrudur, müasir LISP dilini sözün tam mənasında funksional dil adlandırmaq olmaz (yalnız təmiz LISP belə olmuşdur). O, prosedur dillərə xas bəzi xüsusiyyətlərə də malikdir.

LISP dilində ancaq iki verilənlər tipi – *atomlar* və *siyahılar* istifadə olunur. Atomlar ənənəvi dillərdəki adi identifikatorlara çox bənzəyir; bundan başqa, ədədi sabitlər də atomlar hesab olunur. Mötərizələrə alınmış elementlər ardıcıllığını siyahılar adlandırırlar; bu elementlər istər atomlar, istərsə də başqa siyahılar ola bilər. Məsələn, L = (A (B) C (D E F) (G H)) siyahısı beş elementdən ibarətdir. Onlardan birinci və üçüncü – A və C atomları, qalanları isə siyahılardır.



Diqqət edin ki, ikinci element (B) atom deyil, bir B atomundan ibarət olan siyahıdır.

LISP elementar (yaxud sistem) funksiyalar yığınından ibarətdir ki, onlardan da ən əsasları (CAR L), (CDR L) və (CONS M L) funksiyalarıdır. CAR funksiyası L siyahısından birinci elementi ("başı") seçir, CDR funksiyası L siyahısından "başı" uzaqlaşdırır ("quyruğu" saxlayır). Bizim misalda (CAR L) = A, (CDR L) = ((B) C (D E F) (G H)). Bu iki funksiya siyahını hissələrə "ayırır", CONS funksiyası isə M elementini L siyahısının "başına" yerləşdirərək yeni siyahı yaradır: (CONS M (B C D)) = = (M B C D)

Göstərilən üç funksiya ilə dilin elementar funksiyalar yığını, aydındır ki, tükənmir. Bundan başqa, dil elementar funksiyalardan daha mürəkkəb funksiyaları qurmağa imkan verən rahat vasitələrə də malikdir.

LISP dilində verilənlərin və proqramların yazılışının vahid forması qəbul edilmişdir. Məsələn, (F X Y) ifadəsi ya üç elementdən ibarət siyahı kimi, ya da X və Y arqumentlərinə malik F funksiyasına müraciət kimi qəbul edilə bilər. Bunun nəticəsində istifadəçinin proqramı iş prosesində təkcə yeni funksiyaların və proqramların yaradılmasına deyil, həm də, dərhal onları yerinə yetirməyə qadirdir. Dilin bu imkanı onun unikal özəlliyidir (axı imperativ dildəki proqramda bütün buyruqlar onun işləməsinə qədər yaradılmalıdır və iş prosesində dəyişdirilə bilməzlər) və proqramçıların əlində çox güclü silahdır.

PROLOG

Süni intellekt məsələlərinin həllinə alternativ yanaşma Prolog (Programming in logic) dilində reallaşdırılan *məntiqi proqramlaşdırma* konsepsiyası oldu. Onu XX əsrin 70-ci illərində Marsel universitetindən Alen Kolmerye, Filipp Russel və Edinburq universitetindən Robert Kovalski işləyib hazırladılar. Onlar öz ideyalarını demək olar ki, 10 il inkişaf etdirsələr də, çox da tanınmadılar. Hər şey 1981-ci ildə, "yapon çağırışı"nın – beşinci nəsil EHM-nın yaradılması ilə bağlı Yaponiya hökümətinin irimiqyaslı layihəsinin meydana çıxmasından sonra dəyişdi. Gələcək kompüterlərin intellektual səviyyələrinin kəskin şəkildə artırılması və onlarla təbii dilə maksimum yaxın olan dildə ünsiyyətin mümkünlüyü layihənin əsas məqsədləri idi. Prolog dili yaradılan sistemlərin əsası oldu ki, bu da ona olan ümumi diqqətin artmasına və onun populyarlığının bütün dünyada kəskin şəkildə artmasına səbəb oldu.

Məntiqi proqramlaşdırma nədir? İmperativ və funksional dillərdə proqramçı bilir ki, proqram nəyi yerinə yetirməlidir. Onun vəzifəsi kompüterə hesablamaları necə (hansı ardıcıllıqla) yerinə yetirmək barədə göstərişlər verməkdir. Məntiqi proqramlaşdırma köklü şəkildə həm imperativ, həm də funksional proqramlaşdırmadan fərqlənir. Prolog dilində yazılmış proqramda nəticələrin necə hesablanması barədə göstərişlər yoxdur. O, ancaq nəticənin tələb olunan şəklini təsvir edir. Nəticənin alınması dilə daxil edilmiş məntiqi çıxış mexanizmlərinin (*rezolyusiyalar metodu* deyilən metoda əsaslanan) işidir. Prolog dilində proqram buyruqlardan ibarətdir. Buyruqlar *faktlara* və *qaydalara* bölünür. Fakt həmişə doğru qəbul edilən şərtsiz müddəadır. Dilin buyruqları şəklində yazılmış faktlara misallar göstərək:

fil (cumbo).
qurbağa (flik).
güclüdür (cumbo, flik).

Bu onu göstərir ki, cumbo obyekti fildir, flik obyekti qurbağadır, eləcə də fil qurbağadan güclüdür.

LISP 20 ildən çox süni intellekt sahəsində şəksiz hökmranlıq etmişdir. Ancaq bu gün də çoxsaylı rəqiblərin meydana çıxmasına baxmayaraq o, yenə də bu sahədə məsələlərin həlli üçün ən geniş yayılmış dildir.



Prolog ən müxtəlif məsələlərin həlli üçün istifadə olunur. Onun tətbiq sahələrindən ən vacibi ekspert sistemlərinin yaradılmasıdır.

Buyruqların başqa növü daha mürəkkəbdir. Prolog dilində qayda bu şəkildə yazılır: Y: – X, Z... Bu yazılış müəyyən məntiqi çıxarışı təyin edir və belə oxunur: "Y doğrudur, əgər X və Z və ... doğrudursa". Deməli, qaydanın sağ tərəfindəki vergül məntiqi VƏ (konyunksiya) kimi başa düşülür.

güclüdür (cumbo, flik) :- fil (cumbo), qurbağa (flik).

qaydası onu göstərir ki, cumbo obyekti flik obyektindən güclüdür, əgər cumbo fil, flik isə qurbağadırsa. Başqa sözlə, qaydadan faktlar arasındakı məntiqi münasibətləri təsvir etmək üçün istifadə olunur.

Prolog dilində proqramçının işinin məqsədi hər hansı bir vəziyyəti, yaxud predmet sahəsini tam şəkildə təsvir edən faktlar və qaydalar yığınını yaratmaqdır. Bundan sonra Prolog-sistemə sualları *məqsəd* adlanan cümlələr şəklində vermək olar (məsələn, klaviaturadan daxil etməklə). Prolog-sistem məqsədi yoxlamalı, onu ya təsdiqləməli, ya da təkzib etməlidir.

Prolog dilində proqramın prinsipial xüsusiyyəti ondan ibarətdir ki, onun işinin nəticəsi sistemə (ola bilsin, proqramçının özünə də) məlum olmayan yeni faktların alınması ola bilər. Məhz dilin bu xüsusiyyəti süni intellekt sistemlərinin tədqiqatçılarının və yaradıcılarının ona diqqətini təmin edir.



Karl Hyuitt.

Verilənlər bazasının idarəolunması sistemlərinin (VBİS) və təbii dillərdə mətnlərin emalı sistemlərinin yaradıcıları, eləcə də başqa mütəxəssislər Prolog dilinə (ümumiyyətlə, məntiqi proqramlaşdırmaya) böyük ümidlər bağlayırlar. Hətta çoxları hesab edir ki, məhz Prolog gələcəkdə daha sadə və daha etibarlı proqram təminatının yaradılmasında əsas dil olacaqdır. Proqramlaşdırmanın əsaslarının öyrədilməsinə məhz Prolog dilindən başlanılmasının gərəkliyi fikrinə bir çox tədqiqatçılar və pedaqoqlar artıq bu gün əmindirlər.

Eyni zamanda, belə fikir də var ki, indiyə qədər Prolog nəhəng olsa da, yalnız bir eksperimentdir. İş ondadır ki, dildən istifadə edərkən, hələlik qaneedici həlli olmayan xeyli problem yaranır. Bu problemlərdən ən başlıcası proqramların yerinə yetirilməsinin effektivliyinin artırılmasıdır. Bəli, ümumiyyətlə, məntiqi proqramlaşdırma dillərində proqramların yaradılması üsulları hələ sona qədər formalaşdırılmayıb.

PLANNER

LISP və Prolog simvolların emalı üzrə iki ən populyar dildir. Bununla belə, geniş yayılmamış digər dillərin yaradıcıları tərəfindən də maraqlı fikirlər təklif olunmuşdur. Belə dillərdən biri indi nadir hallarda xatırlanan Planner dilidir. O, öz adını ("planlaşdırıcı" kimi tərcümə olunur) əsasən robotların hərəkətlərinin planlaşdırılması sistemlərinin yaradılması üçün nəzərdə tutulmasına görə almışdır. Planner 1971-ci ildə Massaçusets Texnologiya İnstitutunda Karl Hyuitt tərəfindən yaradılmış və simvollu informasiyanın emalı ideyalarının inkişafında ciddi mərhələ olmuşdur.

Planner dili LISP dilinin bütün imkanlarına malikdir, yəni faktiki olaraq LISP onun altçoxluğudur. Lakin hazır funksiyalar yığını Planner dilində xeyli

genişdir, onların özləri isə LISP dilinin uyğun funksiyalarından olduqca güclüdür. Məsələn, ELEM və REST funksiyaları siyahıdan ixtiyari elementləri ayırmağa və çıxarmağa imkan verməklə CAR və CDR funksiyalarını ümumiləşdirir: [ELEM 1 L] = A, [REST 2 L] = = (C (D E F) (G H)). Bundan başqa, yazılara müxtəlif tipli mötərizələrin daxil edilməsi (LISP dilində ancaq dairəvi mötərizələrdən istifadə etmək olar ki, bu da proqramın oxunuşunu çətinləşdirir) proqramı daha anlaşıqlı edir. Ən başlıcası isə dil əlavə imkanlarla zənginləşdirilmişdir.

Onlardan birincisi örnəyə görə verilənlərin axtarışı və analizidir. Oxşar mexanizm Snobol dilində uğurla sınaqdan çıxarılıb və indi geniş şəkildə istifadə olunur. Məsələn, Norton Commander-də *.doc tipli faylların axtarışı verildikdə, məhz bu əməliyyat yerinə yetirilir.

İkinci əlavə imkan, süni intellekt sistemlərinin yaradılması zamanı Planner dilinin olduqca effektiv tətbiqinə imkan verir: məsələnin həll olunduğu mühitin təsvirini qurmaq və dəyişdirmək olar. Mühitin təsviri verilənlər bazası adlanır və verilmiş mühitdə doğru olan, ayrı-ayrı müddəalardan (faktlardan) ibarətdir. Verilənlər bazasında olan faktlardan əlavə, proqramda məsələdə istifadə olunan anlayışlar arasında məntiqi münasibətləri və həmçinin yaranmasına icazə verilmiş əməliyyatların təsvirini vermək olar. Belə təsvirlərə teoremlər deyilir. Hər bir teoremin şərtlər siyahısı vardır ki (ilkin şərtlər), bu şərtlər teoremin tətbiqinə qədər yerinə yetirilməlidirlər. Teoremin tətbiqi nəticəsində verilənlər bazasında bəzi faktlar daha doğru olmur və onlar bazadan uzaqlaşdırılmalıdırlar, digərləri isə, əksinə, doğru olurlar və bazaya əlavə olunmalıdırlar. Hər iki növdən olan faktlar siyahısı teoremlərin təsvirlərində də olur.



Planner dilinin ən vacib yeniliyi geri qayıtmalar rejimi olmuşdur. O, proqram işləyən zaman öncədən qəbul olunmuş qərarların məqsədə aparıb çıxarmayacağı təqdirdə həmin qərarlardan imtina etməyə imkan verir. Məsələnin həlli zamanı variantların seçilməsindən istifadə ümumiyyətlə süni intellekt üçün xarakterikdir. Planner dilinin özəl geri qayıtmalar rejimi (onu geridönmələrlə izləmə mexanizmi də adlandırırlar, *ing.* – backtracking) proqramçını belə seçimləri özünün reallaşdırılması zərurətindən azad edir.

Süni intellekt sahəsində işlər 50 ilə yaxındır ki, aparılır. İlkin ruh yüksəkliyi, universal süni zəkanın yaradılmasında sürətli və həlledici uğurların gözlənilməsi çoxdan öz yerini ayrı-ayrı, ənənəvi olaraq insanın təfəkkür qabiliyyətinin iştirakı tələb olunan məsələlərin avtomatlaşdırılması üzrə yorucu işlərə vermişdir. İntellektual sistemlərin qurulmasında proqramlaşdırma dillərinin rolu olduqca böyükdür. Müasir süni intellekt dilləri çox problemləri uğurla həll etməyə imkan verir, ancaq onların imkanlarının məhdudluğu aşkardır. Deməli, gələcəkdə hansı dillərin istifadə olunacağını söyləmək çox çətindir – LİSP-ə bənzər funksional, yoxsa Prolog-a bənzər məntiqi dillərin. Bəlkə bu gün heç kimin ağlına belə gəlməyən tamamilə yeni ideyalara əsaslanan yeni dillər meydana gələcək?



PROQRAMÇILAR VƏ PROQRAMLAŞDIRMA

AVQUSTA ADA BAYRON, LEDİ LAVLEYS

Qızım, sevimli yavrum, istəkli Adam mənim,
Anana bənzəmisən, yeganəm, əziz balam?
O ayrılıq dəmində üzümə gülər sənin
Mavi gözlərindəki ümid işığı müdam.

C.Bayron



Corc Qordon Bayron.

1980-ci ilin iyulunda ABŞ Müdafiə Nazirliyində Ada adlı proqramlaşdırma dili işlənib hazırlandı. Bu dilə aid kitablar əski Sovetlər Birliyində də tərcümə olundu. Kitablardan birinin adı – "Язык программирования Ада" ("Ada proqramlaşdırma dili") mənfi təsəvvürlər doğururdu (*rus.* "ad" – cəhənnəm deməkdir. Kitabın adı "Cəhənnəmin proqramlaşdırma dili" kimi də yozula bilər). Lakin qaranlığın şər qüvvələrinə və cəhənnəmə bu dilin heç bir aidiyyəti yoxdur. Bu dil dünyada ilk proqramçı Avqusta Ada Lavleysin şərəfinə belə adlandırılmışdır.

Avqusta Ada Bayron 1815-ci ilin dekabrın 10-da anadan olmuşdur. Bir ay sonra onun anası Anna İzabella Milbenk (1792-1860) əri, məşhur ingilis romantik şairi Corc Noel Qordon Bayronu (1788-1824) atır və ata evinə qayıdır. Bayronla həyat yoldaşının ailə münaqişəsi ali cəmiyyətdə böyük əkssəda doğurur və şairin qərəzli təqibinə səbəb olur – o, vətənini tərk etmək məcburiyyəti qarşısında qalır. Londona bir daha qayıtmamaq fikrilə İtaliyaya üz tutan Bayron heç ağlına belə gətirmirdi ki, kibernetikanın gələcək əfsanəsini bələkdə qoyub gedir.

Ada Bayronun böyüyüb tərbiyə aldığı evdə bir qanun hökm sürürdü: gözdən düşmüş şair xatırlanmamalıdır. Onun bütün kitabları ailə kitabxanasından yığışdırılmış və hətta Avqusta Ada adı qısaldılaraq Ada edilmişdi (Avqusta adı ona Bayronun ögey bacısının şərəfinə verilmişdi).

Avqusta Ada üzdən atasına oxşayırdı, ancaq maraqlarını anasından miras almışdı. Ailə həyatının ən yaxşı günlərində ərindən "Paralelloqramlar şahzadəsi" ləqəbini alan Anna İzabella Bayron riyaziyyatla maraqlanırdı. Qızına atasından irsən qala biləcək romantik meyllərin əksinə olaraq o, qızında analitik qabiliyyətləri inkişaf etdirmək istəyirdi.

Bir neçə musiqi alətində gözəl ifa edən və bir neçə xarici dildə danışan qızın əsas marağı riyaziyyata idi. Onu bu işdə anası və anasının alim dostları – professor de Morqanın ailəsi və ər-arvad Sommervillər də dəstəkləyirdi (Meri Sommervil astronom Pyer Laplasın "Göy mexanikası haqqında traktatı"nı fransız dilindən tərcümə etməklə məşhurlaşmışdı). Ada artıq 13 yaşında uçan aparatların sxemlərini çəkirdi. Professor Oqastes de Morqan öz şagirdinin bacarıqları haqqında o qədər yüksək fikirdə idi ki, onu hətta məşhur italyan qadın-riyaziyyatçı Mari Anyezi (1718-1799) ilə müqayisə edirdi. Eyni zamanda qız gizli olaraq şeirlər yazırdı və bundan hər hansı irsi xəstəlik kimi utanırdı. O, öz poetik meyllərini xeyli gec gerçəkləşdirdi.

Avqusta Adanın ailə həyatı valideynlərininkinə nisbətən daha xoşbəxt olmuşdur. 1835-ci ilin iyulunda o, sonradan 1-ci qraf Lavleys olan Uilyam 18-ci lord Kinqə ərə getdi. O, öz arvadının elmi məşğuliyyətini dəstəkləyir və ona kömək edirdi. Ər-arvad kübar həyat tərzi sürür və müntəzəm olaraq "bütün Londonun" iştirak etdiyi gecələr və qəbullar təşkil edirdilər. Belə məclislərin daimi iştirakçılarından biri olan, populyar



Avqusta Ada Bayron.

"Ekzamenator" jurnalının redaktoru Olbani Fonblank ev sahibəsinin belə bir portretini yaratmışdı: "O, heç kəsə oxşamırdı və nəinki poetik, həm də riyazi və metafizik istedada malik idi. Hər bir işin məğzini qətiyyətlə və tez tutmaqda özünü göstərən tamamilə kişilərə xas qabiliyyətlə yanaşı ledi Lavleys incə qadın xarakterinin bütün xüsusiyyətlərinə malik idi. Onun davranışı, onun zövqü, onun təhsili, xüsusən mükəmməl musiqi təhsili, sözün əsl mənasında incə idi və səthi müşahidəçi onun zərif qadın füsunkarlığı arxasında nə qədər daxili qüvvənin və biliyin cəmləşdiyini heç zaman duya bilməzdi. O, yüngüllük və bayağılığa dözmədiyi dərəcədə həqiqi intellektual cəmiyyətdən zövq alırdı və məhz buna görə də elm, incəsənət və ədəbiyyat sahəsində tanınmış şəxslərlə yorulmadan tanışlıq axtarırdı".

Payız günlərinin biri Meri Sommervelgildəki nahar zamanı Ada ilk dəfə Kembric universitetinin riyaziyyat kafedrasının professoru Çarlz Bebicin analitik



Oqastes de Morqan.



Çarlz Bebic.

maşını haqqında eşitdi ("Çarlz Bebbic" məqaləsinə bax). Tezliklə onu məşhur alimlə tanış etdilər. Xanım Sofiya de Morqan gənc Adanın Bebbicin emalatxanasına ilk gəlişini öz memuarlarında belə təsvir edir: "Qonaqların bir hissəsi bu möcüzəli cihaza, necə deyərlər, vəhşi insanların ilk dəfə güzgü gördüyü, yaxud atəş səsi eşitdiyi kimi təəccüblə baxdığı bir vaxtda hələ çox gənc xanım Bayron maşının işini başa düşməyə nail oldu və bu ixtiranın əhəmiyyətini yüksək qiymətləndirdi".

Elmə olan ümumi maraq Ada və Bebbici bir-birilə uzunmüddətli məhsuldar əməkdaşlığa bağladı.

1840-cı ildə Bebbic analitik maşın haqqında mühazirələr oxumaq üçün Turinə dəvət olundu. İtaliyada onu vətənindəkindən daha yaxşı başa düşdülər və mühazirələr hay-küylü uğur qazandı. Dinləyicilərdən biri, gənc mühəndis Luici Menabrea bu mühazirələrin konspektini hazırlayaraq dərc etdirdi. Menabrea dərin mütəfəkkir idi və öz oçerkini maşınlara fövqəlbəşəri qabiliyyətlər "vermiş" və "verən" bütün insanların eşitməli olduğu sözlərlə bitirirdi: "Maşın düşünən varlıq deyil, o, sadəcə ona verilmişləri icra edən bir avtomatdır".

Bebbic özünün "Filosofun həyatından səhifələr" əsərində yazırdı: "Bu məqalə dərc olunduqdan bir qədər sonra qrafinya Lavleys mənə dedi ki, o, Menabreanın oçerkini tərcümə edib. Mən ona yaxşı tanış olan bu məsələ haqqında niyə onun özünün müstəqil bir məqalə yazmadığını soruşdum. Buna cavab olaraq ledi Lavleys bildirdi ki, belə fikir onun ağlına gəlməyib. Onda mən ona Menabreanın oçerkinə müəyyən şərhlər əlavə etməyi təklif etdim. Bu fikir dərhal qəbul olundu". Şərhlərin planı Bebbiclə birgə hazırlandı və o, bu haqda öz "...səhifələr"ində qısaca xatırladır: "Biz istifadə oluna biləcək müxtəlif illüstrasiyaları müzakirə etdik; mən bir neçəsini təklif etdim, seçimi isə o, tam müstəqil etdi". 1843-cü ildə Ada Lavleys məqaləni ətraflı şərhlərlə zənginləşdirərək ingilis dilinə tərcümə etdi (həcminə görə şərhlər əsas mətndən böyük idi). Bundan başqa, o, maşınların praktiki istifadəsinə aid bir neçə misal göstərirdi və müasir dildə desək kifayət qədər mürəkkəb alqoritm əsasında Bernulli ədədlərinin hesablanması proqramını tərtib etmişdi. "İşçi oyuq" və "dövr" terminləri məhz Ada tərəfindən təklif olunmuşdur.

Onun işilə tanış olduqdan sonra Bebbic Menabreanın məqaləsinin tərcüməsini və Adanın ona şərhlərini dərc etmək üçün "Teylorun elmi qeydləri" adlı sanballı elmi jurnalın redaktoru ilə razılıq əldə etdi. Analitik maşının alqoritmik əsasına həsr olunmuş bu iş şübhəsiz böyük maraq kəsb edir.

Ada Lavleys Bebbicin fikirlərini riyazi və "texnoloji" dillərə çevirmişdi. Bir neçə fraqmentə baxaq: "Maşını şüurla



təchiz etmək olmaz, o, yalnız təqdim olunan təsəvvürləri reallaşdırır. Bu təsəvvürlər perfokartlarda əks olunur, onlar hərəkətlər ardıcıllığını yerinə yetirən müxtəlif mexanizmlərə ötürülür...", "İntellektual iş bütövlükdə hesablama üçün zəruri ifadələrin hazırlanması ilə məhdudlaşır...", "Maşına əsl rəqəmlər fabriki kimi baxmaq olar".

Menabreanın məqaləsi başlıca olaraq məsələnin texniki tərəfinə toxunduğu halda, ledi Lavleysin şərhləri əsasən riyazi məsələlərə həsr olunmuşdur. Buna görə də hal-hazırda məqalənin özü şərhlərdən fərqli olaraq yalnız tarixi maraq kəsb edir, çünki müasir hesablama maşınları fərqli texniki prinsiplər əsasında qurulub; məqaləyə verilmiş şərhlər isə müasir proqramlaşdırmanın əsasını qoymuşdur.

Ada Lavleys riyaziyyat və hesablama maşınlarından tutmuş at və musiqiyə qədər müxtəlif fitri istedada, dərin biliklərə və maraqlara malik insan idi. O, hesab edirdi ki, zaman gələcək analitik maşın musiqi əsərləri yazacaq, rəsmlər çəkəcək, praktiki və elmi fəaliyyətdə istifadə olunacaq. Onun proqnozlarının düzgünlüyü və dəqiqliyini indi qiymətləndirmək olar.

Adanın Bebicə verdiyi başqa bir təklif az qala onun elmi karyerasını məhv edəcəkdi. Ledi Lavleys əmin idi ki, maşın artıq praktiki məsələləri həll edə bilər, məhz at yarışlarında məğlubiyyətsiz proqnozlar verə bilər. Lakin, ya maşın düzgün işləmirdi, ya təbiətdə



Bayronların Nottinhemşirdəki ailə sərdabəsi.

nəsə baş verirdi, atlar onlar üçün əvvəlcədən düşünülmüş sistem üzrə qaçmaqdan imtina edirdilər. Ada nəinki özünün, ərinin də pullarını uduzdu. Nə qədər kədərli səslənsə də ailəni tam müflisləşməkdən Adanın xərçəng xəstəliyindən vaxtsız ölümü qurtardı. O, 1852-ci il noyabrın 27-də, atası Corc Qordon Bayron kimi 36 yaşında vəfat etdi. Vəsiyyətinə uyğun olaraq o, Xaknell kilsəsinin nəzdində olan Nottinhempşirdəki Bayronların ailə sərdabəsində atasının yanında dəfn olundu.

Anasının Adanı təsirindən qorumaq istədiyi atasının taleyi onun yeganə qızının taleyində təkrar olundu. Onlar özləri də oxşar idilər: özgə uzaq ölkənin azadlığı naminə ölməyə, yaxud heç kəsin qəbul etmədiyi ixtira üçün hər şeyi qurban verməyə hazır olan yanar gözlü idealistlər.

Əksər ensiklopediyalarda Avqusta Ada Kinq Lavleys ingilis riyaziyyatçısı kimi verilir. Onun əsas elmi əsəri kimi Menabreanın "Bebbicin analitik maşınının elementləri" məqaləsinin tərcüməsi və bu məqaləyə şərhləri göstərilir.

PROQRAMÇILAR

Bu ən gənc və çox populyar peşələrdən biridir. *Proqramçı* – EHM üçün proqram yazan adama deyilir.

XX əsrin 60-cı illərində proqramların hazırlanmasında 3 nəfər iştirak edirdi: məsələni qoyan və onun həlli üçün zəruri verilənləri müəyyənləşdirən alim; məsələnin həlli alqoritmini fikirləşən və onu xüsusi dildə təsvir edən riyaziyyatçı-alqoritmçi; verilmiş alqoritmi maşın dilində kodlaşdıran və böyük proqramı ləng işləyən kompüterin balaca



yaddaşına daxil etməyə çalışan proqramçı.

Hər bir proqram insan düşüncəsinin nailiyyətidir. Onun səmərəliliyinə alqoritmin müəllifi və proqramçı cavabdehdirlər. Riyaziyyatçı həqiqi ədədlər matrisinin mümkün ölçüsünü bilməli idi ki, verilənlərin emalını həyata keçirsin. Kodlaşdırma həm də diqqət tələb edir, özü də bu diqqət aviadispetçerdən tələb olunandan az olmur.

Yüksək səviyyəli proqramlaşdırma dillərinin meydana gəlməsilə proqramçıların vəzifələri artmağa başladı: indi o özü alqoritmlər fikirləşir və onları münasib dildə kodlaşdırır. Səmərəli alqoritmin yaradılması prosesi özünün yaradıcı başlanğıcını saxlamışdır. Kodlaşdırma isə kompüterə həvalə oluna biləcək yorucu əməliyyata çevrilmişdir.

EHM-lərin sayı artdıqda tapşırıqların da dairəsi dəyişdi, bu isə kəskin proqramçı çatışmazlığına gətirib çıxardı. İndiki zamanda yaradıcı yanaşma və aralıq nəticə yalnız alimlərin tələblərinə cavab verir, maliyyəçilər və menecerlərə isə zəmanətli göstəricilərlə təmin olunmuş qısa müddətdə hazır həllər gərəkdir. Buna görə də proqram məhsullarının yaradılmasının xüsusi texnologiyası işlənib hazırlanmışdır. Məsələn, əvvəllər sistem tam şəkildə təsvir olunurdu, layihə təfərrüatı ilə ifadə olunurdu, yalnız bundan sonra sistemin gerçəkləşdirilməsinə başlanılırdı. Sistem üzrə sənədləşmə isə son növbədə aparılırdı. Düzdür, bu şəkildə yalnız müşayətçi proqram məhsulları – redaktorlar, əməliyyat sistemləri və s. yaradılır. Proqram sisteminin yaradılması və bazara çıxarılmasında proqramçının əməyi, adətən, ümumi sərf olunmuş əməyin 10-15%-ni təşkil edir.

Menecerlər, rəssamlar, dizaynerlər, texniki yazıçılar, testləyicilər və proqram sistemi üzərində işləyən qalan işçi heyət arasında yalnız proqramçılar əsl yaradıcılardır. Ən uğurlu proqram məhsullarını proqramçılar özləri üçün nəzərdə tutur və hazırlayır, sonradan onlar başqaları üçün də faydalı olur. Lakin bu yaradıcı şəxsiyyətlər hansısa bir yolla idarə olunmalıdırlar.

Baş proqramçı briqadasının yaradılması bir neçə proqramçının birgə əməyinin təşkili sahəsində göstərilən cəhdlərdən biri oldu. 5-7 nəfərdən ibarət kollektivə daha təcrübəli proqramçı rəhbər təyin olunurdu. O, gündəlik tapşırıqları müəyyənləşdirir və iclaslar keçirirdi: proqramçılar öz proqramlarının fraqmentlərini təqdim edirdi, onlar müzakirə olunur, tənqid olunur və təsdiq edilirdi. Belə qrupun işinin nəticəsi həqiqətən də zəmanətli olurdu, lakin səmərəli deyildi və baha başa gəlirdi. Bəzən bir proqramçı bütün kollektiv üçün nəzərdə tutulmuş işi görürdü.



"Larry" adlı məşhur kompüter oyununun seriyalarının birində qəhrəman proqramçılar işləyən bir müəssisəyə düşür. Onlar displey arxasında əyləşib başlarını qaldırmadan işləyirlər. Masaların arasında əllərində qamçı olan zorba menecerlər gəzişir və ekrandan gözünü çəkən proqramçını vururlar. Ola bilsin ki, bu oyunda proqramçılar kompaniyası menecerinin gizli arzusu gerçəkləşib.

Belə kollektiv kosmik gəminin ekipajına bənzəyir. Vacib proqram sisteminin yaradılmasında işin səmərəliliyinin həlledici faktoru kollektivdəki daxili psixoloji mühit olacaq.

Proqramçı peşəsi insandan xarakterində müəyyən xüsusiyyətlərin olmasını tələb edir. Proqramçılar bir qayda olaraq tək işləyir və hərənin öz iş stili vardır. Bəzən ümumi layihənin gerçəkləşməsi üçün bu stilləri uyğunlaşdırmaq çətin olur. Burada hər şey rəhbərin peşəkarlığından asılıdır: iş düzgün bölüşdürülübmü və bu zaman proqramçılar uğurlu seçilibmi. Axı sistemin arxitekturasında buraxılan səhvlər bütün layihə üçün acınacaqlı nəticələr verə bilər.

Lakin bu gün bütün proqramçıların yaradıcılıq əzabı çəkdiyini fikirləşmək doğru deyil. Müasir proqramlaşdırma sistemləri əksər hallarda sistemin parametrik köklənməsini həyata keçirməyə imkan verir. Belə ki, verilənlər bazasının yaradılması zamanı yalnız hesabatlar və sahələrin informasiya ilə doldurulması nəzərdə tutula bilər, bununla belə bu işin icrası zamanı proqramçı heç bir sətir də kod yazmayacaq. Onu əvvəllər olduğu kimi verilənlər bazası proqramçısı adlandırırlar. Belə "proqramçılar" indi kompüterlərin istifadə olunduğu bütün fəaliyyət sahələrində əksəriyyət təşkil edir. Lakin evdə videomaqnitafonu "proqramlaşdıran" adam özünü proqramçı adlandırmır, baxmayaraq ki, heç də evdəkilərin hamısı bu işi görə bilmirlər. Xəstənin özünün təmizləməyi bacarmadığı çox da dərin olmayan yarasını bağlayan tibb bacısı çox yəqin ki, özünü cərrah adlandırmaz.



REKURSİYA

Proqramlaşdırmada ən geniş yayılmış üsul əvvəllər yaradılmış və təsvir olunmuş proqramlara müraciətdir, yaxud başqa sözlə, altproqramların (prosedura-ların və ya funksiyaların) çağırılmasıdır. Bu təbiidir: əgər hansısa əməliyyatlar qrupunun icrası, yaxud eyni cür təyin olunan qiymətlərin hesablanması proqramda dəfələrlə yerinə yetirilməlidirsə, bu əməliyyatlarının təsvirini (hesablamaları) bir dəfə qurmaq, altproqrama ad vermək, sonra onu lazım olan sayda çağırmaq daha məqsədəuyğundur.

Həqiqətən də, əgər LOGO dilində ekranda tərəfi 5 olan kiçik kvadrat formasında iz qoyan KVADRAT prosedurunu yazsaq

```
BU KVADRAT
  TƏKRARLA 4 [İRƏLİ 5 SAĞA 90]
SON
```

onda özünün yerinə yetirilməsi gedişində birinci proseduru çağıran digər





KÜNC proseduru KVADRAT proseduruna çox oxşayır və yalnız İRƏLİ komandasında addımın qiymətilə fərqlənir (buna görə də *KÜNC* prosedurunda kvadratın tərəfləri bir addım çox alınır). Əgər KVADRAT prosedurunu parametrli prosedur kimi təsvir etsək, bu fərq də itəcək:

```
BU KVADRAT :K
   TƏKRARLA 4 [İRƏLİ :K SAĞA 90]
SON
```

(LOGO dilində dəyişən kəmiyyətdən bilavasitə əvvəl iki nöqtə qoyulur).

İndi KVADRAT 5 proqramının çağırışı KVADRAT proseduru kimi, KVADRAT 6-nın çağırışı isə *KÜNC* proseduru kimi işləyəcək.

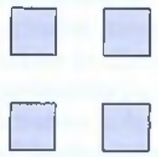
KVADRAT_GENİŞ proqramının *KÜNC* proqramına müraciət etdiyi misalda qeyri-adi heç bir şey yoxdur: bir prosedur özündən fərqli digərini çağırır. Çağıran və çağırılan proqramları KVADRAT :K parametrli prosedura dəyişsək K parametrinin 5 qiymətində nə baş verər?

```
BU KVADRAT :K
   TƏKRARLA 4 [İRƏLİ :K SAĞA 90]
   KVADRAT :K + 1
SON
```

KVADRAT proqramının KVADRAT proqramını çağırması paradoks deyilmi?

Paradoks yoxdur. Üstəlik *A* prosedurunun *A* prosedurunu çağırması halına (eyni adlı proseduru, lakin faktiki parametrin fərqli qiymətilə) proqramlaşdırmada tez-tez rast gəlmək olar və bunun xüsusi adı vardır – *rekursiya*. Deməli, yuxarıda KVADRAT rekursiv proseduruna aid misal verilmişdir.

KVADRAT 5 çağırdıqda bu prosedurun necə işlədiyinə diqqət yetirək. Prosedurun gövdəsindəki birinci sətir tərəfi bir – KVADRATLAR prosedurunun köməyilə, məsələn, aşağıdakı şəkli almaq olar:



```
BU KVADRATLAR
KVADRAT
TƏKRARLA 3
 [
   QƏLƏM QALDIR
   İRƏLİ 10
   QƏLƏM ENDİR
   KVADRAT
   SAĞA 90
 ]
SON
```

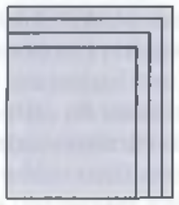
Çağıran proqrama aid bir nümunə – KVADRAT_GENİŞ; bu halda *KÜNC* proseduru *çağırılan proqram* rolunu oynayır:

```
BU KVADRAT_GENİŞ
   TƏKRARLA 4 [İRƏLİ 5 SAĞA 90]
   KÜNC
SON
BU KÜNC
   TƏKRARLA 4 [İRƏLİ 6 SAĞA 90]
SON
```

Prosedurun parametri elə dəyişən kəmiyyətdir ki, o, prosedurun daxilində istifadə olunsun və onun qiyməti prosedur çağırılarkən verilsin. Formal parametr (və ya təsvir etmə parametri) – dəyişənin adıdır, faktiki parametr (və ya çağırış parametri) isə onun qiymətidir.

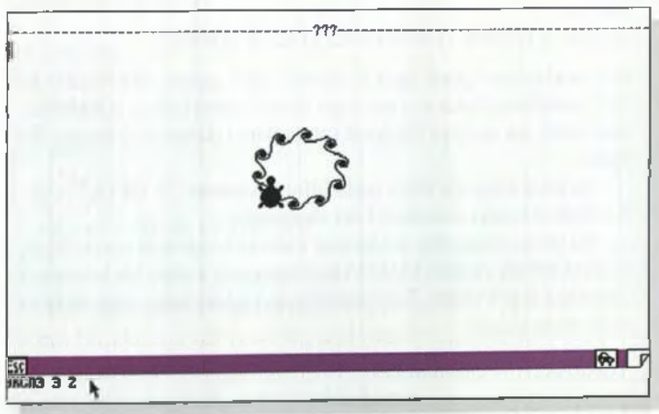


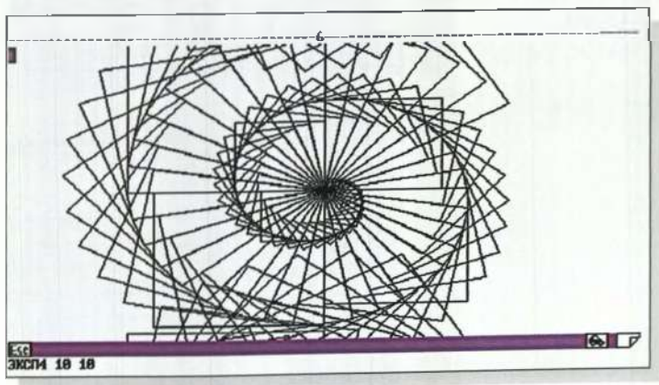
5 olan kvadrat çəkəcək, sonra isə idarəetmə gövdədəki ikinci sətrin komandasına keçəcək. Növbəti hərəkət – KVADRAT proseduruna müraciət olacaqdır. Doğrudur, əvvəlcə faktiki parametr hesablanır: əvvəlki 5 qiymətinin üstünə 1 gəlir və o, 6-ya bərabər olur. KVADRAT proseduru mövcud kvadratın üstündən bir az geniş kvadrat çəkir. Yenidən parametrin qiyməti dəyişir və yenə də KVADRAT proseduru çağırılır və bu dəfə daha geniş kvadrat çəkilir. Artıq aydındır ki, belə demək olar: "və s".



Belə rekursiya sonsuz adlanır: bir dəfə başlandıqdan sonra o, heç zaman bitmir. Əgər kompüter xarab olamasa və ya proqramçı proqramın icrasını dayandırmasa və əgər biz nəzəri yox, praktiki tərəfdən baxsaq, bu proqramın təxminən sonsuz adlandırılması daha məqsədəuyğundur. İş onda deyil ki, K: parametrinin hansısa kifayət qədər böyük qiymətindən sonra kvadratların çəkilməsi prosesi nəzərə çarpmayacaq – şəklin çəkilməsi ekrandan kənarda davam edəcək, məsələ ondadır ki, sonsuz rekursiyanı ancaq sonsuz yaddaşı olan kompüterlərdə icra etmək olar. Müasir kompüterlərin yaddaşı çox böyük olsa da, məhduddur.

Hər hansı bir prosedura müraciət etdikdə yaddan çıxarmaq olmaz ki, prosedurun icrasından sonra idarəetməni çağırışın olunduğu nöqtəyə qaytarmaq lazımdır. Çağırılan prosedura girişlərin "qediyyatı" və giriş nöqtələrinin yadda saxlanılması prosesi prosedurun rekursiv olub-olmamasından asılı deyildir. Qayıdış nöqtələrinin yadda saxlanılması üçün





KUKARAÇA

Bu dil o qədər sadədir ki, burada hətta dəyişən anlayışı yoxdur. Buna baxmayaraq rekursiyalar Kukaraçada çox fəal tətbiq olunur.

İcraçı bir neçə qat-mərtəbədə yerləşdirilmiş damalı sahələrdən ibarət mühitdə fəaliyyət göstərir. O, damalarla mühiti tərk etməmək şərtilə SOLA, SAĞA, YUXARI, AŞAĞI (bir mərtəbə çərçivəsində) İRƏLİ, GERİ (bir mərtəbədən digərinə) komandaları ilə hərəkət edir. Kukaraça onun qarşısına çıxan kubiklərin (o cümlədən bir neçə kubikin) icra olunan komanda istiqamətində yerini dəyişə bilər. Kubiklərin üzərinə hərflər (və ya klaviaturadan daxil oluna bilən istənilən simvol) qoyulmuşdur. Kubiklərin bəziləri çevrilmişdir: onların üz tərəfiində hərfin əvəzinə sual işarəsi göstərilmişdir. Kukaraça kubiki itələməklə kubikin qonşu damasında hansı hərfin olduğunu müəyyənləşdirə bilər. Belə itələmədən sonra hərf haqqında suala cavab şərti komanda tərəfindən verilir:

ƏGƏR <*şərt*> ONDA <*komanda-1*> ƏKS HALDA <*komanda-2*>

burada <*şərt*> – icraçı tərəfindən axtarilan hərfdir.
Məsələn,

```
SAĞA
ƏGƏR A ONDA YUXARI ƏKS HALDA AŞAĞI
```

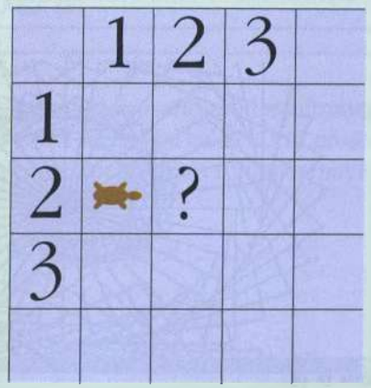
Komandalarına görə əgər üzərində sual işarəsi olan kubikdə "A" hərfidirsə, Kukaraça yuxarıya (ikinci sətrin birinci damasına), əks halda isə aşağıya (üçüncü sətrin ikinci damasına) istiqamətlənir.

Şərtlərə inkar da daxil edilə bilər. Məsələn, "A DEYİL" – "A" hərfindən başqa istənilən hərf deməkdir.

Bu dildə idarəedici strukturun məhdudlaşdırıcısı lazım deyil, çünki ONDA və ƏKS HALDA budaqlarında yalnız bir komanda yazmağa icazə verilir. Təkrarlanma sayı əvvəlcədən məlum olan dövr belə yazılır:

TƏKRARLA *n* <*komanda*>

burada *n* – təkrarların sayıdır.



kompüter tərəfindən avtomatik olaraq ayrılan yaddaş hissəsi *təxirə salınmış* (*açılmış*) prosedurlar *steki* adlandırılır. Bu stekdə icra olunan proqram tərəfindən çağırılmış prosedurların (və ya rekursiv çağırışlar olan halda rekursiv prosedur nümunələrinin) siyahısı saxlanılır. Stek elə təşkil olunur ki, sadalanan prosedurlardan axırıncısı ən birinci əlçatan prosedur olur. Bir-biri tərəfindən çağırılmış prosedurların zənciri nə qədər uzundursa, *təxirə salınmış* (*açılmış*) prosedurlar steki o qədər çox yer tutur. Adi çağırışlarda ən mürəkkəb (rekursiv olmayan) proqramların icrası üçün stekdə kifayət qədər yer olur. Lakin hər bir sonsuz rekursiya gec-tez steki dolduracaq. Bax elə buna görə də rekursiya müasir kompüterlərdə kifayət qədər uzun müddət icra oluna bilsə də, bu proses əbədi deyil.

Lakin nəzəri cəhətdən sonsuz rekursiya nə qədər maraq doğursa da, təbii olaraq bir sual yaranır: sonlu rekursiya mümkündürmü? Bu suala cavab vermək asandır. Rekursiv prosedura aid misalda bilavasitə rekursiv çağırışdan əvvəl qısaldılmış şərti komanda daxil edək:

```
BU KVADRAT :K
TƏKRARLA 4 [İRƏLİ :K SAĞA 90]
ƏGƏR K < 100 ONDA KVADRAT :K + 1
SON
```

Proqramın belə "təkmilləşdirilməsi" 100-cü kvadrat çəkildikdən sonra rekursiv çağırışların zəncirini dayandırır. Sonu olan (şərti başlanğıcda yoxlanılan) rekursiv prosedur ön şərti olan dövrü xatırladır. Bu cür rekursiyadan proqramçılar tez-tez istifadə edirlər.

Rekursiya mexanizmi o qədər də sadə hesab olunmur. Ola bilsin ki, məhz bu səbəbdən onun bütün proqramlaşdırma dillərində istifadəsinə icazə verilmir. LOGO-da rekursiya elə təqdim olunur ki, o, əsas idarəedici struktur təəssüratı bağışlayır. Pascal dilinin təsvirində o mümkün idarəedici struk-

turlardan biri kimi verilir, vaxtilə geniş yayılmış BASİC proqramlaşdırma dilində isə rekursiyalara ümumiyyətlə icazə verilmir. Bu misallardan aydın olur ki, rekursiyanın bu və ya digər dilə daxil edilib-edilməməsini həmin dilin "mürəkkəbliyi" ilə əlaqələndirmək olmaz.

Aşağıda verilmiş bir neçə anlayış və misal Kukaraça icraçısının idarəetmə dili çərçivəsində göstərilir. Məsələn,

```
BU GƏZİNTİ
   TƏKRARLA 9 SAĞA
   TƏKRARLA 9 AŞAĞI
   TƏKRARLA 9 SOLA
   TƏKRARLA 9 YUXARI
GƏZİNTİ
SON
```

proqramı icraçını məcbur edəcək ki, 10x10 ölçülü sahəni uzun-uzadı dolansın. Doğurdan da, belə proqramı yarımçıq kəsmədən dayandırmaq çətindir.

Dildə istifadəsinə icazə verilən ön şərtli dövr (dövr NƏ QƏDƏR Kİ) bu şərtdən şərti komandalarda olduğu qaydada istifadə edir. Məsələn, aşağıdakı komandalarda olduğu kimi:

```
NƏ QƏDƏR Kİ BOŞDUR AŞAĞI
SAĞA
```

Nə qədər ki, xanalar boşdur (BOŞDUR şərti) Kukaraça sütunla aşağı enir və BOŞDUR şərti pozulduqda o dövrdən sonrakı SAĞA komandasına keçir. Bu, Kukaraça ən yaxın kubikə toxunduqda baş verir.

Eyni məsələ – ən yaxın boş olmayan sahənin müəyyənləşdirilməsi də – sonlu rekursiya vasitəsilə həll olunur:

```
BU ENİŞ
   AŞAĞI
   ƏGƏR BOŞDUR ENİŞ
SON
```

Növbəti məsələdə – Kukaraçanın, sonunda proqramın qurtardığını göstərən ulduz (*) nişanı olan kubikin durduğu



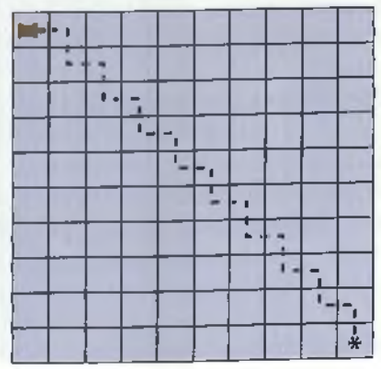
Kukaraça rekursiv prosedurdan istifadə etmədən məsələni həll edir.

pilləkənlə enişində çağıran və çağırılan proqramlar arasında münasibətlər başqa cür qurulur.

Bu münasibətlərin "aydınlaşdırıldığı" aşağıdakı proqrama baxaq. Burada PİLLƏKƏN və PİLLƏ prosedurları özlərinə müraciət etmirlər.

```
PİLLƏKƏN
BU PİLLƏKƏN
   PİLLƏ
SON
BU PİLLƏ
   AŞAĞI
   SAĞA
   ƏGƏR * DEYİL PİLLƏKƏN
SON
```

Elə görünür ki, onlar rekursiv proqramın bundan əvvəl verilmiş sadələşdirilmiş tərifinə uyğun gəlmir. Lakin rekursiya anlayışını təbii yolla genişləndirmək olar.



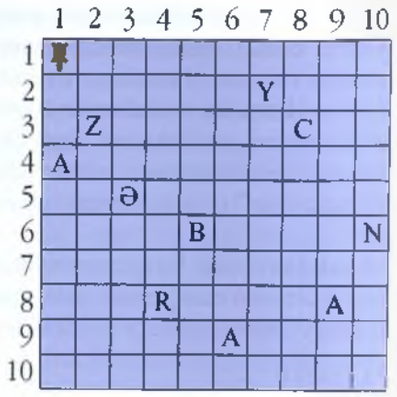


Bu məsələnin həllində Kukaraça 3 rekursiyadan istifadə edir.

Əgər *A* proqramı *B* proqramını çağırırsa və *B* proqramı da yenidən *A* proqramına müraciət edirsə (bu müraciətlər digər çağırılan proqramlar ardıcıllığı ilə də mümkündür), bu iki prosedur (*A* və *B*) *dolayı* rekursiya təşkil edirlər (*birbaşa* və yaxud *bilavasitə* rekursiyadan fərqli olaraq). Bu tərifdə birbaşa rekursiyaya dolayı rekursiyanın xüsusi halı kimi baxılır.

Birbaşa və dolayı rekursiyanın bundan əvvəlki bütün misallarında rekursiv çağırış həmişə prosedurun sonuncu komandası olurdu. Bu halda rekursiya təkrar olunan prosesslərin idarəetmə üsullarından biri kimi fəaliyyət göstərir. Əgər çağırılan prosedurun təsvirində rekursiv çağırışın ardınca digər komandalar gəlirsə, onda ilk baxışdan gözlənilməyən əlavə effektlər yaranır.

Mühitin çoxmərtəbəli strukturunun işə düşdüyü bir məsələyə baxaq. 10x10 sahəsində (birinci mərtəbədə), birinci və axırıncı sətri çıxmaq şərtilə, hər bir sütunda bir kubik olmaqla nizamsız şəkildə 10 kubik səpələnimişdir. Başlanğıcda (1, 1) damasında yerləşdirilmiş Kukaraça sahəni nizama salmalıdır – bütün kubikləri birinci sətrə yığmalıdır. Bu məsələnin əsas hissəsi sütunun ixtiyari yerində duran kubiki sütunun birinci sətrində yerləşdirən prosedurdan ibarətdir. Birinci sətirdən kubikə qədər enmək çətin deyil: Kukaraçanın kubikə



toxunmasını təyin edən şərtlə dolayı rekursiyadan (və ya NƏ QƏDƏR Kİ dövründən) istifadə etmək kifayətdir. Kubiklə ilk sətrə qayıtmaq daha çətindir. Bunun üçün aşağıya doğru neçə addım edilmişsə yuxarıya doğru da o qədər addım edilməlidir. Lakin Kukaraçada sayğac ola bilməz, çünki dəyişən yoxdur! Bəs icraçının addımlarını necə saymaq və yadda saxlamaq lazımdır? Burada *təxirə salınmış rekursiya* adlanan üsul köməyə gəlir.

SÜTUN və ARXASINAKEÇMƏ prosedurları kubiki itələyir və onu birinci sətrə qaytarır:

```
BU SÜTUN
    AŞAĞI
    ƏGƏR BOŞDURSA ONDA SÜTUN
    ƏKS HALDA ARXASINAKEÇMƏ
    YUXARI
SON

BU ARXASINAKEÇMƏ
    İRƏLİ AŞAĞI AŞAĞI GERİ YUXARI
SON
```

ARXASINAKEÇMƏ – adi prosedurdur və bu prosedur vasitəsilə Kukaraça aşağı mərtəbə boyunca tədqiq olunan damanın arxasına keçir. SÜTUN – dolayı-rekursiv prosedurdur. O, aşağı istiqamətdə edilən hər bir "boş" (itələnmədən) addım zamanı çağırılır. Onun xüsusiyyəti – axırıncı YUXARI komandasıdır. İş ondadır ki, SÜTUN prosedurunun ilkin AŞAĞI komandası və ƏGƏR şərti komandası SÜTUN prosedurunun hər bir çağırışı zamanı

təkrar olunur, lakin bu zaman YUXARI komandasının icrası o vaxta qədər təxirə salınır ki, şərti komandanın ƏKS HALDA hissəsində olan komandalar yerinə yetirilib qurtarsın.

Məsələn, üçüncü sətirdə duran kubik üçün SÜTUN prosedurunun komandaları necə icra olunur?

1. SÜTUN prosedurunun başladılması. AŞAĞI komandası yerinə yetirilir. Kukaraça ikinci sətrə enir.

2. Kukaraça əvvəlki hərəkəti zamanı kubikə rast gəlmədiyindən, BOŞ şərti doğru olur və rekursiv prosedura ikinci çağırış baş verir. Birinci çağırışın bir hissəsini təşkil edən YUXARI komandasının icrası təxirə salınır – o, təxirə salınmış komandalar stekində saxlanılır.

3. İkinci çağırış zamanı yenə AŞAĞI komandası icra olunur. Kukaraça üçüncü sətrə keçir.

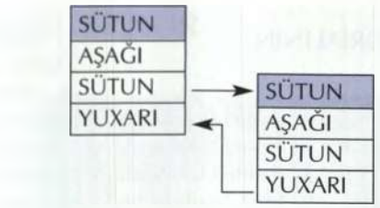
4. Burada o, kubikə rast gəlir. BOŞ şərti pozulur. Ona görə də, icraçı qonşu sahə üzrə kubikin arxasına keçir və kubiki yuxarıya – beşinci sətrə itələmək üçün rahat olan bir mövqedə durur.

ARXASINAKEÇMƏ proseduru daxilində YUXARI komandası icra olunur. Kukaraça 4-cü sətrə qalxır və kubiki 3-cü sətrə itələyir.

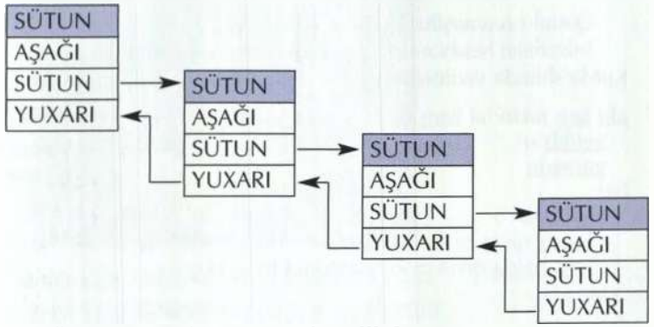
5. İndi SÜTUN prosedurunun çağırışına aid olan YUXARI komandası icra olunur. Kukaraça 3-cü sətrə, kubik isə 2-ci sətrə qalxır. İkinci rekursiv çağırış başa çatır.

6. Lakin birinci rekursiv çağırış tamamlanmamışdır, çünki YUXARI komandası icra olunmamış qalır. İnterpretator onu təxirə salınmış komandalar stekindən çıxarır və icraçıya ötürür. Kukaraça kubiki birinci sətrə itələyərək özü ikinci sətirdə durur. Birinci rekursiv çağırış (stekdə sonuncu çağırış) tamamlanır, onunla birlikdə SÜTUN prosedurunun icrası da başa çatır.

Təsvir olunmuş mərhələlər şərti olaraq sxem şəklində əks olunub:



5-ci sətrin kubiki üçün sxem aşağıdakı kimi olacaqdır:



SÜTUN prosedurunu təsvir etdikdən sonra onu bizim məsələni həll edən DÜZÜM proseduruna daxil etmək çətin deyil:

```
BU DÜZÜM
    TƏKRARLA 9 (SÜTUN İRƏLİ YUXARI
    SAĞA GERİ)
    SÜTUN
SON
```



N ƏDƏDİNİN FAKTORİALININ HESABLANMASI

"*n*!" ("*n*" nida işarəsilə) kimi işarə olunan verilmiş *n* ədədinin faktorialının hesablanması proqramı rekursiv funksiyaya misal ola bilər. Tam *n* ədədinin faktorialı 1-dən *n*-dək olan bütün tam ədədlərin hasilidir və aşağıdakı düsturla hesablanır:

$$F(n)=1\cdot 2\cdot 3\cdot \dots \cdot (n-2)\cdot (n-1)\cdot n$$

Qəbul olunmuşdur ki, 0! = 1.

Faktorialın hesablanması proqram-funksiyası aşağıda KuMir dilində verilmişdir:

```
alq tam faktorial (tam n)
  verilib n
  gərəkdir
baş
| əgər n <= 1
| | onda qiym := 1
| | əks halda qiym := n * faktorial (n – 1)
| tamam
son
```

Bu funksiya onu çağırmış proqrama *n*! ədədi qiymətini qaytarır. İşin əvvəlində proqram hələ bilmir ki, faktorial (*n*) nədir, ona görə də onun hesablanmasını ən azı növbəti rekursiv çağırışa qədər təxirə salır. Lakin ola bilsin ki, faktorialın hesablanması yenə də təxirə salınsın. Və ancaq *n*-ci çağırış zamanı kompüter təxirə salınmış komandalar stekindən ilkin olaraq faktorial (*n*), ondan sonra faktorial (*n* – 1) və sair çıxarmaqla faktorialı hesablamağa başlayacaq. Burada söhbət yenə də təxirə salınmış rekursiyadan gedir.

```
a := 5
b := 6
çapet (faktorial (a) + faktorial (b)) : 2
```

komandalarına görə kompüter ekrana aşağıdakı ədədi qiyməti çıxarır:

$$\frac{5!+6!}{2}=\frac{120+720}{2}=420,$$

çünki

$$5!=1\cdot 2\cdot 3\cdot 4\cdot 5=120,\ 6!=120\cdot 6=720.$$

(Bu misalda istifadə olunmuş fiqurlu mötərizələr çərçivəyə alınmış komandalar qrupunu proqramlaşdırma dilinin razılaşmalarına uyğun olaraq vahid bir komanda hesab etməyə imkan verir).

Yuxarıda göstərilən bütün misallarda rekursiv proqramlar qismində ancaq prosedurlar çıxış edirdi. Lakin rekursiyalar funksiyaların proqramlaşdırılmasında da eyni uğurla istifadə olunur.



Fikirləşmək lazım deyildir ki, insanlar rekursiya ilə yalnız informatikada rastlaşırlar. Müasir riyaziyyat da rekursiv funksiyalarla zəngindir. Gündəlik həyatda da rekursiyalara çoxlu örnəklər tapmaq olar. Rəssamlar rekursiv rəsmlər çəkməyi xoşlayırlar: məsələn, rəsmdə ekrana baxan televiziya tamaşaçısı göstərilir, ekranda isə başqa bir televiziya tamaşaçısı ekrana baxır. Həmin ekranda isə başqa televiziya tamaşaçısı görünür və s.

Öz itini sevən keşiş haqqında tanınmış şer-pritça ədəbi rekursiyaya gözəl örnəkdir:

Keşişin bir iti vardı,
O, itini sevərdi.
İt bir gün ət parçasını yedi,
Keşiş onu öldürdü.
Öldürüb də basdırdı,
Və üstündə də yazdı:
Keşişin bir iti vardı...

ETİBARLI PROQRAMLAR NECƏ YAZILIR

Hər bir proqrama informasiyanın dəyişdirilməsi prosesinin təsviri kimi baxmaq olar: hər hansı ilkin informasiya vardır və ondan nəticə əldə olunmalıdır. Bu sxemə təkcə hesablama məsələləri deyil, oyunlar da daxil olmaqla interaktiv proqramlar aiddir. Bu zaman istifadəçinin hərəkətləri giriş informasiyası, kompüterin işi isə – nəticə hesab olunur.

Müasir dünyada maşınların köməyilə getdikcə daha mürəkkəb məsələlər həll olunur. Buna görə də proqramların etibarlılığına tələbat artır, yeni proqramlar mümkün ilkin verilənlər üçün düzgün nəticə verir və səhv olan halları qeydə alır. Proqramın düzgünlüyünü yoxlamaq üçün xüsusi üsullar mövcuddur. Onlar iki qrupa bölünürlər: *isbat üsulları* və *test üsulları*.

Əgər proqramın düzgünlüyü isbat olunubsa, onu testləşdirmək nəyə lazımdır? Birincisi, bəzən düzgünlüyün isbatının özü də düzgün olmur. İkincisi, proqramın mətnində səhvlər ola bilər. Düzgünlüyün isbatı zamanı müəllif proqramı yazdığı zaman nəzərdə tutduqlarına əsaslanır. Kompüter isə nə yazılıbsa onu icra edir. Proqramın etibarlılığına əmin olmaq üçün çoxlu test lazımdır.

Testin əsas məqsədinin proqramın düzgün işləməsinə əmin olmaq olduğunu hesab edənlər yanılırlar. Həqiqətdə testləşdirmə səhvlərin axtarışıdır.



Testlərə bu baxışların arasındakı fərq testlərin tərtibinə və onların nəticələrinə psixoloji yanaşmadadır.

Birinci halda testi aparan etiraf edir ki, proqram işləyir və bunu təsdiq etməyə çalışır. Buna görə də testlərin sayı ixtisar edilir (nəyi yoxlayasan: hər şey aydındır), testlərdə kələk olmur və onlar eyniliklə proqram tərəfindən işlənən vəziyyətlər yığımını təsvir edirlər. Belə testləşmə, proqramları yazan və testləşdirən zaman nəzərdə tutulmamış vəziyyətləri ağlına belə gətirməyən gənc müəlliflərə daha çox xasdır.

İkinci yanaşmada hər şey dəyişir. Testi keçirənin məqsədi proqramı sındırmaq, onun səhv olduğunu sübut etmək, işlənmənin mürəkkəb olduğu halları



ÜÇBUCAQ HAQQINDA MƏSƏLƏ

Üç natural ədəd verilmişdir. Uçbucağın verilmiş tərəflərinə (bərabərtərəfli, bərabəryanlı, müxtəlif tərəfli) və bucaqlarına (iti bucaq, düz bucaq, kor bucaq) görə üçbucağın növünü müəyyənləşdirməli.

Məhdudiyyətlər: 0 < a, b, c < 1000.

Giriş: 3 4 5

Çıxış: düzbucaqlı, müxtəlif tərəfli üçbucaq.

Yaxşı testlər dəsti hazırlamaq üçün heç olmasa ümumi şəkildə mümkün halları və səhvləri təsəvvür etmək lazımdır.

Giriş verilənləri bilərəkdən düzgün seçilmişdir, yəni şərtlərdə verilmiş məhdudiyyətləri ödəyirlər.

Testlərin hazırlanması şərtlərin təhlilindən başlayır. Asanca görmək olar ki, şərtlərdə çox əhəmiyyətli bir məqam nəzərə alınmamışdır: verilən tərəflər, ümumiyyətlə, üçbucaq əmələ gətirməyə bilər. Bu hal üçbucağın xassəsinin pozulması zamanı olur: hər bir tərəf digər iki tərəfin cəmindən kiçik olmalıdır.

Beləliklə, proqramın korrekt giriş verilənlərinə mümkün reaksiyalarının aşağıdakı siyahısı alınır:

- mövcud deyil;
- bərabərtərəfli, itibucaqlı;
- bərabəryanlı, itibucaqlı;
- bərabəryanlı, korbucaqlı;
- müxtəliftərəfli, itibucaqlı;
- müxtəliftərəfli, düzbucaqlı;
- müxtəliftərəfli, korbucaqlı;

(Siyahıda "bərabəryanlı, düzbucaqlı" variantı yoxdur, çünki natural ədədlərdən ibarət tərəflərin olduğu təqdirdə bu variant mümkün deyil).

Aydındır ki, test zamanı sadalanan cavablardan hər birini heç olmasa bir dəfə almaq lazımdır.

Bu məsələnin həllində hansı xarakterik səhvlər edilə bilər? Daxil edilən verilənlərin simmetriyasına xüsusi diqqət yetirək. Verilmiş üç ədədin hər biri bərabər hüquqludur, çünki şərtdə hansısa xüsusi verilmə qaydası qeyd olunmayıb. Bu halda həll zamanı tez-tez səhvlər yaranır. Məsələn, üçbucağın bərabərsizliyi hər üç tərəf üçün yoxlanılmıya bilər. Praktiki nəticə: eyni ədədlərin müxtəlif qaydada yerləşdirilməsindən ibarət testlər lazımdır. İki tərəfin cəminin üçüncü tərəfə bərabər olduğu hal mütləq ayrıca nəzərdən keçirilməlidir. Bundan başqa mümkün diapazonun sərhədində yerləşən qiymətlər də yoxlanılmalıdır.

Aşırı dolma səhvinə tez-tez rast gəlmək olur. Bu səhv proqramda 16-bit ədədlərdən istifadə olunduqda baş verir, çünki belə ədədlər kvadrata yüksəldildikdə (üçbucağın bucaqlarının müəyyənləşdirilməsi üçün tərəfləri kvadrata yüksəldirlər) 16 bitə yerləşmirlər. Son bölmədəki testlər elə seçilib ki, bu səhv dərhal aşkarlansın.

Üçbucaq haqqında məsələ üçün nümunəvi testlər dəsti cədvəldə verilmişdir:

Giriş	Gözlənilən nəticə	Şərhlər
3 20 100 20 100 3 100 3 20	mövcud deyil mövcud deyil mövcud deyil	Üçbucağın bərabərsizliyi bütün tərəflər üçün yoxlanılıbmı?
999 1 1 1 1 999 1 999 1	mövcud deyil mövcud deyil mövcud deyil	Mövcud olmayan "bərabəryanlı" üçbucaqlar
2 3 5 30 50 20 500 200 300	mövcud deyil mövcud deyil mövcud deyil	Cırlaşmış üçbucaqlar
1 1 1 999 999 999	bərabərtərəfli, itibucaqlı bərabərtərəfli, itibucaqlı	Sərhəd qiymətləri. Eyni zamanda düzgün üçbucağı testdən keçiririk
500 500 300 400 700 400	bərabəryanlı, itibucaqlı bərabəryanlı, korbucaqlı	Müxtəlif bucaqlı və tərəfləri müxtəlif qaydada daxil olunan bərabəryanlı üçbucaqlar
32 48 56 30 40 50 256 368 600	müxtəliftərəfli, itibucaqlı müxtəliftərəfli, düzbucaqlı müxtəliftərəfli, korbucaqlı	Müxtəlif tərəfli üçbucaqlar üçün analojidir
255 256 257	müxtəliftərəfli, itibucaqlı	Aşırı dolmanın yoxlanılması

tapmaq və onları proqrama təklif etməkdən ibarətdir. Aydındır ki, burada iş sadə testlərlə məhdudlaşmır. Ən dolaşıq giriş verilənləri, onların ən gözlənilməz kombinasiyaları istifadə olunur. Əgər proqram belə təzyiqlərə tab gətirirsə, onda doğurdan da onun düzgün yazıldığına şübhə etməmək olar.

Bütün test üsullarını iki qrupa bölmək olar: qara qutu və ağ qutu prinsipləri üzrə testləşdirmə.

Qara qutu o obyekt hesab olunur ki, xarici təsirlərə necə də olsa cavab vermək qabiliyyəti olsun və onun daxili strukturu məlum olmasın. Məsələn, hər bir məişət cihazı əksər insanlar üçün adi qara qutudur. Hamı bilir ki, müəyyən düyməni basdıqda televizorun səsi artacaq, lakin çoxları bilmir ki, belə vəziyyətdə hansı fiziki proseslər baş verir.

Qara qutu üsulu ilə testləmə proqramın həll etməli olduğu məsələnin təhlilinə əsaslanır. Məsələnin şərtlərini öyrənərək müxtəlif mümkün vəziyyətləri müəyyənləşdirirlər, onların hər biri üçün giriş verilənlərinin nümunəsini yaradırlar və mütləq düzgün nəticəni tapırlar. Bununla da alınan testlər dəsti verilmiş məsələni həll edən hər bir proqrama tətbiq edilə bilər.

Ağ qutu üsulu ilə testləşdirmə içində ayrıca bloklar seçilən konkret proqramın mətninə əsaslanır. Hər bir belə bloku vahid tam kimi testləşdirmək olar (qara qutu) və ya daha kiçik bloklara bölmək olar (ağ qutu).

Adətən, proqram üzərindəki iş prosesi zamanı müəllif onu ağ qutu üsulu ilə testdən keçirir. İş bitdikdə isə proqramı ciddi və sərt şəkildə mütləq qara qutu üsulu ilə yoxlayırlar.

Testləşdirmədə ən çətin iş testlər dəstinin tərtib olunmasıdır. Bu yaradıcı, çətin formallaşdırılan bir əməliyyatdır, lakin bunun üçün ümumi prinsiplər mövcuddur:

- testlər məsələnin şərtləri əsasında tərtib olunur və konkret proqramın xüsusiyyətlərini nəzərə almamalıdır;
- əgər məsələ müxtəlif hallara bölünürsə, onların hamısını yoxlayacaq testləri tapmaq zəruridir;
- testlər dəsti tərtib olunarkən testlərdə məsələnin müxtəlif həll yollarının nəzərə alınması arzu olunandır, o cümlədən düzgün olmayan həllərin nəzərə alınması, belə ki, bilərəkdən verilmiş səhv həllin cavabı da səhv olmalıdır;
- hər tip verilənlər üçün müəyyən kritik qiymətlər mövcuddur. Bu qiymətlərin giriş verilənlərində necə emal olunmasının yoxlanılması və proqram tərəfindən həmin verilənlərdə nəticələrin düzgünlüyünün yoxlanılması zəruridir;
- əgər giriş verilənlərinin mümkün qiymətləri məhduddursa, proqramın işini sərhəd nöqtələrində və ona yaxın nöqtələrdə testdən keçirmək lazımdır. Məsələn, əgər məsələnin həllində söh-





bət 1-dən 100-dək diapazondakı N tam ədədindən gedirsə, N=1, 2, 99, 100 hallarında proqramın işi nəzərdən keçirilməlidir.

• testlərə, mütləq məsələ üçün mümkün qiymətlər (mənfi, müsbət və sıfır), o cümlədən qadağan olunmuş (ədəd və ya mətn olmayan) qiymətlər daxil edilməlidir.

Bəziləri üçün testləşdirmə sındırmaq və yaratmamaq kimi destruktiv fəaliyyətdir. Həqiqətdə isə araşdırmanın savadlı aparılması bacarığı, proqramdakı gizli səhvlərin müəyyənləşdirilməsi çox qiymətli vərdişdir, ona görə də belə bir vərdişi olan mütəxəssis proqramçılar kollektivində çox lazımlı bir adamdır.

GÖZƏL PROQRAMLAR NECƏ YAZILIR?

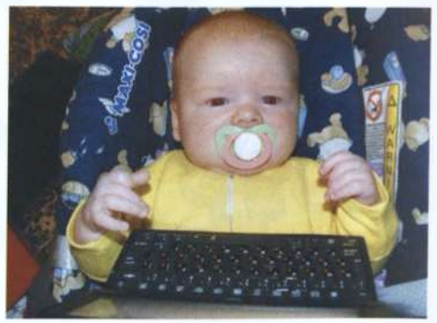
İşlənib-hazırlanma prosesində hər bir proqram bir neçə dəfə düzəlişə məruz qalır və hər dəfə adam gərək bu proqramı oxusun, strukturu araşdırsın, çətinliklər yarana biləcək yerləri axtarsın. Buna görə də proqramlar nə qədər sadə yazılarsa, onları araşdırmaq bir o qədər asan olar.

Proqramlaşdırmağa təzə başlayan proqramçılar çoxt vaxt hesab edirlər ki, proqramın aydınlığı üçün xüsusi olaraq çalışmağa ehtiyac yoxdur: axı proqramı onlar özləri fikirləşmişlər və deməli proqram haqqında hər şeyi bilirlər və tərtibata vaxt sərf etmək nəyə lazımdır? Təəssüflər olsun ki, bu yanlış fikirdir və çox geniş yayılmışdır.

Proqramçı öz proqramını nə qədər yaxşı anlasa da, müəyyən bir həddə var ki, ondan sonra bütün detalları yadda saxlamaq mümkün deyil. Əgər iş uzunmüddətli fasilələrlə aparılırsa, təfərrüatları yada salmaq o qədər də asan olmur. Həyatda çox vaxt da belə olur ki, proqramı bir adam başlayır digəri isə tamamlayır.

Təcrübəli proqramçılar proqramın görünüşünə həmişə diqqət yetirirlər. Proqram oxunaqlı alınırsa onu asanlıqla oxumaq, anlamaq və düzəltmək olar və deməli, proqramın düzgün və faydalı olması şansı çox olar.

Proqramların tərtibi qaydalarını adətən proqramlaşdırma üslubu adlandırırlar. Eyni hərəkətləri demək olar ki, həmişə müxtəlif yollarla yerinə yetirmək olar. Hər bir proqramçının üslubunu təşkil edən öz həvəs və vərdişləri var. Vaxt keçdikcə onun mütləq öz təkrarolunmaz "dəsti-xətti" əmələ gəlir, alətlərdən istifadə etmək üslubu yaranır ki,



bu üslubla proqramçını kağız üzərində imzanı tanıdığımız kimi tanımaq mümkündür. Əgər üslub belə fərdidirsə onu müzakirə etməyə dəyərmi? Gəlin müqayisə aparaq.

Hamı bilir ki, hər bir insanın xətti onun barmaq izləri kimi unikaldır. Lakin barmaq izlərindən fərqli olaraq xətti qiymətləndirmək olar: gözəldir – gözəl deyil, aydındır – aydın deyil. Əli yazı üçün xüsusi olaraq hazırlayırlar: nümunələr üzrə xüsusi tapşırıqlar diqqəti, şüuru və təsəvvürü inkişaf etdirməyə kömək edir.

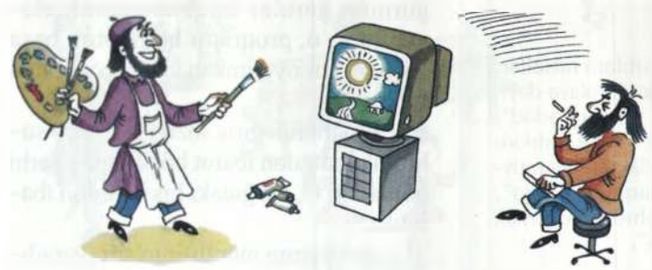
Bunu yaxşı və pis olan proqramlaşdırma üslubuna da tətbiq etmək olar. Birinci sinfə gedənlər yazı nümunəsindən istifadə etdiyi kimi, təzə başlayan proqramçılar da müəyyən standartlara riayət etsələr yaxşıdır.

Proqramlaşdırma üslubu öz normaları olan danışıq üslubu ilə müqayisə oluna bilər (hətta stilistik səhvlər anlayışı mövcuddur). Hər bir insanın sevdiyi sözlər var, cümlələri qurmaq tərzi var, öz fikrini izah etmək üsulu var. Ekspert-ədəbiyyatşünaslar mətnin bir hissəsi əsasında onun müəllifini müəyyən edə bilirlər.

Danışıq üslubunu da qiymətləndirmək olar! Axı yaxşı və pis danışıq üslublarının olması hamıya məlumdur.

Daha bir qeyri-adi müqayisə – aşpazlıq. Proqramlaşdırma üslubu ədviyyatlara bənzəyir. Yeməyin qida dəyəri əsas istifadə olunmuş ərzaq məhsulları ilə müəyyən olunur. Alınan kalorilər və toxluq hissiyyatı ədviyyatlardan asılı deyildir, lakin dad hissiyyatı əsasən onlardan asılıdır. Əgər aşpaz ədviyyatları əlavə etməyi unutmuşsa, yaxud daha pis halda, onlardan düzgün istifadə etməyibsə, ümumiyyətlə hazırlanmış yeməkdən doymaq olar, lakin ondan həzz almaq mümkün deyil.

Daima yadda saxlamaq lazımdır ki, proqramı kimsə oxuyacaq və elə etmək



lazımdır ki, oxuma zamanı çətinliklər yaranmasın.

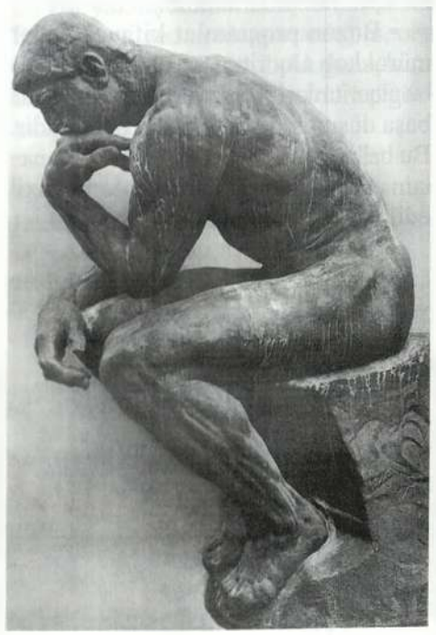
Aşağıdakı tövsiyələr yaxşı üslubun nə olduğunu başa düşməyə imkan verir:

Aydın düşünən, aydın da ifadə edir!

Proqramı yazmamışdan əvvəl dəqiq müəyyənləşdirilməlidir: hansı məsələ qoyulmuşdur və o, necə həll olunacaq. Fikirlərdə aydınlıq yoxdursa – heç bir üsul və fəndlər kömək etməyəcək.

Şərh etməyi unutmayın!

Bütün proqramlaşdırma dilləri proqrama şərhlərin daxil edilməsinə imkan verir – şərhlər ancaq insan üçün lazım olan izahedici mətndir. Kompüter şərhləri oxumur, o, sadəcə olaraq onları



Mütəfəkkir. Heykəltəraş Ogüst Roden.

Pis şərhlərə misallar: "sayğac", "əlavə dəyişən", "həqiqi ədəd", "sətir". Yaxşı şərhlərə misallar: "tapılan uyğunluqların sayı", "siyahıda ən böyük ədəd".



görmür. Şərhlər üslubun önəmli elementidir, o, proqramı həqiqətən başa düşülən etməyə imkan verən çox güclü bir alətdir.

• Proqramın giriş sözü – bir neçə səhifəlik mətndən ibarət böyük giriş şərhi olmalıdır. O, aşağıdakı hissələrdən ibarətdir:

– proqramın müəllifinin adı (yaradıcılıq anonim olmamalıdır!);

– proqramın adı və təyinatı;

– həll olunan məsələnin imkan daxilində təfərrüatlı təsviri;

– məsələnin ilkin verilənlərinin və məsələnin tələb olunan nəticələrinin təsviri;

– həllin əsas ideyasının təsviri;

– məsələ və onun həllilə bağlı ədəbiyyata və digər mənbələrə istinad;

– proqramın ilkin versiyasının yazılma tarixi və sonrakı dəyişikliklərin tarixçəsi.

• Hər dəyişənin, obyektin, kəmiyyətin təsviri onların mahiyyətini aydınlaşdıran şərhlə müşayiət olunmalıdır.

• Hər bir prosedura kiçik giriş şərhi vermək lazımdır – bu konkret prosedur nə üçün nəzərdə tutulub.

• Bəzən proqramlar kifayət qədər mürəkkəb alqoritmlərə əsaslanırlar və bu alqoritmləri proqramın mətni əsasında başa düşmək o qədər də asan deyildir. Bu halda, bütövlükdə alqoritmin mənasını aydınlaşdıran yaxşı şərhlərin daxil edilməsi çox faydalı ola bilər.

• Şərhlərə etibar etmək olmaz.

Bu məsləhət proqramı yazanlar üçün yox, oxuyanlar üçündür.

Yadda saxlamaq lazımdır ki, kompüter şərhlərdə deyil, proqramda yazılanları icra edir. Yaxşı şərh proqramçının niyyəti, proqramın özünün təhlili isə proqramçının düşüncənin necə gerçəkləşməsi haqqında məlumat verir.

Digər tərəfdən bəzi hallarda ətraflı izah proqramda bəyan olunmuş hərəkətlərdən fərqli hərəkətlərin icra olunduğunu görməyə imkan verir.

Dəyişənlərin başa düşülməsi proqramın başa düşülməsinə açardır.

Hər bir proqram dəyişənlərlə ifadə olunmuş informasiyanı emal edir. Buna görə də ilk növbədə proqramda istifadə olunan dəyişənlərin mənasını anlamaq lazımdır.

• Bütün dəyişənlər təsvir olunmalıdır.

Bəzi proqramlaşdırma dilləri – KuMir, Pascal, C – proqramçıları dəyişənləri təsvir etməyə məcbur edir, əks təqdirdə proqram sintaktik baxımdan qeyri-düzgün hesab olunur.

Basic və bəzi başqa dillər proqramçıya daha çox sərbəstlik verir: dəyişənlərin istifadəsi avtomatik olaraq onlar haqqında məlumatların verilməsini təmin edir, lakin dəyişənin tipi və digər zəruri məlumatlar kontekstdən müəyyənləşir.

Lakin bu çox aldadıcı sərbəstlikdir. Məcburi təsvirlər proqramçını intizamlı edir, məlumatların strukturunun daha da mükəmməl tədqiqini stimullaşdırır. Belə proqramları oxumaq olduqca asandır: bütün dəyişənlər haqqında məlumat bir yerdə toplanmışdır və əgər oxucu hansısa dəyişənin nə olduğunu unudubsa, o, bu məlumatları haradan tapacağını bilir.

• Dəyişənlərin adlarını diqqətlə seçmək lazımdır.

Təsvir və şərhlər – bu çox yaxşıdır, lakin proqramı oxuyarkən şərhlər uzaqda – proqramın mətninin başlanğıcında qalır ora çox tez-tez qayıtmaq istəmirsən. Dəyişənin adı isə tamam başqa məsələdir.

Paskal dilində proqramdan tipik fraqment:

```
k: = k + 1; {k-nı 1 artırırıq}
```

Şərh proqramın mətninə qətiyyən heç nə əlavə etmir: oxucu onsuz da görür ki, k dəyişəni 1 artır (əgər görmürsə, onda burada ona hətta ən ətraflı şərh belə kömək etməz). Yaxşı olardı ki, artımın nəyə görə baş verməsi izah ediləydi. Məsələn, "növbəti uyğunluq tapıldı" və ya "növbəti elementə keçirik".

Adı dəyişəndən ayrılmaq olmaz, o həmişə dəyişənin yanındadır və onun proqramdakı mənasına uyğun gəlir.

Birhərfli adlardan və "hərf və rəqəm" tipli konstruksiyalardan qaçmaq lazımdır – onları qarışdırmaq asandır. Yaxın mənalı verilənlərə yaxın adlar vermək lazımdır, amma heç bir əlaqəsi olmayan verilənləri oxşar adlarla adlandırmaq olmaz.

Parçala və hökm sür!

Hədsiz-hüdudsuz olanı əhatə etmək mümkün deyil, buna görə də proqramın prosedurlara düzgün bölünməsi – böyük məsələlərin uğurlu həllinə açardır (burada prosedurlar dedikdə həm də funksiyalar, alt-proqramlar və prosedurların müxtlif proqramlaşdırma dillərində digər reallaşdırılma formaları başa düşülür).

• İstənilən prosedur başlanğıc şərhdə təsvir olunmuş öz çox da böyük olmayan məsələsini həll edir. İdeal şəkildə o, bir qısa cümlə ilə təsvir edilməlidir. Əgər ifadə olunma uzun və mürəkkəb alınırsa, deməli məsələnin bölünməsi axıra kimi aparılmayıb: hələ bir və ya bir neçə proseduru ayırmaq və yazmaq olar.

• Prosedurun həcmi elə olmalıdır ki, onu vahid tam şəklində qavramaq, onun işinin bütün məntiqini birdəfəyə başa düşmək çətin olmasın. Adətən təsvirləri və başlanğıc şərhi olmayan prosedur ekrana tam şəkildə yerləşməlidir. Əgər prosedur bir ekrana yerləşmirsə, onu daha kiçik hissələrə bölmək haqqında fikirləşməyə dəyər.

Mətn rahat oxuna bilinməlidir.

Kompüter üçün o qədər də vacib deyil ki, sizin proqramınız sətirlərə necə bölünüb və proqramda boşluq simvolları harada qoyulub. Kompilyator leksemləri (proqramın təşkil olunduğu minimal elementlər: adlar, əməliyyat işarələri, köməkçi sözlər) bir-birindən ayırmalıdır, yerdə qalanlar onun üçün əhəmiyyət kəsb etmir.



İnsan isə əksinə, mətnin tərtibinə xüsusi diqqət yetirir, axı mətnin qavranması onun tərtibatından asılıdır.

• Alqoritmik strukturu "seçdirmək" lazımdır.

Təcrübəli proqramçılar proqramı həmişə "pilləkənvari" yazırlar. Hər dəfə mətndə alqoritmik struktur (dövr, budaqlanma, seçim və sair) istifadə ediləndə mətn sağa sürüşdürülməlidir. Məsələn,

```
<dövrdən əvvəlki əməl>
| while <dövrün şərti> do
| begin
| | <dövrün gövdəsi>
| end
<dövrdən sonrakı əməl>
```

Bu yazılış konstruksiyanın harada başladığını və harada qurtardığını, ona hansı komandaların daxil olduğunu, hansılarının isə daxil olmadığını dərhal görməyə imkan verir.

• Sürüşmə tabeolma deməkdir.

Bu qayda – əvvəlkinin izahlı genişlənməsidir.



Hətta ciddi isbat olunmuş teorem vardır ki, istənilən proqramı goto-dan istifadə etmədən yazmaq olar (əlbəttə ki, əgər dildə if-then-else və while konstruksiyaları vardırsa). Peşəkar proqramçıların təcrübəsi göstərir ki, goto-dan istifadə çox vaxt proqramçının proqramın strukturunu savadlı şəkildə qura bilməməsinin nəticəsidir.

Ümumiyyətlə, hər dəfə bir fraqmentin digərinin tabeçiliyində olduğunu göstərmək istəyəndə, tabeçilikdə olan fraqmenti sağa sürüşdürməklə yazırlar. Bununla da aydın olur ki, bu sərbəst əməl deyil, əvvəlkinin bir hissəsidir.

• Boşluqlara qənaət etmək lazım deyil. Onlar oxunuşu və proqramın başa düşülməsini asanlaşdırır. Əgər bir sətirdə bir neçə operator yazılırsa, onların arasında boşluqlar qoymaq lazımdır. Bir çox hallarda boşluqların əməl işarələrində istifadə edilməsi yaxşı olardı.

Sehirli ədədlərdən qaçmaq lazımdır.

Proqramdakı ədədləri aşkar şəkildə istifadə etməyə ehtiyac yoxdur. Proqramın mətnində belə ədədə (məsələn, 179) rast gəlindikdə, oxucuya çox vaxt tamamilə aydın deyildir ki, bu ədəd nəyi göstərir və haradan çıxdı.

Belə bir ədədi dəyişmək zərurəti meydana çıxanda proqramçıya çətin olur. Tutaq ki, proqram 100 elementdən ibarət massivi emal edir. Aydındır ki, 100 ədədilə tez-tez rastlaşacağıq. Əgər indi 200 elementdən ibarət giriş massivi verilsə, proqramçı bütün proqram boyu



100 ədədini 200 ədədilə dəyişdirməli olacaq. Bu azmış kimi hardasa 99 ədədini 199 ədədilə, 101 ədədini 201 ədədinə dəyişmək lazım gələcək, hardasa məlum olacaq ki, 100 – elementlərin sayı deyil, suyun qaynama temperaturudur və dəyişdirilməyə ehtiyacı yoxdur. Bir sözlə, çox zəhmət tələb edən və heç də xoş olmayan bir iş yerinə yetirilir.

Ədədlərə adlar vermək və yalnız adlarla işləmək lazımdır. Pascal dilində bunun üçün sabitlər anlayışı mövcuddur. Sabitlər mexanizminin nəzərdə tutulmadığı dillərdə (KuMir, Basic) əlavə dəyişən daxil edirlər və ona proqramın lap əvvəlində qiymət mənimsədirlər.

Əlbəttə, 0 və ya 1 kimi ədədləri sabit etməyə ehtiyac yoxdur – bu, proqramı yalnız mürəkkəbləşdirər.

Addımbaaddım.

Məlumdur ki, əməllərin yazıldığı ardıcıllıqla yerinə yetirildiyi xətti proqramları araşdırmaq daha asandır.

Bu o demək deyil ki, dövrlərdən və budaqlanmalardan qaçmaq lazımdır – onlarsız heç bir mürəkkəb alqoritmi qurmaq mümkün deyildir. Ən yaxşısı, konstruksiyaların qısa olmasıdır, bu halda onlardan hər birinə ümumi xətti ardıcıllıqda ayrıca əməl kimi baxıla bilər.

• goto olmadan.

goto operatoru yerinə yetirilmənin ardıcıllığını pozaraq proqramı azca qısaldır. goto-dan sonra növbəti sətri yox, keçidin göstərildiyi yeri oxumaq lazımdır.

Lakin keçid olan hissələrdə təhlükə gözlənilə bilər. Səhvlər daha çox məhz orada baş verir. Səbəbi aydındır: proqramın eyni yerinə müxtəlif yerlərdən müxtəlif vəziyyətlərlə düşmək olar. Bütün bu vəziyyətlərə (şərtlərə) nəzarət etmək çox çətindir, problemlər də buradan yaranır.

• Dövrləri yarımçıq kəsmək lazım deyil.

Yerinə yetirilmə ardıcıllığının pozulması üsullarından biri də – dövrlərin

vaxtından əvvəl dayandırılmasıdır. Bəzi dillərdə hətta bunun üçün xüsusi vasitələr də vardır: break – C dilində, çıxış – KuMir dilində. Turbo Pascal dilinin yaradıcıları C dilində olan eyni adlı operatorun analoqu olaraq "break" prosedurunu hazırlamışlar.

Dövr – alqoritmik strukturu başa düşmək üçün ən mürəkkəb elementlərdəndir. Adətən hesab olunur ki, dövrün başlığı onun davamının və sonunun şərtlərini tam olaraq təsvir edir. Vaxtından əvvəl dayandırılma bu qaydanı pozur, buna görə də ən yaxşısı ondan istifadə etməməkdir.

• Proqram təbii olaraq sona çatmalıdır. Proqramın ortasında işin dayandırılmasına gətirib çıxaran komandalar qoymaq olmaz. Belə komandaları prosedurlara qoymaq xüsusilə təhlükəlidir. Başa çatma təbii şəkildə olmalıdır – prosedur idarəetməni çağırış nöqtəsinə qaytarmalıdır.

Bu qaydadan yeganə istisna – proqramın sonrakı işini mənasız edən qaçılmaz səhvin aşkar edilməsidir.

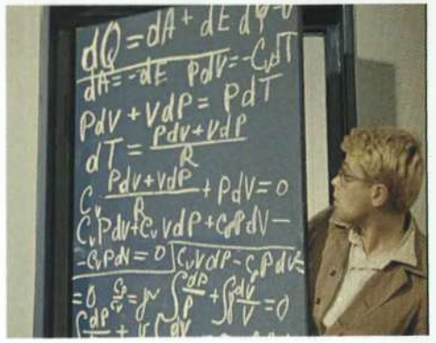
Məntiqi kəmiyyətlərin istifadəsi.

Bəzi dillərdə məntiqi kəmiyyətlər (boolean – Pascal dilində, mənt – KuMir dilində) üçün xüsusi işarələmələr mövcuddur. C dilində məntiqi kəmiyyətlər ədədi kəmiyyətlərin köməyilə göstərilir, amma onlarla həm də bütün vacib olan əməliyyatlar yerinə yetirilir.

Təzə başlayan proqramçılar tez-tez məntiqi kəmiyyətlərin mahiyyətini dərk etməyərək o yerdə ki qısa və aydın yazmaq olar, onlar çox uzun ifadələr yaradırlar.

Məsələn, Pascal dilində yazılmış proqramlarda belə müqayisəyə rast gəlinir: if a = true then... Daha sadə variant: if a then... Məntiqi kəmiyyət – heç bir əlavə müqayisəyə ehtiyacı olmayan şərtdir.

Daha bir misal: if a = true then a:= false else a:= true ifadəsini a:= not a şəklinə salaq. Necə deyərlər, fərqi hiss edin!



Bu isə məntiqi kəmiyyətin C dilində uğursuz istifadəsinə misaldır: if (a > b) return 1; return 0. Bu fraqmenti belə vermək yaxşı olar: return (a > b).

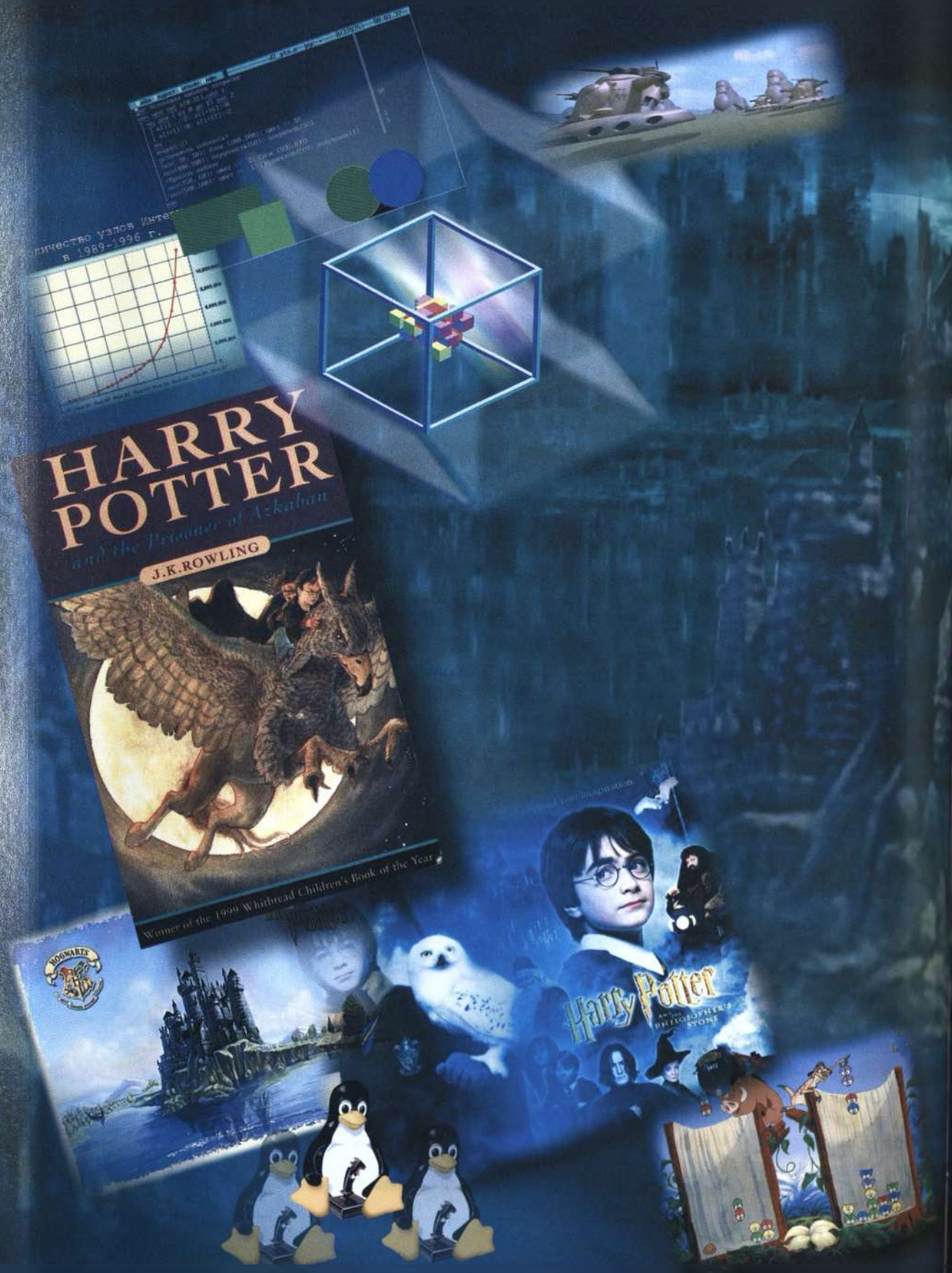
Axırıncı yazılış göstərir ki, funksiyanın nəticəsi – ədəd deyil, məntiqi kəmiyyətdir, qiyməti doğru və ya yalan ola biləcək bir şərtdir.

* * *

Əlbəttə, burada üslubun diqqət tələb edən bütün incəlikləri göstərilməmişdir. Lakin, əgər proqramçı ciddi olaraq heç olmasa yalnız bu qaydalara əməl edəcəksə, onda çox tezliklə onun proqramlarına həvəslə baxılacaq, onları oxumaq və başa düşmək asan olacaqdır. Bir müddətdən sonra isə o, özü qaydalar düşünməyə başlayacaq və öz şəxsi üslubunu formalaşdıracaqdır.



İNFORMASİYANIN SAXLANMASI VƏ İŞLƏNMƏSİ





İNFORMASİYANIN SAXLANMASI

KOMPÜTER VƏ KİTAB ÇAPI

Dünya mədəniyyəti və elmi-texniki inkişafın iki dövrü arasındakı hədd keçildi. Bəşəriyyət qlobal kompüter texnologiyaları dövrünə qədəm qoydu.

Kitab çapı üzrə birinci təcrübə hələ 1041-1048-ci illərdə Çində Bi Şenom tərəfindəın başlanmışdı. Buna baxmayaraq, kitab çapının ixtiraçısı Avropada ilk olaraq çap dəzgahı yaradan İohann Qutenberq (təxminən 1394-1468-ci illər) hesab olunur. Onun təklif etdiyi çap naboru elementlərdən – *literlərdən* təşkil olunmuş mətnin istənilən sayda bir-birinin eyni olan izlərini almağa imkan verirdi. Yeni səhifə çap etmək üçün onları digərlərinə dəyişmək mümkün idi. Qutenberq həm də liter hazırlamaq üçün alət yaratdı. Əvvəlcədən üzərinə hərfin şəkli döyülmüş, çıxarılıb taxıla bilən alt qapağı olan yumşaq metaldan hazırlanmış içiboş milin üzərinə xüsusi xəlitə (ərinti) tökülürdü. Bərkimiş xəlitə hazır litermilin üzərində hərfin əks olunmuş formasını verirdi.



İlk çapçı İvan Fyodorov.

İohann Qutenberq mətbəə xəlitəsinin və mətbəə boyasının alınma reseptini hazırladı. Öz ixtiraları ilə o, kitabların istehsalı prosesini ucuzlaşdırdı və şriftlərin standartlaşdırılması yolunu saldı.

Lakin Qutenberq poliqrafiya üsulu ilə yalnız mətni hazırlayırdı. Çap olunan şəkillər və naxışlar ilk dəfə olaraq alman çapçısı Peter Şöffer tərəfindən 1457-ci ildə "Məzamir" kitabının səhifələrində meydana çıxmışdır. Metala həkk olunmuş illüstrasiyalar və nabor formalı mətn ilk dəfə bir səhifədə florensiyalı mətbəəçi Nikolo di Lorenso tərəfindən 1477-ci ildə çap olunmuşdu. 1564-cü ildə İvan Fyodorov (1510-1580) Moskvada ilk rus çap kitabı olan "Apostol"u nəşr etdi. Sonrakı illərdə çap dəzgahının modernləşdirilməsi əvvəllər əllə yerinə yetirilən ayrı-ayrı proseslərin mexanikləşdirilməsinə, həm də ardıcıl olaraq ağacdan olan hissələrin metal hissələrlə əvəz olunmasına gətirdi.

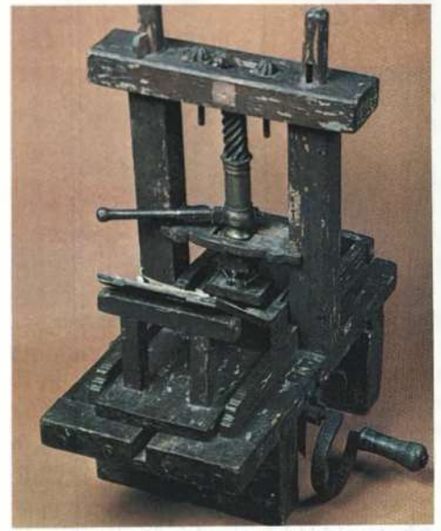
Kitab çapı texnologiyasının – informasiyanın çoxaldılması texnologiyasının genişləndirilməsi, nəinki çoxlu miqdarda uçuz başa gələn kitabların nəşr olunmasına, həm də periodik nəşrlərin – jurnal və qəzetlərin buraxılmasını təşkil etməyə imkan verdi. Bundan sonra, cəmiyyətdə informasiyanın saxlanması və ötürülməsi texnologiyalarını kökündən dəyişən digər kəşflər də oldu – fotoqrafiya, kino, teleqrafiya, telefoniya, radio və televiziya.

Kitab çapının üstünlükləri göz qabağındadır: tirajlama və üzçıxarma əməliyyatları avtomatlaşdı və insanın iştirakı olmadan yerinə yetirilməyə başladı. Lakin informasiyanın işlənməsi, nadir istisnalar olmaqla, insanlar tərəfindən yerinə yetirilir və demək olar ki, avtomatlaşdırılmamışdır.

İkinci minilliyin sonlarında aydın oldu ki, cəmiyyətin informasiya təminatında böhran halları başlayır:

• Toplanmış informasiyanı yenidən işləmək çətinlik törədir. Kitab çapı dövründə cəmiyyətdə dövr edən informasiyanın həcmi sel kimi artır. Bu zaman informasiyanın işlənməsi əvvəlki kimi, "əl işi" olaraq qalır və yalnız insan tərəfindən yerinə yetirilir. Milyonlarla adam informasiyanın sistemləşdirilməsi və lazım olan informasiyanın axtarışı ilə məşğuldur. Axtarış getdikcə daha çox xərc və vaxt tələb edir və etibarsız olur. Daha əvvəllər ixtira olunmuş lazımi texnologiya və qurğuların yenidən kəşfi daha ucuz başa gəlir, nəinki mövcud informasiyanı dəryada axtarıb tapmaq;

• Toplanmış informasiyanın növləri müxtəlifdir. İnformasiyanın müxtəlif növləri – mətnlər, görüntülər, səs – müxtəlif şəkildə təsvir olunub, yazmaq, oxumaq və köçürmək üçün tamamilə fərqli qurğular, müxtəlif saxlanma şəraiti və insan tərəfindən işlənmək metodikası tələb edir;



İvan Fyodorovun çap dəzgahı.

• Toplanmış informasiyanı yadda saxlamaq çətindir. İnformasiyanın köhnə saxlama texnologiyaları yeni tələblərə cavab vermir. İnformasiya daşıyıcıları – kitablar, jurnallar, fotoşəkillər, kino və maqnit lentləri – vaxt keçdikcə köhnəlir və ya istismar müddətində aşınır. Çap olunmuş mətnlərin, fotoşəkillərin, audio yazıların surətinin çıxarılması əməliyyatları müəyyən təhriflər yaradır, lakin bir neçə dəfə üzçıxarmadan sonra bu təhriflər nəzərə çarpacaq dərəcəyə çatır. Yazan və oxudan aparatları necə təkmilləşdirirsən təkmilləşdir, informasiyanın analoq şəkildə təsviri zamanı onun təhrif olunması qaçılmazdır.

Təbii çıxış yolu, qoyulmuş üç problemin həll olunmasındadır. Kompüterlərdən informasiyanın işlənməsinin universal aləti kimi istifadə olunması bu prosesi avtomatlaşdırmağa və bir çox sahələrdə informasiyanın işlənməsini insanın iştirakı olmadan yerinə yetirməyə imkan verir. İnformasiyanı saxlayarkən və ötürərkən onun analoq formadan rəqəmli formaya keçirilməsi nəinki ancaq uzunmüddətli saxlama problemini həll edir, həm də lazım olan informasiyanın axtarışını daha əlverişli edir.

KAĞIZ

Kağızın hazırlanmasının qədim üsulu.



Özünün ilk mətnlərini insan çubuqla qum üzərində və kömürlə mağara divarında, yaş gil lövhəciklər üzərində və daş lövhələr üzərində cızırdı... Lakin onların hamısı ya uzunmüddətli deyildi, ya tez qırılırdı, ya da istifadə üçün münasib deyildi (deyirlər ki, Qədim Çində bəzi əsərlərin daşınması üçün bir neçə araba lazım idi, çünki onlar bambuk lövhəciklər üzərində yazılırdı).

Yazı üçün əlverişli olan materialı ilk olaraq qədim Misir əhalisi aşkar etmişdir. Bu bizim qamışa oxşar papirus idi. O, Nil çayı vadisində bitirdi və hətta mədəni bitki kimi yetişdirilirdi. Ondan kəndir, həsir, ayaqqabı, qayıqlar və onlar üçün yelkənlər düzəldirdilər. Onun gövdələrindən misirlilər uzun zolaqlar kəsirdilər, yapışqanla birləşdirirdilər, məngənə altına qoyur və günəşin altında qurudurdular. Bu yolla yazı üçün əlverişli nazik, yumşaq və möhkəm vərəqlər alınırdı, onları yapışqanla yapışdırır və uzun lülələr şəklində bükürdülər. Təsadüfi deyil ki, Avropa dillərində "kağız" sözü bizə papirusu xatırladır: fransız və alman dillərində – papier, ingilis dilində isə paper.

Məşhur Norveç alimi və səyyahı Tur Heyerdal (1914-2002) "Ra" papirus qayıqlarında Atlantik okeanını üzüb keçmişdir.





Təqribən b.e.ə. VII əsrdə papirus bükümləri Misirdən Yunanıstana, oradan isə Romaya daxil oldu. Papirus az tapılırdı və ona görə də başqa ölkələrdə yazı üçün müxtəlif materiallar tapırdılar. Çində bu ipək idi, Rusiyada məişətdə tozağacı qabığından istifadə olunurdu – onun üzərində məktublar, iş sənədləri – vəsiyyətnamələr, qaimələr və s. yazılırdı (məşhur Novqorod tozağacı qabığındakı sənədlər bizim dövrümüzə qədər əcdadların gündəlik həyatından bir çox məlumatlar gətirib çıxarmışdır). Perqam çarlığında xüsusi qaydada hazırlanmış qoyun və dana dərisindən – perqamentdən istifadə olunurdu. Əfsuslar olsun ki, o, çox baha idi və bundan başqa, rütubətin təsirinə məruz qalırdı. Antik dövrlərdən başlayaraq, gündəlik həyatda üzü mumla örtülü taxta parçalarından istifadə olunurdu. Onların üzərində iti uclu metal vasitə ilə hər hansı hesablamalar və məlumatlar yazılırdı (metal çubuğun digər başı yastı olurdu, onun köməyilə mumu hamarlayırdılar).

B.e.-nın II əsrində Çində bitki məmulatından kağız almaq üsulu kəşf olundu. Kağızı bambukdan, qamışdan və ağac qabığından alırdılar. Sonralar onun hazırlanması sirrini ərəblər (onlar kəndir lifindən və kətandan istifadə edirdilər) və təqribən X əsrdə avropalılar öyrəndilər. Kağızın keyfiyyəti o qədər yüksək idi ki, o dövrün əlyazmaları bizim günlərə qədər çox yaxşı vəziyyətdə gəlib çatmışdır. Kağız ağ rəngini və elastikliyini saxlamışdır. O, perqamentdən çox ucuz idi və buna görə də tez bir zamanda onu sıxışdırıb aradan çıxartdı. Lakin kağız hələ uzun müddət zənginlik göstəricisi hesab olunurdu – o yalnız ən mühüm məqamlarda istifadə olunurdu. Yalnız bir neçə əsr keçdikdən sonra kağız həqiqətən kütləvi məhsul oldu, bu isə kitab çapının sürətlə artmasına, qəzet işinin inkişafına şərait yaratdı.

Min illər ərzində kağız insana sədaqətlə xidmət etdi. Kağız istehsalının texnologiyası və mətnin kağız üzərinə köçürülməsi qaydaları təkmilləşdi.

Əvvəlcə qaz lələyini qələmlə, sonra isə polad qələmi mürəkkəbə batırmaqla, avtoqələmlə, diyircəkli qələmlə, flomasterlə yazmağa başladılar. Paralel olaraq texniki vasitələr inkişaf edirdi – çap dəzgahı, yazı maşını, son on ildə isə kompüterə qoşulmuş çap qurğuları – printerlər.

Kompüterlərin meydana gəlməsilə kağızsız texnologiyaların tətbiqinin vacibliyindən danışmağa başladılar. Axı yazmaq, düzəliş aparmaq və mətnləri saxlamaq, onları dəyişmək kağız üzərinə köçürmədən də mümkündür. Baxmayaraq ki, əvvəlki kimi tamamilə kağızsız keçinmək mümkün deyil, bəşəriyyət çox sürətlə yazı qabiliyyətini itirir. Ona görə də xəttə görə insanın xarakterini müəyyən edən qrafoloq peşəsi görünür ki, çox keçmədən keçmişdə qalacaq. İnsanlar demək olar ki, bir-birinə kağız üzərində məktub yazmağı yaddan çıxarmışlar, jurnalistlər redaksiyaya öz reportajlarını elektron poçt vasitəsilə göndərirlər, şairlər və yazıçılar isə əlyazması nə olduğunu unutmuşlar. Öz əsərlərini indi onlar artıq disketlərdə doc formatlı fayl şəklində saxlayırlar. Biz bu gün Puşkinin, Lev Tolstoyun əlyazmalarını görməsəydik və Mixail Bulqakovun "Master və Marqarita" əlyazması üzərindəki onillik işindən faylın sonuncu versiyası qalsaydı... nə qədər kasıb olduğumuzu təsəvvür etmək çətin olmazdı.

Vaxtilə məktəblərdə hüsnxət dərslərinin olmasını artıq heç kim xatırlamır. Müasir uşaq üçün bir neçə cümləni aydın yazmaq problemə çevrilmişdir. Təsa-

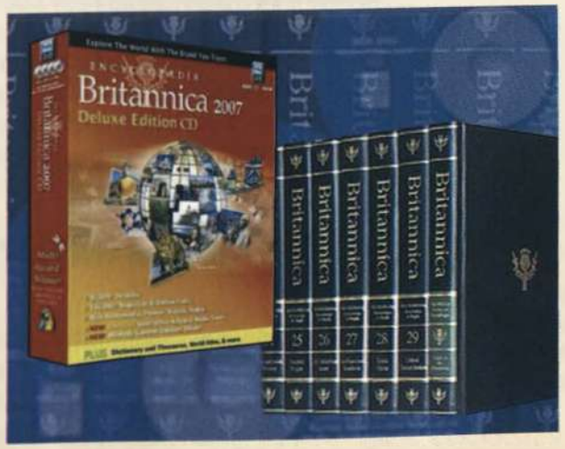
ALİMLƏR ÜÇÜN KİTABXANA

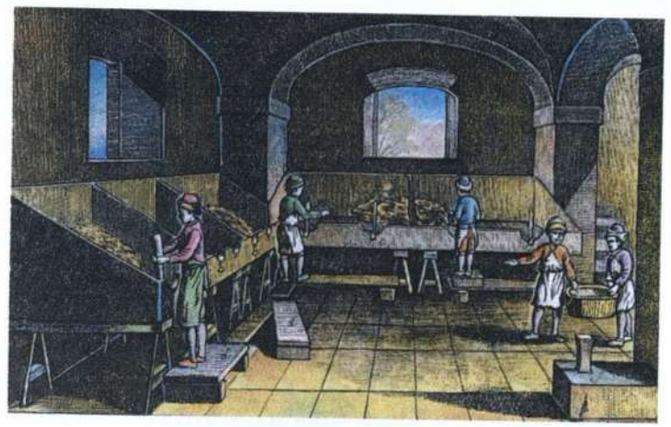
Keçmiş minillikdə dəfələrlə alimlər lazım olan məlumatın saxlanması və kitablarda, jurnallarda axtarılması üsullarına müraciət etmişlər. Belə ki, varlı bir insan olan və yeni elmi kəşflərlə maraqlanan Çarlz Darvin özünəməxsus kitabxanaya malik idi. Bütün ədəbiyyatı o, iki nüsxədə əldə edirdi. Aldığı jurnallardan və kitablardan ancaq onu maraqlandıran suallara aid səhifələri kəsirdi, qalanlarını isə məhv edirdi. Lazım olan vərəqləri o, sistemləşdirir və saxlayırdı.

Bu cür yanaşma fasiləsiz balanslaşdırma tələb edir, çünki ilk baxışda lazımsız görünən və sonradan gərək ola bilən hissəni atmaq təhlükəsi mövcuddur. Bundan başqa, şəxsi kitabxananın təşkilinin bu metodu lazım olan məqalənin və ya kitabın axtarılması zamanı o qədər də kömək etmir.

XX əsrdə istifadəni asanlaşdırmaq üçün dəfələrlə informasiyanı sistemləşdirməyə cəhd göstərmişlər. Kitabxanada məlumatın axtarılması nəinki bir neçə saat və gün, hətta aylar və illər tələb edirdi. Məlumatı ensiklopediyalarda axtarmaq daha rahatdır, çünki burada məqalələr əlifba sırası ilə düzülmüşdür. Elmi informasiyanın axtarılmasında referativ jurnallar alimlər üçün çox yaxşı kömək idi. Keyfiyyətli axtarış isə mətinləri rəqəmli formaya köçürmədən mümkün deyil.

Akademik İvan Vasilyeviç Obreimov "İnformasiyanın rəqəmli kodlaşdırılması haqqında" kitabçasında sitat maşınının yaradılmasını təklif etmişdi. Bu, elmi işçiyə kitabxanada saxlanılan informasiyanın bir hissəsini sistemləşdirməklə və rəqəmli formaya keçirməklə kitabxana yaratmağa imkan verir. Bu qismən mexaniki maşının köməyilə axtarışın sürəti saatda 225 sitata çata bilərdi. Aydındır ki, bu cür sürət və rahatlıq müasir kompüterli kitabxana və ensiklopediyalarla heç cür müqayisəyə gələ bilməz, belə ki, 32-cildlik "Britannica" ensiklopediyası dörd CD-ROM-da yerləşdirilmişdir.





Kağız istehsalı üçün materialın hazırlanması

düfi deyil ki, bir çox alimlər həyəcan təbili çalırlar: sübut olunub ki, insan beyni xüsusən yazı zamanı sürətlə inkişaf edir!

Lakin tərəqqilə mübarizə aparmaq mənasızdır, həm də lazım deyil. Onun nailiyyətlərindən istifadə etmək daha düzgündür – bunu xüsusi qələm və kağızdan ibarət qeyri-adi yazı dəstinin yaradılması barəsində məlumat vermiş İsveç alim və mühəndisləri qərara almışlar.

Xarici görünüşünə görə rəqəmli avtoqələm adi qələmdən heç nə ilə fərqlənmir. Lakin onun daxilində mürəkkəb üçün kiçik balondan başqa, miniatür rəqəmli kamera, mikroprosessor, yaddaş və ötürücü qurğu var. Buna baxmayaraq avtoqələmin çəkisi 50 qramdan azdır! İşə başlamaq üçün heç bir düyməni basmaq lazım deyil, yalnız qapağı çıxarmaq kifayətdir. Kağız üzərində qələmucunun hərəkəti zamanı kamera bir saniyədə 100-ə qədər çəkiliş edir. Onlar prosessor tərəfindən emal edilir, alınan məlumat isə mobil telefonun və ya kompüterin yaddaşına göndərilir. Kağız üzərində yaranan əlyazma mətni eyni zamanda monitorun ekranında öz əksini tapır, lazım gəldikdə isə onu çap mətninə çevirmək olar.

Belə qələmlə xüsusi kağız üzərində yazmaq lazımdır. Xarici görünüşünə görə "rəqəmli kağız" adi kağızdan yalnız öz rəng çalarlarına görə fərqlənir və "bəyaz gecə" adlanır. Xüsusi texnologiyadan istifadə etməklə, hər bir vərəq üzərinə çoxsaylı mikroskopik nöqtələr qoyulur. Bu nöqtələr bir-birindən 0,3 mm məsafədə yerləşirlər. Qələm vərəq üzərində hərəkət etdikcə, mürəkkəb bu nöqtələri rəngləyir. Şəkilləri emal edən prosessor birmənalı olaraq, onların koordinatlarını müəyyən edir (vərəq üzərində nöqtələr elə yerləşmişdir ki, bunun üçün onların, minimum miqdarı kifayətdir) və bu yolla rəqəmli təsvir formalaşdırılır.

Bu texnologiyanın istifadə olunması mobil telefonun displeyinə və ya elektron poçt ilə nəinki yığılmış məlumatları, həm də əlyazma mətnlərini göndərməyə imkan verir. Məsələn, ondan əl ilə doldurulmuş anketlərin, ərizələrin, həkim reseptlərinin göndərilməsində istifadə etmək olar (ola bilsin ki, tezliklə "elektron imza" sözbirləşməsində itirilmiş olan "imza" sözü özünün ilkin mənasını alacaq. Şübhəsiz, digər çoxsaylı tətbiqlər də olacaqdır. Deməli, əlyazmalar dövrü informasiya əsrində də davam edəcəkdir.

NSLOIO BXEQBXUİPİÇĞBRİ

Məqalənin adı tilsim deyil, yalnız digər kodda yazılmış "mətnin kodlaşdırılması"-dır.

Əgər kompüterin yaddaşında saxlanılmış mətni çıxışa (ekrana, yaxud çapa) vermək üçün müxtəlif kodlaşdırmalardan istifadə edilərsə, onda çıxarışın nəticəsi müxtəlif olacaq. Mümkün kodlaşdırmaların sayı çoxdur. Mətnin bir neçə kompüterdə oxunması üçün bu kompüterlərin hər birinin dəstəklədiyi kodlaşdırma mövcud olmalıdır. Standart kodlaşmaların seçim məsələsi belə yaranır.

MƏTNİN STANDART KODLAŞMALARI

Çıxış zamanı simvolun özünə çevrilən ədəd simvolun kodu adlandırılır. Dilin simvolları ilə ədədlər yığımı arasında uyğunluq cədvəlini mətnin kodlaşdırılması cədvəli və ya sadəcə olaraq kodlaşdırma adlandırırlar.

Ən birinci standart kodlaşdırma ASCII (American Standard Code for Information Interchange) kodlaşdırmasıdır. Onu XX əsrin 50-ci illərinin sonunda yaratmağa başladılar, 1968-ci ildə bu kodlaşdırma ABŞ-nın Milli Standartlaşma İnstitutu tərəfindən (ANSI) təsdiqləndi.

ASCII kodlaşdırması simvolları kodlaşdırmaq üçün 0-dan 127-ə qədər olan ədədlərdən istifadə edir. Cədvəldə ASCII mətn simvollarının 32-dən 126-a qədər kodları göstərilmişdir. Qalan ədədlər isə (0–31 və 127) idarəedici simvolların (mətn olmayan) kodlaşdırılmasında işlədilir (məsələn, səs siqnalının simvolu, yaxud özündən sonra gələn simvolları yeni sətrə keçirən sətrin sonunu göstərən simvol).

Kod	Simvol
32–47	! " # $ % & ' ( ) * + , - . /
48–63	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
64–79	@ A B C D E F G H I J K L M N O
80–95	P Q R S T U V W X Y Z [\ ] ^ _
96–111	` a b c d e f g h i j k l m n o
112–126	p q r s t u v w x y z { | } ~

0-dan 127-dək ədədlərin ikilik say sistemində yazılışı sayı 7-dən çox olmayan rəqəmlərdən ibarətdir.

Bu o deməkdir ki, ASCII-də simvolun kodu kompüter yaddaşında 7 bit yer tutur (kompüterdə ünvanlanan ən kiçik informasiya vahidi – bayt yaddaşda 8 bit yer tutur). Simvolun kodunu yerləşdirmək üçün yaddaşda 1 bayt yer ayrılması daha münasib hesab olunmuşdur. Asanlıqla hesablamaq olar ki, bir baytda 256 ədəd (0–255) yerləşdirmək olar. Bu o deməkdir ki, bir baytda ASCII-in simvollarından başqa daha 128 simvol da kodlaşdırmaq mümkündür. Məsələn, rus, yunan, yaxud hər hansı digər dillərin əlifbalarının simvollarını, riyazi simvolları (kök, inteqral, cəm və s. işarələri), psevdoqrafik simvolar (cədvəllər çəkmək üçün vertikal, horizontal və digər xətlər) və s.

Mətnin yeni yaradılan 8-bitlik kodlaşdırılmasının əsasında ASCII kodlaşdırılması durur, əlavə simvolların kodlaşdırılması üçün isə 128-dən 256-ya kimi, həmçinin 0-dan 31-ə kimi (ASCII-nin idarəedici simvollarının əvəzinə) ədədlər götürülmüşdür.

RUS KODLAŞDIRMALARI

Rusiyada üç rusdilli kodlaşdırma daha geniş yayılmışdır: KOI-8, alternativ kodlaşdırma (daha çox DOS kodlaşdırması da adlanır) və Windows.

KOI-8 (8 bit informasiya ilə mübadilə üçün kodlaşdırma) UNIX əməliyyat sisteminin idarəsi altında işləyən kompüterlərdə standart rus kodlaşdırma cədvəlidir. Bundan başqa, KOI-8 faktiki olaraq, qlobal İnternet şəbəkəsində rusdilli mətnləri vermək üçün standart olmuşdur.

Alternativ kodlaşdırma MS DOS əməliyyat sistemində geniş istifadə edilir. Onun adı 80-ci illərdə Dövlət standartı (DÜST) tərəfindən iki – "əsas" və "alternativ" kodlaşdırmanın təsvirindən yaranmışdır. "Əsas" kodlaşdırma əlverişli olmadığına görə istifadədən çıxmışdır.



Windows kodlaşdırması MS Windows əməliyyat sistemində istifadə olunur. Bu kodlaşdırma KOI-8 və alternativ kodlaşdırmadan yalnız simvolların kodlaşması ilə deyil, həm də kodlaşdırılmış simvolların toplusu ilə də seçilir. Məsələn, bu topluda psevdoqrafika simvolları olmaya da bilər, əvəzində cürbəcür dırnaq simvolları vardır.

8-BİTLİ KODLAŞDIRMALARIN PROBLEMLƏRİ

Mətnlə işləyən proqram hətta hər üç rus kodlaşdırmasını dəstəkləsə belə, ancaq o zaman mətni ekranda və ya çap qurğusunda düzgün göstərə bilər ki, mətnin məhz hansı kodlaşdırmada yazıldığını bilsin.

Bəzi proqramlar özləri mətnin kodlaşmasını onun başlanğıcındakı bəzi hərflərin daxilolma tezliyinə görə hesablaya bilirlər. Ancaq bu üsul standart olmayan mətnlər üçün işləməyə bilər (məsələn, əgər mətnin başlanğıc hissəsi, çox vaxt proqramların yazılışında olduğu kimi, yalnız latın hərflərilə yazılmışsa). Bu zaman istifadəçi mətndə istifadə olunan hər bir kodlaşdırmanı əllə seçməli olur.

Mətnin yazıldığı dil məlum olmayan halda isə, böyük miqdarda 8-bitli milli kodlaşdırma mövcud olduğundan mətnin daxil edilməsində istifadə edilən kodlaşdırmanın müəyyən edilməsi məsələsi kifayət qədər mürəkkəb olur.

BEYNƏLXALQ KODLAŞDIRMALAR

Mətnin beynəlxalq kodlaşdırılması üçün UCS (Universal Character Set – simvolların universal toplusu) yaradılmışdır. Bu kodlaşdırmaya bütün dünya dillərinin simvolları, eləcə də bəzi başqa işarələr, məsələn, sözlərin tələffüzünün yazılışında istifadə edilən IPA fonetik əlifbası daxildir.

Hal-hazırda UCS simvollarının yığılması üçün iki beynəlxalq kodlaşdırma mövcuddur: Unicode və ISO/IEC 10646. Bu kodlaşdırmalarda bir simvolu kompüterdə göstərmək üçün 2 bayt yaddaş istifadə edilir. Bu, 65536 simvolu kodlaşdırmağa imkan verir ki, bu da UCS həcmindən çoxdur.

Unicode və ISO/IEC 10646 kodlaşdırmaları kodlaşdırılmış simvolların tərkibinə və onların kodlarına görə tamamilə eynidir. Fərq yalnız praktiki tətbiqdədir. Unicode kodlaşdırmasında tətbiqi proqramlar və əməliyyat sistemləri arasında mətnləri köçürmək üçün çoxlu alqoritm və spesifikasiyalar vardır.

Gələcək – beynəlxalq kodlaşdırmanındır. Unicode Windows və UNIX kimi populyar əməliyyat sistemlərində getdikcə daha geniş istifadə edilir. Həm də göstərilən əməliyyat sistemlərinin rəqabət qabiliyyəti verilən kodlaşdırmanın istifadə dərəcəsindən çox asılıdır, çünki bu, informasiya mübadiləsi prosesini kifayət qədər asanlaşdırır.

MƏTN REDAKTORLARI

KOMPÜTERLƏR ÇAPÇILIQDA

Mətnin kompüterə daxil edilməsi imkanının meydana gəlməsi yazı sahəsində çapçılığın ixtirası kimi bir inqilab yaratdı.

Mətnin EHM-ə daxil edilməsi mətbəədə kitabın yığılması işinə bənzəsə də (yəni bir dəfə yığmaqla istənilən qədər üzünü çap etmək olar), onun bir sıra üstünlükləri vardır. Səhv aşkar edilərkən mətnin görünüşünü pozmaq lazım deyil – çatışmayan simvolları qoymaq üçün o biri simvolların yerini dəyişmək və ya əksinə, artıqları sildikdən sonra qalanları bütün sətir boyunca yenidən yerləşdirmək lazım gəlmir. Yerdəyişmələr kompüterin yaddaşında gedəcək və çapda əks olunmayacaq.

Tanınmış Kolumbiya yazıçısı, ədəbiyyat üzrə Nobel mükafatı laureatı, unudulmaz "Yüz il tənhalıqda" romanının müəllifi Qabriel Qarsia Markes hələ 20-ci əsrin 80-ci illərində kompüterdə əsərlərini ilk yaradanlardan idi. Hal-hazırda kompüterdə mətnlərin yığılması onun ən geniş yayılmış imkanlarından biridir.

REDAKTORLARIN TƏKAMÜLÜ

Mətni kompüterin yaddaşına daxil etmək və daxil olunmuş mətnin üzərində dəyişiklik aparmaq (redaktə) üçün *redaktor* adlanan xüsusi kompüter proqramlarından istifadə edilir.

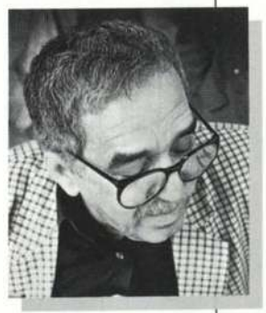
İlk redaktorlar komanda redaktorları idi: kompüterə mətnlə nə etmək lazım olduğu barədə əmrlər verilirdi. İstifadəçi kompüterin monitorunda mətni, hətta mətnin çapa verilməsi komandası yerinə yetirildikdən sonra da görə bilirdi.

Məsələn, 1975-ci ildə yaradılmış TECO redaktorunda mətnlə işin səciyyəvi seansı təqribən belə görünürdü:

```
! DİSPLAY CURRENT LİNE'S ASCII CODE
, 20 PER DISPLEY-LINE. !
! D F KOENIG, 1989-11-24. !
@^U.L [^[^A Line length = ^A^[

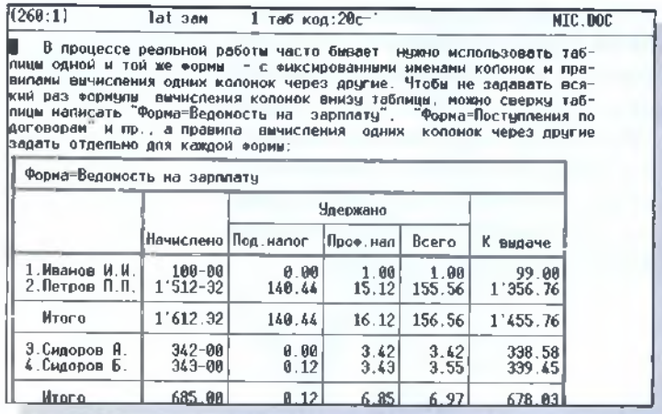
MN0U. 0^|QN<Q. 0A: =^|^A
^A^| ( (%. 0/10) * 10)
Q.0"E^| ( (Q.0/20) * 20) – Q.0"E^[^A
^A^[ ^[ [^A  ^A^[ ^[ '^[ '^[ >^[^A
^A^[ [
```

Göstərilən fraqmentdə hər bir simvol redaktorun hər hansı komandasını, məsələn, hansısa simvolun mətnin verilmiş sətrinin hər hansı bir mövqeyinə qoyulmasını göstərir.



Qabriel Qarsia Markes.

Proqramçıların belə bir oyunu vardı: öz soyadını komandalar kimi redaktora daxil etmək və bu komandalar dəstini redaktora tətbiq etdikdə redaktə olunan mətnlə nə baş verəcəyini tapmaq.



(260:1) lat эам 1 таб код:20с-' MIC.DOC

В процессе реальной работы часто бывает нужно использовать таблицы одной и той же формы - с фиксированными именами колонок и правилами вычисления одних колонок через другие. Чтобы не задавать всякий раз формулы вычисления колонок внизу таблицы, можно сверху таблицы написать "Форма=Ведомость на зарплату", "Форма=Поступления по договорам" и пр., а правила вычисления одних колонок через другие задать отдельно для каждой формы:

Форма=Ведомость на зарплату

	Начислено	Удержано			К выдаче
		Под.налог	Проф.нал	Всего	
1.Иванов И.И.	100-00	0.00	1.00	1.00	99.00
2.Петров П.П.	1'512-32	140.44	15.12	155.56	1'356.76
Итого	1'612.32	140.44	16.12	156.56	1'455.76
3.Сидоров А.	342-00	0.00	3.42	3.42	338.58
4.Сидоров Б.	343-00	0.12	3.43	3.55	339.45
Итого	685.00	0.12	6.85	6.97	678.03

DOS üçün "MikroMir" mətn redaktorunun ekranından bir şəkil.

Komandalar ardıcıllığının yazılışı simvolların əcaib görünüşlü mənasız yığımını xatırladır. TECO kimi komanda redaktorları həm də sətir redaktorları adlanır (*ing.* line editors). Onların əsas mənfi cəhəti ondadır ki, redaktə zamanı dəyişdirilməli olan simvolun və sətrin nömrəsini bilmək və komandada göstərmək lazımdır.

Redaktorların sonrakı nəsli – ekran redaktorları (screen editors) və ya WYSIWYG-redaktorlarıdır (*ing.* what you see is what you get – "nə görürsən, onu da alırsan"). Onların istifadəsi zamanı mətn ekranda çapa çıxacağı şəkildə əks olunur. Mətndə düzəliş olunacaq yer xüsusi işarə – kursor ilə göstərilir, onu da mətn boyunca yuxarı, aşağı, sağa, sola hərəkət etdirmək mümkündür. Kursor cari simvoldan solda/aşağıda yerləşən nazik şaquli/üfüqi xətt, yaxud simvolun üzərində düzbucaqlı formasında olur, ona görə də simvolun özünün təsviri neqativ alınır.



REDAKTORLAR MƏTNİN VİZUALLAŞDIRILMASI VƏ DƏYİŞDİRİLMƏSİ ÜÇÜN SİSTEMLƏRDİR

Bütün müasir redaktorlar mətnin daxil edilməsi və dəyişdirilməsi zamanı onun təsvirini monitorun ekranında saxlayır. Redaktorun əsas işi – yığılmış mətni göstərməkdir; istifadəçi hər hansı bir klavişi basdıqda bunun mətni necə dəyişdirəcəyini müəyyənləşdirməkdir; mətnin ekranda təsvirində düzəliş etməkdir.

Bundan başqa, müasir redaktorlar çoxlu sayda başqa funksiyalar da yerinə yetirirlər: yığılan mətni yoxlayır, onu formatlaşdırır, çapa verir, elektron poçtla göndərir və s.

MENYU NƏYƏ GƏRƏKDİR

Redaktorun hər bir komandasını hər hansı müəyyən klavişi basmaqla çağırmaq olar. Redaktorda nə qədər çox müxtəlif funksiyalar (komandalar) olarsa, onları çağırmaq üçün bir o qədər çox müxtəlif klaviş tələb olunur.

Klavişlərin kombinasiyasını yadda saxlamaq üçün xeyli güc və vaxt sərf etmək lazım gəlir, ona görə də redaktorla rahat işləmək üçün menyudan istifadə edilir.

Son vaxtlar menyular getdikcə daha çox standartlaşdırılır. Redaktorun ekranının yuxarı sətirdə bir neçə açar söz yerləşir: "Fayl" "Düzəliş", "Görünüş", "Format" və digərləri. Hər bir sözə istifadəçidən gizlin menyu uyğundur. O, yalnız çağırıldıqda işıqlanır: "siçan"ın göstəricisini uyğun sözün üstünə gətirdikdə və "siçan"ın düyməsini, yaxud Alt klavişini basdıqda menyu işıqlanır. Menyuda açar-sözlə bağlı komandalar sadalanır.



FAYLLARLA İŞ

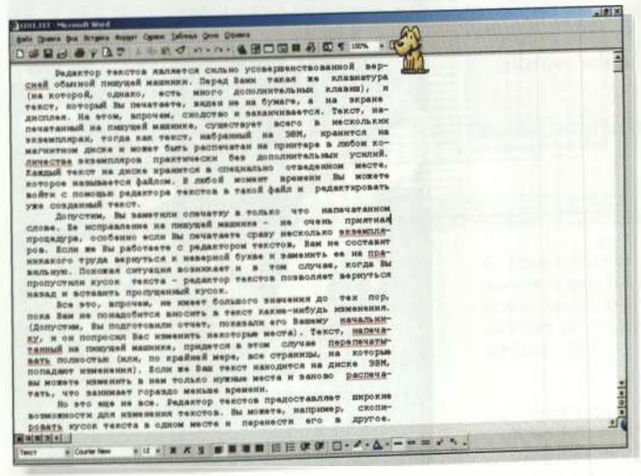
"Fayl" sözü informatikada "qovluq" kimi anlaşılan file ingilis sözündən yaranmışdır. Qovluq-fayl özündə mətn, şəkil, kompüter proqramı və s. saxlaya bilər.

Mətnlə iş "Açmaq" komandası ilə başlayır. Qovluq açıldıqdan sonra onda saxlanılan mətni görmək və redaktə etmək olar. Dəyişdirilmiş mətni həm ilkin faylda diskə yazılmış ilkin formatda saxlamaq ("Saxlamaq" komandası), həm də başqa faylda və/və ya formatda ("Necə saxlamaq..." komandası) saxlamaq mümkündür. Bu ona görə edilir ki, mətndə edilən düzəlişlər xətdə gərginliyin gözlənilmədən kəsilməsi və ya kompüterin dayanması zamanı itməsin, çünki redaktə seansı vaxtı mətndə edilən dəyişikliklər fayla dərhal yazılmır, əvvəlcə onlar aralıq yaddaşda saxlanılır.

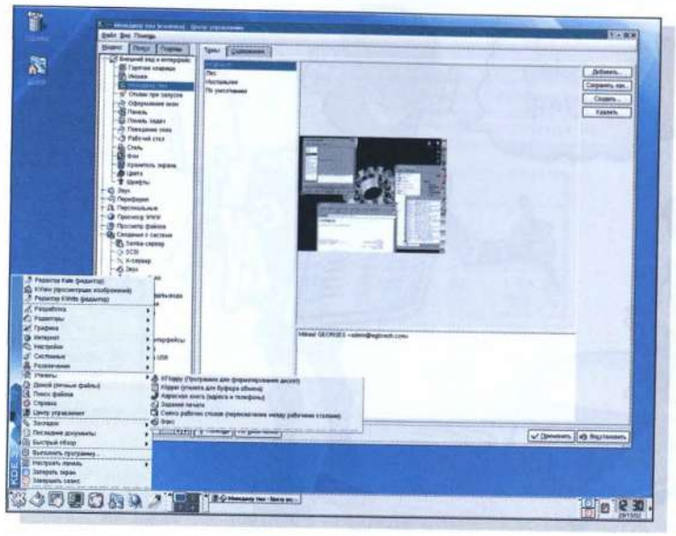
Adətən, redaktə edilən mətndən çıxarkən redaktor faylda edilən düzəlişləri saxlamaq lazım gəlib-gəlmədiyini soruşur. Əgər mətn korlanmışsa və onu ilkin variantına qaytarmaq lazımdırsa, düzəlişləri yadda saxlamadan faylı qapatmaq mümkündür.

REDAKTORLARIN UZLAŞMASI

İstifadəçinin redaktoru mənimsəməsi – uzun, çətin bir prosesdir. Redaktoru dəyişdikdə yeni anlayışlar, yeni komandaları, klavişlərin yeni funksiyalarını, menyunun yeni bəndlərini təzədən öyrənmək lazım gəlir. Belə perspektiv hətta yeni redaktordan imtina etməyə gətirib çıxara bilər. İstifadənin belə mürəkkəbliyi istehsalçıya əlverişli deyil, ona görə də istifadəçi interfeysi standartlaşdırılır.



Microsoft Word redaktorunun ekranından bir şəkil.



Linux-un X-Windows ofis sisteminin ekranından bir şəkil.

Məsələn, "proqram Microsoft Office ilə uzlaşır" anlayışı meydana gəlmişdir. Bu o deməkdir ki, proqram elə hazırlanıb ki, artıq bir çox istifadəçiyə tanış olan Microsoft Office komplektinin proqramlarına oxşayır.

Bütün mətn redaktorları yalnız bir iş – mətnin yaradılması üçün nəzərdə tutulmuşdur. Ona görə onların komandalar və funksiyalar toplusu oxşardır.

Onlar bir-birindən yalnız xüsusi imkanları ilə fərqlənirlər (məsələn, orfoqrafiyanın yoxlanması və ya özünün elektron cədvəllərinin olması). Deməli, redaktorun seçimi istifadəçinin həll etməli olduğu məsələlərə əsasən və ya onun şəxsi zövqü ilə müəyyən edilir.

MAKETLƏŞDİRMƏ

Müasir insan üçün böyük miqdarda kitab, jurnal, qəzet adi bir şeydir. Yalnız XVI əsrdə, insanlar kağızda mətni çap edə bilən maşını icad edəndə, eyni bir kitabı çox nüsxədə hazırlamaq üçün real imkan yarandı. Çapa hazırlama prosesi mürəkkəb və çox zəhmət tələb edən idi. Çap olunmalı görüntü təmizlənirdi (şəkil olmayan yerlərin səthindən üst təbəqə götürülürdü), sözlərin və durğu işarələrinin düzgün sırasının alınması üçün hər bir liter dəqiq öz yerində yerləşdirilirdi. Gələcək kitabın sətirləri bu sıradan ibarət olurdu. Belə mətndə səhvi tapmaq çox çətin idi, axı mətn gələcək səhifənin güzgüdə görünən əksi kimi yığılırdı. Sonrakı əməliyyat yığılmış mətnin qabarıq hissələrinə xüsusi mətbəə boyasının çəkilməsindən ibarət idi. Sonra, nə qədər ki, boya qurumayıb, mətn yığılmış lövhəni təmiz kağız vərəqə bərk sıxırdılar. Bununla da basma (ottisk) – gələcək kitabın səhifəsi alınırdı. Kompüterlərin meydana gəlməsilə çap prosesində, xüsusilə kitabın çapa hazırlanmasında – *maketləşdirmədə* inqilabi dəyişikliklər baş verdi.

Kitabın maketləşdirilməsi nədən başlayır? Formatın seçimindən. Müxtəlif formatlar vardır. Kitabların bir-birindən təkcə ölçülərinə görə yox, həm də formalarına görə fərqləndiyini görmək üçün kitab mağazısına girmək kifayətdir. Məsələn, çay haqqında kitab dəm çaydanı formasında ola bilər – müasir texnologiya buna imkan verir.

Kitabın formatından asılı olaraq rəssam maket qurur, yəni sahələrin ölçülərini, mətnlərin, illüstrasiyaların qarşılıqlı yerləşməsini, şriftlərin uyğunluğu, rəng seçimini, üz qabığında və arxa tərəfdə böyük illüstrasiyaları – *şmustitulları* hazırlayır, müxtəlif dizayn elementlərini, məsələn, mətnin təkrarlanan və ya oxucunun xüsusi diqqətini tələb edən elementlərini fikirləşib tapır. Bəzən mətndəki kiçik kvadratlar oxucunun diqqətini haşiyəyə, sitata və ya əsas mətnin mənasını izah edən, onu anlamağa kömək edən illüstrasiyalı maraqlı materiala cəlb edir. Maket nə qədər harmonikdirsə, onu oxumaq da bir o qədər sadə və xoşdur.

İndi qrafik redaktor və başqa xüsusi proqramların köməyi olmadan əllə işlənmiş maketə çox nadir hallarda rast gəlmək olar. Lakin əgər rəssam onu kompütersiz etsə belə, yaradılan maket yenə də xüsusi *usta–səhifələri*, şablonlar şəklində kompüterə köçürülür və *səhifələmə proqramında* onlara uyğun kitabın bütün səhifələri toplanır.

Kitabın mətni asanca oxunmalıdır. Bunun üçün rəssamlar cürbəcür şriftlər fikirləşmişlər. Məsələn, başlıqlar yaxud reklam üçün xüsusi şriftlər vardır. Belə mətn gərək o saat diqqəti cəlb etsin və çağırsın: "Məni oxu !". Populyar jurnallarda həm mətnə, həm də ümumi tərtibata, dizayna bir o qədər də diqqət verilmir. Moda jurnallarının parlaq səhifələri bəzən oxucunun görmə qabiliyyətində güclü gərginlik yaradır. Belə nəşrlərin yaradıcıları qarşılarına oxucunu məzmunlu mətnlə cəlb etmək məqsədini qoymurlar – qoy oxucu, sadəcə, gözəl şəkillərə baxsın. Əksinə, "kitab" şriftilə çap olunmuş mətni uzun müddət yorulmadan oxumaq olar. Məktəb dərslikləri üçün şriftə xüsusi diqqət ayrılır. Pis seçilmiş şrift kitabı nəinki çətin mənimsənilən edir, həm də ziyanlıdır. Belə şrift görmə qabiliyyətinin pisləşməsinə, məktəblinin yorulmasına gətirib çıxara bilər.

Mətn redaktorlarının ayrılmaz komandaları simvolun daxil olunması və silinməsidir. Bundan başqa, redaktorlarda mətnin yığılmasını asanlaşdıran (mətn üzrə hərəkəti sürətləndirən, mətnin fraqmentlərini köçürməyə imkan verən) komandalar vardır.



XX əsrin əvvəllərində mətbəədə mətnin yığılması.

MƏTN REDAKTORU

Redaktoru çağırdıqda ekranda onunla işləmək üçün pəncərə açılır. Pəncərədə, adətən, aşağıdakılar olur:

- Redaktorun və redaktə olunan mətnin adı olan *başlıq*.
- *Menyu* – başlığın altında açar sözlərdən ibarət siyahı olan zolaq.
- *Alətlər zolağı* – redaktorun bəzi komandalarına uyğun gələn düymələr olan zolaq. Düymə "siçan"ın çıqqıltısı ilə "basılır" və bununla uyğun komanda çağırılır.
- *Formatlar* – şriftlərin adları olan boks, şriftlərin ölçüsü olan boks və şriftin parametr və formatını göstərən bir neçə düymə yerləşən zolaq. Bu boksların sağ tərəfində düymə yerləşir, onu basdıqda pəncərə açılır, şriftlərin siyahısı və ya ölçüləri görünür və istifadəçi "siçan"ı çıqqıldatmaqla uyğun sətri seçə bilir.
- *Xətkeş.*
- *Mətni redaktə pəncərəsi* – mətnin təsvir olunduğu və içərisində yanıb-sönən kursoru olan ağ düzbucaqlı. Əgər mətn redaktə pəncərəsinə tam yerləşmirsə, mətni fırlatma zolaqları yaranır: sağda – şaquli fırlatma zolağı; aşağıda – üfüqi fırlatma zolağı.
- *Status zolağı* – çoxlu sayda faydalı informasiyanın, məsələn, redaktorun işi barəsində, mətndə kursorun yerləşmə koordinatları: sətrin və simvolun mövqeyi barəsində informasiyanın saxlandığı yer.

Bu kitabın hazırlanmış səhifələrinin, mətnin maketləşdirilməsi proqramının ekrandan fotosu.

Kitablar nadir hallarda illüstrasiyasız olur. Bəzən yazılmış mətnə illüstrasiya vermədən həmin yazıda nə isə izah etmək, sadəcə, mümkün deyildir. Kitab illüstrasiyaları yaratmağın çoxlu yolları vardır.

Boya, karandaş, yaxud nə isə başqa bir şeylə kağızda və ya başqa şey üzərində çəkmək olar, ya da kompüter proqramları vasitəsilə şəkil çəkmək, eləcə də təsvirli xüsusi fayllar yaratmaq imkanları vardır. Daha bir üsul vardır – fotoşəkildən, başqa kitabdan, yaxud *slayd-skanerlər* tətbiq etməklə birbaşa neqativdən görüntüləri daramaqla fotoşəkillərdən istifadə etmək üsulu.

Kollajlar yaratmaq, yəni, müxtəlif təsvirləri bir birinə uyğunlaşdırmaq mümkündür, məsələn, fotodan təsviri götürüb, həmin təsvirdən kompüterdə yalnız bir adamı "kəsib" ayırmaq və onu təzə fonda – palmalı okean sahilində və ya hotelin girişində vermək olar.

Əsas materiallar – mətn və illüstrasiyalar yığıldıqdan və maket hazırlandıqdan sonra kitabın hazırlanmasına başlamaq olar. Bu mərhələdə səhifələyici işə başlayır. O, ayrı-ayrı hissələri birləşdirərək bütov şəklə – oxucunun gördüyü kitab şəklinə gətirir. Dizaynerin fikirləşdiyinə uyğun olaraq (bəzən iki ixtisas – dizayner və səhifələyici birləşdirilir), mətni "doldurur", şəkilləri, imzaları, şərhləri gələcək kitabın səhifələrində elə yerləşdirir ki, o həm gözəl, həm də başadüşülən olsun. Bu, yaradıcı prosesdir, həmişə asan olmur...

Səhifələyicinin kompüteri onun əsas alətidir. Və çox vaxt bu ənənəvi IBM PC yox, Macintosh olur. ABŞ-da belə kompüterlər, xüsusilə təhsil və poliqrafiyada geniş yayılmışdır. Azərbaycanda hesab edilir ki, bu kompüterlər xüsusi olaraq nəşriyyat işi və rəssamlar üçün yaradılmışdır. Görünür, burada maşının etibarlılığı və interfeysin rahatlığı əsas rol oynayır, çünki yaradıcı insan üçün proqramın "ilişib qalması" səbəbindən eyni bir işi bir neçə dəfə görməli olması arzuolunmazdır.

Beləliklə, əla maşında səhifələyici əvvəlcə kompüter faylı şəklində olan virtual kitabı yaradır. Sonra bu fayl nəşriyyata düşür (müasir nəşriyyatlarda əllə yığım artıq yoxdur), orada başqa "sehrbazlar" kitabı çap edirlər.

ELEKTRON CƏDVƏLLƏR



"Bazar ertəsi 6 dərs və 2 saat müstəqil məşğələ, çərşənbə axşamı 5 dərs və 3 saat müstəqil məşğələ, çərşənbə günü 5 dərs və 3 saat müstəqil məşğələ, cümə axşamı 6 dərs və 2 saat müstəqil məşğələ, cümə günü 5 dərs və 2 saat müstəqil məşğələ".

Burada söhbət şagirdin məşğələlərindən gedir. Əgər bu yazıda cümə axşamı üçün dərslərin sayını axtarıb tapmaq lazımdırsa, onda bütün mətnə baxmaq lazım gələcək. Bu isə əlverişli deyil, ona görə də məşğələləri cədvəl şəklində yazırlar (*lat.* tabula – "məlumatların sütunlar üzrə yazılması").

Bu halda qoyulan məsələni həll etmək böyük zəhmət tələb etmir. Sadəcə, cədvəlin "Cümə axşamı" sətrilə "Dərslərin sayı" sütununun kəsişməsindəki xananı tapmaq lazımdır.

Həftənin günü	Dərslərin sayı	Müstəqil məşğələ saatları
Bazar ertəsi	6	2
Çərşənbə axşamı	5	3
Çərşənbə	5	3
Cümə axşamı	6	2
Cümə	5	2

Cədvəlin xanalarındakı rəqəmlər birbirilə əlaqəli ola bilər. Məsələn, sağdakı cədvəldə dördüncü sütunun elementləri ikinci və üçüncü sütunların elementlərinin cəmindən ibarətdir. "Cəmi" sətri isə ondan əvvəlki sətirlərin cəmidir.

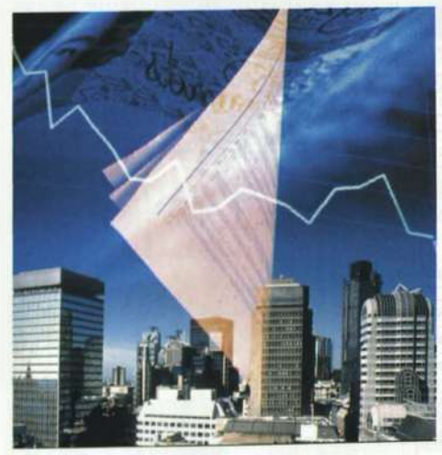
Nə qədər ki, cədvəl kağız üzərindədir, ədədlərin bir-birilə olan əlaqəsindən istifadə etmək mümkün deyildir. Lakin əgər cədvəl kompüterdə yerləşirsə, bəzi xanaların doldurulmasını maşına həvalə etmək olar. Məsələn, ikinci cədvəl üçün kompüter axırıncı "Cəmi" sətrinin xanalarını sütunlardakı verilənləri cəmləməklə, axırıncı "Saatların ümumi sayı" sütununu, əvvəlki iki sütunun verilənlərini cəmləməklə hesablaya bilər.

Cədvəlin avtomatik doldurulmasında uduş nədən ibarətdir? Birincisi, cədvəlin doldurulmasına sərf edilən vaxta önəmli şəkildə qənaət edilməsi (Tutaq ki, cədvəlin doldurulmasına 4 dəq. sərf edilir: hesablamaya – 1 dəq., 18 ədədin daxil edilməsinə – 3 dəq. Onda avtomatik doldurma zamanı yalnız 10 ədədin daxil edilməsinə vaxt sərf edilir, bu isə 2 dəq.-dən azdır). İkincisi, səhvlərin sayı azalır. Məsələn, ikinci cədvəldə 10 ədədi daxil edəndə səhv etmək ehtimalı 18 ədədin daxil edilməsindəkindən daha azdır. İnsandan fərqli olaraq kompüter hesablamalarda səhv etmir (əlbəttə, əgər elektrik cərəyanı kəsilməzsə və ya güclü elektromaqnit mancəsi olmazsa).

Cədvəllərdən verilənləri oxuyan və cədvəlləri yeni hesablanmış verilənlərlə dolduran proqramları cədvəllərin işlənməsi proqramları adlandırırlar. 20 rəqəmli ədəddən ibarət (45 987 801 734 897 235 947 və ya 0,06 237 891 209 875 611) 1 mln. xanası olan cədvəlin işlənməsi proqramın nə qədər vaxta qənaət etməyə imkan verdiyini təsəvvür etmək belə çətindir. Xüsusilə, hesablama, düsturlar həddən artıq mürəkkəb və onların sayı çoxdursa! Və belə bir proqramın hesabına nə qədər daxiletmə və hesablama səhvindən qaçmaq olar!

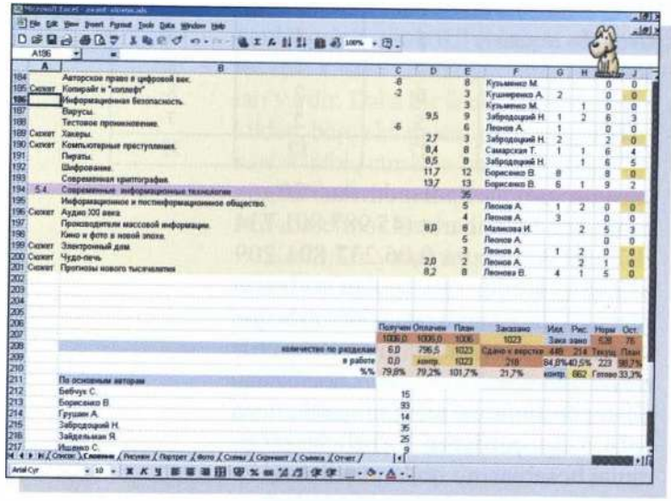
Həftənin günü	Dərslərin sayı	Müstəqil məşğələ saatları	Saatların ümumi sayı
Bazar ertəsi	6	2	8
Çərşənbə axşamı	5	3	8
Çərşənbə	5	3	8
Cümə axşamı	6	2	8
Cümə	5	2	7
Cəmi	27	12	39

Əlbəttə, cədvəllərin işlənməsi proqramının yaradılması müəyyən qədər vaxt və əmək tələb edir, lakin bir dəfə yazılmış proqram cədvəlin hər dəfə doldurulması zamanı işi tezləşdirir (cədvəl isə bəzən min və ya milyon dəfələrlə doldurulur).



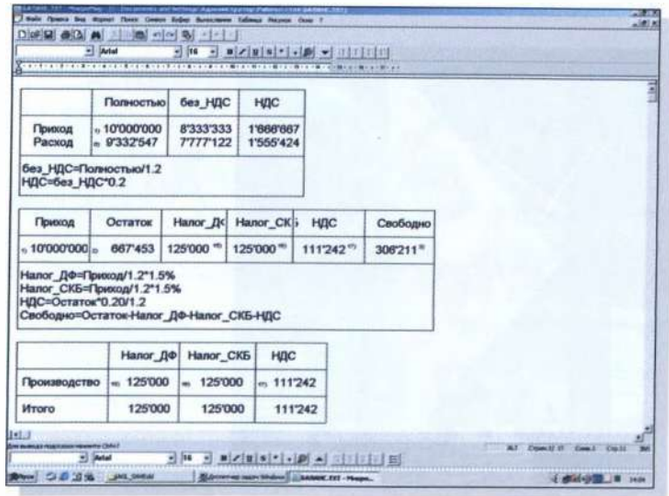
ELEKTRON CƏDVƏLLƏR

Cədvəllərin doldurulması proqramının işlənməsi prosesi necə baş verir? İstifadəçi proqramçıya cədvəlin doldurulması alqoritmini, yəni xanalara yazılmalı olan ədədlərin hesablanması düsturlarını deyir. Proqramçı bu düsturları proqramlaş-



Excel elektron cədvəllər proqramının ekranından bir şəkil.

dırma dilinə çevirir, eləcə də giriş verilənlərinin (istifadəçi tərəfindən yığılan ədədləri) proqrama daxil edilməsini və hesablama nəticələrinin proqramdan çıxışa verilməsini (alınmış ədədlərin cədvələ yazılışını) reallaşdırır.



MikroKalk elektron cədvəllər proqramının ekranından bir şəkil.

Cədvəlin formatı dəyişdikdə (məsələn, əlavə sütun yarandıqda) və ya cədvəlin doldurulmasının alqoritmi dəyişdikdə (düsturun dəyişməsi) cədvəlin işlənməsi proqramını dəyişdirmək lazım gəlir, deməli istifadəçi yenə də proqramçının köməyinə müraciət etməlidir.

Prosess sadələşdirmə tələb edirdi. Yeni düşünülmüş elektron cədvəllər proqramçının iştirakını istisna etməli, cədvəlin işlənməsi alqoritminin proqramlaşdırılmasını istifadəçinin özünün yerinə yetirməsinə, alqoritmdə dəyişikliklər olduqda isə onun istifadəçi tərəfindən yenidən proqramlaşdırılmasına imkan verilməli idi.

Elektron cədvəl (*ing.* "spread-sheet" – "cədvəl prosessoru") – cədvəldə saxlanılan verilənlərin istifadəçinin verdiyi düsturlar üzrə avtomatik işlənməsi üçün istifadə olunan proqram və ya proqram sistemidir.

Müəyyən xana, sətir və ya sütunlarda yerləşən verilənlər üzrə hesablamalar aparmaq üçün cədvəlin hər bir xanasına, sətrinə və ya sütununa düstur təyin etmək olar. Hər hansı bir xananın içindəkinin istənilən dəyişikliyi zamanı cədvəl avtomatik olaraq yenidən hesablanır.

İlk elektron cədvəl 1979-cu ildə yaranmışdır. Onu çox uğurlu adlandırmışdılar – "VisiCalc" (*lat.* "visio" – "görmə", "görünmə" və "calculo" – "saymaq", "hesablamaq"). Adından da göründüyü kimi, VisiCalc görünən, əyani hesablamalar aparır.

VisiCalc-dan sonra tezliklə başqa elektron cədvəllər də yaranmağa başladı. Artıq 1981-ci ilin fevralında vaxtilə SSRİ-də populyar olan SuperCalc elektron cədvəli işlənib hazırlandı. 1983-cü ilin əvvəllərində isə indiyədək ən rəqabətədavamlı elektron cədvəllər üçlüyünə daxil olan (Microsoft kompaniyasının Excel və Borland firmasının Quattro proqramları ilə yanaşı) Lotus firmasının "1-2-3" elektron cədvəli meydana gəldi.

N	*a*	*b*	*c*	*x1*	*x2*	K.s.
1	1	2	1	–1.00	–1.00	1
2	1	–2	1	1.00	1.00	1
3	1	–1	3	###.##	###.##	0
4	2	4	1	–1.71	–0.29	2
5	2	8	1	–3.87	–0.13	2
6	2	12	1	–5.92	–0.08	2
7	2	16	1	–7.94	–0.06	2
8	2	20	1	–9.95	–0.05	2

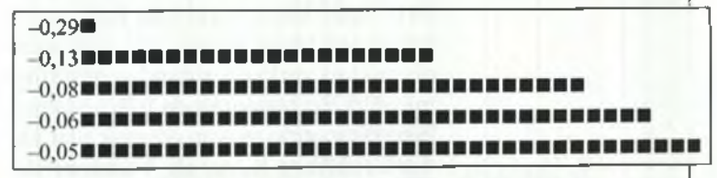
*D = b*b - 4*a*c*
*x1 = (–b – SQRT(D)) / (2*a)*
*x2 = (–b + SQRT(D)) / (2*a)*
Köklərin sayı =
D > 0 olduqda : 2
D = 0 olduqda : 1
əks halda: 0

90-cı illərin əvvəllərində Rusiyada da elektron cədvəllər işlənib hazırlanmışdır. Onlardan biri, "MikroMir" mətn redaktoruna qoyulmuş "MikroKalk", MDU-nun mexanika-riyaziyyat fakültəsində hazırlanmışdır. "MikroKalk" istifadə üçün çox sadə idi və kompüter resurslarına tələbkar deyildi. Bu da proqramı həm biznes, həm də "məişət" hesablamalarında (məs., mənzil kirəsi və ya ev heyvanlarının qidalanma rasionunun hesablanmasında) populyar etmişdi.

ELEKTRON CƏDVƏLLƏRİN PROQRAMLAŞDIRILMASI

Yuxarıdakı cədvəldə $ax^2+bx+c=0$ kvadrat tənliyinin əmsallarına görə ("*a*", "*b*" və "*c*" sütunları) köklərinin hesablanması proqramlaşdırılmışdır. Cədvəl həm kökləri ("*x*1" və "*x*2" sütunları), həm də onların sayını hesablayır.

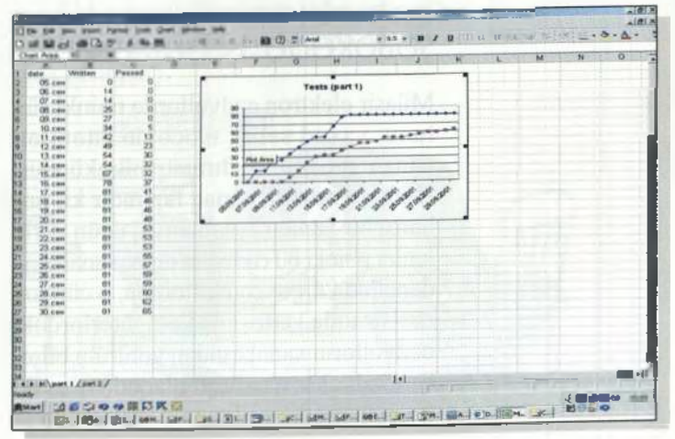
Üçüncü tənliyin kökləri yoxdur, çünki diskriminant (düstürlarda müvəqqəti D dəyişəni) mənfidir. Ona görə də bu halda üçüncü sətrin "*x*1" və "*x*2" xanalarında ədəd əvəzinə "#" işarələri çıxır ki, onlar da hesablamalarda səhvlərin olduğunu göstərir: "kvadrat kök" funksiyası (düstürlarda SQRT kimi göstərilmişdir) mənfi ədədlər üçün təyin olunmamışdır.



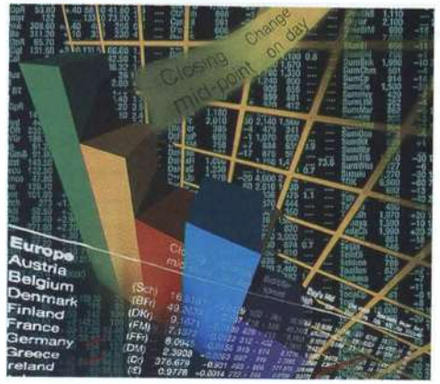
QRAFİK İMKANLAR

Hesablamaların avtomatlaşdırılması informasiyanın elektron formada təqdim edilməsinin yeganə üstünlüyü deyildir. Kompüterə qrafik təsvirlərin qurulmasını da həvalə etmək olar.

Hərf-rəqəmli displeylərin istifadəsi dönəmin ekranda qrafik və *histoqram* (sütunlardan ibarət qrafik) çəkmək üçün yalnız simvollar, məsələn, kiçik kvadrat-



Excel elektron cədvəllər proqramında qrafiklərin qurulması.



lardan istifadə edilirdi. Şəkildə elektron cədvəldə qurulmuş üfüqi histoqram göstərilmişdir. Onun qurulması üçün verilənlər 4-8 tənliklərinin "*x*2" kökünün qiymətləri yerləşən sütundan götürülmüşdür. Bu histoqramda üfüqi sütunların qiymətinə əsasən görmək olar ki, 4-cü tənlikdə b əmsalı 4 addımı ilə 4-dən 20-yə qədər dəyişəndə *x*2 kökü əvvəlcə çox, sonra isə get-gedə az artır.

Qrafiki displeylərin meydana gəlməsilə ekranda ayrı-ayrı nöqtələri seçdirmək və qrafik qurmaq mümkün olmuşdur. Şəkildə iki qrafik elə proqramlaşdırılmışdır ki, onların hər biri cədvəlin iki sütunundan birindəki verilənləri əks etdirir. Cədvələ yeni verilənlərin daxil edilməsi qrafiklərin dərhal dəyişilməsinə gətirib çıxarır.

CƏDVƏLLƏRİN ŞƏBƏKƏDƏ İŞLƏNMƏSİ

Müasir elektron cədvəllərdə nəinki sütunu, yaxud sətiri, eləcə də istənilən xananı proqramlaşdırmaq mümkündür. Həm də nəzərə almaq lazımdır ki, hər hansı bir xananı hesablamaq üçün verilənlər nəinki bu cədvəlin digər xanalarından, hətta digər cədvəllərdən, o cümlədən uzaqda yerləşmiş kompüterlərdəki cədvəllərin xanalarından götürülə bilər.

Cədvəldə qarşılıqlı əlaqədə olan iki qrupun: proqramçıların və testləyicilərin (proqramları sınaqdan çıxaranların) hər hansı bir layihə üzərindəki işlərinin nəticələri verilmişdir. Birinci sütunda birinci qrup tərəfindən hazırlanmış proqramların miqdarı, ikinci sütunda isə bu proqramlardan neçəsinin ikinci qrup tərəfindən uğurla sınaqdan çıxdığı göstərilmişdir.

Göstərilən işçi qruplar bir-birindən uzaq məsafədə, məsələn müxtəlif binalarda və hətta müxtəlif şəhərlərdə yerləşə bilərlər. Elektron cədvəllər imkan verir ki, eyni bir vaxtda cədvəlin eyni bir nüsxəsi doldurulsun. Şəbəkədə cədvəlin doldurulması iki üsulla aparılır.

Birincisi, qruplara cədvəllə işləməyə icazə verilə bilər. Onda həm proqramçılar, həm də testləyicilər öz kompüterlərindən verilən cədvələ girib onlara ayrılmış sütunun xanalarını doldura bilərlər.

İkincisi, cədvəlin xanalarını elə proqramlaşdırmaq olar ki, elektron cədvəl xanaları doldurmaq üçün ədədləri göstərilən yerdən özü oxusun. Bu o deməkdir ki, hər bir işçi qrup öz kompüterlərində işin göstəricilərinin qiymətini hər hansı bir şəkildə (məsələn fayl və ya cədvəl) saxlayacaqlar. Elektron cədvəl isə yekun cədvəli açaraq ona təqdim olunmuş digər kompüterlərə daxil olub xanaları doldurmaq üçün qeyd olunan yerdən verilənləri oxuyacaq.

Müasir elektron cədvəllərdə mətnlərin redaktəsi zamanı təqdim olunan imkanlar mükəmməl mətn prosessorlarının imkanlarına tamamilə yaxınlaşdırılmışdır. Cədvəllərdə nəinki hesablama aparmaq, həm də mətni gözəl yerləşdirmək olar. Bir çox firmalarda qiymət siyahıları, hesablar və buna oxşar digər sənədlər cədvəllərdə yerləşdirilir.

Praktiki olaraq bütün elektoron cədvəllərdə kifayət qədər sadə proqramlaşdırma dili quraşdırılmışdır. Uyğun proqramların yazılışında bu dillərdən istifadə etməklə elektron cədvəllərdən sadə verilənlər bazasına qədər istənilən ofis proqram məhsulunu almaq olur.

VERİLƏNLƏR BAZASINI İDARƏETMƏ SİSTEMLƏRİ

Öz qohumlarına və dostlarına qarşı diqqətsizlik göstərən adamlara tez-tez rast gəlinir: bayramlarada təbrik etməyi yaddan çıxarırlar, ad günlərini unudurlar... Belələrinə lazımdır ki, ad, soyad, telefon nömrələrindən başqa yubileylər, ad günləri və s. mühüm tarixlər də qeyd olunan yazı kitabçası tutsunlar. Baxmayaraq ki, heç bu da problemin həllinə kömək etməyəcək.

Adətən, yazı kitabçalarında 30 səhifə kəsilmiş olur və həmin kəsiklərdən Azərbaycan əlifbasının hərflərinin birindən təşkil olunmuş "adlar" görünür. Onların ardıcıllığı yazı kitabçasının "mündəricatıdır". Məsələn, "B" səhifəsində aşağıdakıları yazaq:

Babayev A.	766 66 64
Balaş (bərbər)	766 56 55
Bağırlı O. (11 may)	756 77 66
Botanika bağı (bağban Həsən)	733 45 54

və s.

Soyadı "B" hərfilə başlayan dostu tapmaq üçün bütün səhifəyə baxmaq lazımdır. Amma bağban Həsəni tapmaq bir az çətindir: onun adı ilk tanışlıq zamanı Botanika bağının telefonu ilə birgə yazılıb. Bir neçə vaxt keçdikdən sonra Həsənin adını "H" hərfi yazılmış səhifədə axtarmaq istəyəcəksiniz. "B" səhifəsindən "H" səhifəsinə Həsən (bağban) 733 45 54 qeydinin təkrar yazılması hansısa anda sizdə əsəbilik yarada bilər.

Əgər Həsən telefon nömrəsini dəyişsə, onda iki səhifədə düzəliş aparmaq lazım olacaq. Ən çətini isə bağban Həsən haqqında olan məlumatın iki yerdə olduğunu yadda saxlamaqdır.

Bağırlı Orxanı ad günü münasibətilə təbrik etmək yenə də yaddan çıxa bilər. Yatmamışdan qabaq bütün yazı kitabçasına baxıb sabah dost-tanışlardan kiminsə ad gününün olduğunu müəyyənləşdirməyə çox adam vərdiş etməyib.

Bu problemi necə həll etməli? Gündəlik (hər bir səhifəsi ilin bir gününə uyğun 366 səhifəlik kitabça) götürüb hamını, istədiyin məşhur adamlar da daxil olmaqla ad gününə uyğun səhifəyə köçürmək olar. Və onda: "Orxan, ad günü münasibətilə bütün görkəmli adamları: səni, Şah İsmayıl Xətaini, Rual Amudseni, Robert Şeklini təbrik edirəm" (bu sırada kimin olmağının fərqi yoxdur, əsas odur ki, eyni gündə doğulmuş olsunlar).



"TƏRƏZİ" (VƏ/VƏ YA) ALQORİTMİ BARƏDƏ

Alqoritm iki nizamlı çoxluğun "birləşməsi" əməliyyatını effektiv həyata keçirməyə imkan verir.

1. Bir çoxluğun növbəti elementini ikinci çoxluğun (birinci elementlərdən başlayaraq) növbəti elementilə müqayisə edirik.

2. Elementlər bərabər deyillər – kiçik elementi yekun çoxluğa yazırıq; elementi götürülmüş çoxluqdan müqayisə üçün növbəti element seçilir (digər çoxluqda element dəyişilmir).

Əks halda (yəni bərabərdirlər) – çoxluqların birindən elementi yekun çoxluğa yazırıq və müqayisə üçün hər iki çoxluqdan növbəti elementi seçirik.

3. Əgər hər iki çoxluğun bütün elementlərinə baxılmayıbsa, onda 1-ci addıma keçirik.

Analoji alqoritm iki nizamlı çoxluğun "kəsişməsi" əməliyyatını aparmağa imkan verir.





Əllə bunu etmək çətin olduğundan, yaxşı olar ki, kompüterdən istifadə edilsin.

Kompüter yazı kitabçasını hazırlayanda hansı problemlər çıxır?

Hər şeydən əvvəl adın və ya soyadın ilk hərfi üzrə adamı axtaranda səhifəyə ardıcıl baxışın yorucu olması. Əgər "adlar" iki, üç hərfli olsa, onda siyahı qısa olacaq və proses tez gedəcək. Hər adama bir səhifə ayrılarsa, axtarış daha da sürətlənəcək.

Bundan başqa, bütün məlumatlara tam baxmadan istənilən mövcud, yadda saxlanılmış informasiyanı tapmaq olmaz, çünki ani olaraq ancaq yaradılan "mündəricat" tapılır. Məsələn, B hərfinə uyğun "Botanika bağı" axtarılanda "bağban Həsən"in də bu səhifədə olması heç də məlum olmur.

Daha bir problem təhlükəli təkrarlamaların yaranmasıdır ki (bir yazı kitabçasında və ya başqa "mündəricatları" olan ikinci, üçüncü kitabçalar tutulanda), bu da bir sıra səhvlərə gətirib çıxara bilər: bir yerdə düzəldilib, digər yerdə isə unudulub.

Belə məsələləri həll etmək məqsədilə *verilənlər bazası* düşünülüb. Dar mənada – bu, xüsusi şəkildə təşkil olunmuş informasiya saxlanılan yerdir ki (verilənlər bazasının *faylları*), burada nəinki lazım olan məlumatları tez tapmaq olar, hətta düzəlişlər aparmaq və onları uzaqlaşdırmaq da olar. *Verilənlər Bazasının İdarəolunması Sistemi* (VBİS) – sadaladığımız əməliyyatları yerinə yetirməyə imkan verən proqramlar toplusudur. Geniş mənada verilənlər bazası – faylların və proqramların birləşdirilmiş adıdır.

Verilənlər bazasının hər bir faylında informasiya *yazılar* adlanan kəsilməz porsiyalarla saxlanılır. Yazı öz növbəsində *sahələrdən* ibarətdir. Sahələr ancaq rəqəmlərdən ibarət *ədədi*, istənilən işarədən (hərf, rəqəm, durğu işarələri) ibarət olan *simvollu*, müəyyən formatı olan (tarix, pul, telefon nömrəsi və s.) *xüsusi* sahələrdən ibarətdir. Bir bazanın bütün yazılarında sahələr toplusu eynidir və bazanın yaradılmasına qədər müəyyən olunur (bu mürəkkəb prosess *layihələndirməyə* aid edilir).

Şəxsi telefon soraqçası üçün verilənlər bazasını layihələndirdikdə soraqçanı təşkil edən hər bir yazıya şəxsin adını, soyadını, telefon nömrəsini, doğulduğu günü və ayı daxil etmək lazımdır. Əgər bir ildən sonra ata adını, iş yerini və doğulduğu ili də nəzərə almaq lazımdırsa, onda əlavə sahələr ayırmaq olar, bu zaman mövcud yazılarda artıq bu sahələrin qiymətləri boş qalacaq.

Hər bir yazının faylda yerləşmə mövqeyindən asılı olaraq unikal nömrəsi olur. Bu isə yazıları ixtiyari ardıcıllıqla saxlamağa, uzaqlaşdırmağa və boşalan yerlərə yenilərini daxil etməyə imkan verir. Verlənlər faylından yazını oxumaq üçün onun nömrəsi yetərli informasiyadır. Hərçənd ki, axtarılan yazını yalnız bütün yazılara ardıcıl baxmaqla tapmaq olar.

İndeks adlandırılan "mündəricatlar" mövcud olduqda sürətli axtarış həyata keçirilir. Sahə üçün indeksin olub-olması verilənlər bazası layihələndirilərkən müəyyənləşdirilir. Məsələn, əgər tanışları ada görə axtarmaq nəzərdə tutulmursa, "Ad" sahəsi üçün indeks yaratmaq mənasızdır.

İndeks sahənin qiyməti və yazının unikal nömrəsi cütlüyündən ibarət olur. Cütlükdə birinci hissə *açar*, ikinci isə *istinad* adlanır.

Soyad	
Eminbəyli	**172**
Əliyev (Açar)	150
Bağırov	160
Eminbəyli	**134**
Qurbanova	122 (İstinad)
Zeynallı	099

Eminbəyli soyadlı adamı axtarmaq üçün "Soyad" indeksinin açarlarının siyahısına baxmaq, verilənlər faylında yazıların unikal nömrələrini (172, 134) tapmaq və onlara görə iki adam haqqında yazıları oxumaq kifayətdir.

Əgər verilənlər bazasında 4000 adam barəsində informasiya saxlanılırsa, onda hər bir axtarışda indeksdəki bütün açarlara, yəni 4000 yazıya baxmaq tələb oluna bilər. Bu, verilənlər faylında eyni miqdarda yazıya baxmaqdan daha tez həyata keçirilir.

Axtarışı sürətləndirmək üçün indeksdə (*indekslər faylında*) açar – istinad cütlüyünü açarların artan sırasında saxlamaq və təkrarlanan açarları açar-istinadlar siyahısı cütlüyü ilə əvəz etmək lazımdır. Bu zaman istinadlar siyahısı da artan qaydada yerləşdirilməlidir.

Soyad	
...	
Bağırov	160
...	
Əliyev	150
...	
Zeynallı	099
Eminbəyli	**134 172**
...	
Qurbanova	122
...	

Əgər axtarış üçün heç bir xüsusi alqoritm tətbiq olunmasa belə, onun vaxtı orta hesabla iki dəfə azalacaqdır.

Yox, əgər məşhur "*yarıya bölmə*" (*binar axtarış*) alqoritmindən istifadə olunarsa, 4000 adam arasında axtarılanı tapmaq üçün 12-dən çox olmayan yazı oxumaq lazım gələcək ($11<\log_2 4000<12$), yəni sürət 340 dəfə artacaq!

İndeksdə açarların artan sıra ilə saxlanması axtarışı göstərilən qiymətdən böyük və ya kiçik qiymət diapazonunda asanlıqla həyata keçirməyə imkan verir.

Fərz edək ki, "Gün" və "Ay" sahələrindən alınmış "Ad günü" indeksi aşağıdakı kimidir:

Ad günü	
...	
0228 (21 mart)	244
0321	056 084 142
0329	**135**
0331	**044 179**
0402 (19 aprel)	**127 278 299 305 432**
0419	021 093
0423	112

"25 martdan 5 aprelə qədər kimin ad günüdür?" sualına cavab üçün "Ad günü" indeksində 0325-dən 0405-ə kimi bütün açarları tapmaq lazımdır. Bir başa axtarışla VBİS 0329 qiymətli açarını (başlanğıc qiymətə nəzərən ən yaxın böyük açarı) tapacaq, sonra isə 0419 (sonuncu qiymətə ən yaxın böyük açarı) açarına çatana kimi, təkrarlamaya yol verməmək üçün, bütün açarları ardıcıl oxuyaraq, istinadlar siyahısı üzərində "birləşdirmə" aparmaqla istinadları toplayacaqdır. Nəticədə istinadlar siyahısı: 044 127 135 179 278 299 305 432 alınacaqdır ki, bunun da köməyilə verilənlər faylından tələb olunan yazını oxumaq olar.

Axtarışın başqa variantı isə – eyni zamanda bir neçə suallı cavablandıran bir neçə indeks üzrə yazılar tapılır. Məs., "Mart ayında anadan olmuş Qurbanovlar". Bunun üçün "Soyad" indeksində "Qurb"-la başlayan açarlar, "Ad günü" indeksində isə 0301-dən 0331-ə kimi açarlar tapılır.





Onda VBİS birinci indeksə görə aşağıdakı istinadlar siyahısını: 044 135 172, ikinci indeksə görə isə: 044 056 084 135 142 179; sonra isə siyahılar arasında "Kəsişmə" əməliyyatı aparmaqla nəticədə 044 135 cavabını tapacaq.

Verilənlər bazası faylına yazıların artırılması/uzaqlaşdırılması indekslərin dəyişməsilə müşayət olunur. Əgər verilənlər faylında, məsələn, 122 saylı "Qasımov, 4 mayda doğulmuşdur" yazısı daxil edilsə, indeksi olan bütün sahələr üzrə ardıcıllığı gözləməklə açar-istinad cütlüyü yaradılacaqdır:

Soyad	
...	
Bağırov	160
...	
Əliyev	150
...	
Zeynallı	099
Eminbəyli	134 172
...	
Qarayev	110 117
Qurbanova	120
Qasımov	**122**
...	

Ad günü	
...	
0228	244
0321	056 084 142
0329	135
0331	044 179
0402	127 278 299 305 432
0419	021 093
0423	112
0405	**122**
...	

Əgər açar artıq indeksdə vardırsa, onda ancaq istinadlar siyahısına istinad əlavə olunacaq. Artırmada olduğu kimi, açar-istinad cütlüyü uyğun indekslərdən uzaqlaşdırılır. Əgər açara bir istinad uyğundursa, onda cütlük tam silinəcəkdir, əks halda yalnız istinadlar siyahısından istinad silinəcəkdir. Məs.: 115 saylı yazı, "Qasımova", 19 apreldə doğulmuşdur:

Soyad	
...	
Bağırov	160
...	
Əliyev	150
...	
Zeynallı	099
Eminbəyli	134 172
...	
Qarayev	110 113
Qasımova	118 119
Qurbanova	120
Qarayev	122
...	

Burada 115 saylı yazıya istinad olunmuşdur

Ad günü	
...	
0228	244
0321	056 084 142
0329	135
0331	044 179
0402	127 278 299 305 432
0419	118 119
0423	112
0405	122
...	

Və burada 115 saylı yazıya istinad olunmuşdur

Ancaq bundan sonra verilənlər faylından yazı uzaqlaşdırılır.

İndekslər faylında dəyişikliklər prosesi verilənlər faylına yazıların əlavə olunmasına nisbətən daha çox vaxt aparır.

Bunu, Qarayev soyadı əvəzinə səhvən daxil edilmiş Qaraxanov soyadı misalında göstərək. Əgər VBİS hər indeksi ayrı fayla yazırsa, onda verilənlər faylında bir sahənin dəyişməsi (Qaraxanov soyadının Qarayevə dəyişdirilməsi) "Soyad" indekslər faylında dəyişikliyə səbəb olacaq.



Soyad	
...	
Qarayev	**110 113**
Qasımova	118 119
Qurbanova	120
Qasımov	122
Qaraxanov	132
...	

Yeni cütlük əlavə etmək və mövcud olan cütlükdə dəyişiklik aparmaq lazım gələcəkdir. Adətən, indekslər faylının həcmi verilənlər faylının həcmindən çox böyük olur.

Nizamlanmış fayllarda qonşu yazılar qrupu yaratmağa əsaslanan başqa axtarış üsulu da mövcuddur. Belə qruplarda birinci yazı faylın ayrı-ayrı hissələrinə "mündəricat" kimi göstərici rolu oynayan cədvəldə saxlanılır. "Qaraxanov" yazısını tapmaq üçün (Q hərfi üzrə axtarış) indeks faylının lazımlı hissəsinə düşmək, sonra isə binar axtarışdan istifadə etmək olar:

Soyad	
...	
Kərimov	970
Qasımov	110 113
...	
Qaraxanov	**132**
...	

"Q" qrupu

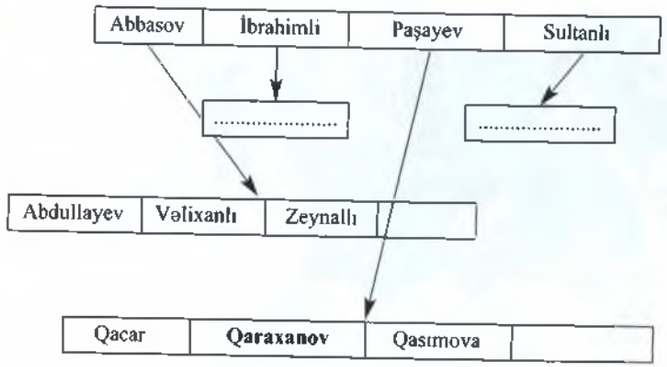
Binar axtarışın sürəti nə qədər yaxşı olsa da, indekslər faylında cütlüklərin əlavə olunması və uzaqlaşdırılması uyğun göstəriciləri vermir, çünki, ardıcıllığı gözləmək tələb olunur və buna da kompüterin kifayət qədər vaxt resursu sərf olunur. İndekslərin B-ağac şəklində qeyri-xətti saxlanılması formasından istifadə olunması daha məqsədəuyğundur.

B-ağac başı üstə çevrilmiş ağac şəklindədir: yarpaqlar aşağıda, yeganə kök isə yuxarıda yerləşir. Yarpaqlarla kök arasındakı budaqlanma aralıqlarını ağacın *düyünləri* adlandırırlar. Yarpaqlar aşağı səviyyəli düyünlər, kök isə yuxarı səviyyəli düyündür. B-ağacın düyünlərində açarlar yerləşir, məsələn, "Soyad".

Bu misalda B-ağac iki səviyyəli alınmışdır. Yuxarı səviyyə bölünmə nöqtələri olan açarlardan ibarətdir. Bu düyündə hər bir açar bir səviyyə aşağıda yerləşən düyünü göstərir. "Abbasov" açarı verilənlər faylına istinad olmaqla, həm də "Abbasov"dan "İbrahimli"yə kimi bütün açarların sonrakı səviyyədə olduğunu göstərir.

"Qaraxanov" yazısını ağacın kökündə axtaran zaman əvvəlcə görəcəyik ki, "Paşayev" açarı B-ağacın növbəti səviyyəsinui göstərir və "Qaraxanov" açarını ağacın düyünlərində axtarmaq lazımdır (belə ki, "Qaraxanov" əlifbaya görə "Paşayev"lə "Sultanlı" arasında yerləşir). Sonra isə növbəti düyündə verilənlər faylında yazıya istinad olan "Qaraxanov" açarı tapılacaq.

B-ağaca elementlərin əlavə olunması və uzaqlaşdırılması sadə və mürəkkəb məsələ ola bilər. Məsələn, yeni "Ziyadov" açarını əlavə etmək asandır,



çünki, "Zeynallı" açarının yanında boş yer var. Ancaq "Qarayev" açarını əlavə etmək üçün bu düyündə yer olmadığından düyünlərin yeni bir səviyyəsini

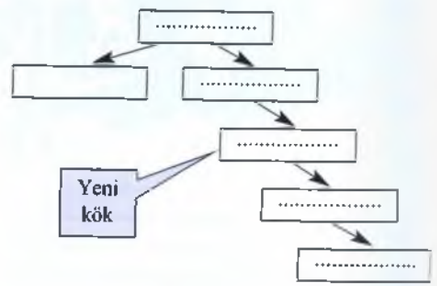


yaratmaq lazımdır. Düyünü yuxarı və aşağı düyünlərə bölmək lazım gələcək:

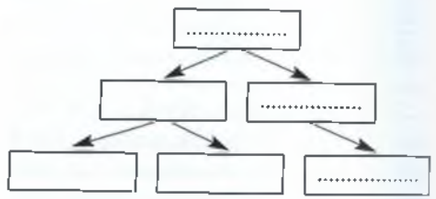


B-ağacı balanslaşdırılmış vəziyyətdə saxlamaq lazımdır ki, onun "budaqları" bərabər "böyüsünlər". Belə olmasa bəzi hallarda axtarış çox vaxt tələb edəcəkdir.

Təsvir olunmuş şəkildə "yeni kökü" (şəkildə qeyd olunmuş düyün) "tutub çəksək",



onda ağacın sol tərəfi yüngülcə dəyişəcək:



Belə ağacda axtarılan açarın istənilən qiymətlərində axtarış daha səmərəli olacaq. Balanslaşdırılmamış ağacları balanslaşdırmağa imkan verən xüsusi alqoritmlər mövcuddur.



DOĞRAMA

Çoxluq tipli verilənlər strukturunun doğrama-realizasiyasında olduğu kimi verilənlər faylında axtarış zamanı *dəhrələmədən* istifadə daha münasibdir. Hər yazıya, hər hansı bir sadə prosedurun köməyilə effektiv şəkildə hesablana bilən bir ədəd qarşı qoyulur. Soyadları indekslərinin qurulması üçün əlifbanın bütün hərflərini nömrələmək, yəni hər hərfə 1-dən 32-ə kimi bir ədəd qarşı qoymaq kifayətdir:

A-1; B-2; ... Z-32

Bundan sonra soyadına hərflərin (1-dən 32-dək) kodlarının ardıcıllığı kimi baxılır. Məsələn, Babayev soyadı cədvəldə belə yazılır:

2	1	2	1	31	6	30

"uzunluq" isə 7-yə – hərflərin sayına bərabər olacaq.

```
alq tam indeks (tamcəd soyad [1:20], tam uzunluq)
verilib
gərəkdir
baş tam K
|   qiym:=0
|   db K üçün 1 dən uzunluq dək
|   |   qiym := qiym * 32 + soyad [K]
|   ds
son
```

Alınan qiyməti – unikal ədədi yazıya istinad hesab etmək olar. Amma belə ədəd çox böyük olacaq. Babayev soyadına uyğun misalda indeks təxminən 10^{10} olacaq! Belə yanaşma yaddaşın həddən artıq böyük həcmdə olmasını tələb edir. Bu isə səmərəli deyil, çünki hərflərin kombinasiyalarının heç də hamısı soyadı əmələ gətirmir. Adətən, bu problemi həll etmək üçün psevdounikal ədədin hesablanması alqoritmindən istifadə edilir və alqoritm *doğrama-funksiyası* adlandırılır. Soyadı üçün ayrılmış (*doğrama-cədvəl*) cədvəlin ölçüsünün modulu üzrə hesablanmış soyadındakı hərflərin kodlarının cəmini saymaq asandır:



```
alq tam doğrama-funksiyası (tamcəd soyad [1:20], tam uzunluq, N)
verilib
gərəkdir
baş tam K
|   qiym:=0
|   db K üçün 1 dən uzunluq dək
|   |   qiym:=mod (qiym+soyad [K], N)
|   ds
son
```

(*mod (x, y)* funksiyası *x* və *y* tam ədədlərinin bölünməsindən alınan qalığı qaytarır və onu həm də modul adlandırır və *x mod y* kimi yazılır). Müxtəlif açarların tez-tez üst-üstə düşməməsi üçün (axı bu ədədlər unikal yox, psevdounikal hesab olunurlar) doğrama-cədvəlin dörddə birisini boş saxlamaq lazımdır. Buna baxmayaraq, əgər üst-üstə düşmə – *kolliziya* baş verərsə, onda həmin cədvəldə ardıcıl seçmə yolu ilə müvəqqəti boş yer tapmaq olar. Daha effektiv olması üçün birinci dəhrələmə-funksiyasına analoji olaraq ikinci dəhrələmə-funksiyasını seçib, ikinci dəhrələmə aparmaq məqsədəuyğundur:

```
alq tam ikinci dəhrələmə-funksiyası (tamcəd soyad [1:20], tam uzunluq, N, M, addım)
verilib
gərəkdir
baş tam K, i
|   qiym:=0
|   db K üçün 1 dən uzunluq dək
|   |   qiym := mod (qiym + soyad [K], M)
|   ds
|   qiym := mod (qiym * addım + dəhrələmə-funksiyası
|     (soyad, uzunluq, N), N)
son
```

M ədədi *N* ədədi ilə qarşılıqlı sadə ədəd kimi seçilir. Adətən, *M* = *N*-2. Əgər doğrama zamanı kolliziya baş verərsə, 1-ə bərabər "addım"la ikinci doğrama-funksiyasının qiyməti hesablanır. Əgər kolliziya yenə də təkrar olunarsa "addım" 2-yə bərabər götürülür və s. İdeya ondan ibarətdir ki, "modula görə" kolliziyadan qaçmaq mümkün olsun. Bunun üçün ardıcıl indekslər kimi dəhrələmə+ikinci dəhrələmə, doğrama+2 ikinci doğrama, doğrama+3 ikinci doğrama, ... indekslərinin ardıcıllığı götürülür. Doğrama zamanı əlifbaya görə bir-birinə yaxın olan soyadlar cədvəl boyu ixtiyari şəkildə yerləşdirilə bilər. Ümumiyyətlə desək, onlar əlifbaya görə düzülməli deyillər. "Qaraxanov" yazısının axtarışı nəticəsində ona yaxın soyadlar – Qarayev və ya Q haqqında sorğulara cavab almaq mümkün deyil.

RELYASİON VERİLƏNLƏR BAZASI

Əlaqəli verilənlərin kiçik qruplarını adətən yazı adlandırırlar. Məsələn, "Məhsul-qiymət" verilənlər faylında məhsulun adından ibarət qrupu yazı əmələ gətirir. Yazının elementi sahə adlanır.

Yazılar, elementlər və açarlar arasındakı əlaqələrin xarakterini verilənlərin təşkilinin üç əsas tipi müəyyənləşdirir. Faktiki olaraq kompüterlərin meydana gəlməsilə verilənlər bazasında iyerarxiyalı *modeldən* istifadə olunurdu.

Bunu ərzaq mağazasının misalında izah etmək olar. Birinci səviyyəyə əsas element kimi ərzaqların kateqoriyası çıxış edən şöbələr cədvəli daxil edilir. Lazımı kateqoriyanı, məsələn, "süd məhsulları"nı seçməklə sonrakı ikinci səviyyəli cədvələ keçmək olar. İkinci səviyyəyə detallaşdırma olmadan şöbələr üzrə ərzaqların adlarını sadalayan cədvəllər aiddir. Məs., "Süd" 1№-li şöbəyə – "Süd məhsulları" şöbəsinə aiddir, ancaq konkret ad və qiymət iyerarxiyanın sonrakı səviyyəsindəki cədvəldə saxlanılır. Verilənlər bazasının belə strukturu satıcıya rahatdır, lakin alıcı üçün yaramır, çünki onu ilk növbədə ərzaqların qiyməti maraqlandırır. Məsələn, "Pınar" 1,5% südü neçəyədir?

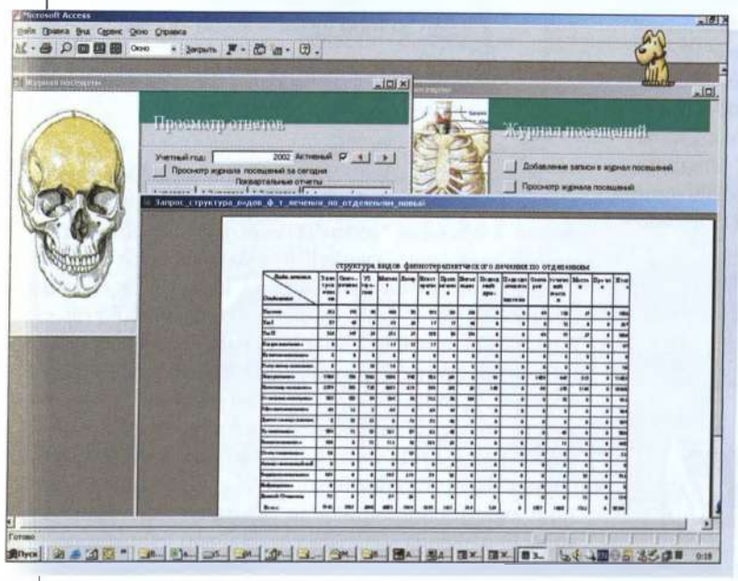
Şöbə	Məhsulların kateqoriyası
1	Süd məhsulları
2	Ət məhsulları
3	Baqqaliyyə malları
4	Balıq məhsulları
5	Çörək-bulka məmulatları

Ət məhsulları	Ət Yarımfabrikatlar Quşlar ...

Süd məhsulları	Süd Xama Kəsmik Pendir Kefir Yağ ...

Məhsul	Qiymət
"Pınar" 0,5 % "Pınar" 1,5%	1 man. 60 qəp. 1 man. 80 qəp.
"Sevimli dad" 3,2%	1 man. 50 qəp.
...	...

Verilənlər bazasının ekranından bir şəkil.



sualına verilənlər bazası ancaq bütün iyerarxiya səviyyələrini keçməklə "cavab verə" bilər. Nə qədər ki, verilənlər bazasının iyerarxiyalı strukturunu yan keçməklə birinci və ikinci səviyyələr arasında göstəricilər əlavə olunmayıb, ərzağın qiyməti haqqında suala cavab almaq çətindir (əməliyyatın yerinə yetirilmə müddəti baxımından).

Verilənlər bazasının müəyyən suallara tez cavab verə bilməsinə şərait yaradan verilənlərin təşkilinin sadəliyi iyerarxiyalı verilənlər bazalarına imkan verdi ki, "aşağı sürətli" EHM-lərdə geniş yayılsınlar. Bundan başqa, verilənlər bazasının layihələndirilməsi mərhələsində onun hansı suallara effektsiz cavab verəcəyini əvvəlcədən nəzərə almaq mümkün deyil.

BİR GİQABAYT NƏ DEMƏKDİR

Bir giqabaytda 230 bayt var və yaxud müasir kitablarda yazıldığı kimi (nəzərə alsaq ki, 1 kbayt 1024 yox, 1000 baytdır) bu təqribən milyard bayt yaddaş deməkdir. Belə yaddaşda aşağdakıları saxlamaq mümkündür:

- 500 min səhifəlik mətn, yəni təqribən 1 min roman – bütöv bir kitabxana;
- təqribən 1 min ən yüksək keyfiyyətli rəngli slayd;
- 5000-ə qədər 1280 rastrli rəngli rəqəmsal fotoşəkillər (JPEG formatında 1024);
- 100 saatdan çox danışığın audioyazısı (bu yazı keyfiyyətcə telefon danışığına bərabərdir);
- 10 saatdan çox MP 3 formatlı musiqi yazısı. Bu keyfiyyətinə görə kompakt disk yazısından fərqlənmir (bu texnologiya FM radioyayımlarında istifadə edilir);
- həmin MP 3 formatında müddəti 20 saatdan çox musiqi yazısı. Düzdür, həcm artdıqca mahnının keyfiyyəti pisləşir və bu yaxşı kassetli maqintafonun yazı keyfiyyətinə uyğundur.
- CD-keyfiyyətli ən azı 2 saatlıq musiqi fraqmenti;
- Sıxma alqoritmindən istifadə olunmadan, təqribən 15 saniyəlik yüksək keyfiyyətli rəngli film;
- Görüntünün və səsin MPEG sıxılma alqoritmindən istifadə edilməklə iki saatdan çox videoyazı (yazılma keyfiyyəti televiziya verilişlərinin yazılma keyfiyyətindən azca aşağıdır);
- nəhəng firmanın bank hesabı ilə 5 min ildən çox aparılmış əməliyyatların protokolları.

1 Gbayt informasiya saxlayan qurğunun qiyməti halhazırda 1 ABŞ dollarından çox deyil (müqayisə üçün: 3 CD ROM-da 2 Gbayt informasiya yerləşir). Kompüter şəbəkəsində 1 Gbayt informasiyanın 1 il ərzində saxlanması cəmi bir neçə ABŞ dolları təşkil edir. Gözləmək olar ki, 10-15 ildən sonra bütün deyilənlər doğru olacaq və giqabaytların yerini terabaytlar tutacaq (1Tbayt=2^{40} bayt).



Şəbəkə modeli iyerarxiyalı modeldən daha çevik olması ilə fərqlənir. Bu modeldə lazım olan fayla dərhal müraciət etməyə imkan verən fayıllararası çoxqat əlaqələr mümkündür. Mağaza misalında birinci və üçüncü səviyyələr arasında istinad yaratmaq mümkündür. Lakin bunun üçün çoxlu xüsusi proqramçı fəndlərindən (verilənlər bazasını əsaslı sürətdə mürəkkəbləşdirən xırda fəndlərdən) istifadə etmək tələb olunur, çünki baza yaradılarkən onun bütün mürəkkəb strukturunu (səviyyələrarası əlaqələrlə birlikdə) ətraflı şəkildə təsəvvür etmək lazımdır. "Şəbəkə bazası verilənləri itirməyin ən doğru yoludur" ifadəsi verilənlərin şəbəkə bazasını çox mürəkkəb struktur kimi xarakterizə edir.

Hesab olunur ki, *relyasion verilənlər bazasının strukturu* sorğulara münasibətdə daha böyük çevikliyə malikdir. Burada elementlərin iyerarxiyası yoxdur, verilənlər bazasında informasiyanın axtarışı zamanı hər bir element *açar* kimi istifadə olunur. *Yazı* dedikdə, ayrı-ayrı elementlərin məcmusu (onlardan biri baş element sayılır) deyil, cədvəlin sətri başa düşülür.

Bizim misalda artıq cədvəl belə görünür:

Məhsul	Şöbə	Məhsulların kateqoriyası	Qiyməti	Ölçü vahidi
"Pınar" 0,5 %	1	Süd məhsulları	1 m. 60q.	1 litrlik paket
"Pınar" 1,5%	1	Süd məhsulları	1 m. 80q.	1 litrlik paket
"Sevimli dad" 3,2%	1	Süd məhsulları	1 m. 50q.	1 litrlik paket
Kambala	4	Balıq məhsulları	4 man.	1 kq
Nərə	4	Balıq məhsulları	10 man.	0,5 kq-lıq qab

"Pınar" 0,5 % süd məhsulları 1 man. 60 qəp. 1 litrlik paket yazısını "Məhsul", "Şöbə" , "Məhsulların kateqoriyası", "Qiymət" və "Ölçü vahidi" elementləri arasında əlaqə kimi başa düşmək olar. Sonralar bu əlaqə "Tədarükçünün adı", "Saxlanma müddəti", "Anbardakı miqdarı" və s. kimi elementlərlə artırıla bilər.

Relyasion verilənlər bazasının cədvəllər yığını olması haqqında təsəvvür dəqiq deyildir (burada ikinci dərəcəli indekslər nəzərə alınmır). Real verilənlər bazalarında yazıların sayı həddindən artıq çoxdur, buna görə də bütün cədvəli bir vərəqdə çap etmək mümkün deyildir. Verilənlər bazası ilə kompüterdə iş, cədvəllərlə kağız üzərində işdən fərqlidir və daha çox kartoteka ilə işi xatırladır: kağız kartlarda olduğu kimi, yazıların bazada yerlərini dəyişmək, bazaya əlavə etmək və bazadan uzaqlaşdırmaq asandır.



Edqar Frenk Kodd.

Cədvəldəki əlaqələrin tam yığınından yeni sadələşdirilmiş əlaqələr cədvəlini (süd şöbəsindəki məhsullar haqqında) asanlıqla qurmaq olar:

"Pınar" 0,5 %	1 man. 60 qəp.
"Pınar" 1,5%	1 man. 80 qəp.
"Sevimli dad" 3,2%	1 man. 50 qəp.

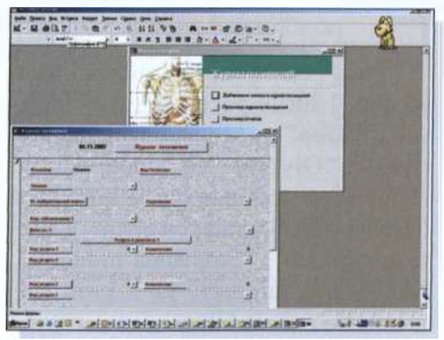
Və ya məhsulun qiyməti haqqında məlumat vermək olar:

Kambala	4 man.	1 kq

Verilənlər bazasının təşkilinin relyasion prinsipi sorğuların xarakteri haqqında əvvəlcədən heç nə demək mümkün olmadığı halda daha münasibdir.

Sadə hallarda (telefon sorğu kitabçasının reallaşdırılması və ya ərzaq mağazası misalındakı kimi) verilənləri təsvir etmək üçün bir cədvəl kifayətdir, daha mürəkkəb hallarda isə (elektron depozitarıdə-müxtəlif müəssisələrin səhmdarları haqqında informasiya saxlanılan yer) öz aralarında istinadla bağlı olan bir neçə cədvəl tələb oluna bilər.

Fərdi kompüterlərdə daha geniş istifadə olunan relyasion verilənlər bazası ilk dəfə IBM firmasının əməkdaşı E.F.Kodd tərəfindən 1970-ci ildə təklif edilmişdir. Riyaziyyatçı bu model prinsiplərini çoxluqlar nəzəriyyəsinə əsaslanaraq işləyib hazırlamışdır.



Verilənlər bazası ekranından bir şəkil.

Bu prinsiplər ümumi şəkildə aşağıdakılardan ibarətdir:

• Verilənlər *münasibətlər* adlanan sətirlər şəklində nizamlanmış yığındır;

• Sətirlər bir-birindən heç olmazsa bir sahənin qiymətinə görə fərqlənirlər.

• Əməliyyatlar tam şəkildə münasibət üzərində yerinə yetirilir, nəticə də münasibətdir. Bu prinsip qapanma prinsipi adlanır.

İnformasiya axtarış sistemlərini işləyib hazırlayanlar, adətən, informasiyanın axtarışı, əlavə edilməsi və ya dəyişilməsi zamanı bazada saxlanılmış verilənlərin ekranda necə görünəcəyinə çox diqqət yetirirlər. Kompüterin ekranında verilənlərin göstərilmə formatı ekran forması adlanır. Hətta bir verilənlər bazası ilə işlədikdə bir neçə ekran forması tələb oluna bilər. Məsələn, ərzaq mağazasının verilənlər bazasına dəyişikliklər edərkən dəyişiklikləri yalnız bir sətirdə, lakin genişləndirilmiş formatda görmək, şöbənin məhsullarının siyahısını nəzərdən keçirərkən isə hər bir mal haqqında qısa informasiya olan bir neçə sətri birdən görmək məqsədəuyğundur.

Ekran formasını verilənlər bazasının hər hansı bir hissəsinə baxmaq üçün insanın istifadə etdiyi bir pəncərə kimi təsəvvür etmək olar. Bu pəncərəyə "rəngli şüşələr" qoyula bilər: bəzi sahələr ekranda bir qədər dəyişilmiş şəkildə təsvir edilir (verilənlər bazasında saxlanıldığı kimi yox, insanın görmək istədiyi şəkildə).

Yazıları yalnız onların daxil edildiyi ardıcıllıqda göstərən verilənlər bazasının idarə olunması sistemi praktiki olaraq səmərəsizdir. Ona görə də, bütün VBİS-lər yazılara baxmaq üçün onları sahmana salmağa imkan verir. Mətn sahələri kimi, ədədi sahələri də müqayisə

etmək olar. Telefon çitabçasında yazıları əlifba sırası ilə yerləşdirmək daha təbiidir: iki sözdən, birinci hərfi "kiçik" olan söz əvvəlcə gəlir (yəni əlifba üzrə əvvəl gəlir). Tarixləri müqayisə etmək lap asandır – daha əvvəlki vaxta aid olan tarix o birisindən "kiçik" sayılır.

Relyasion verilənlər bazasında yazıların düzülüş qaydasını vermək üçün açar sahələr tətbiq edilir (adətən, yazının birinci sahəsi və yaxud birinci gələn bir neçə sahəsi). İstifadəçi baxımından yazılar elə ardıcıllıqla yerləşirlər ki, açar sahələrin tərkibi bazanın əvvəlindən axırına doğru artır.

Əgər açar sahələr bir neçədirsə, onda yazıların düzülüşünü müəyyən edən zaman mətnlərin leksikoqrafik müqayisəsində tətbiq edilən metoddan istifadə olunur. Əvvəlcə yazılar birinci açar sahənin tərkibinə görə nizamlanır; birinci açar sahənin tərkibinə görə eyni olan yazılar ikinci açar sahəyə görə yerləşdirilirlər və s.

Lazım olan informasiyanı tez tapmaq və ona ekranda baxmaq – verilənlər bazasının əsas üstünlüyüdür, lakin işlə əlaqədar olaraq verilənlər bazasının vəziyyəti barədə hesabatı da çap etmək lazım gəlir. Ekranın ölçüləri məhdud olduğundan ona baxmaqla bütün verilənlər bazası haqqında təsəvvürə malik olmaq çox çətindir. Ekrana bir və ya bir neçə yazı barədə məlumat çıxdığı halda, hesabata verilənlər bazası yazılarının böyük bir qrupu (hətta bazanın bütün yazıları) barədə məlumat daxil olur. Buraya, həmçinin, bir neçə avtomatik hesablana bilən verilənlər də əlavə olunur.

Bütün informasiya sistemləri axtarış və hesabatların tərtibi zamanı kifayət qədər mürəkkəb şərtlərdən istifadə etməyə imkan verir. Çox zaman yazı eyni zamanda bir neçə sadə şərti ödəməlidir.

Hesabi ifadələr hesabi əməliyyatların köməyilə yaradıldığı kimi, məntiqi əməliyyatların ("və", "və ya", "deyil") köməyilə məntiqi ifadələr (mürəkkəb şərtlər) tərtib oluna bilər. "Süd" şöbəsində "Xama"nı axtarmaq üçün onun qiymətinin 2 manatdan çox olmadığı şərtini də əlavə edək.

Relyasion modeldə verilənlər münasibətlər vasitəsilə təsvir olunurlar, lakin verilənlərin yaddaşda fiziki şəkildə necə yerləşməsi heç yerdə göstərilmir. Verilənlərin təsvirinin konseptual səviyyəsinin məntiqi səviyyədən ayrılması verilənlər bazasının proqramlaşdırılmasında inqilabi çevriliş oldu, belə ki, əvvəllər proqramlaşdırma verilənlərin saxlanılması üçün nəzərdə tutulan qurğuların (disklərin) idarə olunmasına çevrilirdi.

Hesabatları tərtib edərkən çox vaxt xüsusi sorğu dilindən istifadə olunur. "Xama" haqqında sorğu təxminən belə görünür:

Ərzaq = "xama" və qiymət<2 OLMAQ ŞƏRTİLƏ ərzaq mağazası CƏDVƏLİNDƏN BÜTÜN YAZILARI SEÇMƏK.

Kambala	4 man.	1 kq

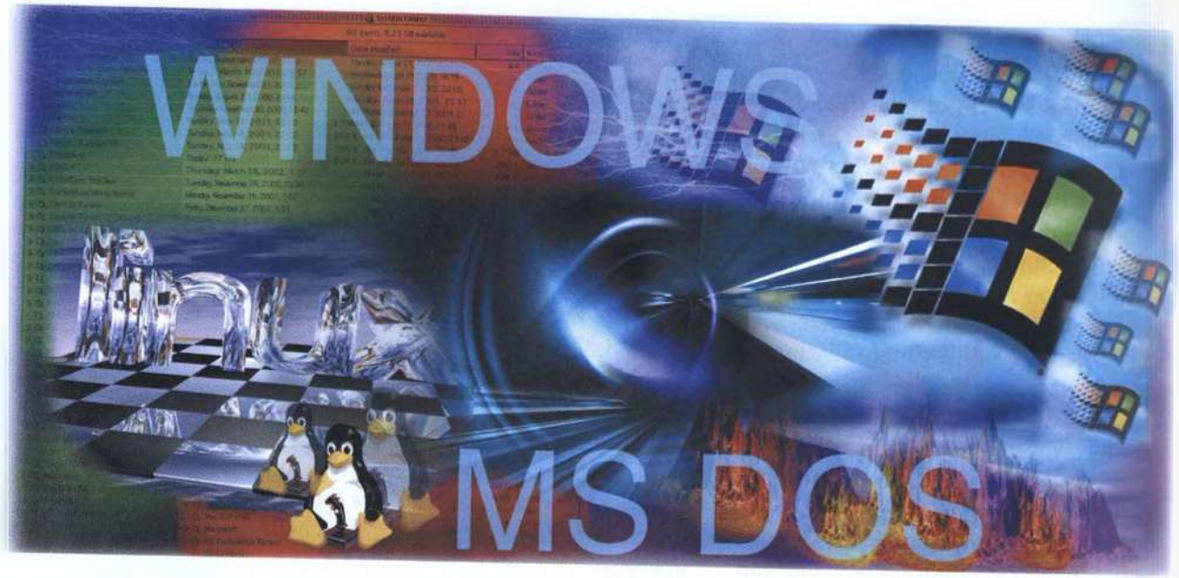
cədvəli aşağıdakı sorğunun nəticəsi olacaq:

ərzaq ="Kambala" OLMAQ ŞƏRTİLƏ ərzaq mağazası CƏDVƏLİNDƏN ərzaq, qiymət, ölçü vahidi SAHƏLƏRİNİ SEÇMƏK.

Əgər verilənlər bazasında "Kambala" yazısı bir neçədirsə, onda sorğu nəticəsində də bir neçə sətir alınacaq. Sorğu dilində nəyi almaq lazım olduğunun göstərilməsi, onu necə almaq lazım gəldiyini göstərməkdən vacibdir.

Əlbəttə, real verilənlər bazasında həm verilənlər bazasının layihələndirilməsi prosesi, həm hesabatların, həm də formaların özlərinin və sorğuların alınması prosessi çox mürəkkəbdir. Əsil isformasiya-axtarış sistemini yaratmaq üçün çox işlər görmək lazımdır, çünki bütün sistemin işinin səmərəliliyi verilənlərin cədvəllər üzrə necə paylanmasından asılıdır. Sadə hallarda, müasir layihələndirmə vasitələri relyasion verilənlər bazasının ümumi qurulma prinsiplərinə əsaslanan kiçik sistemləri yaratmağa imkan verir.

Adətən, praktikada hesabatlara hər hansı bir sahə üzrə verilənlərin sadəcə olaraq cəmi əvəzinə daha mürəkkəb statistikanın daxil edilməsi tələb olunur. Ona görə də sistemlərdə hər bir hesabat tipi üçün onun generasiyasını təmin edən xüsusi proqramı yazmaq tələb oluna bilər (bir çox hallarda kifayət qədər sadə).



KOMPÜTERİN ƏMƏLİYYAT SİSTEMİ

ƏMƏLİYYAT SİSTEMLƏRİNİN MEYDANA GƏLMƏSİ



EHM-in paket rejimində işi üçün perfokart dəstinin yüklənməsi.

İlk EHM-lər yalnız riyazi məsələlərin həlli üçün istifadə edilirdi, maşın kodlarında yazılmış hesablama alqoritmləri isə onlar üçün proqramlar rolunu oynayırdı. Proqramları kodlaşdırarkən proqramçı EHM-i sərbəst şəkildə idarə etməli və öz proqramının yerinə yetirilməsini təmin etməli olurdu. Lakin elektronikanın inkişafı nəticəsində avadanlıqlar təkmilləşdi, proqramlarda isə hesablama ilə yanaşı, informasiyanın kompüterlərə giriş-çıxışı üçün müstəqil altproqramlara bölünən alqoritmlər meydana gəldi. *Giriş-çıxış kitabxanaları* – EHM-in proqramlaşdırmasını asanlaşdıran xidməti altproqramlar yığını bu cür meydana çıxdı. Maşın vaxtının dəyəri çox yüksək olduğundan, bir neçə proqramın eyni zamanda yerinə yetirilməsinə tələbat yarandı: bir proqram giriş-çıxış əməliyyatını yerinə yetirərkən, digəri EHM-in prosessorundan istifadə

edirdi və əksinə. EHM-də yerinə yetirilən proqramları və proqramda istifadə olunan verilənləri *tapşırıq*, bir neçə proqramın eyni vaxtda yerinə yetirilməsini isə *tapşırıq növbəsi* adlandırdılar.

İlk vaxtlarda prosessorda məsələlərin birindən digərinə *keçid* alqoritmləri proqramın özünə altproqram şəklində daxil edilirdi. Bu cür altproqramlar yığını *monitor* və ya *supervizor* adlanırdı. Bundan başqa, yığına giriş-çıxış kitabxanaları, operativ yaddaşın paylanması altproqramları, ümumiyyətlə, hesablama məsələlərilə birbaşa bağlı olmayan bütün alqoritmlər daxil idi.

Daha bir problem ondan ibarət idi ki, EHM-də yerinə yetirilən proqramlarda tez-tez səhvlər olurdu və bunun nəticəsində ya proqram həmişə prosessoru məşğul edirdi ("asılı qalırdı"), ya da ki, öz işinin nəticələrini səhvən digər proqramların yerləşdiyi operativ yaddaşa yazırdı. Nə qədər ki, yalnız bir proqram yerinə yetirilirdi, bu hal dözülən idi, lakin bir neçə proqram eyni zamanda yerinə yetirilərdə, onlardan birinin sıradan çıxması bütün proqram dəstinin işinin pozulmasına səbəb olurdu. Qeyd etmək lazımdır ki, hesablama texnikasının inkişafı konkret məsələlər üçün nəzərdə tutulan EHM-dən bir çox məsələlərin həllini nəzərdə tutan universal EHM-ə aparan yolla getmişdir, buna görə də yaranan problem təxirə salınmadan özünün həll olunmasını tələb edirdi.

Proqramın yaddaşını digər proqramların təsadüfi istifadəsindən qoruyan xüsusi aparat mexanizmlərinin yaradılması ilə çıxış yolu tapıldı. Bu mexanizmlərin idarə olunmasını artıq proqramların özünə həvalə etmək düzgün olmazdı: qorumanı təmin edən alqoritmlərin yerləşdiyi yaddaşın özü proqramdan qorunmalıdır, əks halda proqramın yerinə yetirilməsi zamanı baş verən qəzalar yaddaşı korlaya bilərdi. Ona görə də, yaddaşı qoruma mexanizmlərini idarə edən xüsusi proqrama tələbat yarandı. *Rezident monitor* belə yarandı – monitora yaddaşın qorunmasını idarə edən altproqramlar əlavə edildi və onların hamısı ayrıca bir proqram şəklində tərtib edildi.



ƏMƏLİYYAT SİSTEMİ NƏDİR



Əməliyyat sistemi haqqında təsəvvür zaman keçdikcə inkişaf etmişdir. Q.Katsanın "Əməliyyat sistemləri. Praqmatik yanaşma" kitabından (1973-cü il) götürülmüş tərif: "Əməliyyat sistemi – proqram və verilənlərin mütəşəkkil yığınıdır. Bu yığın hesablama sistemi resurslarının idarə olunması, EHM üçün proqramların yaradılmasının asanlaşdırılması, həmçinin EHM-in idarə edilməsi üçün nəzərdə tutulmuşdur".

Q.Katsanın verdiyi tərif əməliyyat sisteminə avadanlıq nöqteyi-nəzərindən yanaşmadır: EHM avadanlığı mövcuddur, bu avadanlıqda proqramları yerinə yetirmək lazımdır və bu məsələni sadələşdirmək üçün isə əməliyyat sistemi mövcuddur. Əməliyyat sisteminə müasir baxış həm də EHM resurslarının istehlakçısı kimi istifadəçinin baxışını əks etdirir.

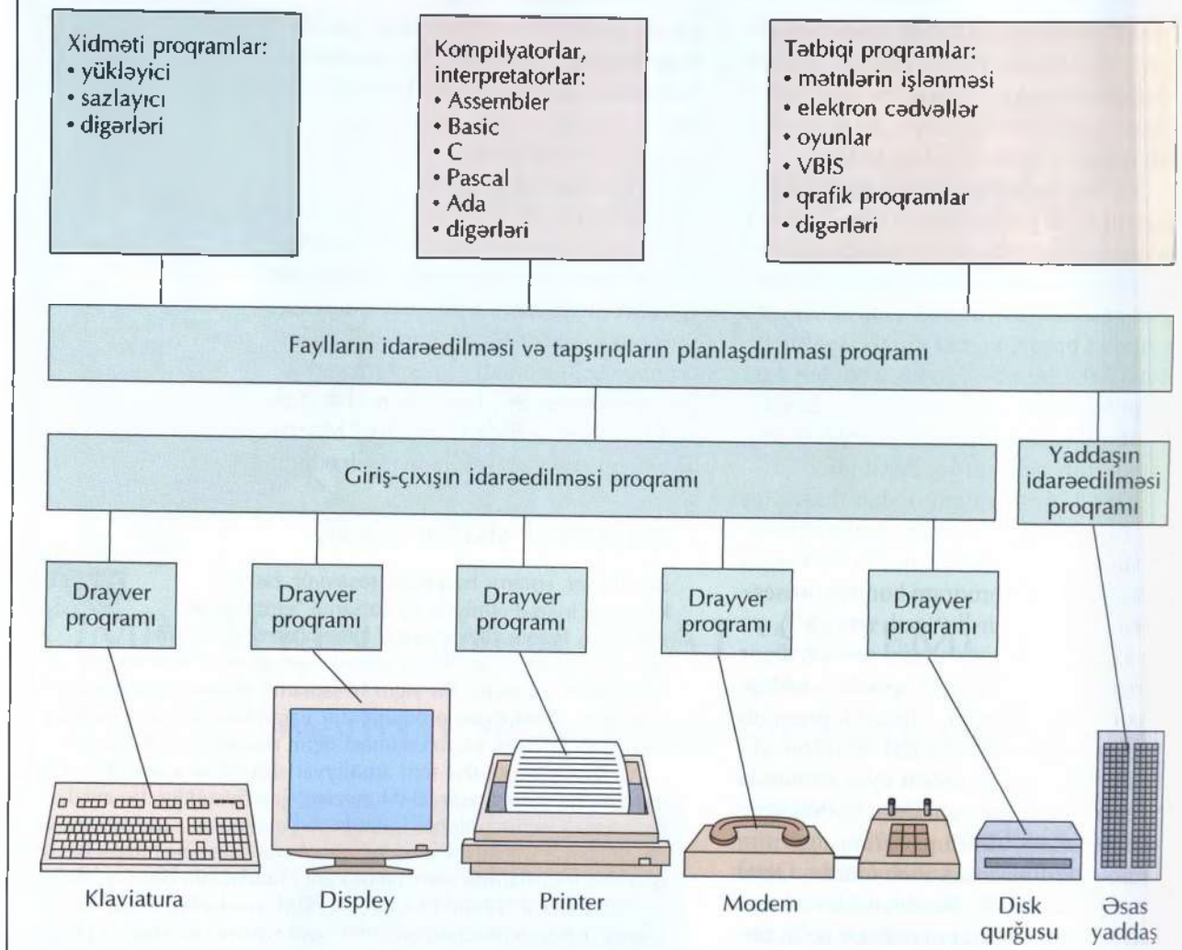
"Əməliyyat sistemi istifadəçilə EHM avadanlığı arasında vasitəçi rolunu oynayan proqramdır. Əməliyyat sistemi istifadəçiyə proqramın rahat və effektiv şəkildə yerinə yetirilməsinə lazımi şərait təmin etmək üçün nəzərdə tutulmuşdur" (A.Zilberşats, P.B.Helvin. "Əməliyyat sistemlərinin prinsipləri", 1998-ci il).

"Əməliyyat sistemi – tətbiqi proqramların yerinə yetirilməsi prosesini idarə edən və istifadəçilə EHM avadanlığı arasında interfeys rolunu oynayan proqramdır. Əsasən əməliyyat sisteminə qarşı üç tələb irəli sürülür: rahatlıq, effektivlik və inkişaf perspektivi" (U.Stallinqs. "Əməliyyat sistemləri", 1995-ci il).

Nəhayət, Endryu Tanenbaumun "Müasir əməliyyat sistemləri" (1992-ci il) adlı kitabında əməliyyat sisteminin iki tərifi verilir: birisi avadanlıq nöqteyi-nəzərindən, digəri isə istifadəçi nöqteyi-nəzərindən.

"Əməliyyat sistemi genişləndirilmiş EHM kimi. Əməliyyat sisteminin funksiyası istifadəçiyə əsas EHM-ə nisbətən asan proqramlaşdırılan və ona ekvivalent genişləndirilmiş və ya virtual EHM-in təqdim olunmasıdır...

Əməliyyat sistemi resursların inzibatçısı kimi. Əməliyyat sisteminin məqsədi verilən resursdan kimin istifadə etməsini izləmək, resurslara sorğuları cavablandırmaq və onların istifadəsini protokollaşdırmaq, həmçinin istifadəçilərin müxtəlif məsələləri həll etmək üçün resurslara müraciəti zamanı yaranan konfliktləri həll etməkdir".



Rezident monitor – bu gün qəbul edildiyi kimi, əməliyyat sisteminin rüşeymini təşkil edir. İndi tətbiqi proqramlar artıq yalnız öz alqoritmlərini reallaşdırmaqla məşğuldurlar və monitora yalnız köməkçi alqoritmlər üçün müraciət edirlər. Müraciət zamanı onlar *tətbiqi proqram interfeysi* adlanan qaydalar yığınından istifadə edirlər. İlk dəfə olaraq tətbiqi proqramların bir EHM-dən digərinə *köçürülməsinin* mümkünlüyü barəsində danışmağa başladılar. Proqram interfeysinin meydana gəlməsi nəticəsində EHM-lə iş üsullarının avadanlıqdan ayrılması və EHM-lərin insana "yaxınlaşması" prosesi başladı.

Proqram interfeysi avadanlıqda heç cür ifadə edilə bilməyən mücərrəd anlayışların yaradılmasına imkan verdi. Məsələn, *fayl* və müvafiq olaraq *fayl sistemi* anlayışı – informasiyanın saxlanmasının təşkili üçün interfeys dəsti və verilənlərin strukturu meydana gəldi. Həmçinin resurs istifadəçisi və EHM-in iş vahidi kimi *prosess* (yaxud *məsələ*) anlayışı yarandı.

Rezident monitor həmişə EHM-in operativ yaddaşında yerləşir (bu səbəbdən də onu "rezident" adlandırırlar), çünki onun altproqramları bütün digər proqramların yerinə yetirilməsi üçün lazımdır. Ona görə də həmin proqramlar üçün "tətbiqi proqramlar", yaxud "istifadəçi proqramları" terminlərini işlətməyə başladılar.

Əlbəttə, "fayl" və "proses" anlayışları rezident monitorun yaranmasından xeyli əvvəl mövcud idi. Lakin onun ayrıca xüsusi proqram kimi meydana gəlməsi tətbiqi proqramların xidməti altproqramlarla iş qaydasını təsbit etməyə imkan verdi. Həqiqətən də, nə qədər ki monitorlar proqramların daxilində yerləşirdi, proqramın hazırlanması zamanı fayllarla iş altproqramının birinin digərilə (məsələn, daha səmərəli olanla) əvəz edilməsinə həmişə imkan var idi. Rezident monitor aşkar və qeyri-aşkar şəkildə "fayl" anlayışını yenidən təsvir etməklə, həmçinin fayllarla və proseslərlə işləmə qaydalarını təsbit etməklə yanaşı, bu anlayışların mahiyyətini də dəqiqliklə ifadə etdi.

Əməliyyat sistemlərinin inkişafındakı növbəti mərhələ iş qaydalarının artmasının nəticəsi kimi əməliyyat sistemlərinin özlərinin mürəkkəbləşməsilə xarakterizə olunur. Rezident monitora tez-tez müraciət olunan əməliyyatların – faylların köçürülməsini, mətnlərin redaktə edilməsini, proqramların proq-

Prosesə proqram sayğacı da daxil olmaqla kompüterin aparat registri; prosessin ünvan sahəsinin – prosesin oxumaq, yazmaq və yerinə yetirmək üçün istifadə etdiyi EHM-in yaddaşının təsviri; prosesin işlədiyi faylların təsviri və proqramın yerinə yetirilməsi üçün zəruri olan bütün digər informasiyalar aid edilir.

ramlaşdırma dilindən maşın koduna kompilyasiyasının yerinə yetirilməsini asanlaşdıran köməkçi proqramlar və s. əlavə edildi. Onların hamısı birlikdə *əməliyyat sistemi*, köməkçi proqramlar dəsti – redaktorlar, kompilyatorlar, fayllarla iş proqramları isə *sistem utilitləri* adlandırıldı. Bu zaman, sistem utilitlərini və tətbiqi proqramları ayıran hədd demək olar ki, şərti idi. "Rezident monitor" termini *əməliyyat sisteminin nüvəsinə* çevrildi və nüvə də öz növbəsində, bir neçə mühüm hissəyə bölündü:

– proseslərin planlaşdırıcısı;
– yaddaşın idarəedilməsi altsistemi;
– fayl sistemi;
– giriş-çıxış əməliyyatlarının idarəedilməsi altsistemi;
– tətbiqi proqram interfeysi.

PROSESLƏRİN PLANLAŞDIRICISI

Proseslərin planlaşdırıcısı – prosesləri idarə edən altsistemlərin ürəyi hesab olunur. Bu planlaşdırıcılar proseslər arasında prosessor vaxtının bölünməsi alqoritmlərini həyata keçirirlər, daha doğrusu, prosessorda bir prosesin öz növbəsini gözləyən digər proseslə əvəzlənməsini həyata keçirir. Adətən, bu proseduru ifadə etmək üçün *prosesin kontekstinin dəyişdirilməsi* terminindən istifadə edilir. Bu cür kontekst dəyişmə alqoritmi hansı prosesə nə vaxt prosessor vaxtının verilməsini müəyyən edən bir çox parametrlərdən asılıdır.



Planlaşdırıcının işini əks etdirmək və ümumilikdə əməliyyat sisteminin işini başa düşmək üçün *stratetegiya* və

Kvant anlayışı bölünməzlik deməkdir. Yerinə yetirilmə kvantı o mənada bölünməzlikdir ki, nə qədər ki, prosessorda yerinə yetirilən proses planlaşdırıcı tərəfindən ona ayrılan müddətdən tam istifadə etməmişdir, o prosessoru digər prosesə verməyəcək.

mexanizm anlayışlarını fərqləndirmək lazımdır. Strategiya "nə etmək", mexanizm isə "necə etmək" lazım olduğunu müəyyən edir.

Məsələn, planlaşdırıcının işinin strategiyası ondan ibarət ola bilər ki, proses aşkar (xüsusi altproqrama müraciətlə) və ya qeyri-aşkar (giriş-çıxış əməliyyatı yerinə yetirilərkən, proqram başa çatdırılarkən) şəkildə əməliyyat sisteminə müraciət edib onu xəbərdar edir ki, prosessor boşdur və ondan digər proseslərin yerinə yetirilməsi üçün istifadə oluna bilər.

Bu halda deyilir ki, əməliyyat sistemi *kooperativ çoxtapşırıqlıq* rejimində fəaliyyət göstərir, daha doğrusu, proseslər, bəlkə də, ümumi "səsvermə" yolu ilə birlikdə hansı tapşırığın növbəti addımda yerinə yetiriləcəyini müəyyən edir. Kooperativ çoxtapşırıqlıq, məsələn, ilk EHM-lərin monitor proqramlarında,



DEC firmasının Alpha kompüteri.

həmçinin fərdi kompüterlərin əməliyyat sistemlərində – MS DOS, Windows 3.1 və ondan əvvəlki versiyalarda istifadə olunmuşdur.

Digər strategiya – *sıxışdırılan çoxtapşırıqlıq* ondan ibarətdir ki, prosessorda hər bir prosesin başqasına mane olmadan zəmanətli şəkildə yerinə yetirilməsi təmin olunur. Planlaşdırıcı prosesin yerinə yetirilməsini dayandıra və onu digər proseslə əvəz edə bilər. Bu zaman o, hər bir prosesə *yerinə yetirilmə kvantı* adlanan əvvəlcədən müəyyən edilmiş müddəti ayırır və proses də bu müddət ərzində prosessorda fasiləsiz olaraq yerinə yetirilir. Adətən, yerinə yetirilmə kvantı elə seçilir ki, proses prosessoru o qədər də çox məşğul etməsin, ancaq ayrılan vaxt ərzində hansısa bir faydalı iş görə bilsin.

Müasir proseslər üçün yerinə yetirilmə kvantının tipik qiyməti 10 ms təşkil edir. Yalnız olduqca yüksək sürətli prosessorlar üçün, DEC firmasının Alpha prosessoru üçün kvant 1 ms həddindədir.

Kvant başa çatdıqdan sonra planlaşdırıcı prosessorda yerinə yetiriləcək növbəti prosesi müəyyən etməklə yenidən planlaşdırma alqoritmini həyata keçirir. *Vaxtın bölünməsi* xüsusi əməliyyat sistemlərində bütün proseslər eyni hüquqlu hesab olunurlar və növbə ilə öz kvantlarını alırlar. İstifadəçiyə elə gəlir ki, prosessor bütün prosesləri eyni vaxtda yerinə yetirir. Bu cür alqoritm *dairəvi sistem üzrə yenidən planlaşdırma alqoritmi* (*ing.* "round robin scheduling") adlanır.

Yenidən planlaşdırma alqoritmində proseslərin bərabərliyinə nail olunması, ilk baxışdan sadə görünsə də, əslində, elə də sadə məsələ deyildir. Tutaq ki, bir proses mətn redaktoru, digəri isə qiymət vergüldən sonra milyonda bir dəqiqliklə hesablayan proqramdır. Əməliyyat sistemi nöqteyi-nəzərindən mətn redaktorunun iş alqoritmi və hesablama alqoritmi təxminən bu cür qurulur:


```
alq Mətn redaktoru
baş
  db nə qədər ki çıxış klavişi basılmayıb
    klaviaturadan simvolun daxil
    edilməsini gözləmək
    simvolu emal etmək
    simvolu ekranda göstərmək
  ds
son

alq Hesablama π
baş
  db nə qədər ki π ədədi müəyyən
edilməyib
    hesablamaq
  ds
son
```

Beləliklə, hesablamanı yerinə yetirən bir proses həmişə prosessoru məşğul edir, yaxud digər prosesin olmasını nəzərə alaraq, onun yerini tutmaq istəyir (daim prosessorun resursu uğrunda mübarizə aparır). Digər proses – mətn redaktoru – əsasən, daxiletmə əməliyyatını yerinə yetirir və prosessor uğrunda təsadüfi hallarda rəqabət aparır. Həqiqətən də, hətta peşəkar operator belə mətni dəqiqədə 600 simvol, yaxud 100 ms-da 1 simvol sürətilə yığır. Əgər yerinə yetirilmə 10 ms təşkil edirsə, onda mətn redaktoru növbəti simvolu daxil etməyi gözləməklə, faktiki olaraq 10 kvant müddətində prosessor uğrunda rəqabət aparmır. Əgər proseslər eynihüquqlu olarsa, planlaşdırıcı redaktora bir yerinə yetirmə kvantı verəcək, bundan sonra isə növbəti kvantı hesablama prosesinə keçəcək. Müəyyən olunmuşdur ki, əgər daxiletmə ilə ekranda göstərilmə arasında yubanma 20 ms-i (və yaxud 2 yerinə yetirilmə kvantını) keçmirsə, insan hesab edir ki, sanki simvol ekranda ani olaraq görünür. Əgər redaktor bir kvant vaxtı ərzində simvolu işləyə bilmirsə, prosessor vaxtını daha bir kvant müddətində gözləməli olur. Nəticədə istifadəçi simvolu daxil edərkən yubanmanı hiss edir.

Bunu düzəltmək üçün proses-redaktora birdən çox kvant (on kvant – onun prosessorda yerinə yetirilmədiyi müddətə bərabər olaraq vermək daha düzgün olardı) vermək lazımdır. Belə görünür ki, bərabərliyin təmin edilməsi üçün prosesləri qeyri-bərabər hüquqlu etmək lazımdır.



Bu cür ədalətə nail olmaq üçün əməliyyat sisteminə *prosesin prioriteti* anlayışı daxil edilir. Bu prioritet prosessorda prosesin nə qədər uzun müddət yerinə yetirilmədiyini göstərən tam ədədlə ifadə edilir.

Planlaşdırıcı növbəti kvant qurtardıqdan sonra yerinə yetirilən prosesin prioritetini azaldır və bundan sonra rəqabət aparan proseslər arasından ən böyük prioritetə malik olan prosesi seçən yenidən planlaşdırma alqoritmini başladır.

Misal üçün, proses-redaktorun simvolun daxil edilməsini gözlədiyi və hesablama prosesinin yerinə yetirildiyi on kvant müddəti keçəndən sonra, hesablama prosesinin prioriteti 10-a, proses-redaktorun prioriteti isə sıfıra bərabər olacaq. Ona görə də, növbəti 10 kvant müddətində proses-redaktorun prioriteti yüksək olacaq və planlaşdırıcı ona üstünlük verəcək. Sonra proseslərin prioritetləri



PRİORİTETLƏRİN HESABLANMASI DÜSTURU

Bəzən verilən prosesin digərlərindən prioritetli olduğunu aşkar şəkildə göstərmək lazım gəlir. Bunun üçün prosesin müəyyən prioriteti daxil edlir ki, onun əsasında da dinamik prioritet hesablanır. İkincisi, dinamik prioritetlərin daha çevik şəkildə hesablanması üçün düsturun özünü bir qədər mürəkkəbləşdirirlər.

UNIX BSD 4.4 kimi əməliyyat sistemi üçün düstur bu cür yazılır:

$$P_{din} = P_0 + \left[\frac{T}{4}\right] + 2 \times P_{fiks},$$

burada P_{din} – prosesin dinamik prioriteti; P_0 – istifadəçi proseslərinin təməl sistem prioriteti; P_{fiks} – prosesin qeyd olunmuş prioriteti; T – öz növbəsində, aşağıdakı rekkurent düsturlarla hesablanan prosesin yerinə yetirilmə vaxtıdır:
əgər proses prosessoru məşğul edərsə,

$$T = \frac{2 \times T_{or}}{2 \times T_{or} + 1} T + P_{fiks},$$

əgər proses gözləmə vəziyyətində olarsa,

$$T = \left[\frac{2 \times T_{or}}{2 \times T_{or} + 1}\right]^{T_{gözlə}} \times T,$$

T_{or} – prosessorun orta yüklənmə müddəti; $T_{gözlə}$ – prosesin gözləmə vəziyyətində olduğu vaxtdır.

eyni olacaq və planlaşdırıcı növbəti prosesi təsadüfi qaydada seçəcək.

Təsvir edilən alqoritm *dairəvi sistem üzrə əks əlaqəli planlaşdırma alqoritmi* adlanır. Planlaşdırıcının bu cür alqoritmdə işləməsi *interaktiv tapşırıqlar*, yəni informasiyanın daxil edilməsini təmin



edən tapşırıqlar, interaktiv olmayan tapşırıqlar, məsələn, hesablama alqoritmləri üzərində üstünlüyə malik olur. Bu planlaşdırma alqoritmi Windows və UNIX əməliyyat sistemlərində tətbiq edilir.

Tamamilə başqa tipli alqoritmlərə *real vaxt sistemlərinin yenidən planlaşdırması* alqoritmləri aiddir. Real vaxt əməliyyat sistemləri informasiyanın emalı məsələlərinin həlli üçün tətbiq edilir. Burada emalın özünün sürətilə informasiyanın daxil olması sürəti müqayisə ediləbilən olmalıdır. Buna yaxşı nümunə kimi xəstəxanada pasientlərin sağlamlıq vəziyyətinə nəzarət edən avtomatlaşdırılmış idarəetmə sistemini göstərmək olar. Belə ki, burada mütəmadi olaraq pasientlərin vəziyyətinin dəyişməsinə tez reaksiya verilməsi tələb olunur. Yaxud da sənaye müəssisələrində manipulyatorun hərəkətini ani olaraq dəyişən robotların avtomatlaşdırılmış idarəetmə sistemini misal göstərmək olar.

Verilmiş müddətdə məlumatın emalının uğursuzluğunun labüd səhvlərə gətirib çıxardığı real vaxt sistemləri *sərt real vaxt sistemləri* adlanır. Yuxarıda xatırladılan xəstələrin vəziyyətinə nəzarət və istehsalın idarə olunması sistemləri əməliyyat sistemlərinin məhz bu sinfinə aiddir.

Bəzən hadisənin verilmiş vaxt intervalında zəmanətlə başa çatdırılması elə də vacib olmur, yaxud, heç olmasa, labüd səhvlərə gətirib çıxartmır. Məsələn, kompüterdə səs faylının səsləndirilməsi – *yumşaq real vaxt* məsələsidir. Səsləndirmə alqoritmi aşağıdakı kimi görünür:

```
alq Səsləndirmə
baş
 | db nə qədər ki faylın sonu deyil
 |  | verilənlərin növbəti porsiyasını
 |  | oxumaq
 |  | oxunan porsiyanı səs kartının
 |  | registrinə yazmaq
 | ds
son
```




Rusiya Uçuşları İdarəetmə Mərkəzi.

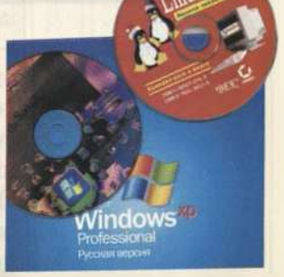
Burada da müəyyən vaxt intervalında növbəti verilənlər porsiyasını səs kartının registrlərinə çatdırılıb yazmaq gərəkdir, yoxsa səs kartı heç nə səsləndirməyəcək. Səsin məqbul keyfiyyətdə səsləndirilməsi üçün (rəqəmli şəklə salınma tezliyi – 44 Khz, stereo, 16-bitlik səs kartı, yəni 44 Khz tezliklə 4 bayt) verilənləri diskdən 200 kbayt/san sürətlə oxumağa və yazmağa nail olmaq lazımdır. Alqoritmdə verilənlər porsiyasını 8 kbayta bərabər hesab etmək olar, yəni dövrün bir tam addımını 8/200 = 1/25 san müddətində yerinə yetirməyə nail olmaq lazımdır. Əgər ayrılmış vaxtda bunu etmək mümkün olmazsa, onda bu, səsin 1/25 san qədər itirilməsinə gətirib çıxaracaq. Vəziyyət arzuolunan olmasa da, qaçılmaz da deyil. İnsanların bir çoxu səsin itməsini, sadəcə, hiss etməyəcək. Deməli, yumşaq real vaxt rejimində fəaliyyət göstərən əməliyyat sistemi imkan daxilində müəyyən edilmiş vaxt intervalında məlumatın tətbiqi prosesə çatdırılmasını təmin etməlidir.

Planlaşdırıcının iş mexanizmi aşağıdakı qaydada qurula bilər. Sistemdəki bütün proseslər heç vaxt dəyişilməyən yerinə yetirilmə prioriteti əldə edir. Yenidən planlaşdırma alqoritimində həmişə ən böyük prioritetli proses seçilir. Sərt real vaxt sistemlərində yenidən planlaşdırma alqoritmini istənilən hadisə baş verdikdə yerinə yetirmək lazımdır. Bununla da zəmanət verilir ki, hadisəni gözləyən proses yerinə yetirilməyə kifayət qədər yüksək prioritet şərti daxilində buraxılır.

ƏMƏLİYYAT SİSTEMLƏRİNİN ÖLÇÜLƏRİ

Əməliyyat sistemlərinə (ƏS) yeni funksiyalar daxil edildikcə onların həcmi genişlənməyə başladı: 1963-cü ildə Massaçusets Texnologiya İnstitutunda tətbiq edilən Compatible Time Sharing System vaxtın bölünməsi əməliyyat sistemi 32 min yaddaş sözündən (hər biri 36 bit olmaqla) ibarət idi. 1964-cü ildə IBM firmasının istehsalı olan OS/360 artıq milyondan çox komandası olan koddan ibarət idi. 1975-ci ildə Massaçusets Texnologiya İnstitutunda Bell Laboratories ilə birlikdə hazırlanmış Multiks "monstr"ı 20 milyondan çox komandaya malik idi.

Fərdi EHM-lər üçün yaradılan əməliyyat sistemləri isə, əksinə, yığcam idi: MS DOS-un ilkin versiyaları 50 min komandadan artıq deyildi, 80-ci illərin ortalarında UNIX 7.5 Bell Laboratories isə təqribən 100 min maşın komandasından ibarət idi. Vaxt keçdikcə, fərdi EHM-lər üçün nəzərdə tutulan UNIX olduqca yığcam olaraq qaldı, amma Windows-95, 98, 2000 seriyaları barədə bunu demək olmaz.





IBM-360 kompüterinin maşın zalı.



Boeing-707-nin kabinəsi.

Yumşaq real vaxt sistemlərində, əvvəllər olduğu kimi, prosessor vaxtı yerinə yetirilmə kvantlarına bölünür və yenidən planlaşdırma alqoritmi kvant müddəti başa çatdıqdan sonra başladılır. Əgər hadisə münasib olmayan vaxtda baş verirsə (məsələn, növbəti yerinə yetirilmə kvantının başlanğıcında), yəqin ki, hadisəni emal etməyə imkan olmayacaq, lakin prosessor vaxtı da boş yerə, daim yenidən planlaşdırma alqoritminin yerinə yetirilməsinə sərf olunmayacaqdır. Əməliyyat sistemi yardımçı proqramdır və gərək olmayan yerdə əməliyyat sisteminin öz alqoritmlərinin yerinə yetirilməsilə prosessorun resursunu məşğul etməyə dəyməz.

YADDAŞIN İDARƏ EDİLMƏSİ ALTSİSTEMİ

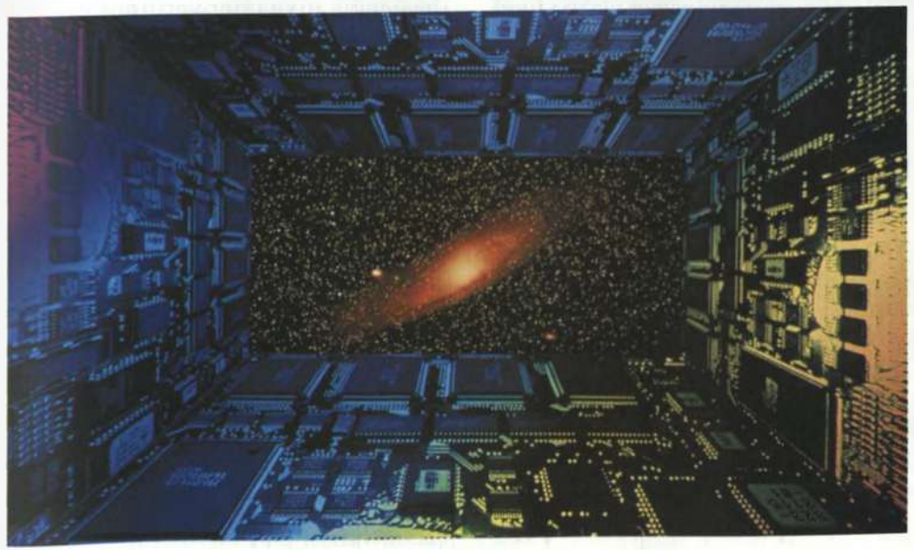
Əməliyyat sisteminin digər mühüm hissəsinə – *yaddaşın idarə edilməsi altsisteminə* qarşılıqlı əlaqədə olan iki məsələ həvalə edilib: EHM-də yerinə yetirilən proseslərin bir-birindən qorunması və EHM-in yaddaşının proseslər arasında bölünməsi.

Prosessor yaddaşla ünvanlar vasitəsilə əlaqə yaradır: yaddaşın hər bir baytının, onu birqiymətli olaraq təyin edən nömrəsi və ya *fiziki ünvanı* mövcuddur (birinci yaddaş baytının ünvanı – 0, ikincisinin – 1 və s.). Yaddaşın bütün baytlarının ünvanlarından ibarət toplu *fiziki ünvanlar sahəsi* adlandırılır. Yaddaşın idarə edilməsi məsələsini həll etmək üçün *virtual yaddaş* adlanan aparat vasitəsindən istifadə edilir. Onun funksiyası əlavə ünvan sahəsini və ünvanların bir sahədən digərinə dəyişdirilməsi mexanizmlərini təmin etməkdir. Həmin əlavə ünvan sahəsi *virtual ünvan sahəsi*, onun daxilindəki ünvanlar isə *virtual ünvanlar* adlanır. Fiziki ünvan sahəsindən fərqli olaraq, virtual ünvan sahəsi sadəcə heç nəyə uyğun olmayan ünvanlar dəstidir. Lakin virtual ünvanlar sahəsindən olan ünvanlara müəyyən qaydalarla uyğun olaraq fiziki ünvanlar mənimsətməklə, onların mövcudluğuna məna vermək mümkündür. Bu halda virtual ünvan real yaddaş baytının nömrəsi olacaq.

Birinci məsələ – prosesləri bir-birindən qorumaq, yəni bir prosesin yaddaşını digər prosesin təsadüfi müdaxilələrindən qorumaqdır. Hər bir prosesə özünün virtual sahəsi – sıfırdan prosessorun imkan verdiyi maksimal nömrəyə qədər olan ünvanlar toplusu və hər bir proses üçün unikal olan virtual ünvanların fiziki ünvanlara çevrilməsi qaydası qarşı qoyulur. Prosesin virtual sahəsini ifadə etmək üçün *prosesin ünvan sahəsi* termini işlədilir. Öz ünvan sahəsində fəaliyyət göstərən prosesin digər prosesin ünvan sahəsinə nəsə yazmaq imkanı yoxdur. Həqiqətən də, proses tərəfindən tanınan istənilən ünvan prosesin ünvan sahəsinə aiddir, digər ünvanlar isə aid deyil.

Virtual yaddaş mexanizmindən istifadə etmək üçün EHM-ə prosesin nömrəsi və virtual ünvan cütlüyünü yaddaşdakı fiziki ünvana çevirən xüsusi bir qurğu – *yaddaşı idarəedən dispetçer* əlavə etmək lazımdır. Müasir prosessorlar üçün yaddaşı idarəedən dispetçer prosessorun tərkibində yerləşən ayrıca blokdur, lakin hələ XX əsrin 90-cı illərinin əvvəllərində dispetçer daha çox EHM-in tərkibində ayrıca mikrosxem kimi yerləşirdi.

Adətən, çevrilmə qaydaları fiziki və virtual ünvanların diapazonları arasında uyğunluğun saxlandığı cədvəl formasında yazılır. Ünvanlar hərəsində bir neçə min ardıcıl ünvan olan dəstlərdə, məsələn, tərkibində 4096 ünvan yerləşən dəstlərdə qruplaşdırılır ki, bu da 4 kbayt yaddaşa və ya 4 kbayt virtual ünvan sahəsinə uyğundur. Bu cür ünvanlar



"Virtual yaddaş", "virtual ünvan" terminləri ingilis dilindəki "virtual memory" və "virtual address" terminlərinin hərfi tərcüməsidir. Lakin ingilis və Azərbaycan dillərindəki "virtual" sifətinin mənası bir-birinin əksidir. İngiliscə virtual "faktiki", "həqiqi" mənasını verir. Azərbaycan dilində isə virtual sifəti "mövcud olmayan", lakin "mümkün olan" ifadələrinin sinonimi kimi işlədilir. Məsələn, virtual reallıq. Demək olar ki, ingiliscə virtual memory termini ünvanlar yığını dəsti və onların fiziki yaddaş ünvanlarına çevrilməsi qaydaları deməkdir. Azərbaycan dilində olan "virtual yaddaş" termini isə heç nəyi təyin etməyən mücərrəd yaddaş mənasını verir.



dəsti *səhifə* adlanır. Əgər bu fiziki ünvanlar səhifəsidirsə, onda onu *fiziki səhifə*, yaxud *yaddaş səhifəsi* adlandırırlar. Uyğun olaraq, virtual ünvan səhifəsini *virtual səhifə* adlandırırlar. Səhifələri yaddaş baytlarının nömrələndiyi qaydada nömrələmək lazımdır, bu zaman səhifənin ünvanı alınır. *Səhifələr cədvəli* adlanan yaddaşı idarəedən dispetçer cədvəlində fiziki səhifələrin ünvanları saxlanır, dispetçer isə cədvəldəki virtual səhifələrin ünvanlarına əsasən müvafiq fiziki səhifənin ünvanının cədvəldə yerləşdiyi yeri hesablayır.

Virtual yaddaş mexanizmindən istifadə planlaşdırıcıda cüzi dəyişikliklərin edilməsini tələb edir, məhz prosesi (konteksti) dəyişdirən alqoritmə səhifələr cədvəlinin dəyişdiricisini əlavə etmək lazımdır. Yəni bir prosesi digərilə əvəz edərkən dispetçerə yeni prosesin səhifələr cədvəlinin yaddaşda yerləşmə ünvanını göstərmək lazımdır. Səhifələr cədvəlinin dəqiq formatı dispetçerin arxitekturası ilə (müasir prosessorlarda – prosessorun arxitekturası ilə) müəyyən edilir.

Səhifələr cədvəlinin yaddaşda yerləşməsi ona gətirib çıxarır ki, hər dəfə virtual ünvan fiziki ünvana çevrilərkən dispetçer informasiya üçün yaddaşa müraciət etməli olur. Prosessorun işinin sürəti ünvana müraciət etmə sürətindən xeyli yüksəkdir, ona görə də ünvanlar çevrilərkən prosessor işsiz qalır. Prosessorun işsiz qalma hallarını azaltmaq üçün belə edirlər: prosessorda böyük olmayan həcmdə yaddaş – *ünvanların çevrilməsi buferi* yaradılır. Həmin buferdə virtual və fiziki səhifələr arasındakı müəyyən uyğunluqlar, yəni əslində səhifələr cədvəlinin ayrı-ayrı elementlərinin surətləri saxlanılır. Virtual ünvanı fiziki ünvana çevirərkən prosessor ünvanları çevirmə buferində uyğunluq axtarır və uyğunluq olmadıqda artıq yaddaşdakı səhifələr cədvəlinə müraciət edir. Praktikada bu sadə üsulun köməyilə yaddaşa müraciətlərin sayı nəzərəçarpacaq dərəcədə azaldılır.

Proseslərin bir-birindən qorunması mexanizmindən həm də fiziki yaddaşı proseslər arasında bölmək üçün də istifadə edilir. Məsələni qısaca olaraq belə ifadə etmək olar: fiziki ünvanın məhdud sahəsində proseslərin mümkün qədər çox ünvan sahələrini yerləşdirmək.

Hər bir prosesin ünvan sahəsi eyni cür tətbiq olunmur, onlarda aşkar istifadəsiz yerlər – "deşiklər" var ki, bu deşiklər üçün fiziki yaddaşın ayrılması lazım deyil. Həmçinin ünvan sahəsinin müəyyən bir hissəsi, yəqin ki, tez-tez işlədilməyəcək. Məsələn, yalnız proqramın işinin başlanğıcında (işçi dəyişənlərə başlanğıc qiymətlər verilməlidir) və sonunda istifadə edilən (yaddaşı boşaltmaq, nəticəni çıxışa vermək və s. lazımdır) istifadə olunan kod və verilənlər.

Müvafiq olaraq, yalnız işləri bitirərkən istifadə edilən ünvan sahələrinin hissələri üçün fiziki yaddaşı, belə yaddaş tələb olunanadək ayırmamaq olar, yalnız işin başlanğıcında tətbiq olunan yaddaşı isə istifadə etdikdən sonra geri götürmək olar.

Beləliklə, hər bir zaman intervalında proses iş üçün öz ünvan sahəsinin bir hissəsini istifadə edir. Müəyyən zaman intervalında yaddaşın istifadə edilən virtual səhifələrini prosesin *işçi çoxluğu* adlandırırlar. Aydındır ki, işçi çoxluq müvəqqəti zaman intervalının davam-

etmə müddətindən asılıdır (o, nə qədər çox olarsa, prosesin işçi çoxluğu bir o qədər artıq olar). Əgər zaman intervalı kimi proqramın bütün işləmə müddəti götürülərsə, işçi çoxluq tətbiq edilən ünvan sahəsinə bərabər olacaqdır.

Fiziki yaddaşa qənaət etmək üçün onu yalnız yerinə yetirilən proseslərin işçi çoxluğundan olan virtual səhifələr üçün ayırmaq olar. İşçi çoxluğu müəyyənləşdirərkən yerinə yetirilmə kvantına bərabər zaman intervalını seçmək daha əlverişlidir. Yaddaşın ən yaxşı bölüşdürülməsi üçün alqoritmi belə təsvir etmək olar: prosesə növbəti yerinə yetirilmə kvantını ayırarkən bu kvant müddətində fiziki yaddaşda prosesin işçi çoxluğunun bütün səhifələrini yerləşdiririk.

Bu alqoritm, həqiqətən də ən yaxşı alqoritm olsa da, təəssüf ki, onu tətbiq etmək olmaz. İşçi çoxluğa hansı səhifələrin daxil olduğunu yalnız kvant müddətinin gedişində bilmək olar, alqoritmdə isə onu başlanğıcda bilmək tələb olunur.

Gələcəkdə nələr baş verəcəyini söyləmək çətindir, lakin artıq əldə edilən təcrübədən istifadə etməyə dəyər. Praktikada, demək olar, həmişə elə təəssürat yaranır ki, işçi çoxluğu bir neçə yerinə yetirilmə kvantı müddətində əsaslı surətdə dəyişmir. Müvafiq olaraq, prosesin işinin başlanğıcında onun işçi çoxluğunu boş hesab etmək və fiziki yaddaşı virtual səhifə üçün lazım gəldikcə ayırmaq, həmçinin hər hansı bir zaman intervalı ərzində müraciət edilməyən ünvan səhifələrinin hissələri üçün yaddaşı məcburi qaydada götürmək olar. Ustəlik, əgər interval kimi sonsuzluğu götürsək, yalnız iş üçün boş fiziki yaddaş qalmayacağı təqdirdə biz fiziki yaddaşı məcburi qaydada götürəcəyik. Həmin anda yaddaş ən "köhnə" virtual səhifədən, daha doğrusu, daha çox müddətdə müraciət olunmayan virtual səhifədən götürülür.

Səhifələrin qorunması mexanizmi virtual səhifəyə nə vaxt fiziki yaddaş ayırmaq lazım olduğunu müəyyən etməyə imkan verir. Prosesin başladıldığı an hələ boş olan virtual və fiziki səhifələr arasındakı uyğunluq cədvəlini hazırlamaq olar. Əlavə olaraq hələ informasiyanı da saxlamaq lazım gəlir: çevirmə cədvəlində uyğunluğun hələ qurulmadığı virtual səhifələri prinsipcə belə uyğunluğun olmalı olduğu ünvan sahəsindəki "deşiklərdən" fərqləndirmək lazımdır.

Əgər prosessor virtual ünvana müraciət edərkən virtual səhifə üçün uyğunluq tapmırsa, *müstəsna vəziyyət – kəsilmə* yaranır. Kəsilmə yaranarkən prosessor xüsusi altproqramı – *kəsilmə emalçısını*

PROSESİN DAXİLİNDƏ QORUNMAQ ÜÇÜN VİRTUAL YADDAŞDAN İSTİFADƏ EDİLMƏSİ

Virtual yaddaş mexanizmi prosesin özünü özündən qorumaq üçün də istifadə edilir. Məsələn, proqram kodu seqmentini təsadüfi yazılardan qoruyurlar. Prosesləri bir-birindən qorumaq üçün, sadəcə, bütün mümkün ünvanları tutmaq və beləliklə, kənar yaddaşa müraciət ehtimalını aradan qaldırmaq kifayətdir. Prosesin daxilində qorunmaq üçün bu texnika işləmir, çünki həmişə proqramın koduna və verilənlərinə müraciət etmək tələb olunur. Lakin hər bir seqmentin öz xüsusiyyəti var: kod seqmentini yalnız oxumaq və yerinə yetirmək kifayətdir, verilənlər seqmentinə yalnız oxumaq və yazmaq (amma yerinə yetirmə üçün yox!) üçün müraciət olunur.

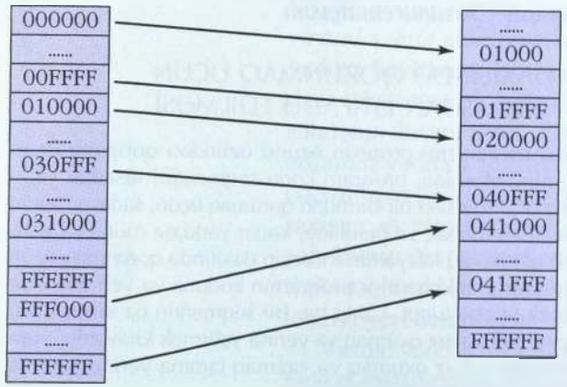
Seqmentə müraciət edilməsinə həddqoymaq üçün seqmentin idarəedici informasiyanın xüsusi baytında bitlər – müraciət əlamətləri saxlanılır: oxuma, yazma və yerinə yetirmə. Təbii ki, prosessor bu bitləri tanımalı və müraciətləri yoxlamalıdır. Beləliklə, kod seqmentinin səhifələri üçün "oxuma", "yerinə yetirmə" adlı müraciət bitləri, verilənlər və stek seqmenti səhifələri üçünsə "oxuma" və "yazma" adlı müraciət bitləri qoymaq olar. Bu halda, proqram verilənləri təsadüfən kod sahəsinə yaza bilməz və kod seqmentində kodun yerləşdiyini müəyyən edəcək təsadüfən verilənləri kodun əvəzinə yerinə yetirə bilməz, çünki bu halda proqram işləməyəcək.



VİRTUAL YADDAŞ MEXANİZMİNİN İŞİNƏ DAİR NÜMUNƏ

Misalın sadəliyi üçün tutaq ki, kompüterdə 16 Mbayt yaddaş vardır. Onda yaddaşın sonuncu baytı onluq say sistemində 16777215 (16x1024x1024 – 1), yaxud onaltılıq say sistemində 0xFFFFFF ("x" hərfi onu göstərir ki, ədəd onaltılıq say sistemində yazılıb) olan fiziki ünvana malikdir. Bu halda fiziki ünvan sahəsi belə görünür: 0x000000 – 0xFFFFFF (yəni 0-dan 0xFFFFFF-ə qədər olan ünvanlar dəsti).

Virtual ünvan sahəsi də 0x000000-dan 0xFFFFFF-ə qədər (həcmi 16 Mbayt-dır) nömrələnir. Onda virtual ünvanlara aşağıdakı qaydada fiziki ünvanlar qarşı qoyulur:



• 0x000000-dan 0x00FFFF-ə qədər olan virtual ünvanlar 0x010000-dan 0x01FFFF-ə qədər olan fiziki ünvanlara uyğundur (yəni 0 virtual ünvanı 0x10000 fiziki ünvanına, 1 virtual ünvanı 0x10001 fiziki ünvanına və s., nəhayət, 0xFFFF virtual ünvanı 0x1FFFF fiziki ünvanına uyğundur);

• 0x010000 – 0x030FFF virtual ünvanları 0x020000 – 0x040FFF fiziki ünvanlarına uyğundur;

• 0xFFF000 – 0xFFFFFF virtual ünvanları 0x041000 – 0x041FFF fiziki ünvanına uyğundur;

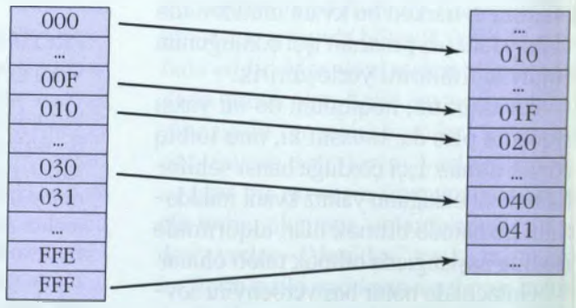
• 0x031000 – 0xFFEFFF virtual ünvanları heç bir fiziki ünvana uyğun deyildir.

İndi prosessor, məsələn, 0x025367 virtual ünvanlı baytı oxuya bilər. Qaydalara görə, bu virtual ünvana 0x35367 fiziki ünvanı və müvafiq olaraq 217961-ci yaddaş baytı uyğun gəlir ($35367_{16} = 217961_{10}$).

İndi isə virtual ünvana görə hesablanıb tapılmalı olan fiziki ünvanı idarə edən dispetçer üçün qaydanı yazmaq lazımdır, həm də mümkün qədər daha yığcam yazmaq lazımdır, çünki dispetçer bu qaydanı hardasa yadda saxlamalıdır. Əgər hər bir virtual ünvan üçün cədvəldə ona uyğun fiziki ünvan saxlanılarsa, onda bunun üçün 48 Mbayt yaddaş = 16 Mbayt yaddaş x 3 bayt (ünvanın ölçüsü) tələb olunacaq; bu isə yaddaşın həcmindən üç dəfə (!) çoxdur.

Ona görə də yaddaşın 4 kbaytlıq səhifələrə bölünməsindən (onları 0-dan başlayaraq nömrələməklə) istifadə edilir. Ünvanı bilərək səhifənin nömrəsini asanlıqla hesablamaq olar: 16-lıq say sistemində sonuncu üç rəqəmi atmaq lazımdır. Məsələn, onaltılıq say sistemində 0x35367 ünvanı onluq say sistemində 3516-cı, və yaxud 5310-cu səhifədə yerləşir.

Səhifələrlə yaddaşın uyğunluq qaydaları aşağıdakı kimi olacaq:



• 0x000 – 0x00F virtual səhifələri 0x010 – 0x01F fiziki səhifələrinə uyğundur;

• 0x010 – 0x030 virtual səhifələri 0x020 – 0x040 fiziki səhifələrinə uyğundur;

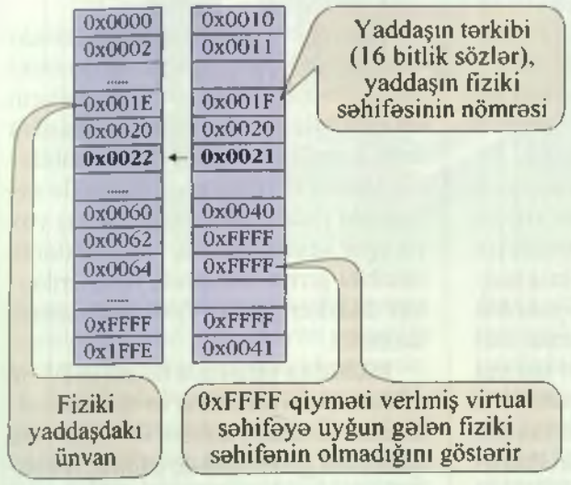
• 0xFFF virtual səhifəsi 0x041 fiziki səhifəsinə uyğundur;

• 0x031 – 0xFFE virtual səhifələri heç bir fiziki səhifəyə uyğun deyildir.

Bu halda səhifələrin uyğunluğunu yadda saxlamaq üçün yaddaşda birölçülü cədvəl (massiv) kifayətdir.

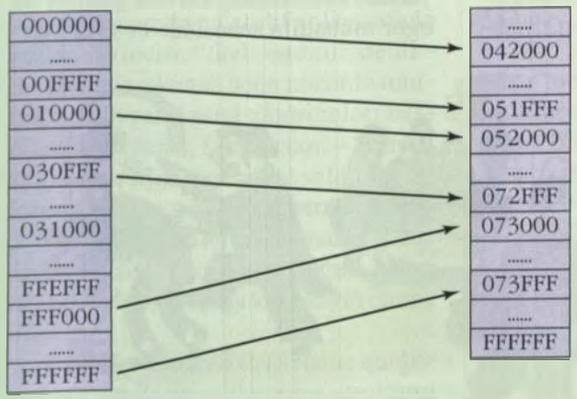
Tutaq ki, cədvəl EHM-in fiziki yaddaşının başlanğıcında yerləşir. Verilən virtual səhifə üçün ona uyğun fiziki səhifənin ünvanını hesablamaq üçün virtual səhifənin nömrəsini (səhifənin nömrəsi 2 baytla kodlaşdırılır) ikiyə vurmaq lazımdır.

Məsələn, prosessor 0x113A5 virtual ünvanı üzrə baytı oxumalıdır, bu halda axırdan üç rəqəmi atmaqla virtual səhifənin nömrəsi alınır: 0x11. 0x11-i ikiyə vuraraq 0x22 almaqla cədvəldə bu ünvan üzrə qiyməti oxumaq olar və nəticədə fiziki ünvanın nömrəsi alınır – 0x21. Sonra yenidən virtual ünvanın üç son rəqəmini bu ünvana əlavə etməklə, fiziki ünvanı alırıq ki, buna əsasən də baytı oxumaq lazımdır – 0x213A5.



Yaddaşdakı cədvəl cəmi 8 kbayt və ya 16 Mbayt-lıq yaddaşın cəmi 0,5%-ni tutur.

İndi isə iki prosesə – *A* və *B*-yə baxaq. Onların hər birinin xüsusi virtual ünvan sahəsi vardır: 0x000000 – 0xFFFFFF. Yalnız hər iki prosesin ünvan sahəsi üçün fiziki yaddaşın müəyyən edilməsi qalır. Tutaq ki, *A* prosesinin ünvan sahəsi artıq məlum olan cədvəldən istifadə edir. *B* prosesi üçün yaddaş ayırarkən, ona diqqət yetirmək lazımdır ki, sərbəst fiziki yaddaşın iki sahəsi vardır: müvafiq olaraq 0x000000 – 0x00FFFF və 0x042000 – 0xFFFFFF ünvanlarından ibarət yaddaş sahələri. Birinci yaddaş sahəsi *A* prosesinin səhifələr cədvəlinin yerləşdirilməsi üçün istifadə edilir. İkinci yaddaş sahəsində isə *B* prosesi yerləşəcək:



• 0x000000 – 0x00FFFF virtual ünvanları 0x042000 – 0x051FFF fiziki ünvanlara uyğun olacaq;

• 0x010000 – 0x030FFF virtual ünvanları 0x52000 – 0x072FFF fiziki ünvanlara uyğun olacaq;

• 0xFFF000 – 0xFFFFFF virtual ünvanları 0x073000 – 0x073FFF fiziki ünvanlara uyğun olacaq;

• 0x031000 – 0xFFEFFF virtual ünvanlara hec bir fiziki ünvan uyğun olmayacaq.

B prosesinin səhifələr cədvəlini yerləşdirmək üçün yenə də 8 kbayt yaddaş tələb olunur. Prosesin virtual yaddaşının üç ünvan dəstindən (onları həm də seqment adlandırırlar) istifadə edilməsi təsadüfi deyil. Məsələ ondadır ki, adətən, hər bir proses-proqramın yaddaşı aşağıdakı qaydada bölünür:

• prosessor tərəfindən yerinə yetirilən komandalar üçün yaddaş – *kod seqmenti;*

• dəyişənlər üçün yaddaş – *verilənlər seqmenti;*

• altproqramların müvəqqəti dəyişənlərinin, altproqramlardan qayıdış ünvanlarının saxlandığı stek üçün yaddaş – *stek seqmenti*.

Misaldakı prosesin virtual sahəsi uyğun olaraq üç seqmentdən ibarətdir:

0x000000 – 0x00FFFF – kodlar seqmenti;

0x010000 – 0x030FFF – verilənlər seqmenti;

0x031000 – 0xFFEFFF – yaddaş sahəsində istifadə edilməyən "deşik";

0xFFF000 – 0xFFFFFF – prosesin stek seqmenti.

Verilənlər seqmenti, proqramın yerinə yetirilməsi prosesində dəyişilə bilər, çünki yeni verilənlər üçün yaddaş tələb olunur. Verilənlər seqmentini kodlar seqmentindən (dəyişməyən seqment) dərhal sonra, stek seqmentini isə ünvan sahəsinin sonunda yerləşdirmək olar.

* * *

Praktikada səhifələr cədvəli daha mürəkkəb görünür. Hazırkı prosessorların çoxu 32 mərtəbəlidir. Bu, onu göstərir ki, ünvan 32 mərtəbədən, yaxud 4 baytdan (bir baytda 8 mərtəbə var) ibarətdir. Müvafiq olaraq, 32 mərtəbəli prosessorun işləyə bildiyi virtual ünvan sahəsi 2^{32} bayt uzunluğa, yaxud 4 Gbayt həcmə malikdir. Lakin fiziki yaddaşın həcmi daha böyük olmalıdır. Məsələn, 32 mərtəbəli "Intel Pentium III" prosessorunda fiziki ünvanın uzunluğu 36 mərtəbədir. 64 bitlik prosessorların imkanları daha böyükdür, virtual ünvan sahəsinin uzunluğu 2^{64} bayt və ya 16 Pbaytdır. Bəşəriyyət hələ bu qədər fiziki yaddaşla nə etmək lazım gəldiyini müəyyən etmədiyindən, fiziki ünvanın real mərtəbəsi hələ aşağıdır. 64 mərtəbəli "Alpha" prosessorunun fiziki ünvanı 48 mərtəbədən ibarətdir.

yerinə yetirməyə başlayır. Belə halda bu səhifələrin qorunmasının pozulması üzrə kəsilmə olacaqdır. Emalçı əvvəlcə səhvin baş verib-vermədiyini, virtual səhifə ilə hər hansı bir fiziki səhifə arasında uyğunluğun yaradılmalı olub-olmadığını aydınlaşdırmalıdır. Əgər lazımdırsa emalçı fiziki səhifəni ayırır və cədvəldə uyğun dəyişikliyi aparır.

Ən köhnə səhifəni müəyyən etmək üçün yaxşı olardı ki, prosessor virtual səhifəyə müraciət edərkən yaddaşın müəyyən bir hissəsinə, EHM-in qoşulması anından başlayaraq yerinə yetirilən komandalarla bərabər hesablanan cari zamanı da yazardı. Əməliyyat sistemi müəyyən zaman intervalından sonra bütün proseslərin səhifələr cədvəlini nəzərdən keçirir və fiziki səhifəni azad edir. Praktikada səhifəyə müraciətə sərf olunan bütün vaxtı yadda saxlamaq olduqca baha başa gəlir və əməliyyat sistemi effektivliyi az olan alqoritmlərdən istifadə edir.

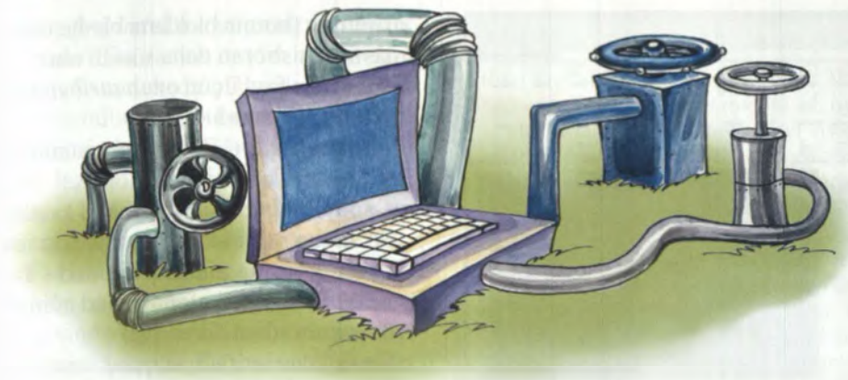
Yaddaşın bölüşdürülməsi alqoritminin işinin nəticəsində fiziki səhifələr azad olunur. Belə səhifələrdə yerləşən verilənlərə yenə də tələbat ola bilər və buna görə də informasiyanı yadda saxlamaq lazımdır. Bir qayda olaraq, EHM-dəki diskdə azad edilmiş fiziki səhifələrin tərkibini yadda saxlamaq üçün yer ayırmaq olar. Diskdəki həmin yer *mübadilə sahəsi* adlanır. Azad edilmiş səhifələrin tərkibinin yadda saxlanması *səhifələrin mübadiləsi* adlanır. Bu prosesi ifadə etmək üçün bəzən ədəbiyyatda ingilis dilindən gələn "svopinq" termini işlədilir (*ing.* swapping – "mübadilə").

Səhifələrin mübadiləsindən istifadə edərkən yaddaşın paylanması alqoritmi bir qədər mürəkkəbləşir. Kəsilmələrin emalçısında fiziki səhifəni ayırmazdan öncə aydınlaşdırmaq lazımdır ki, səhifənin tərkibi bundan əvvəl mübadilə sahəsində yadda saxlanmışdır, yoxsa yox və əgər saxlanmışdısa, onda səhifənin tərkibini geriyə köçürmək, yəni, verilənləri diskdən geriyə – yaddaşa oxumaq lazımdır.

Mübadilə sahəsinin həcmi EHM-in yüklənmə dərəcəsindən və operativ yaddaşın həcmindən asılıdır. Əvvəllər belə düsturdan geniş istifadə edilirdi: mübadilə sahəsinin həcmi fiziki yaddaşın həcmindən iki dəfə böyükdür. İndi hətta fərdi EHM-lərin fiziki yaddaşı bir neçə giqabayta çatır. Ona görə də mübadilə sahəsinin həcmi yaddaşın həcmilə eyni, yaxud ondan da az götürülür.

Əgər mübadilə sahəsində boş yer yoxdursa, onda nə etməli? Bu problemin iki həll üsulu mövcuddur. Bunlardan konservativ üsul adlanan birinci üsul ondan ibarətdir ki, yaddaşın növbəti porsiyasını ayırarkən mübadilə sahəsində ehtiyat yer saxlamaq, əgər bu yer yoxdursa, onda yaddaş ayırmamaq. Bu onu göstərir ki, əgər mübadilə sahəsində boş yer yoxdursa, əməliyyat sistemində yeni prosesi başlatmaq olmaz, artıq işləyən proseslər isə bu zaman əlavə yaddaş almayacaq.





Praktikada bu ona gətirib çıxarır ki, mübadilə sahəsindən səmərəli istifadə edilmir: mübadilə sahəsində boş yerlər mövcuddur, lakin ondan istifadə etmək olmaz, çünki ehtiyat üçün saxlanılır.

İkinci üsul ondan ibarətdir ki, nə qədər ki, fiziki yaddaşda və mübadilə sahəsində boş yer var, əməliyyat sistemi öz üzərinə "yüksək öhdəliklər" götürür və virtual yaddaş ayırır (ingilis dilində bu strategiya overcommitment adlanır). Əgər mübadilə sahəsində boş yer yoxdursa, arzuolunmaz əməliyyat yerinə yetirmək lazım gələcək: yeri məcburi qaydada boşaltmaq, bu isə hər hansı bir prosesi yarıda məhv etmək deməkdir.

Əvvəlki təcrübə əsasında gələcəkdə nə baş verəcəyini xəbər verməyə cəhd göstərən başqa alqoritmləri də işləyib hazırlamaq mümkündür. Məsələn, yaddaşı ən "köhnə" səhifədən götürmək əvəzinə, ən az istifadə edilən səhifədən götürmək olar (Bunlar eyni şey deyil!)

Yaddaşın paylanması alqoritmləri sahəsində daima tədqiqatlar aparılır, burada həmişə təkmilləşdirmə imkanı var.

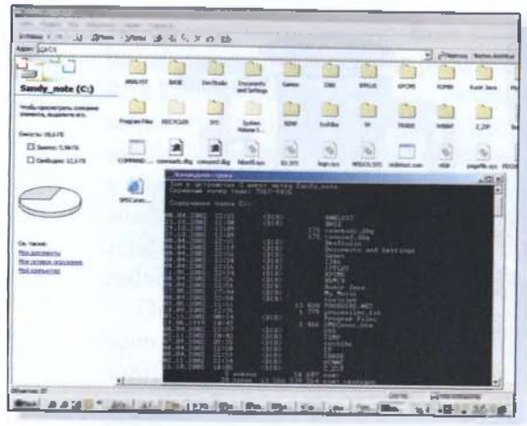
FAYL SİSTEMİ

Əməliyyat sisteminin daha bir mühüm hissəsi *fayl sistemidir*. Bu termin vasitəsilə iki tamamilə müxtəlif anlayış ifadə edilir. Birincisi, "fayl sistemi" dedikdə fayllarla işləmək üçün nəzərdə tutulan əməliyyat sistemi alqoritmləri başa düşülür. İkincisi, fayl sistemi – yaddaşsaxlama qurğularında, məsələn, disklərdə faylların saxlanması strukturudur.

Ən ümumi (və ən anlaşılmaz) şəkildə *faylı* yazıların nizamlanmış yığını, *yazını* isə baytların ardıcıllığı kimi təyin etmək olar.

Fayl sisteminə yaddasaxlama qurğularında faylların saxlanması strukturu kimi baxmaq daha münasibdir. Məsələn, diskə *bloklar* massivi kimi, hər bir bloka isə müəyyən uzunluqlu, adətən, 512 bayt götürülən baytlar massivi kimi baxılır. Yəni diskin hər bir baytına birbaşa müraciət etmək olmaz: yalnız blokların ünvanlarına müraciət oluna bilər. Blokun ünvanını 32-bitlik ədəd hesab etmək olar. Bu uzunluqlu ünvan-

Yazı anlayışı vacib anlayışdır, çünki fayllarla məhz yazı terminində işləyirlər. Məsələn, mətn faylları ilə simvollar sətrinin yığınları ilə işlədikləri kimi işləyirlər; mətn faylında yazı – sətirdir. Videofayl kadrlardan ibarətdir, ona görə də videofaylda yazı – kadrdır. Bəzən faylla baytların ardıcıllığı kimi (bu halda tez-tez *baytlar axını* terminindən istifadə edilir) işləmək daha əlverişlidir, bu zaman hər bir baytı ayrıca yazı hesab etmək olar.

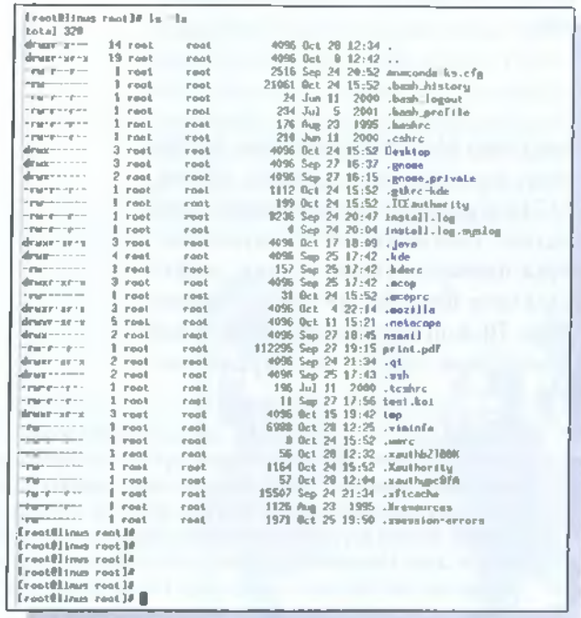


Windows-un fayl sistemi.

lar, həcmi $2^{32} \cdot 512 = 2$ Tbayta qədər yaddaşı olan disklə işləmək üçün kifayətdir.

Müasir fayl sistemlərində ölçüsü *fiziki blokun* ölçüsünün bir neçə mislinə bərabər olan *məntiqi blok* anlayışından istifadə edilir. Məsələn, hazırda məntiqi blokun ən geniş yayılmış ölçüsü 1 kbaytdır, həmçinin 4 və 8 baytlıq ölçülərə də rast gəlmək mümkündür. Məntiqi blok anlayışının daxil edilməsinin vacibliyi diskin xüsusiyyətilə izah olunur – diskdən bir-birinin ardınca gələn blokların oxunması (həmin blokların bir-bir oxunmasına) nisbətən daha sürətli olur.

UNIX-in fayl sistemi.



Hər bir fayl üçün onun *atributlarını* saxlamaq lazımdır:

- faylın adı (UNIX fayl sistemində bu, sadəcə, ədəddir);
- faylın tipi: adi fayl, yaxud kataloq;
- faylın sahibi kimdir və o hansı istifadəçi qrupuna aiddir (iki ədəd – fayl sahibinin identifikatoru, yaxud nömrəsi və qrupun identifikatoru);
- fayldan istifadə hüququ – oxumaq, yazmaq və yerinə yetirmək üç istifadəçi kateqoriyası üçün: – faylın sahibi, faylın sahibilə eyni qrupdan olan istifadəçi və digər istifadəçilər (9 bit: üç kateqoriyadan olan istifadəçi üçün hər biri üç bit olmaqla qorunma informasiyası);
- faylla işləmək haqqında informasiya – faylın yaradıldığı tarix, onda edilən sonuncu dəyişiklik və ondan sonuncu istifadə tarixi (1970-ci ilin yanvar ayının 1-dən cari ana qədər hesablanan saniyələrin sayı kimi, inteqrasiya olunan 3 böyük ədəd, yaxud hərdən deyildiyi kimi, Dövrün başlanğıcından – bu tarix UNIX-in yarandığı tarix sayılır). 32-bitlik ədəddə 1970-ci ildən 2038-ci ilə qədər olan müddəti kodlaşdırmaq olar);
- fayla istinadların sayı, yaxud başqa sözlə, faylın simvolik adlarının miqdarı;
- faylın yerləşdiyi blokların nömrəsi – fayl bloklarının siyahısı;
- faylın yerləşdirildiyi blokların nömrələri – faylın bloklarının sayı.
- faylın baytlarla ölçüsü, çünki blokların siyahısı kifayət deyildir; faylın ölçüsü blokun ölçüsünə mütənasib olmaya bilər.

Blokların siyahısı istisna olmaqla, bütün atributlar elə də çox yer tutmur – cəmi 32 bayt. Ona görə də diskdə qeyd edilən həcmdə sahə ayırmaq və həmin sahədə eyni zamanda bütün fayllar haqqında informasiya saxlamaq olar. Bu sahə *indekslər massivi* adlanır. Müvafiq olaraq, *fayl indeksi* – onun atributlarının yığınıdır. Bu zaman faylın adı həmin massivdə indeksdir. Diskdə faylların sayı məhduddur və bunu diskdə fayl sistemini yaradarkən əvvəlcədən bilmək lazımdır. Lakin sadəcə olaraq, kifayət qədər böyük ədəd seçmək və bunun yetərli olacağına ümid etmək olar.

Faylın adına görə axtarılması adi əməliyyatdır, çünki faylın adı indeks massivində fayl indeksinin nömrəsidir. Amma istifadəçi faylların simvollu adları ilə işləyir və ona görə də faylın simvollu adı ilə indeks nömrəsi arasında uyğunluğu saxlamaq lazımdır. Ən sadəsi bu informasiyanı kataloqda yerləşdirmək-dir. Kataloqda da yalnız bu informasiya saxlanılacaq. Yəni kataloq simvollu ad və indeks nömrəsi şəklində olan yazılardan ibarət fayldır. Qaldı xüsusi kataloqu – *iyerarxiyanın başlığını* ayırmaq, çünki fayl sistemi iyerarxiyalı struktura malikdir. İyerarxiyanın başlığını *fayl sisteminin kökü* adlandırırlar. Onun üçün xüsusi indeks (məsələn, birinci) və xüsusi ad ayırmaq olar. UNIX-in fayl sisteminin kökünün ənənəvi adı belədir – "/". İndi fayl sistemində istənilən fayla onun simvolik adı ilə müraciət etmək olar.

Məsələn, "/MənimSənədlərim/Müxtəlif/MənimFaylım.txt" adlanan faylı tapmaq üçün aşağıdakıları etmək lazımdır:

1) fayl sisteminin kökünün bloklarını taparaq, birinci indeksi oxumaq;

2) yazı tapılana qədər "MənimSənədlərim" simvollu adının yerləşdiyi kökün bloklarını oxumaq. Tutaq ki, bu ada 17 nömrəli indeks uyğundur;

3) bu kataloqun bloklarını taparaq, 17 nömrəli indeksi oxumaq;

4) "Müxtəlif" simvollu adının yerləşdiyi yazı tapılana qədər 17 nömrəli faylın bloklarını oxumaq. Tutaq ki, bu ada 103 nömrəli indeks uyğundur;

5) bu kataloqun bloklarını taparaq 103 nömrəli indeksi oxumaq;

ADLAR

Bəzi fayl sistemlərində, məsələn, MS DOS-un və ya Windows-un FAT-ında faylın adı adi simvollu ad, məsələn, MənimFaylım ola bilər. UNIX əməliyyat sisteminin fayl sistemlərində faylın adı – sadəcə, ədəddir, yəni bütün fayllar nömrələnmişdir. Bu zaman, elə simvollu adlar da var ki, onlar həmin faylın adına istinad rolunu oynayır və bu cür istinadlar istənilən qədər ola bilər. MənimFaylım və SəninFaylın kimi simvollu adlar eyni adlara istinad edə bilər. RSX və VMS fayl sistemlərində fayl tam şəkildə simvollu adla, uzantı və versiya ilə müəyyən olunurdu. Yəni MənimFaylım.txt;1, MənimFaylım.txt;2 şəklində olan fayllar nəzərdə tutulur. Burada MənimFaylım – ad, txt – uzantıdır, 1 və ya 2 – versiyadır. İndi uzantılı və versiyaya oxşar adlara CD-ROM-disklərində faylların saxlanması üçün istifadə edilən ISO9660 fayl sistemində rast gəlmək olar.

Müasir fayl sistemləri *iyerarxiyalı struktura* malikdir. Fayllar *kataloqlarda* və ya *qovluqlarda* saxlanılır. Ona görə də faylı əldə etmək üçün yalnız onun haqqında informasiyaya malik olmaq kifayət deyil: fayla çatmaq üçün keçilməli olan bütün kataloqların siyahısını əks etdirən *marşrut adını* bilmək lazımdır. Məsələn, MS DOS-un FAT-ında faylın marşrut adı belə görünür:

D:\MənimSənədlərim\Müxtəlif\MənimFaylım.txt

Burada D – diskdir, bu diskdə MənimFaylım.txt adlı fayl yerləşir; MənimSənədlərim – qovluğun adıdır, onun içərisində Müxtəlif adlı qovluq (bu qovluqda isə lazım olan fayl) yerləşir; \-simvolu marşrut adının elementlərini bir-birindən ayırır.

UNIX ƏS-nin fayl sistemində həmin marşrut adı belə yazılacaq:

/mnt/disk/MənimSənədlərim/Müxtəlif/MənimFaylım.txt

Burada /mnt/disk/ – diskin fayl sisteminin başlandığı kataloq; /-simvolu – marşrut adının elementlərini bir-birindən ayıran simvoldur.

Nəhayət, VMS fayl sistemində marşrut adı aşağıdakı qaydada yazılır:

DISK$D:[MƏNİMSƏNƏDLƏRİM.MÜXTƏLİF]MƏNİMFAYLIM.TXT;1

Burada DISK$D – faylın yerləşdiyi diskin adıdır; "[" və "]" simvolları arasında fayla gedən yolda rast gələn kataloqların siyahısı yerləşdirilir; "." simvolu kataloqları bir-birindən ayırır.



6) "MənimFaylım.txt" simvolik adının yerləşdiyi yazı tapılana qədər 103 nömrəli faylın bloklarını oxumaq. Tutaq ki, bu ada 356 nömrəli indeks uyğundur;

7) bu faylın bloklarını taparaq 356 nömrəli indeksi oxumaq;

8) tələb olunan faylın bloklarını oxumaq.

UNIX ƏMƏLİYYAT SİSTEMİNDƏ FAYLLAR

Yanaşma ondan ibarətdir ki, fayl sistemində faylın səciyyəvi uzunluğunu götürüb diskin həcmindən asılı olaraq lazım olan sayda indeks yaratmaq. Məsələn, müasir UNIX əməliyyat sistemlərində faylın ənənəvi uzunluğu 10 kbaytdır. Deməli, fayl sisteminin hər on məntiqi blokuna (əgər məntiqi blokun ölçüsü 1 kbayt təşkil edirsə) bir indeks ayırmaq lazımdır.

Blokların siyahısı ilə bağlı məsələ daha mürəkkəbdir. İstifadəçi faylın hər bir baytını ünvanlamağı bacarmalıdır. Faylda ünvanlama ənənəvi olaraq oxuma/yazma göstəricisinin cari mövqeyinə nəzərən sürüşmə kimi qurulmuşdur; həm irəli, həm də geriyə sürüşmə mümkündür, buna görə də ünvan 32-mərtəbəli işarəli ədəd olacaq. Faylın maksimal həcmi isə 2 Gbayt – düz 32-mərtəbəli ədəd vasitəsilə ünvanlamanın mümkün olduğu qədər olacaq.

Blokların siyahısında 2 milyona qədər element ola bilər (bloklar siyahısında elementlərin sayı = faylın həcmi/blokun ölçüsü = 2 Gbayt/1 kbayt = $2 \cdot 2^{20}$), siyahının saxlanması üçünsə 8 Mbayt lazımdır ($2 \cdot 2^{20}$ blokun 32-mərtəbəli ünvanında 4 bayt = 8 Mbayt).

Bir neçə ilk blokların nömrələri birbaşa faylın indeksində saxlanılır. Çünki hesab edilir ki, tipik fayl on blok yer tutur, ona indeksdə on blok üçün yer ayrılır. Sonra *dolayılıq bloku* anlayışı (səhifələr kataloquna analoji olaraq) tətbiq edilir. Bu blokda faylın yazısı yox, blok nömrələrinin siyahısı saxlanılır. *Dolayılıq* bloku indeks faylında 11-ci blok olacaq. Blokun ölçüsü 1 kbayt, ünvanın ölçüsü isə 32 bit həcmində qəbul olunduğundan *dolayılıq* blokunda cəmi 256 blokun nömrəsini saxlamaq olar. Cəmi 266 blok və ya faylın ilk 266 kbaytı.

Ümumiyyətlə, bir *dolayılıq* blokunda 256 kbaytlıq fayl haqqında informasiya saxlamaq olar. 2 Gbayt ölçüsündə olan faylın saxlanması üçün bizə 8 min *dolayılıq* bloku ($2 \cdot 2^{20}$ blok siyahıda 256 blok = $8 \cdot 2^{10}$ blok dolayılıq blokunda) lazımdır ki, bu da indeksdə 32 bayt tələb edir. Bu, sadəcə, blok nömrələrinin saxlanması üçün tələb olunan 8 Mbayt-dan azdır, lakin eyni zamanda, hər bir indeksdə bütün dolayılıq bloklarının saxlanması üçün tələb ediləndən hələ çoxdur.

İkinci dolayılıq bloku birinci dolayılıq blokunun nömrələrini (onlar artıq faylın bloklarının nömrələrini özündə saxlayırlar) özündə saxlayır və faylın indeksində 12-ci blokdur. İkinci dolayılıq blokunda birinci dolayılıq blokunun ilk 256 nömrələrini saxlamaq olar. Cəmi $10 + 256 + 256^2 = 65\,802$ blok və ya 64 Mbaytdan bir az çox. 2 Gbaytlıq həcmi yadda saxlamaq üçün bizə 32 ikinci vasitəlilik bloku lazım olacaq (siyahıda $2 \cdot 2^{20}$ blok/ ikinci dolayılıq blokunda 256^2 blok = 32 ikinci dolayılıq bloku) bunun üçün isə indeksdə 128 bayt tələb olunur. Üçüncü dolayılıq bloku özündə ikinci dolayılıq blokunun nömrələrini saxlayır. O, indeksdə 13-cü blok olacaq. Üçüncü dolayılıq blokunda 256-ya qədər ikinci dolayılıq bloku saxlanıla bilər. Cəmi, $10 + 256 + 2562 + 2563 = 16\,843\,018$ blok olacaq, yəni faylın maksimal ölçüsü 16 Gbayt təşkil edə bilər. Üçüncü dolayılıq bloku aşkar şəkildə 2 Gbayt həcmində olan faylın saxlanması üçün tamamilə kifayətdir. İndeksdə blokların siyahısının strukturu aşağıdakı kimi alınır:

Faylın ilk on blokunun nömrəsi.

Birinci dolayılıq bloku; → faylın növbəti 256 blokunun nömrəsi.

İkinci dolayılıq bloku; → birinci dolayılıq blokunun növbəti 256 nömrəsi; → Növbəti 256^2 bloklarının nömrələri.

Üçüncü vasitəlilik bloku; → 256 ikinci dolayılıq blokunun nömrələri; → 256^2 birinci dolayılıq bloklarının nömrələri; → növbəti 256^3 fayl blokularının nömrələri.

Blok nömrələrini saxlamaq üçün indeksdə 52 baytı ehtiyat üçün saxlamaq lazımdır (13 blok nömrəsi x 32 bit = 52 bayt). İndeksin ümumi ölçüsü 84 baytdır.

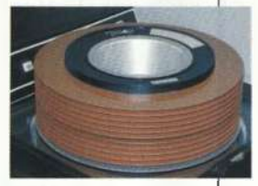
Lakin böyük fayllarla işləyərkən kompakt indekslər məhsuldarlığın azaldılması hesabına başa gəlir. Həqiqətən də, faylın 11-ci kilobaytını oxumaq üçün iki bloku oxumaq lazımdır – blokun nömrəsini öyrənmək üçün birinci dolayılıq blokunu və blokun özünü. Birinci meqabaytdan başlayaraq, hər bir bloku oxumaq üçün üç bloku – ikinci dolayılıq blokunu, birinci dolayılıq blokunu və blokun özünü oxumaq tələb olunur. Nəhayət, yüzüncü meqabaytdan başlayaraq bloku oxumaq üçün dörd bloku – üçüncü dolayılıq blokunu, ikinci dolayılıq blokunu, birinci dolayılıq blokunu və blokun özünü oxumaq tələb olunur.

Müasir disklərin həcmi onlarla gigabaytdan ibarətdir; əgər hər 10 kbayta bir indeks ayrılsa, bizə 1 milyon indeks gərək olacaq, müvafiq olaraq, indeks massivinin ölçüsü 84 Mbayt, yaxud diskin həcminin 1%-dən az olacaq ki, bu da o qədər də çox deyildir.



Bu alqoritmə diqqətlə baxsaq, görmək olar ki, fayl sistemində istənilən fayl üçün diskdən eyni məlumatları oxumaq lazımdır. Həqiqətən, hər bir fayl üçün alqoritmin birinci addımını yerinə yetirmək lazım gələcək. İstifadəçinin faylları, bir qayda olaraq, yalnız ona məxsus olan kataloqda yerləşir, misalımızda bu "MənimSənədlərim" kataloqudur, ona görə də tez-tez 2-ci addımı yerinə yetirmək lazım gələcək. Diskdən oxuma olduqca ləng əməliyyatdır, buna görə də haqlı olaraq tez-tez istifadə edilən faylların axtarılması və oxunmasının sürətləndirilməsinə tələbat yaranır.

Verilənlərin keşləşdirilməsi ideyasından istifadə etmək və onları fayllara tətbiq etmək olar. Yaddaşda faylların tez-tez istifadə olunan blokları üçün yer – *bufer keşi* ayrılır. Bufer keşilə iş alqoritmi (blokların sıxışdırılması alqoritmi) yaddaşın bölünməsi alqoritmilə çox oxşardır. Əgər ən "köhnə" yaddaşdan səhifəni ləğv edən alqoritm bufer keşinə tətbiq edilsə, onda ən "köhnə" blok sıxışdırılıb çıxarılacaq. Lakin daha effektiv həll keşdən ən az istifadə olunan bloku çıxarmaqdır. Bu alqoritmin, əlbəttə, çatışmayan cəhətləri də var, lakin bütün alqoritmlərin əsas problemi elə budur, onlar zaman-zaman təkmilləşməkdədir və bununla da barışmaq lazım gəlir. Blokları diskə yazılarkən də buferləşdirmək olar. Blokların diskə yazılması əməliyyatı oxunma əməliyyatına nisbətən daha ləngdir. Bu, avadanlığın xüsusiyyətlərilə bağlıdır. Birincisi, müasir disklər blokları özləri keşləyir və oxunma diskin keşindən yerinə yetirilə bilər. İkincisi, diskdən oxuyarkən verilənlər yaddaşa yazılır, diskə yazarkən isə yaddaşdan oxunur. Necə qəribə görünsə də, yaddaşdan oxunma əməliyyatı yaddaşa yazma əməliyyatından daha ləngdir. Diskə yazılmanı da keşləmək mümkündür. Amma burada ehtiyatlı olmaq lazımdır: diskə yazılan bloklar keşlənərkən EHM-də hər hansı nasazlıq yaranarsa, onları itirmək riski çox yüksəkdir. Ola bilər ki, onlar keşdə qalsınlar və diskə yazılmasınlar. Problem ondan ibarətdir ki, istifadəçi məntiqlə hesab edə bilər ki, onun verilənləri artıq diskdə yadda saxlanılıb və onları orada tapmayanda çox təəccüblənəcək. Ona görə də bir sıra əməliyyat sistemləri, ümumiyyətlə, yazma əməliyyatını keşləmir. Əməliyyatın yavaş, lakin etibarlı olmasına üstünlük verilir. UNİX əməliyyat sistemi ənənəvi olaraq yazı zamanı blokları keşləyir, ancaq müəyyən bərabər vaxt intervallarında (adətən, 30 saniyədən bir) bütün blokları məcburi qaydada diskə yazır. Bu alqoritmdən 30 ildən çox istifadə edilmə təcrübəsi onun istifadəçilərin əksəriyyəti üçün məqbul olmasını göstərdi.



Maqnit disk.

ƏGƏR ONLARIN SAYI ÇOX OLARSA?

Müasir əməliyyat sistemlərində eyni vaxtda bir neçə fayl sistemilə işləmək tələb olunur. Məsələn, diskdə UNIX və ya Windows əməliyyat sistemilə işləyərkən NTFS fayl sistemi yerləşdirilir, çünki həmin sistemlər xüsusi olaraq disklərdəki fayllarla səmərəli şəkildə işləmək üçün yaradılmışdır, yaxud, necə deyərlər, sürətə görə optimallaşdırılmışdır. CD-ROM disklərilə işləmək üçün faylları daha səmərəli şəkildə saxlamağa imkan verən, yəni saxlanma sahəsinə görə optimallaşdırılmış ISO9660 fayl sistemindən istifadə edilir. Disketlər, məlum olduğu kimi, vaxtilə MS DOS əməliyyat sistemində istifadə edilən FAT fayl sisteminə malikdir. Hamı, sadəcə, adət edib ki, disket FAT-la işləyir. Bundan başqa, lokal şəbəkələrə qoşulan və ya İnternet vasitəsilə qoşulmaq mümkün olan digər EHM-lərin disklərilə işləməyə imkan verən NFS fayl sistemi də mövcuddur.

Fayl sistemləri arasında faylların necə adlandırılmasına və yaxud necə deyərlər, ayrı-ayrı ad sahələrinə dair müxtəlif razılaşmalar mövcuddur. İstifadəçi, təbii ki, vahid ad sahəsilə işləmək, yəni istifadə etdiyi fayl sisteminin tipi haqqında düşünmədən faylları adət etdiyi qaydada adlandırmaq istəyərdi. Əlbəttə, əməliyyat sistemi adların dəyişdirilməsi və onların vahid formaya salınması işini öz üzərinə götürür. Adların dəyişdirilməsi işinə görə əməliyyat sisteminin xüsusi hissəsi – virtual fayl sistemi cavabdehdir. Öz mahiyyətinə görə, virtual fayl sistemi əməliyyat sistemləri və konkret fayl sistemi strukturunda işləyən alqoritmlər arasındakı qatı təşkil edir. Yəni virtual fayl sistemi əməliyyat sistemindən fayllarla işləmək üçün sorğuları (məsələn, verilənləri fayldan oxumaq və ya faylı uzaqlaşdırmaq) qəbul edərək, müvafiq faylın hansı fayl sistemində yerləşdiyini müəyyən edir və sorğunu həmin fayl sisteminin əməliyyatlarına çevirir.

GİRİŞ-ÇIXIŞ ƏMƏLİYYATLARININ İDARƏEDİLMƏSİ ALTSİSTEMİ VƏ QURĞULARIN DRAYVERLƏRİ



Əməliyyat sisteminin bir hissəsi kimi, *giriş-çıxış əməliyyatlarının idarəedilməsi altsisteminin* əsas vəzifəsi disk, klaviatura, videokart, şəbəkə kartı kimi periferiya qurğularından istifadəni təmin etməkdir. "Periferiya qurğuları" adı EHM-in əsas qurğular toplusu – prosessor, operativ yaddaş və informasiyanın daxil edilməsini və çıxışa verilməsini yerinə yetirən yardımçı qurğular yığını kimi təsvir edilməsilə əlaqədar meydana gəlmişdir. Bu qurğuların adlandırıldığı "periferiya" ifadəsi ingilis dilindəki "peripheral" ("ikinci dərəcəli" deməkdir) sözündən götürülmüşdür. Periferiya qurğularının digər ümumişlək adları bunlardır: "xarici qurğular" və "giriş-çıxış qurğuları".

Giriş-çıxış qurğuları çoxdur və əməliyyat sistemindən, onlarla səmərəli işləməsilə yanaşı, hər bir konkret qurğunun özünəməxsus cəhətləri haqqında xüsusi bilgilər olmadan da onlara müraciət üçün vahid qaydalara malik olması tələb edilir. Həqiqətən də, istifadəçi nöqteyi-nəzərindən, klaviaturanın bütün tipləri bir-birlərindən heç nə ilə fərqlənmir. Daha doğrusu, o, deyə bilər ki, bu klaviaturaların korpuslarının dizayn və düymələrinin sayı müxtəlifdir, lakin deməyəcək ki, onların EHM-ə qoşulma tipləri və onlarla işləmə qaydaları müxtəlifdir. Məsələn, klaviatura EHM-ə ardıcıl port, yaxud PS/2 portu, ya da USB şini vasitəsilə qoşula bilər. Göstərilən qoşulma tiplərindən hər biri onlar üçün spesifik olan əlaqə qaydalarına, yaxud, necə deyərlər, *qurğu ilə interfeysə* malikdir.

Fləş-yaddaşlı adi saatların USB vasitəsilə kompüterə birləşdirilməsi.



Qurğularla eyni cür əlaqə yaratmaq üçün əməliyyat sistemlərində vahid əlaqə qaydaları müəyyən edilir və qurğunun hər bir tipi üçün vahid qaydaları bu tip qurğularla spesifik əlaqə qaydalarına keçirən, yəni müvafiq qurğuları idarə edən alqoritmlər toplusu yaradılır. Bu cür alqoritmlər toplusu *qurğu drayveri,* bu alqoritmlərlə işləmə qaydası isə *drayverlə interfeys* adlanır.

Aydın məsələdir ki, bütün drayverlərlə hər tip qurğu üçün vahid və eyni zamanda, səmərəli interfeys yaratmaq mümkün deyildir. Ona görə də drayverlər və əməliyyat sistemlərinin digər hissələri (drayverlə bir neçə interfeys tipi) müvafiq olaraq, bir neçə drayver tipi (müxtəlif interfeyslər – müxtəlif alqoritmlər) arasında əlaqə yaratmaq üçün bir neçə əlaqə qaydası mövcuddur. Adətən, aşağıdakı tiplər göstərilir:

• simvol qurğusunun drayveri, məsələn, klaviaturanın drayveri. Bu cür qurğularla işləmək üçün əsas əməliyyatlar baytlar ardıcıllığının oxunması və yazılmasıdır;

• bloklu qurğuların drayveri, məsələn, diskin drayveri. Bu cür qurğularla işləmək üçün əsas əməliyyatlar blokların, yəni müəyyən uzunluqlu bayt massivlərinin oxunması və yazılmasıdır;

• şəbəkə qurğusu qrayveri, məsələn, İnternetə girmək üçün adapter. Bu cür qurğularla işləmək üçün əsas əməliyyatlar paketlərin, yəni müəyyən strukturlu bayt massivlərinin qəbulu və göndərilməsidir.

Drayverlə, məsələn, simvol qurğusunun drayverilə interfeys necə görünür? Hər şeydən əvvəl, əməliyyat sistemi qurğunun EHM-də olub-olmadığını müəyyən etməlidir. Deməli, drayverdə EHM-in konfiqurasiyasını yoxlayaraq "hə" və ya "yox" cavabını verən altproqram olmalıdır:

mənim_drayverim_soruşmaq( ) hə/yox qaytarır

Əgər qurğu tapılmışsa, onu yenidən *inisiallaşdırmaq* lazımdır, yəni onu elə proqramlaşdırmaq lazımdır ki, "söndürülmüşdür" vəziyyətində olsun. İnisiallaşdırma funksiyası çağırılan anda əməliyyat sistemi hələ qurğularla işləməyə hazır deyil. Məsələn, klaviatura üçün bu o deməkdir ki, o, basılmış klavişin kodunu EHM-ə göndərə bilməz.

mənim_drayverim_inisiallaşdırmaq( )

Əməliyyat sistemi tərəfindən çağırılan növbəti altpoqram qurğu ilə işin başlanması altproqramıdır. Əməliyyat sistemi bu altproqramı çağıraraq, məlumat verir ki, o, qurğu ilə işləməyə, məsələn, klaviaturadan simvolları qəbul etməyə hazırdır.

Müvafiq olaraq, işin başlanması altproqramı qurğunu işə salmalıdır.

mənim_drayverim_açmaq (qurğunu)

İndi oxuma, yazma və idarəetmə altproqramlarını çağırmaq olar.

Oxuma altproqramı çağrıldığı anda qurğu tərəfindən verilən baytları qaytarır. Əgər qurğu hələ heç nə verməyibsə, 0 qaytarılacaq.



Universal USB bağlayıcısı vasitəsilə kompüterə birləşdirilmiş CD- ROM.

mənim_drayverim_oxumaq (buferi) oxunan baytların sayını qaytarır

Yazma altproqramı göstərilən sayda baytı qurğuya yazır, klaviatura variantında bu altproqram olmayacaq.

mənim_drayverim_yazmaq (baytları) uğurlu/uğursuz qaytarır

Qurğuların idarəolunması altproqramı bütün digər əməliyyatları yerinə yetirir. Məsələn, klaviaturalar yandırıla və söndürülə bilən işıq indikatorlarına malikdir.

mənim_drayverim_idarəetmə (komanda, komanda_ parametrləri)

Nəhayət, qurğu ilə iş başa çatdığı anda bu qurğunu söndürən altproqram çağırılır.

mənim_drayverim_söndürmək (qurğunu)

Drayverdə həmçinin kəsilməni emal edən altproqram mövcuddur; o, adətən, qurğuda kəsilmə halları baş verdikdə əsas işi yerinə yetirir. Əgər klaviatura klaviş basılanda kəsilmə yaradırsa, klaviatura drayverini emal edən altproqram klavişin kodunu drayverin daxili simvol buferinə köçürür. Bu buferin tərkibi oxuma altproqramı tərəfindən köçürüləcək.

mənim_drayverim_kəsilmə ( )

TƏTBİQİ PROQRAM İNTERFEYSİ

Tətbiqi proqram interfeysinin köməyilə proqramçının əməliyyat sistemilə iş qaydası müəyyən edilir. Bu qaydaya proqramçıya işləmək üçün təqdim olunan altproqramların – *sistem çağırışlarının* siyahısı, bu altproqramların çağırılması qaydalarının təsviri, o cümlədən altproqramların adları, onların giriş parametrlərinin və onların qaytardığı qiymətlərin sayı, həmçinin müvafiq altproqramın gördüyü işin təsviri daxildir.

Proqramların əməliyyat sistemilə qeyd edilmiş, dəyişməz iş qaydaları toplusu şəklində olan tətbiqi proqram interfeysi rezident monitorlarda giriş-çıxış və idarəetmə köməkçi altproqramlarının yığını kimi yaranmışdır. Həqiqətən də, nə qədər ki monitorlar proqramların daxilində idi, proqramı hazırlayarkən bir giriş-çıxış altproqramını digəri, məsələn, başqa interfeyslə olan daha səmərəli altproqramlarla əvəz etməyə həmişə imkan var idi. Rezident monitoru halında köməkçi altproqramlar istifadəçi proqramından kənarda yerləşir və altproqramı dəyişmək mümkün deyil.

Həmin andan başlayaraq, tətbiqi proqramlar köməkçi alqoritmlər üçün tətbiqi proqram interfeysi vasitəsilə əməliyyat sisteminə müraciət etməklə, yalnız əsas alqoritmi reallaşdırır. Bu, əməliyyat sistemlərinin inkişafında çox mühüm mərhələdir. Bir əməliyyat sisteminin müəyyən proqram interfeysini müxtəlif əməliyyat sistemlərində müxtəlif EHM-



lər üçün yenidən hazırlamaq olar. Bu, o deməkdir ki, tətbiqi proqramlar dəyişiklik edilmədən müxtəlif EHM-lərdə yerinə yetirilə bilər, yəni tətbiqi proqramların bir EHM-dən digərinə *keçirilə bilərliyi* haqqında danışmaq mümkündür. Başqa sözlə, aparaturadan və əməliyyat sistemlərindən asılı olmayan proqramların hazırlanması imkanı yarandı.

Tətbiqi proqramların keçirilə bilmə imkanı (və ya *mobilliyi*) *açıq sistem* konsepsiyasının əsasını təşkil edir. "Sistem" adı altında təkcə əməliyyat sistemini deyil, həm də hər hansı tətbiqi məsələnin avtomatlaşdırılması üçün tətbiq edilən aparatura və proqramlar başa düşülür. Əməliyyat sistemi həmin *informasiya* sisteminin əsas komponentlərindən biridir. Bu sistemin digər komponentləri kimi verilənlər bazasının idarəedilməsi sistemini – VBİS-i, qrafik örtüyü, şəbəkə vasitələrini – tətbiqi proqramlar üçün zəruri olan mexanizmləri göstərmək olar, ümumiyyətlə desək, belə mexanizmlər əməliyyat sistemi tərəfindən təmin olunmaya bilər.

Açıq sistem – komponentləri bir-birilə standart interfeys vasitəsilə qarşılıqlı əlaqədə olan informasiya sisteminin qurulması üsuludur. İstifadəçi nöqteyi-nəzərindən, açıq sistem bir sistemdən digərinə keçmək imkanını təmin edən interfeysləri təyin edən, verilənlərin formatını və s. dəstəkləyən, açıq standartlar yığınına uyğun olmalıdır.

Açıq standart – texnologiyalardan, xüsusi aparat və proqram təminatından, və yaxud konkret istehsalçının məhsullarından asılı olmayan standartdır. Açıq standart istər sistem istifadəçiləri, istərsə də proqramçılar, istərsə də aparatura istehsalçıları üçün eyni dərəcədə irişiləndir. Açıq standart ictimaiyyətin nəzarəti altındadır və ona görə də bu məsələdə maraqlı olan hər

bir kəs onun işlənib hazırlanmasında iştirak edə bilər.

Açıq sistem aşağıdakı xüsusiyyətlərə malikdir:

• *keçirilə bilmə imkanı* (*mobillik*) – avadanlığı dəyişərkən proqram və verilənlərin köçürülməsi, sistemdə dəyişiklik edilərkən hər hansı hazırlıq işləri görmədən istifadəçilərin onlarla işləyə bilmək imkanı;

• *genişlənmə imkanı* (*miqyaslanma*) – sistemə yeni funksiyaların əlavə edilməsi, yaxud mövcud funksiyaların, sistemin hissələrini dəyişmədən dəyişdirilmək imkanı;

• *uyuşqanlıq* (interoperabellik) – bu geniş termindir və digər sistemlərlə qarşılıqlı əlaqə imkanını nəzərdə tutur;

• "*istifadəçiyə səmimilik*".

Əlbəttə, bu cür xüsusiyyətlərə malik olan sistem istər istifadəçilər, istərsə də proqramçılar üçün olduqca cəlbedicidir. Sözsüz ki, açıq sistem konsepsiyası meydana gələnə qədər istifadə edilən proqramlar bu qəbildən olan ayrı-ayrı xüsusiyyətləri özündə əks etdirirdi. Lakin açıq sistemdə, birincisi, bu xüsusiyyətlər qarşılıqlı əlaqəli xüsusiyyətlər kimi baxılır, ikincisi, onların reallşdırılmasına açıq standartların seçilmiş yığını vasitəsilə zəmanət verilir.

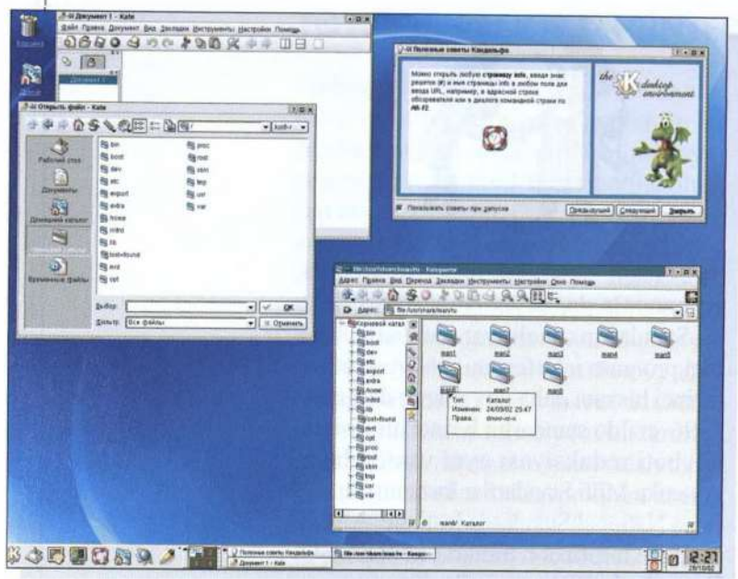
İnformasiya sisteminin komponenti olan əməliyyat sistemi üçün tanınmış beynəlxalq standart POSIX-dir. Bu standartın tam adı ISO/IEC 9945 İnformation Technology – Portable Operating System Interface-dir, amma ədəbiyyatda daha çox ixtisar olunmuş POSIX adı işlədilir. Standartın hazırlanmasına 1985-ci ildə ABŞ-ın Elektrotexnika və Elektronika Mühəndisləri İnstitutunda (Institute of Electrical and Electronics Engineers, IEEE)) başlanıldı və bu standart IEEE Std 1003 kimi tanındı. Bir müddət sonra, 90-cı illərin əvvəllərində həmin standart Ümumdünya Standartlaşdırma Təşkilatı (International Organization for Stan-



dardization – ISO) tərəfindən dərc edildi və bununla da, beynəlxalq status aldı.

Standartın əməliyyat sisteminin tətbiqi proqram interfeysini müəyyən edən birinci hissəsi daha çox maraq doğurur. 1996-cı ildə standartın birinci hissəsinin növbəti redaksiyası eyni vaxtda həm Amerika Milli Standartlar İnstitutu (American National Standards Institute, ANSI) və IEEE tərəfindən birlikdə, həm də ISO Beynəlxalq Elektrotexnika Komissiyası (International Elektrotechnical Commission, IEC) ilə birlikdə müvafiq olaraq ANSI/IEEE Std 1003.1-1996 və ISO/IEC 9945-1:1996 adları altında buraxıldı. Bu standartın "ən yeni" redaksiyalarından biri 2001-ci ildə IEEE tərəfindən IEEE Std 1003.1-2001 adı ilə dərc edildi.

POSIX standartı UNIX əməliyyat sistemi üçün xarakterik olan tətbiqi interfeysi müəyyən edir. Hətta POSIX adının ixtisarı da bu faktı əks etdirir: bir vaxtlar UNIX əməliyyat sistemini hazırlayanlar tələb edirdilər ki, UNIX əməliyyat sisteminin ilkin kodları əsasında yaradılan əməliyyat sistemlərinin adlarında "IX" sonluğu olsun. Buna görə də UNIX-i tez-tez "açıq sistem" terminin sinonimi kimi işlədirlər. Həqiqətdə isə, birincisi, UNIX əməliyyat sisteminin heç də bütün versiyaları həmin standartın tələblərinə uyğun gəlmir. İkincisi, UNIX əməliyyat sistemilə heç bir bağlılığı olmayan bir sıra əməliyyat sistemləri POSIX tətbiqi interfeysini dəstəkləyir. Məsələn, Windows NT və onun sonrakı modifikasiyaları olan Windows 2000 və Windows XP, Open VMS, real vaxt əməliyyat sistemi VxWorks öz xüsusi tətbiqi inter-



Linux əməliyyat sisteminin standart pəncərə sistemi.

feysindən başqa, POSIX interfeysinə də malikdirlər.

Standartlaşdırmaya cavabdeh olan nüfuzlu təşkilatlar tərəfindən buraxılan POSIX beynəlxalq standart hesab olunsa da, mükəmməl açıq informasiya sisteminin qurulması üçün bu yetərli deyildir. Ona görə də POSIX-lə bərabər digər standartlar da geniş yayılmışlar. Bir qayda olaraq, bu cür standartlara POSIX daxil edilir və üstəlik bir sıra özünəməxsus xüsusiyyətlər əlavə olunur. Məsələn, informasiya texnologiyaları bazarının aparıcı şirkətlərini birləşdirən çox məşhur The Open Group konsorsiumu UNIX98 spesifikasiyasını yaratdı. Bu spesifikasiyaya POSIX-lə yanaşı, qrafi interfeys, şəbəkə vasitələri, Java dili və İnternet xitmətləri standartları daxildir.

UNIX ƏMƏLİYYAT SİSTEMİ

Bu gün UNIX haqlı olaraq əməliyyat sistemləri dünyasında etalon hesab edilir. Son 30 il ərzində əməliyyat sistemləri, hesablama texnikası və informasiya texnologiyaları sahəsində əldə edilən istənilən nailiyyət bu və ya digər şəkildə UNIX əməliyyat sisteminin adı ilə bağlıdır. Bu müvəffəqiyyətin səbəbi nədir? UNIX-ə həmişə nəinki öz populyarlığını saxlamağa, həm də bir sıra universal əməliyyat sistemləri arasında lider olmağa imkan verən nədir?

UNIX əməliyyat sisteminin yaradılmasına qədər Massaçusets Texnologiya İnstitutunun, General Elektric şirkətinin və AT&T şirkətinin elmi-tədqiqat mərkəzinin Bell Telephone Laboratories korporasiyasının tədqiqatçılarını birləşdirən "Multics" hesablama sistemi (1964-cü il) layihəsi var idi. Bu layihənin məqsədi – təkcə həftədə 7 gün, sutkada 24 saaat fasiləsiz və dayanmadan işləməklə maksimal spektr imkanlar vermək deyil, həm də tədqiqatçıların sözlərilə desək, bu gün və yaxın gələcəkdə vaxt bölgüsü rejimində fəaliyyət göstərən əməliyyat sisteminin yaradılması olmuşdur. Qarşıya qoyulan vəzifənin, demək olar ki, yalnız yarısını yerinə yetirmək mümkün oldu: Multics əməliyyat sistemi, doğrudan da, etibarlı və dayanıqlı sistem idi, Ford şirkətində 1974-cü ildən 1997-ci ilə qədər istifadə olunmuşdur.

1969-cu ilin aprelində, Multics əməliyyat sisteminin 1-ci versiyasının meydana gəlməsinə bir neçə ay qalmış, AT&T şirkətinin rəhbərliyi layihədən çıxmaq qərarına gəldi və üstəlik əməliyyat sisteminin yaradılması sahəsində apardığı bütün tədqiqatları dayandırdı. Bu xəbər Bell Laboratories-də məyusluq yaratdı, çünki tədqiqatçılar yalnız bir çox məsələləri səmərəli şəkildə həll etməyə deyil, həm də ətrafında geniş istifadəçi qrupu yaratmağa qadir olan əməliyyat sisteminə ehtiyac duyurdular. Vaxt bölgüsü rejimində işləyən əməliyyat sistemlərindən istifadə təcrübəsi göstərdi ki, bir neçə istifadəçinin birlikdə işləməsi təkcə interaktiv iş deyil, hər

şeydən əvvəl ideyaların mübadiləsi üçün qidalandırıcı mühit – insanın elmi fəaliyyətinin əsasıdır.

Bell Telephone Laboratories-in Ken Tompson, Dennis Ritçi, Co Ossanna, Rud Kenedey və Duq Mak-İlroydan ibarət mütəxəssislər qrupu eksperimental əməliyyat sisteminin ayrı-ayrı hissələrini, o cümlədən əməliyyatlar sisteminin fayl sistemini işləyib hazırlamaqda davam edirdilər.

1969-cu il ərzində Ken Tompson öz mahiyyətinə görə Günəş sistemində kosmik cisimlərin hərəkəti fizikasının proqram modelini əks etdirən "Kosmik səyahət" oyunu hazırladı. Laboratoriya EHM-ində bu oyunu oynamaq böyük israfçılıq idi: oyun üçün tələb edilən hesablayıcı resursların bir saat ərzində istifadəsinə çəkilən xərc 75 dollar təşkil edirdi. Xoşbəxtlikdən, Tompson münasib hesablama aparatı – qrafik terminala malik olan PDP-7 EHM-ini tapdı. Serverlə müqayisədə, PDP-7 cib kompüteri kimi görünürdü. Oyunu PDP-7-yə köçirərkən *kross-texnologiya* adlanan texnologiya yaradıldı ki, onun vasitəsilə bir EHM üçün nəzərdə tutulan proqramların digər EHM-dən istifadə etməklə yazmaq mümkün idi.

Əməliyyat sistemləri üçün kross-texnolgiya vacib addım idi. Tədqiqatın əvvəlində hələ mövcud olmayan əməliyyat sisteminin *icra olunan faylını* (əməliyyatlar sisteminin obrazını) yaratmaq tələb olunurdu. Bu, məşhur bir sualı xatırladır: "Hansı daha əvvəl yaranmışdır: toyuq, yoxsa yumurta?" Bu məsələnin həlli digər EHM-də əməliyyat sisteminin icra oluna bilən faylları yaratmağa qadir olan instrumental vasitələri işləyib hazırlamaqdan ibarət idi.

"Kosmik səyahət" üçün xüsusi *icra mühiti* lazım idi. Burada Bell Laboratories şirkətinin fayl sistemi kara gəldi. Proseslərin idarə edilməsi üçün əlavə olunmuş altsistemlə birlikdə bu sadə vaxt bölgüsü əməliyyat sistemini əmələ gətirirdi. Bu əməliyyat sistemi çoxistifadəçi əməliyyat sistemi idi (PDP-7-yə iki terminal qoşulurdu!), lakin eyni vaxtda hər bir istifadəçi yalnız bir proqram başlada bilirdi. Sistem üçün elə də böyük olmayan, fayllarla işləməkdən ötrü (mətn redaktoru, faylların surətini çıxaran, uzaqlaşdıran və çap edən utilitlər) nəzərdə tutulan istifadəçi utilitləri yığınını və komanda interpretatoru – proqramı yaradıldı (bu proqramın köməyilə istifadəçilər bütün qalan proqramları başlada bilirdi). PDP-7 EHM-də əməliyyat sisteminin ilk versiyası məhz belə bir formada meydana çıxdı. Bir qədər sonra, 1970-ci ildə Brayan Kerniqan bu əməliyyat sistemi üçün ad təklif etdi – UNICS (*ing.* "Uniplexed İnformation and Computing Service" – "Sadə informasiya-hesablama xidməti"), bu, Multics-in antonimi idi. Tezliklə bu ad UNIX-ə qədər qısaldıldı.

UNIX-in tarixində ən mühüm an 1973-cü ildə Duq Mak-İlroy və Rad Kənədey tərəfindən *proqram kanallarının* (*komandalar konveyerini* əldə etmək üçün bir proqramın çıxış axını ilə digər proqramın giriş axınının əlaqələndirilməsi) kəşf olunması oldu.

Bundan sonra həmin əməliyyat sistemi müasir anlamdakı UNIX əməliyyat sistemi oldu.

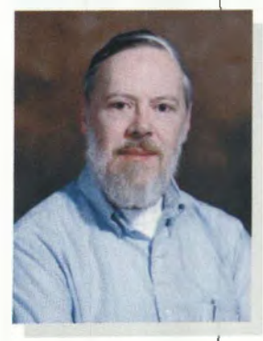
Proqram kanalının işini təsvir etmək üçün növbəti nümunəyə baxmaq olar. Tutaq ki, hər bir fəsli ayrıca faylda yerləşən kitab vardır. Bu kitabın səhifələrini səhifə nömrələrilə çap etmək lazımdır. UNIX-də bu əməliyyat aşağıdakı kimi olacaqdır:

```
cat fəsil1 fəsil2 fəsil3 fəsil4
fəsil5 | mpage -H | lpr
```

Burada cat – fəsil1-dən fəsil5-dək olan faylları oxuyan və onları öz çıxış axınına yazan proqramdır; mpage – mətni formatlaşdıran, səhifələrə bölən və



Ken Tompson.



Dennis Ritçi.



Duq Mak-İlroy.



Bill Coy.

onun səhifələrini nömrələməklə öz çıxış axınına yazan proqramdır; lpr – giriş axınının çapa verilməsidir. " | " işarəsi proqram kanalı vasitəsilə çıxış axınını giriş axını ilə əlaqələndirir.

Proqram kanallarının meydana gəlməsilə UNIX əməliyyat sistemi yeni keyfiyyət mərhələsinə qədəm qoydu: əməliyyat sistemindən o proqramçının tətbiqi məsələlərin proqramlaşdırılmasını əhəmiyyətli dərəcədə asanlaşdıran *instrumental vasitələr yığınına* çevrildi. UNIX hesablama texnikasından istifadə üçün yeni yanaşma – yalnız özünü təmin edən xüsusi təyinatlı vahid sistemi yaratmaq əvəzinə universal proqramların kombinasiyasını yaratmağı təklif etdi.

Proqram kanallarının ixtira edilməsi ilk rabitə peykinin, düyməli telefon aparatının, mobil rabitəni və optik tel rabitəsinin ixtira edilməsi kimi mühüm hadisədir.

1973-cü ildə UNIX əməliyyat sistemi Kerniqan və Ritçi tərəfindən xüsusi olaraq bu sistem üçün yaradılan C proqramlaşdırma dilinə keçirildi. Bu, daha bir inqilabi addım idi: buna qədər əməliyyat sistemləri yaradılarkən belə yüksək səviyyəli dillərdən istifadə edilməmişdi.

1973-cü ilin oktyabr ayında Tompson və Ritçi simpoziumda əməliyyat sistemlərinin qurulması prinsiplərinə dair hesabat təqdim etdilər. Bu hesabat 1974-cü ilin iyul ayında "Communications of the ACM" jurnalında dərc edildi. Həmin vaxt AT&T şirkətinin proqram təminatlarının satışına icazə verilmirdi. Çünki bu şirkət telefon xidmətləri bazarında inhisarçı idi və ABŞ-ın müvafiq federal hökumət qanununa görə, bu halda şirkət digər kommersiya fəaliyyətilə məşğul ola bilməzdi. Ona görə də UNIX-i ilkin kodları ilə birlikdə hər bir kəs ala bilərdi. Məhz bu UNIX-in sonrakı uğurunu müəyyən etdi: ilkin kodların köməyilə (zəruri hallarda lazımi komponentləri əlavə etməklə, yaxud mövcud səhvləri düzəltməklə) proqram təminatının asanlıqla bu və ya digər tələblər üçün kökləmək olardı. Universitetlər UNIX əməliyyat sistemilə maraqlanmağa və onun ilkin kodlarından proqramlaşdırmanın tədrisi prosesində istifadə etməyə başladılar. Həmçinin tədqiqatçıların arzuladıqları kimi, istifadəçilər cəmiyyəti formalaşdı.

1978-ci ildə UNIX mobil əməliyyat sisteminə çevrildi. Elə həmin vaxt Avstraliyanın Vollonqonqa Universitetində proqramçılar Cüris Reynfelds və Riçard Müller UNIX-i İnterdata 7/32 EHM-nə keçirdilər. Bundan bir qədər sonra Prinston Universitetində Tom Layon UNIX-in IBM 360 EHM-nə keçirilməsini başa çatdırdı, Dennis Ritçilə Stiv Conson UNIX-in İnterdata 8/32 EHM-nə keçirilməsinə nail oldular. Bunun üçün onlar C dilini tamamladılar.

1979-cu ilin yanvar ayında UNIX-in ilkin kodlarının dərs vəsaiti kimi istifadəsini qadağan edən növbəti lisenziyalı redaksiyası buraxıldı. Kompüter aləmi iş stansiyaları erasına qədəm qoydu, 2 metr hündürlüyündə olan şkafların əvəzinə kristal şəklində prosessorları olan ucuz EHM-lər istifadə edilməyə başlandı. Müvafiq olaraq, istehsalçılar üçün yeni avadanlığın proqram təminatına istehsal xərclərini azaltmaqdan ötrü mobil əməliyyat sisteminə tələbat yarandı. AT&T artıq UNIX-in yeni kommersiya versiyalarını buraxmağa başladı (bunların arasında daha populyar olanlar 1981-ci ildə buraxılmış UNIX System III və 1983-cü ildə buraxılmış UNIX System III-dür), lakin artıq UNIX adının Bell Telephone Laboratories-lə bağlanmasına son qoyuldu, onların yerini universitetlər tutmağa başladı.

Berkli şəhərinin Kaliforniya Universitetindən olan tədqiqatçılar qrupu UNIX-in BSD (Berkeley Software Distribition) adı ilə tanınan variantını buraxmaqla daha geniş şöhrət qazandılar. 1976-cı ildə Berklinin məzunu Ken Tompson öz akademik məzuniyyətini universitetdə müəllimlik etməklə keçirdi. O, orada tədqiqat üçün nəzərdə tutulan UNIX-i quraşdırdı və Pascal dili üçün kompilyator yazdı. 1977-ci ildə tələbə Bill Coy proqram təminatından ibarət kolleksiya toplayaraq, BSD-nin birinci versiyasını buraxdı.

BSD UNIX-in populyarlığı, əslində, aşağıdakı üç amillə əlaqədar idi:

- demokratik lisenziya: BSD UNIX-i ilkin kodlarla birlikdə hər kəs əldə edə bilərdi;
- şəbəkədə işləmək imkanı ("standart İnternet sistemi" – heç də az əhəmiyyətli xüsusiyyət deyil);
- uğurlu platforma seçimi: VAX – "80-ci illərin ilk yarısının EHM-i". Bütün sadalananlarla yanaşı, BSD UNIX – yaxşı sistemdir, onun fayl sistemi və yaddaşın idarəedilməsi altsistemi uzun müddət UNIX aləmində ən sürətlisi idi.

1982-ci ildə Bill Coy Berklini tərk etdi və SUN Microsystems şirkətinin əsasını qoydu. Bu şirkət indiyə qədər ilkin olaraq UNIX-in idarəsi altında işləmək üçün nəzərdə tutulan (SUN UNIX-in versiyasında Sun əməliyyat sistemi adlanırdı, sonra isə Solaris-ə dəyişdirildi) işçi stansiyaları və serverlər buraxır. SUN-ın UNIX-ə meylliliyi onun 80-ci illərin ikinci yarısında liderlik etməsinə gətirib çıxartdı. 90-cı illərdə birincilik Intel prosessorlarına malik olan fərdi EHM-lərə keçdi. SUN-ın UNIX sahəsində əsas nailiyyəti şəbəkə vasitəsilə əlaqələndirilən uzaqda yerləşən EHM-lərin fayl sistemlərilə işləməyə imkan verən NFS şəbəkə fayl sisteminin yaradılmasıdır.

UNIX-in inkişafındakı növbəti mərhələ Helsinki Universitetinin Təbiət Elmləri fakültəsinin Hesablama Texnikası bölməsinin ikinci kurs tələbəsi Linus Torvaldesin düşünüb yaratdığı Linux-la bağlıdır. Təhsil aldığı müddətdə o, əsasında əməliyyat sistemləri sahəsində tanınmış mütəxəssis Endryu Tanenbaum tərəfindən xüsusi olaraq tədris məqsədləri üçün yaradılan UNIX əməliyyat sisteminə oxşar Minix əməliyyat sisteminin dayandığı əməliyyat sistemləri nəzəriyyəsini öyrənirdi. Torvaldes əvvəlcə istəyirdi ki, sistem Minix-dən bir qədər yaxşı olsun. O, ciddi bir şey iddiasında deyildi – bu, sadəcə, onun üçün hobbi idi. 1991-ci ilin sentyabrında Linux-in 0.01 versiyası sərbəst şəkildə köçürülmək üçün İnternetdə yerləşdirildi.



Linus Torvaldes.



Endryu Tanenbaum.

Linux layihəsinin sürətli inkişafı iki amillə bağlı idi. Birincisi, artıq 90-cı illərin əvvəllərində ilkin inkişaf mərhələsində olan və sərbəst yayılan əməliyyat sistemləri daha mövcud deyildi. Buna görə də, tədqiqatçılar qrupuna daxil olmaq və bu cür layihələrdə iştirak etmək olduqca çətin idi: bu və ya digər əməliyyat sisteminin konkret detalları haqqında bilik ehtiyatına malik olmaq tələb edilirdi. Linux isə elə də böyük olmayan əməliyyat sistemi və yeni ideyaların sınaqdan keçirilməsi üçün ideal poliqon idi. İkincisi, Linux əməliyyat sisteminin yeni versiyaları çox sürətlə buraxılırdı, versiyalar arasında dəyişikliklər çox deyildi, bu da koddakı səhvləri asanlıqla tapmağa və aradan qaldırmağa imkan verirdi. Linux 2.1 nüvələri seriyasını rekord saymaq olar, belə ki, onun əsasında 150-yə yaxın versiya buraxılmışdır.

Hazırda Linux milyonlarla istifadəçisi olan bir sistemdir.



MULTİMEDİA

İNFORMASİYANIN RƏQƏMSAL FORMADA TƏSVİRİNƏ KEÇİD

XX əsrdəki informasiya böhranı göstərdi ki, informasiyanın saxlanmasının ənənəvi, analoq üsulları köhnəlmişdir. Artıq informasiyanın saxlanması və işlənməsi üçün kompüterdən istifadə etməklə verilənlərin rəqəmsal formada təsvirinə keçmək zərurəti yarandı. Minilliklərin sərhədində tamamilə məlum oldu ki, gələcək rəqəmsal texnologiyalarındır.

İnformasiyanın rəqəmsal formaya çevrilməsi heç də həmişə asan məsələ deyil. İnformasiya formaca müxtəlif ola bilər: audio, video, mətn və hətta incəsənət əsəri formasında. İki problem meydana çıxır: birincisi müxtəlif tip informasiyanın təsvir edilməsilə, ikincisi isə informasiyanın kompüterə daxil edilməsinin təşkililə bağlıdır.

Kompüterlər rəqəmsal texnologiyalar əsasında işləyir, burada informasiya fasiləsiz dəyişən kəmiyyətlərin qiymətlərilə deyil, fiziki qurğuların dayanıqlı vəziyyətilə əks etdirilir. İnformasiyanın rəqəmsal formada işlənməsi üçün kənardan verilən siqnallar əsasında bir vəziyyətdən digərinə keçə bilən iki dayanıqlı vəziyyətə malik fiziki qurğular lazımdır.

Ən sadə məsələ, məlum olduğu kimi, mətnlərin təsvir edilməsidir. Burada fikirlər, obrazlar, anlayışlar artıq hərflərdən ibarət sözlər vasitəsilə kodlaşdırılmışdır, onlar da öz növbəsində hərflərdən ibarətdir, hərfləri isə çox asanlıqla rəqəmsal formaya çevirmək olar.

Fiqurlu işarələrin bütöv anlayışları və sözləri, yaxud ayrı-ayrı hecalar və danışıq səslərini əks etdirdiyi heroqlif yazısı ilə bağlı məsələ isə xeyli mürəkkəbdir. Yapon dilində bir neçə min heroqlifdən istifadə olunur. Bu heroqliflər sözlərin köklərini, müxtəlif hecaları və

yazıda işlədilən qrammatik sonluqları yazmaq üçün istifadə olunan onlarla "kan" işarəsindən ibarətdir. Hər heroqlifə xüsusi bir kod vermək mümkün olardı. Lakin səmərəsi olmazdı. Heroqlif mətnlərinin rəqəmsal formada təqdimatı daha çox fonetik prinsipə əsaslanır.

Mətnin klaviaturanın köməyilə kompüterə daxil edilməsi sürətinə və rahatlığına görə (mətn dərhal rəqəmsal formada təsvir edilir) adi yazıdan çox üstündür.

Lakin əgər mətn artıq çap və ya əlyazması şəklində mövcuddursa, mətnin daxil edilməsinin müasir forması qəbuledilməzdir. Birincisi, səhvlərin olması qaçılmazdır, ikincisi, insan əməyindən bu cür səmərəsiz istifadə etmək məqsədəuyğun deyil. Belə mətnlər mətni qrafik informasiya, şəkil kimi oxuyan skanerin köməyilə daxil edilə bilər. Xüsusi tanıma proqramı isə qrafik faylı mətn sənədinə çevirir. Obrazların tanınması məsələsi hələ mükəmməllikdən çox uzaqdır və pis xətt hələ də kompüter üçün tapmaca olaraq qalır.

Müasir mətnlərin işlənməsi sistemləri daxiletmə və redaktə ilə yanaşı elə imkanlara malikdir ki, ilk mətn redaktorlarını yaradanlar bunları yalnız arzu edə bilərdilər. Bunlardan bəzilərini göstərək: mətnin yığılması prosesində orfoqrafiyanın avtomatik yoxlanması; mətnin intellektual səhifələnməsi; tamamilə yeni, kağız "analoqları" olmayan, təkcə mətndən deyil, həm də səslə müşaiyət olunan və animasiyaya malik olan hipermətn və multimedia sənədlərinin yaradılması. Bütün bunlar mətn informasiyasının rəqəmsal formaya çevrilməsi nəticəsində mümkün oldu.

AUDİOİNFORMASİYA

Kompakt-disk və ya DVD-disk səs, görüntü, yazı formasında olan informasiyanı milyardlarla ikilik sıfırlar və vahidlər şəklində saxlaya bilir. Lazer diskinin uzunmüddətliliyi (yüz illərlə) daha çox rəqəmsal texnologiyaların prinsiplərilə izah olunur. Vinil vallarından fərqli olaraq, tozun lazer diskinə təsiri yoxdur, xırda cızıqlar isə səs, görüntü və yazının göstərilməsinin keyfiyyətini aşağı salmır.

Bəzən rəqəmsal səsin mexaniki, cansiz olması barədə fikirlər söylənilir. Müasir audioqurğularda sırf analoq xarakteristikalarına malik olan lampalı siqnal gücləndiriciləri daha ucuz və qənaətcil, həmçinin daha "diskret" tranzistorlarla əvəz olunmuşdur. Yəqin ki, bu "düzgünlük" rəqəmsal formaya salınmış səsin uyğun şəkildə qavranılmasını şərtləndirir. Keyfiyyətli, natural (high-end) səs həvəskarları isə bahalı lampalı gücləndiricilərdən və vinil vallarından istifadə etməkdə davam edirlər.

1992-ci ildə Sony firması yeni səsyazma daşıyıcısı olan mini-diskləri (MD) təqdim etdi.

Səslər böyük sayda müxtəlif tezlikli sinusoidal rəqslərdən ibarətdir. İnsan orta hesabla 50 Hs-dən 18 kHs-dək səsləri qəbul edə bilir. AudioCD-lər 44,1 kHs 16 bit stereo diskretləşdirmə tezliyilə yazılır. Bu, o deməkdir ki, iki streokanalın hər biri üçün siqnal amplitudu saniyədə 44,1 dəfə ölçülür, onların hər birinin rəqəmsal formada saxlanması üçünsə 16 bitdən isitfadə edilir. Belə bir diskretləşmə ilə yazılma zamanı 22,05 kHs-dən böyük tezlikdə olan harmonik rəqslər yazılma zamanı kəsilib atılır. Bu isə, ümumiyyətlə, qorxulu deyildir, çünki bu həddə praktiki olaraq insanın qavrama həddinə bərabərdir. Səsin qavranılmasının psixoakustik xüsusiyyətlərindən istifadə etməklə kodlaşdırma bu itkiləri insanın hiss edə bilməyəcəyi həddə endirməklə, səsin kompakt-diskə daha sıx şəkildə (təxminən 5-7 dəfə) yerləşdirilməsinə imkan verir.

Mətn kimi, səsin rəqəmsal formada işlənməsi də, praktiki olaraq, hər addımda tətbiq edilir. Musiqi mərkəzlərinin əksəriyyəti səs siqnallarının rəqəmsal formada işlənməsi üçün xüsusi mikrosxemi olan rəqəmsal prosessorlara malikdir. Müasir DVD-səsləndiricilər

Tomoqraf.



Əgər siz öz xəttinizi oxuya bilmirsinizsə, lakin mətni dəqiqədə 70 sözdən çox olmaqla, kompüterdə yığırsınızsa, hər şeydən əvvəl siz dahi deyilsiniz, sadəcə, müasir bir insansınız...

Həcmli fotoqrafiyanın yaradılması üçün avadanlıqlar artıq 2002-ci ilin sonunda satışa daxil olmuşdur.

GÖRÜNTÜ

Fotoqrafiyadan və ya qravüradan ibarət olan görüntünü daxil etmək üçün eyni skanerlərdən istifadə etmək olar, çünki bu skanerlər onların keyfiyyətli şəkildə daxil edilməsinə imkan verir. Artıq əvvəlcə adi fotoaparatla şəkli çəkmək, sonra da çıxardılan görüntünü və ya slaydı skanerdən keçirmək mənasızdır. Müasir rəqəmsal fotoaparatlar çəkilişlərin keyfiyyətinə görə analoq fotoaparatlardan geri qalmır. Bunu rəqəmsal video haqqında da demək olar.

Həcmli (fəza) obyektlərin təsviri hələ mürəkkəb problem olaraq qalır (lakin artıq həcmli fotoşəkillərin yaradılması mümkündür). Bu cür təsvirin iki əsas üsulu mövcuddur: baxılan obyektin səthində nöqtələr şəbəkəsinin verilməsi və obyektə dəqiq riyazi təsviri olan müxtəlif həndəsi fiqurlar vasitəsilə yaxınlaşma. Birinci üsul heykəl və insanın özü kimi mürəkkəb obyektləri təsvir edərkən tətbiq edilir. İkinci üsul avtomatik layihələndirmə sistemlərində (ALS), lazımi detalların istehsalı üçün rəqəmli proqramla idarə edilən (RPİ) dəzgahlarda geniş istifadə edilir.

Xüsusi cihazda – tomoqrafda paralel şəkillər seriyası əsasında həkimlər insanın müayinə edilən orqanlarının və bədən hissələrinin üçölçülü görüntüsünü alırlar. Bu cür görüntüləri kompüterlə işləmədən lazer vasitəsilə müasir əməliyyatları yerinə yetirmək, məsələn, ətrafdakı toxumaları yarmadan və ya zədələmədən daxili şişləri kəsib atmaq mümkün deyildir. Kompüter təkcə artıq mövcud olan informasiyanı dəyişməyə deyil, həm də yenisini yaratmağa qadirdir.

Qrafikanın və videonun işlənməsi rəqəmsal informasiyadan isitifadə etməklə əhəmiyyətli dərəcədə asanlaşır. Adobe Photoshop və ya Corel Draw kimi peşəkar qrafik redaktorlar ev şəraitində müxtəlif effektlərin, rəqəmsal filtrlərin, və videomaqnitofonlar da səsin işlənməsi üçün keyfiyyətinə görə kinoteatrlarda quraşdırılan Dolby ACZ sistemlərindən geri qalmayan prosessorlara malikdir. Bir çox radiostansiyalar və telestudiyalar da rəqəmsal yazıya və səsin işlənməsinə keçmişlər. Hətta artıq evlərdə də MD-lərdəki (mini-disklərdəki) yazıların səsləndirilməsində daha çox köhnə, analoq üsullardan (təkrar yazılma zamanı keyfiyyətin itməsilə) yox, rəqəmsal üsuldan (ilkin yazının dəqiq surəti) istifadə edirlər.

Mütəmadi olaraq aparılan tədqiqat istiqamətlərindən biri də səsin tanınmasıdır. Tezliklə səs tanıma proqramlarından mətnin daxil edilməsi üçün istifadə etmək mümkün olacaq.



kökləmələrin geniş çeşidini təqdim edir. Halbuki əvvəllər bunun üçün mürəkkəb və bahalı avadanlıqlardan ibarət bütöv bir fotolaboratoriya tələb olunurdu.

Rəqəmsal texnologiya kinoda da az rol oynamır. Burada söhbət hətta qeyri-xətti rəqəmsal montajın rahatlığından getmir. "Ulduz müharibələri. Epizod I" filminin bir hissəsi rəqəmsal fotokamera ilə çəkilmişdir.

Filmin rejissoru Corc Lukas haqlı olaraq qeyd etmişdir ki, bu rəqəmsal hissəni heç kəs müəyyən edə bilməz. Növbəti film – "Ulduz müharibələri. Klonların hücumu" isə yalnız rəqəmsal texnologiyalardan istifadə etməklə çəkilən ilk filmlərdən biri oldu. Xüsusi effekt-vermə prosesləri də sadələşdi və kompüterin köməyilə yaradılan görüntü real görüntüdən heç nə ilə fərqlənmir.



VEKTOR VƏ RASTR QRAFİKİ

Kompüterdən ilk dəfə görüntülərin yaradılması üçün istifadə edən Ayven Sazerlend bir dəfə belə demişdi: "EHM-ə qoşulan displey mənə Alisanın möcüzələr ölkəsinə açılan pəncərəsini xatırladır... Displeyin köməyilə mən təyyarəni aviadaşıyıcının göyərtəsinə endirdim, elementar zərrəciyin potensial çaladakı hərəkətini izlədim, işıq sürətinə yaxın sürətlə raketlə uçdum və hesablama maşınlarının sirli daxili həyatını müşahidə etdim".

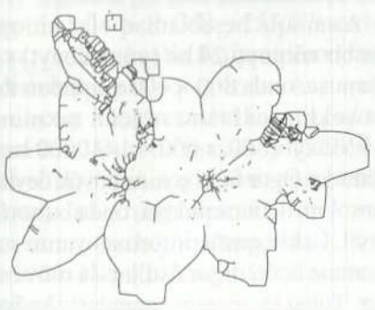
80-ci illərdə fərdi kompüter bazarı faktiki olaraq IBM PC kompüterlərinin klonları tərəfindən zəbt edilmişdi. 1987-ci ildə yaradılmış VGA standartı (*ing.* Video Graphics Array – hərfi tərcüməsi "videoqrafik matris" və ya "massiv") IBM PC-lərin inkişafında dönüş nöqtəsi oldu. İnformasiyanı ekranda təsvirə çevirən videoadapterlər və videokartlar kompüterin monitorunda daha böyük çözümlülükdən (640 × 480 nöqtə) və daha dərin rəng çalarından (16 əvəzinə 256 rəng) istifadə etməyə başladılar.

Ekranda təkcə aşağı multiplikasiya keyfiyyətində olan üslublaşdırılmış şəkilləri yox, həm də fotoşəkilləri görmək imkanı yarandı. Rəngli ekranda ağ siçan canlı kimi görünürdü.

Fərdi kompüterin ekranı nöqtələrin rastrından və ya piksellərdən (*ing.* "picture element") ibarət idi. VGA standartı 256 kbayt-lıq yaddaşla ekomplektləşdirildi. Bu yaddaşda displeydəki (kineskopun ekranındakı) üfüqi açılmış sətirlərdən yaranmış şəkillər element-element olmaqla saxlanılırdı.

Adi çap qurğularında sadə rəsmlərin yaradılması EHM-in meydana gəldiyi ilk illərə təsadüf edir. 1950-ci ildə Massaçusets Texnologiya İnstitutunda Whirlwind kompüteri görüntünü çıxarmağa imkan verən elektron-şüa borulu displeylə təchiz edildi.





Rastr və vektor qrafiklə çəkilmiş gül.

70-ci illərdə Xerox firmasının tədqiqat mərkəzində hazırlanmış çoxpəncərəli interfeys həm bu firmanın kompüterlərində, həm də Apple Computer. Inc. firmasının Lisa EHM-lərində istifadə olunurdu. İnterfeys qrafikanın və mətnin monitorun ekranında eyni vaxtda yerləşdirilməsini tələb edirdi. Yeni sistemlər WYSIWYG (*ing.* What You See Is What You Get – "Nə görürsənsə onu da alırsan") prinsipilə qurulurdu. Bu prinsipə görə, ekranda görüntü çap variantına mümkün qədər daha çox bənzəməli idi.

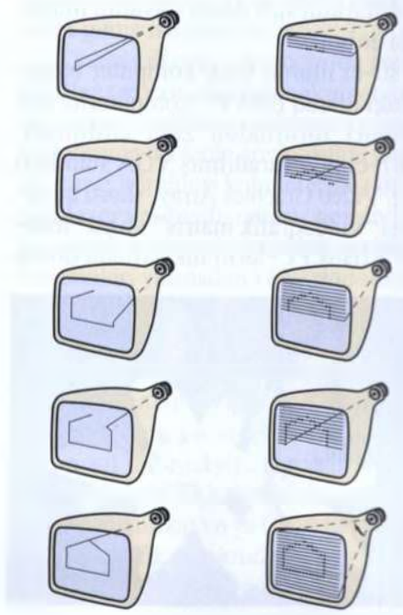
Elektron-şüa borusundakı şüa lüminoforla örtürülmüş ekrana düşür. O, şüa enerjisini ayırır (işığın intensivliyi elektronların kinetik enerjisindən asılıdır). Işıq lüminoforla yalnız bir neçə millisaniyə ərzində şüalandığına görə bütün görüntünü saniyədə 30 dəfəyə yaxın tezliklə, bəzən isə daha tez-tez *regenerasiya etmək* (təzələmək) lazımdır.

Məsələn, *rəng dərinliyini* nöqtədə 8 bit (256 rəng) saxlamaqla, rastrın 800 × 600 nöqtəyə qədər (üfiqi xətt üzrə 800 nöqtə və şaquli xətt üzrə 600 nöqtə) böyüdülməsi üçün yaddaşın həcmini iki dəfə (800 × 600 = 480000 nöqtə, yəni 480 kbayta yaxın) artırmaq tələb olunur. Rastrın ölçüsü 1024 × 768, 1280 × 1024, 1600 × 1200 olarsa, daha böyük həcmdə yaddaş lazım olacaq. 256 rəng şəklin mükəmməl alınması üçün kifayət deyil, ona görə də rəng dərinliyi artırıldı: nöqtədə 16 bit – high color (*ing.* "ən yaxşı rəng"), nöqtədə 24 bit – true color (*ing.* "əsl, həqiqi rəng"). İkinci ona görə belə adlandırıldı ki, oxşar rəng çalarından (16,7 milyon rəng) istifadə etməklə alınan görüntü göz üçün demək olar ki, idealdır.

Asanlıqla hesablamaq olar ki, əgər hər bir nöqtəyə 24 bit (yəni 3 bayt) sərf olunursa, onda 800 × 600 nöqtədən ibarət şəklin saxlanması üçün təxminən 1,4 Mbayt (800 × 600 = 1440000 bayt) lazımdır. Əgər fiqur çox da böyük deyilsə, məsələn, üçbucaqlıdırsa, onda bu sərfəli deyil. Lakin qrafik informasiyanın saxlanması üçün digər üsullar da mövcuddur. Tutaq ki, rəssam karandaşla kağızda şəkil çəkir və onun biləyində bütün əl hərəkətlərini kompüterə ötürən xüsusi qurğu var. Onda çəkilmiş şəkli təkcə nöqtə (rastr) toplusu kimi deyil, həm də rəssamın əl hərəkətlərinin ardıcıllığı kimi yaddaşda saxlamaq (və sonra çəkmək) olar. Qrafik informasiyanın bu cür əks etdirilməsi *vektor metodu* adlanır. Şəkil, parçanın başlanğıc koordinatlarını və nisbi sürüşməsini göstərən vektorlarla verilir. Əgər hesab etsək ki, parçaların koordinatları iki bayt yer tutur, onda üçbucaqlının yaddaşda saxlanması üçün cəmi 2 × 2+2 × 3 × 2 = 16 bayt (başlanğıc nöqtənin koordinatları və üç vektor) tələb olunacaq.

VEKTOR VƏ RASTR DİSPLEYİ. SXEMLƏR

Vektor displeyində elektron şüa istənilən iki nöqtə arasında fasiləsiz hərəkət edərək, dəqiq düz xətlər əmələ gətirir. Bu cür displeylərdə hərflər qısa vektorlardan ibarətdir. Displey ikiölçülü koordinatlarla verilən müxtəlif sadə fiqurları, məsələn, qövs və xətləri də çəkə bilir.



Sonrakı çəkilmə üçün mövqelərin yadda saxlanması displeyin işidir.

Vektor qrafikası displeylərdə əvvəllər olduqca geniş şəkildə tətbiq olunurdu, çünki bunun üçün kiçik həcmdə yaddaş tələb edilirdi. Onun əsas çatışmayan cəhəti bütöv (kəsilməyən) sahələri göstərə bilməməsi idi, belə ki, obyektlər (həm 2-ölçülü, həm də 3-ölçülü) "məftil" karkaslar şəklində təsvir edilirdi. Əgər ekranda çoxlu sayda elementar fiqurlardan ibarət olduqca mürəkkəb vektor təsviri olarsa, onda, o, təzələnməyi çatdırmayacaq və görüntü titrəməyə başlayacaq.

Vektor displeylərindən fərqli olaraq, rastr displeylərində şüa bütün kadr boyunca (televizorda olduğu kimi) hərəkət edir və bu zaman şüalar rəngli displeydə kənarlaşmanı deyil, yalnız şüaların intensivliyini idarə edirlər. Rastr displeylərinin üstün cəhətlərindən biri ondan ibarətdir ki, şəkilin təsviri onun mürəkkəblik dərəcəsindən (yəni, onu təşkil edən obyektlərin sayından) asılı deyil, ona görə də titrətmə problemi yaranmır.

Vektor displeyində şüa xətkeş və reyssına kimi çəkildiyi halda, rastr displeyinin işi *puantilizmin* (bədii təsvirdə nöqtələr şəklində çəkmə manerası) elektron analoquna əsaslanır.

Bu metod tətbiq edilərkən nəinki ayrı-ayrı piksellər gözə çarpacaq, həm də üfüqi və şaquli olmayan bütün xətlərin sonluqları pilləvari şəkildə olacaq. Lakin çözümlülük artırıldıqda, yəni ekranın ölçüsü dəyişir, lakin rastr elementlərinin sayı artırılır, belə "pilləvari" effekt minimuma enəcək. Vektor təsvirləri asanlıqla miqyaslaşdırılır, buna xüsusi proqram təminatından istifadə etməklə, rastr displeyində də nail olmaq çətin deyil.

XX əsrin 90-cı illərinə qədər rastr displeyləri (və fərdi EHM-lər) öz rəqiblərini – vektorlu hesablama texnikalarını praktiki olaraq bazardan sıxışdırıb çıxartmağa nail oldu.



Jorj Söra. Qrand-Jatt adasında bazar günü günortadan sonra. 1884-cü il, İncəsənət İnstitutu. Çikaqo.

Puantilizm (*fransız.* pointiller – "nöqtələrlə yazmaq") XIX əsrdə fransız impressionist rəssamı Jorj Söra tərəfindən irəli sürülmüşdür.

QRAFİK FAYL FORMATLARI

Görüntünü ötürmək və ya fayl şəklində diskdə saxlamaq üçün müəyyən formatlar yaradılmışdır (görüntünün kodlaşdırılmasını əks etdirən metodlar). Onların arasında daha populyar olanlar bunlardır: BMP, PCX, GIF, PNG, TIFF və hamının yaxşı tanıdığı JPEG.

BMP (*ing.* "Bit MaP" – "bit xəritəsi") – Microsoft şirkətinin Windows əməliyyat sisteminin qrafik faylları üçün "doğma" formatdır, çünki bu sistemin daxili formatına daha çox uyğun gəlir.

Rastr massivi adlanan görüntünün formatı hər bir pikselin rənginin kodlaşdırılması üçün istifadə edilən bitlərin sayından asılıdır (pikseldə 1, 4, 8, 16 və ya 24 bit). Məsələn, 256 rəngli görüntü halında hər bir pikselə faylın *rənglər cədvəlindəki* rəngin nömrəsini saxlayan 1 bayt (8 bit) ayrılır. Beləliklə, əgər BMP faylının rəng cədvəlindəki RGB rəng modelində R/G/B = 255/0/0 saxlanarsa, onda pikselin 0 qiymətinə (rastr massivində) parlaq-qırmızı rəng qarşı qoyulacaq. Rənglər cədvəli və ya *palitra* massivdir. Burada massivin indeksi – palitranın rənginin nömrəsi, onun məzmununu isə – rəngi onun üç əsas komponentinin parlaqlıq dərəcəsinə görə təsvir edən üç ədəddir: R/G/B (qırmızı, yaşıl, göy).

Windows 95 və Windows 98-in meydana gəlməsi PCX formatında bəzi dəyişikliklərin edilməsinə səbəb oldu: artıq nöqtədəki bitlərin sayını 32-yə çatdırmaq, görüntünün ölçüsünü isə 16 bitdən 31 bitədək (və ya 2 Gbitədək) böyütmək olur. Ölçünü görüntünün çəkilməsi "istiqamətini" göstərən mənfi ədədlə də vermək mümkündür (sol aşağı küncdən yox, sol yuxarı küncdən).

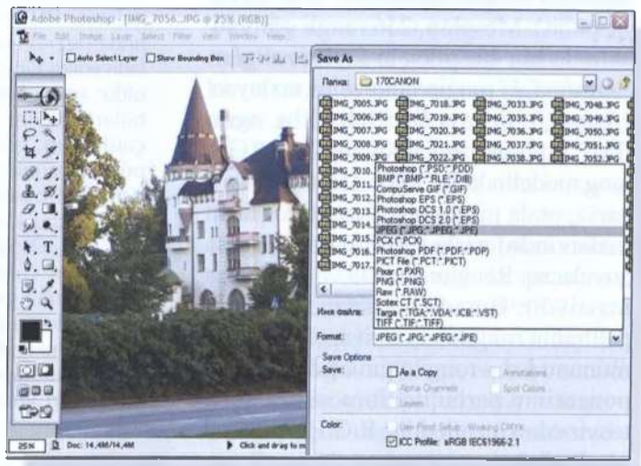
Qrafik faylı anbara (videokarta) çəlləkləri (baytları) daşıyan yük maşınlarının axını kimi təsəvvür etmək olar. Anbara çatarkən çəlləklərin içindəkiləri hissələrə bölərək, müəyyən rəflərə düzürlər. Hər bir çəlləyi açıb lazımi hissələri çıxarmaq və onları rəflərə düzmək az vaxt aparmır. Lakin əgər hər bir rəfə yığılmalı olan hissələr əvvəlcədən ayrıca bir çəlləyə doldurulsaydı, ümumi iş vaxtına qənaət edilmiş olardı. Bu, elə avadanlıqdan asılılıq deməkdir.

ÇÖZÜMLÜLÜK

Çözümlük çap edərkən, originalın həqiqi ölçüsünü saxlamaq tələb olunduqda vacibdir. Fotoşəkil skanerlə oxunarkən görüntünün diskret (rəqəmli) formaya çevrilməsi prosesi baş verir. Görüntü piksellərdə müəyyən ölçünü alır və bu zaman fotoşəklin ilkin həqiqi başlanğıc ölçüsünə dair informasiyanın itirilməməsi üçün 1 sm-də və ya 1 düymdə görüntünün neçə nöqtəsinin yerləşdiyini göstərən çözümlülüyü yadda saxlamaq tələb olunur. Çözümlülük (istər üfüqi, istərsə də şaquli) piksellərdə alınan ölçülərin ilkin görüntünün müvafiq ölçüsünə bölməklə hesablanır.

Əgər görüntü üfüqi xətt üzrə 800 nöqtədən ibarətdirsə, fotoşəklin ilkin vəziyyətindəki uzunluğu isə 12 sm olarsa, onda skanerlə oxunma zamanı çözümlülük 1 sm-də 800/12 = 67 nöqtə olacaq. Çözümlülük nə qədər yüksək olarsa, cap zamanı görüntü daha keyfiyyətli alınacaq, lakin rastr massivi də bir o qədər böyük olacaq, buna uyğun olaraq, fayl bir o qədər çox yer tutacaq.

Pikseldə 16 və 24 bitlik rəng dərinliyi olan BMP fayllarının rəng cədvəli yoxdur. Belə fayllarda rastr massivi piksellərinin qiymətləri bilavasitə RGB rənglərinin qiymətlərini verir. Görüntünün formatındakı piksellər görüntünün aşağıdakı sətrindən başlayaraq soldan sağa yerləşmə qaydasında saxlanılır. Əgər açılma sətrindəki baytların sayı cüt deyilsə, onda rastr massivinin verilənlərini "bərabərləşdirmək" üçün belə sətrə daha bir bayt əlavə olunur.

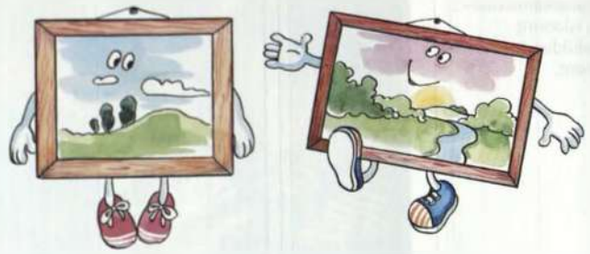


PCX – ZSoft firması tərəfindən hazırlanan və Microsoft şirkətinin MS DOS əməliyyat sistemindəki PC Paintbrush proqramı üçün istifadə edilən standartdır. Lakin PSX formatı BMP qədər populyar ola bilmədi. Buna baxmayaraq PSX, PC texnikasında yayıldı. Xüsusi qrafik redaktorlar – Corel Draw, Ulead PhotoExpress, Adobe PhotoShop – bu formatı dəstəkləyir. PSX aparatdan asılı formatdır. Bu format informasiyanın faylda videokartda olduğu kimi saxlanılması üçün nəzərdə tutulmuşdur. Əvvəllər IBM PC-də yaddaşı fasiləsiz ikiölçülü parçalara (*planlara*) bölünən 16-rəngli EGA adapterləri mövcud idi. Pikselin rənginin tərkib hissələri (adətən, 1 bitdir) planın müvafiq yerlərində yerləşirdi. Müəyyən rəngdə nöqtəni qoymaq üçün dörd yerin hər birində 1 bitdə düzəliş emək tələb olunurdu. Elə alınırdı ki, sanki rəngli təsvir dörd monoxrom ("ağ-qara" ilə qarışdırmamaq) plandan ibarətdir. Köhnə proqramlarla uyğunluğu saxlamaq üçün müasir videoadapterlər oxşar EGA rejimlərdə işləyə bilir. True color halında PCX-də həmişə üç plan olur, bir plan isə yalnız 8-bitlik görüntüdə (məhz bu halda palitra faylın sonunda olacaq) olur. PCX formatı ekranda görünən şəkilə nisbətən görüntünün rastr massivində çox şeyin saxlanmasına daha çox imkan verir. Sol yuxarı sərhəd görüntünün yuxarı küncündən boş yer qoyur, sərhədlər arasındakı fərq isə görünən hissənin ölçüsünü göstərir. Bu, verilən formatın faksimil aparatlarda istifadəsi üçün nəzərdə tutulmuşdur. Bundan başqa, görüntünü ekrana daha düzgün bağlamaq üçün faylda 1 düymdə

144 nöqtəli çözümlülüyü təsvir edən iki sahə mövcuddur.

GIF (*ing.* "Graphics Interchange Format" – "qrafik mübadilə formatı") ilkin olaraq CompuServe şəbəkəsi üçün nəzərdə tutulmuşdu. Bu format PCX və BMP-dən fərqli şəkildə qurulub. GIF-də qrafik informasiya bir-birinin ardınca gələn bloklardan ibarətdir (hələ 16-cı blok oxunmamış demək olmaz ki, 17-ci blok haradan başlayır). Ona görə də, deyirlər ki, informasiya *verilənlər axını* şəklində saxlanılır (fayl oxunarkən informasiya su kimi fasiləsiz olaraq axır). GIF formatı 256-dan artıq rənglə işləmir. Əvəzində hər fayl özündə bir yox, bir neçə görüntü saxlaya bilər. GIF formatının bir-birindən öz kodlarına görə fərqlənən iki versiyası mövcuddur: GIF87*a* və GIF89*a*. GIF87*a* 1987-ci ilin mayında buraxılmışdır. Hər bir fayl başlıqdan, vacib olmayan palitradan və əslində şəkil haqqında informasiyadan ibarətdir. *Qlobal palitradan* faylda yerləşən və xüsusi palitrası olmayan bütün görüntülər üçün istifadə edilir. Şəkillərin palitrası və qlobal palitra tamamilə eyni cür qurulmuşdur (BMP-də olduğu kimi): üç bayt nöqtənin rənginin qırmızı, yaşıl və göy kimi tərkib hissələrini verir.

GIF formatlı fayllar şəbəkədən yüklənərkən özünəməxsus şəkildə görünürlər: əvvəlcə anlaşılmaz olur, sonra getdikcə aydınlaşır. Əgər İnternetə qoşulma sürəti yüksək deyilsə, onda faylın yüklənməsini "canlı şəkildə" müşahidə etmək olar. Əvvəlcə şəklin 0-cı, 8-ci və 16-cı sətirləri ötürülür (yəni hər səkkizinci sətir), ikinci sətirdən başlayaraq növbələşmə 4 sətirdən bir baş verir: 2-ci, 6-cı, 10-cu... Nəhayət, axırda birinci sətirdən başlayaraq bütün tək sətirlər ötürülür: 1-ci, 3-cü, 5-ci... Bu cür görüntü bütün şəkli görməyə imkan verir, hətta görüntünün yalnız yarısı alınmışsa belə.



GIF89*a* versiyası 1989-cu ilin iyul ayında yaradılmışdır. Ona *idarəedici genişlənmə* adlanan yeni informasiya blokları əlavə edilmişdir.

İnformasiya blokları dörd tipə bölünür:

- Qrafikanın idarə edilməsindən hərəkət edən görüntülərin yaradılması üçün istifadə edilir (*ing.* "animated GIF").
- Mətn bloku, mətni görüntünün üzərində yerləşdirməyə imkan verir. Bu, xüsusən, fonu olduğu kimi saxlamaqla, mətni dəyişdirmək istədikdə əlverişlidir.
- Şərhlər görüntü çıxışa verilərkən buraxılır, lakin onlar, ətraflı informasiyanı (yaradılma tarixi, variantlar, eskizlər) göstərmək üçün müəlliflərə faydalıdır.
- Əlavələrdə xüsusi proqramlar vasitəsilə qəbul edilən xüsusi verilənlər saxlanılır. Digər proqramlar sadəcə olaraq bu bloku buraxırlar.

GIF fayllarının çatışmayan cəhətləri kimi rəng dərinliyinin çox olmamasını (cəmi 256 rəng) və rastr massivinin verilənlərinin LZW qapalı alqoritminin köməyilə sıxılmasını göstərmək olar. Belə ki, əgər yaradılan proqramda GIF



BMP və PCX kimi, GIF də diskdə saxlanarkən az yaddaş tutması üçün verilənləri sıxan xüsusi alqoritmlərdən istifadə edir. Görüntü ölçüsü 255 baytdan çox olmayan bloklar axını şəklində saxlanılır. Hər bir blok baytdan (blokun uzunluğu) və verilənlərin özlərindən ibarətdir. Şəkil qurtardıqdan sonra sıfır uzunluqlu blok gəlir.



JPEG alqoritmi vasitəsilə işlənmiş fotoşəkildən bir fraqment.



Sıxılmamış fotoşəkil.

formatından istifadə edilirsə, onda CompuServe şirkətilə ödənişli lisenziya müqaviləsi bağlamaq lazımdır.

PNG (*ing.* "Portable Network Graphic Format" – "qrafiklərin şəbəkə ilə ötürülməsi formatı", amma "pinq" kimi tələffüz olunur) Tomas Buttelin başçılıq etdiyi amerikalı proqramçılar qrupunun (23 nəfərin) təşəbbüsü ilə yaradılmışdır.

Qrafik faylların əksəriyyətinin formatı bir-iki proqramçı tərəfindən hazırlanırdı və bunun nəticəsi kimi yaradılan formatı yeni tələblərə uyğun modifikasiya etmək çətin olurdu.

PNG formatı xüsusi olaraq yaradılan komitə tərəfindən hazırlanmışdır və PNG-nin sənədləşdirilməsinə xüsusi diqqət yetirilmişdir. Ona görə də onun bütün sonrakı modifikasiyaları artıq onlarla işləyən qrafik redaktorların modifikasiya olunmasını tələb etmir.

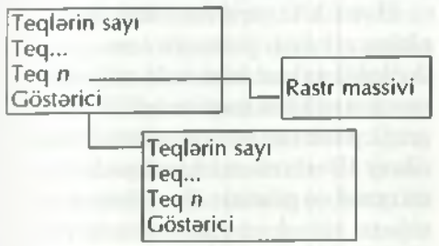
Ümumiyyətlə, PNG formatı GIF-ə alternativ kimi hazırlanmışdı, həm də PNG ilə GIF89*a* bir çox ümumi xüsusiyyətlərə malikdir. Məsələn, onların hər ikisi verilənlər axını kimi təşkil olunmuşdur, görüntünü mərhələli şəkildə çıxarır, əsas rəngi şəffaflaşdıra bilir və s. Lakin PNG təkcə true color görüntüləri (24 bit) yox, həm də pikseldə 48 bit saxlamağa imkan verir. PNG itkisiz sıxmanı təmin edən LZ77 açıq sıxma alqoritmindən istifadə edir (LZW-dən fərqli olaraq LZ77 ödənişli lisenziyalaşdırma tələb etmir).

TIFF (*ing.* "Tagged Image File Format" – hərfi tərcüməsi: "yarlıqlarla" – teqlərlə təchiz edilmiş görüntülərin formatı") skaner vasitəsilə daxil edilən ağ-qara təsvirlərin saxlanmasının standartı kimi 1986-cı ildə Aldus Corporation tərəfindən hazırlanmışdır. Bu, masaüstü nəşriyyat sistemlərində tətbiq edilən rastr qrafiki kimi ən çox dəstəklənən formatdır. TIFF 4.0 versiyası sıxılmamış rəngli görüntüləri saxlamağa imkan verir. 1988-ci ilin avqustunda meydana çıxan növbəti modifikasiya artıq rəng palitrasından və LZW sıxma alqoritmindən istifadə edirdi. 1992-ci ilin iyununda TIFF 6.0 versiyası təkcə RGB-görüntünü deyil, həm də CMYK və YUV kodlaşdırmalarını, həmçinin JPEG sıxma alqoritmini dəstəkləyirdi.

TIFF qrafik faylların ən mürəkkəb formatlarından biri hesab edilir. Fayl *teqlər* və yaxud sahələrin siyahısından və rastr massivlərindən ibarətdir. Siyahıların və massivlərin mövqeyi müəyyən edilməmişdir, yəqin ki, elə buna görə də bu mürəkkəb format sayılır. TIFF-in quruluşu fayl sisteminin quruluşuna bənzəyir: siyahı – kataloqdur, siyahının hər bir elementi (teq) – faylın adıdır, rastr massivi – faylın verilənləridir.

Fayl sistemində olduğu kimi, teq rastr verilənlərinə, kataloq isə növbəti kataloqa istinadı özündə saxlaya bilər.

TIFF 70-dən çox müxtəlif teq tipinə malikdir. Məsələn, teq tiplərindən birisi



görüntünün eni haqqında informasiyanı (piksellərlə), digəri görüntünün hündürlüyü haqqında informasiyanı, növbətisi isə rənglər cədvəlini saxlayır... Həmçinin teqlərin xüsusi tiplərini təyin etmək və ya aydın olmayanlarını nəzərə almamaq olar.

JPEG adı Joint Photographic Experts Group (Fotoqrafiya üzrə birləşmiş ekspertlər qrupu; 1987-ci ildə yaradılmışdır, Ümumdünya Standartlaşdırma Təşkilatına – ISO-ya daxildir) komitəsinin adının baş hərflərindən ibarətdir. Əvvəllər hazırlanmış sıxma metodlarının əksəriyyəti praktiki olaraq yüz minlərlə rəngdən ibarət olan görüntülərin sıxılması üçün əlverişli deyil. JPEG formatı

Qrafik formatların müqayisəli cədvəli

Format	BMP	PCX	TIFF	GIF	PNG	JPEG
Bitlərin/ piksellərin sayı	24	24	24	8	48	24
Rənglərin sayı	16,7 mln	16,7 mln	16,7 mln	256	281,475 mlrd	16,7 mln
Görüntünün ölçüsü	65535x 65535	65535x 65535	4,3 mlrd	65535x 65535	2 mlrd x 2 mlrd	65535x 65535
Sıxma metodları	RLE	RLE	LZW, RLE və s.	LZW	LZ77	JPEG (informasiyanın itirilməsilə)
Görüntülərin sayı	1	1	1 və daha çox	1 və daha çox	1	1 və daha çox

3D-SÜRƏTLƏNDİRİCİSİ NƏYƏ QADİRDİR

Sürətləndiricilərin istifadə etdikləri minimal element üçbucaqlıdır. Təsvir olunan bütün şəkillər əvvəlcədən üçbucaqlılara bölünür. Birrəngli üçbucaqlılar həcmli şəkil effekti yarada bilmir, ona görə də şəkilin aydınlaşdırılması prosesində onların bir çoxunun üzərində teksturalar (bütün səthin üzərinə dərhal yerləşdirilən və materialın zahiri görünüşünü verən görüntülər, məsələn, xalça üzərində şəkil, çimərlikdə qum) yerləşdirilir. Binanın kərpic divarlarını təsvir etməkdən ötrü kərpicləri modelləşdirmək üçün bir çox ayırıcı xətlərin çəkilməsi tələb olunacaqdır, lakin tekstura daha az hesablama resursları işlədir və reallığı daha çox əks etdirir. Teksturanı səthdə yerləşdirərkən perspektivi nəzərə almaq lazımdır. Teksturalı obyektlərin müşahidəçilərə nə dərəcədə yaxın olmalarından asılı olaraq, onların müxtəlif görünüşə malik olmaları üçün bu korreksiya vacibdir. Prosesin sürətləndirilməsi üçün bir neçə eynitipli tekstura seriyasını saxlamaq və teksturanı seçərkən artıq ekranda əks etdirilən detallara əsaslanmaq olar. Müvafiq olaraq, əgər obyekt kiçildilirsə, onda onun tekstura örtüyünün ölçüsü də azalacaqdır.

Teksturanı yerləşdirərkən iki yaxın obyekt arasında sərhəd xətti tez-tez hiss edilir. Bu effektlərin aradan qaldırılması üçün xüsusi filtrasiya metodları tətbiq edilir. Düz xətlərdəki pilləvari sonluqlar rastr displeylərinin çoxdankı problemidir. Görüntülərin dəqiq qıraqlarının alınması üçün qırağın rəngindən fonun rənginə rəvan keçid yaradılır. Obyektlərin sərhədində yerləşən nöqtənin rəngi iki sərhəd nöqtəsinin rəngləri arasında orta bir rəng kimi təyin edilir. Bu zaman, bəzən qıraqların "malalanması" kimi kənar effekt yaranır.



isə informasiyanın itirilməsilə nəticələnən sıxma metodudur. Bu format qrafik faylın standart formatı kimi təyin edilməsə də, onun əsasında çoxlu sayda yeni formatlar yaradılmış və bir sıra köhnə formatlar yaxşılaşdırılmışdır.

ÜÇÖLÇÜLÜ QRAFİKA

Adi zolaqlı rəsmlər parçalar, çevrələr və çoxbucaqlılar əsasında modelləşdirilir. Üçölçülü qrafikdə iki koordinata *z* koordinatı və yeni obyektlər: çoxüzlülər, piramidalar, kürələr, silindrlər və üçölçülü səthlər əlavə edilir. Bütöv (kəsilməz) obyektləri də sadə elementlərdən formalaşdırmaq olar. Üçölçülü səhnənin obyektlərinin verilməsindən sonra onların vizuallaşdırılması problemi yaranır. Əgər görüntü statik deyilsə (məsələn, oyunda və ya trinajorda), onda üçölçülü səhnənin bütün vəziyyətlərini nəinki saniyədə 30 dəfə saymaq, həm də birbirlərini ardıcıl şəkildə əvəz edən şəkilləri ekranda ardıcıl olaraq əks etdirmək lazımdır.

Hətta XXI əsrin texnikası üçün, bu, olduqca bahalı prosesdir, ona görə də üçölçülü səhnə daha sadə obyektlərin məcmusu kimi təqdim edilir. Xüsusi qrafik prosessor isə (videokarta birləşdirilmiş 3D-sürətləndiricisi) şəkli ekranda qurur və göstərir. Fərdi kompüterin müasir videokartı (monitorla birlikdə) xalis rastr displeyindən 3D-sürətləndiricisi olan intellektual qrafik displeyə çevrilmişdir.

Gəlcəkdə nə gözlənilir? XX əsrin kompüter qrafikasına dair bütün ideyalar XXI əsrdə fərdi ev kompüterlərində həyata keçirilmişdir. Belə təsəvvür yaranır ki, son həddə çatılmışdır.

Üçölçülü səhnəni yaradarkən əsas problem hansı obyektlərin göründüyünü və onların necə işıqlandırıldığını müəyyən etməkdir. Obyektlərin qarşılıqlı şəkildə yerləşməsi haqqında informa-

siyaya malik olmaq da vacibdir. Adətən, bu məsələnin həlli üçün *Z-bufer* tətbiq edilir. Z-buferdə rastrın hər bir pikselinə bir qiymət olmaqla, bütün görünən piksellərin (*z* koordinatlarının) uzaqlıq qiyməti saxlanılır. Təsvir olunan obyektin növbəti pikseli hesablanarkən (*x*, *y* koordinatları ilə), alınan *z* koordinatı Z-buferdə saxlanılan *z* koordinatının qiymətilə müqayisə edilir.

Əgər yeni piksel Z-buferdə yazılandan daha çox dərinliyə malikdirsə, deməli, obyektin bu nöqtəsi hansısa başqa bir obyekt tərəfindən örtülmüşdür və ona görə də ekranda görünmür. Əksinə, əgər piksel ön planda peyda olarsa, onda onun *z* koordinatının qiyməti Z-buferə yazılacaq. Bu piksel başqa obyekt tərəfindən örtülənə qədər ekranda da görünəcək. Üçölçülü şəkil hesablanandan sonra Z-buferdə qalan bütün piksellər görüntüdə istifadə ediləcək.

Baxmayaraq ki, Z-bufer böyük yaddaş tutur, 3D-sürətləndiricisində qurulmuş Z-buferləşdirmə şəklin hesablanma sürətini son dərəcə artırır (24-mərtəbəli Z-bufer 640 × 480 çözümlülüyündə 900-ə yaxın kbayt tələb edir). Bu yaddaş 3D-sürətləndiricisinin daxili yaddaşı olmalıdır. Z-buferin mərtəbəliliyi (*z* koordinatında bitlərin sayı) ən mühüm göstəricilərdən biridir; mərtəbə nə qədər yüksək olarsa, *z* koordinatının diskretliyi də bir o qədər yüksək olar və uzaqda yerləşən obyektlərin çəkilməsi dəqiq yerinə yetirilər.

Sürətləndiricilər, adətən, bəzi effektlərin, məsələn, yarış avtomobili yolda şütüyərkən yaranan toz dumanının əks etdirilməsi imkanına malik olur. Təsvir olunan piksellər dumanın rəng dərinliyindən asılı olaraq onun rəngilə qarışır. Həmin alqoritmin köməyilə obyektlər zəif dumana bürünür ki, bu da məsafə illüziyasını yaratmağa imkan verir.

3D-sürətləndiricisinin təmin etdiyi daha bir effekt şəffaflıqdır. Ətraf mühitdə



qeyri-şəffaf obyektlərlə yanaşı, şəffaf və yarımşəffaf obyektlər də mövcuddur. Yarımşəffaf effekti nöqtələrin rənginin ön (yarımşəffaf) və arxa plandakı rənglərin kombinasiyası yolu ilə yaradılır. Adətən, şəffaflıq əmsalı (alpha) 0-dan 1-ə qədər qiymətləndirilir. Bu zaman, rəng aşağıdakı düstur üzrə hesablanır:

pikselin rəngi = (alpha) (ön plandakı pikselin rəngi) + (1 - alpha) (arxa plandakı pikselin rəngi).

(arxa plan pikseli, Z – buferdə yerləşən pikseldir).

Hər bir növbəti kadr formalaşdırılarkən 3D-sürətləndiricisi bütün qurulma yolunu yenidən keçir, ona görə də, o yüksək sürətə malik olmalıdır. Hərəkətə rəvanlıq vermək üçün ikiqat buferləşdirmə tətbiq edilir: buferin biri şəklin göstərilməsi üçün, digəri isə yeni şəklin yaradılması üçün nəzərdə tutulur. Bir buferin tərkibi ekranda əks etdirilərkən digərində hazırlıq işləri görülür. Növbəti kadr hazır olduqda buferlər yerlərini dəyişir. Beləliklə, ekranda həmişə əla şəkil müşahidə edilir.



2D-sürətləndiriciləri artıq 90-cı illərdən fərdi kompüterlərin videokartlarında tətbiq olunur. Onlar siçanın kursorunun ekran boyu hərəkətini təsvir edir və görüntünün ekranın düzbucaqlı pəncərəsində hərəkətini təmin edir.

RƏQƏMSAL FOTOQRAFİYA

Heç kimə sirr deyil ki, görüntülərin kompüterdə işlənməsi bu gün adi hala çevrilib. Saytlar üçün qrafika, fotomontaj, jurnal və qəzetlərin səhifələnməsi – bütün bunlar və bir sıra digər işlər EHM-də yerinə yetirilir.

Rəqəmsal fotonun imkanlarına nümunə.



Hazırda görüntülərin kompüterdə saxlanması üçün bir sıra formatlar mövcuddur. Onların arasında ən geniş yayılanı və istifadə ediləni çox güman ki, JPEG-dir.

Əgər bu cür faylda yerləşən şəkil böyük həcmlidirsə, məsələn, üfüqi və şaquli xətt üzrə minə yaxın nöqtədən ibarətdirsə, onda belə bir şəkli əks etdirən informasiya verilənlərin daşıyıcısında olduqca çox yer tutacaq. Faylların ölçülərini kiçiltmək üçün sıxma metodu tətbiq edilir, özü də bu zaman informasiya itə də bilər, itməyə də bilər.

Sıxma proqramları ilkin görüntünü təhlil edir və "lazımsız" nöqtələri atır. Bu analiz nə qədər kobud olsa, görüntü bir o qədər tez və güclü şəkildə sıxılır. Güclü sıxma zamanı görüntü hiss ediləcək dərəcədə təhrif oluna bilər. Görüntülərin sıxılmasının müasir metodları minimal itkilərlə çox yaxşı nəticələr əldə etməyə imkan verir.

Görüntünün rəqəmsal formata çevrilməsi xüsusi qurğuların – *skanerlərin* köməyilə həyata keçirilir. Skanerin əsas hissəsi işığa həssas elementdir. O, kağızdakı ilkin görüntüdən əks olunan işıq axınını tutur və elektrik siqnalı generasiya edir ki, həmin siqnal ilkin şəkli təsvir edən verilənlər selinə çevrilir. Bu cür axın artıq kompüterdə işlənə bilər.

Adi gün işığı, məlum olduğu kimi, işığa həssas element üçün yetərli deyil, ona görə də daranan sahəni işıqlandırmaq üçün skaner əlavə olaraq güclü lampa ilə təchiz edilir.

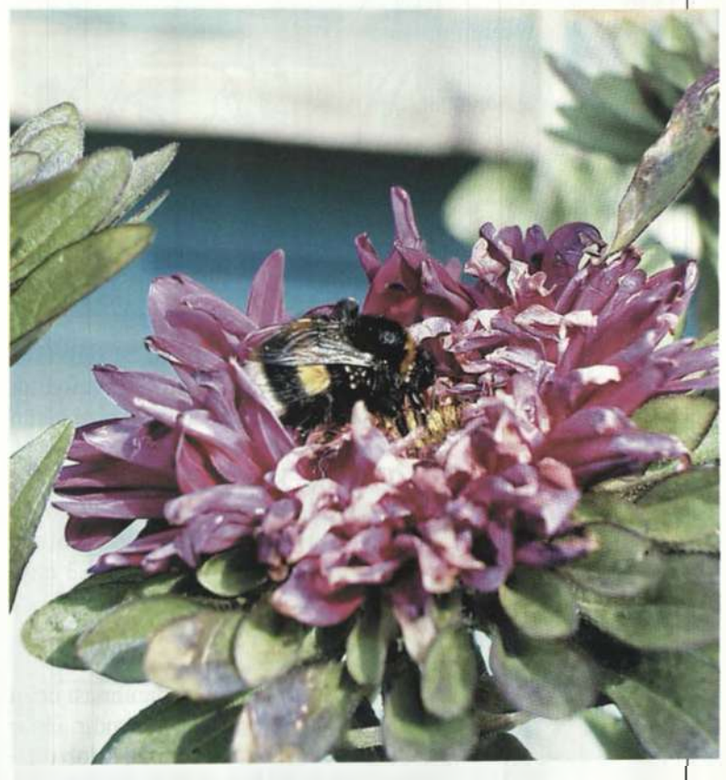
İşığa həssas element kifayət qədər bahalıdır, bu səbəbdən də o, kiçik ölçüdə hazırlanır və görüntü boyu addım-addım hərəkət edən şassi üzərində quraşdırılır, verilənlər isə kompüterə hissə-hissə ötürülür. Kompüterdəki idarəetmə proqramı şəklin ayrı-ayrı hissələrini birinə "yapışdırır" və son nəticəni ekranda müşahidə etməyə imkan verir.

Rəqəmsal görüntüləri yaratmaq üçün istifadə olunan digər bir qurğu rəqəmsal fotokameradır. Onun iş prinsipi adi plyonkalı kameranın iş prinsipinə çox oxşayır, fərq yalnız ondadır ki, plyonkanın əvəzinə işığa həssas matrisdən istifadə edilir. Matris nə qədər çox nöqtə emal edə bilirsə, şəkil bir o qədər keyfiyyətli alınır.

Rəqəmsal kameralar xüsusi məlumat daşıyıcıları ilə təchiz edilir. Həmin daşıyıcılara geniş yayılmış formatda olan fayllar yazılır. Daşıyıcılar disket, kompakt-disk, sərt disk və fləşyaddaş ola bilər. 1280 × 1024 nöqtə ölçüsündə olan (təxminən 9 × 12 standart formatına uyğun gəlir) 35 fotoşəkil 8 Mbayt yer tutur. Daha doğrusu bu ölçüdə olan təxminən 3 min fotoşəkli bir kompakt diskdə yerləşdirmək olar.

Rəqəmsal kameradan istifadə edərkən nəinki fotoşəklin aşkarlanması aradan qaldırılır, həm də şəkil çəkmə daha ucuz başa gəlir. Əvvəla ona görə ki, plyonkalara pul xərcləmək lazım gəlmir; ikinci isə, rəqəmsal kamera ilə çəkərkən, şəklin pis alınacağından çəkinmək lazım deyil, çünki onu dərhal görüb pozmaq (rəqəmsal yaddaşı yeniləri üçün boşaltmaq məqsədilə) olar. Bu səbəbdən də valideynlərin əksəriyyəti öz uşaqlarına rəqəmsal fotoaparatlar hədiyyə edirlər ki, yeni texnikanı öyrənənə kimi onları müflis eləməsinlər.

Rəqəmsal görüntünün emalı üçün bir çox proqramlar mövcuddur. Bu proq-



Rəqəmsal makroçəkiliş.

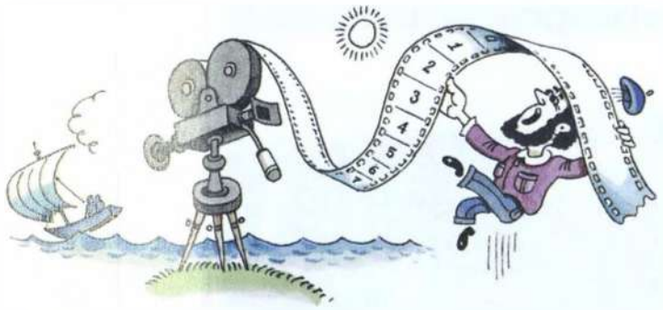
ramlar yaradıcılıq üçün – təsvirin ölçüsündə sadə dəyişiklik etməkdən tutmuş ona fantastik effekt verməyə qədər – geniş imkanlar yaradır.

Rəqəmsal fotoqrafiyanın digər üstün cəhətləri ondan ibarətdir ki, şəkil korlanmır, köhnəlmir, onda cızıqlar qalmır, fotoşəkili cəmi bir neçə dəqiqə ərzində elektron poçt vasitəsilə dünyanın istənilən nöqtəsinə göndərmək və əlbəttə ki, İnternetdə ev səhifəsində yerləşdirmək mümkündür.

RƏQƏMSAL VİDEO

Kompüterdə təkcə oynamaq, müəyyən məsələləri həll etmək, yazı yazmaq yox, həm də bəzən görüntülərinin keyfiyyətinə görə adi videomaqnitofon və ya televizordan əhəmiyyətli dərəcədə üstün olan filmlərə baxmaq olar.

Məlum olduğu kimi, videogörüntü kadrlar ardıcıllığından başqa bir şey



deyil. Filmin hər saniyəsi 24 müxtəlif kadr toplusudur. Beləliklə, bir saniyəlik videonu yadda saxlamaq üçün bizə bütün 24 kadrı saxlamaq lazım gəlir.

Tutaq ki, kadrın ölçüsü 512 × 384 nöqtədən ibarətdir. İstənilən nöqtə 16 milyon rəngdən birinə boyana bilər, yəni onun əks etdirilməsi üçün 3 bayt yaddaş lazım gələcək:

$$512 \times 384 \times 3 = 589824 \text{ bayt.}$$

Deməli, bir kadrın saxlanılması üçün 600 kbayta yaxın yaddaş lazımdır. Əgər həm də səsi yadda saxlamaq tələb olunarsa, onda kadrın ölçüsü böyüyəcək. Bu zaman, məsələn, çox da böyük olmayan sıxlıq ilə iki saatlıq videoyazının saxlanması üçün 10 Gbaytdan artıq yaddaş lazım gələcək ki, bu da hətta müasir informasiya daşıyıcıları üçün o qədər də kiçik deyil.

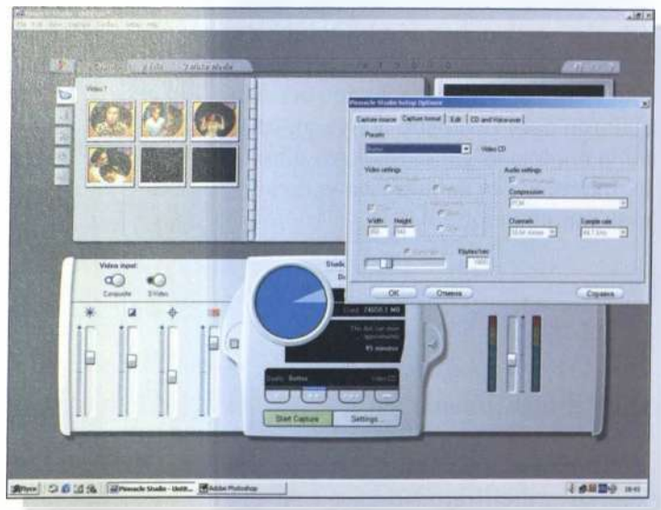
Videonu rəqəmsal daşıyıcılarda saxlamaq üçün videogörüntülərin sıxılması texnologiyasından istifadə edilir. Ən geniş yayılmış sıxma formatı MPEG-dir. Bu format ötən əsrin sonunda Motion Picture Engineering Group-un alimləri tərəfindən hazırlanmışdır (MPEG-in təkmilləşdirilməsilə bağlı işlər indi də davam etdirilir). Hazırda MPEG formatı təsvirin mürəkkəblik dərəcəsindən asılı olaraq, onu 10 dəfədən 100 dəfəyədək və daha çox sıxmağa imkan verir.

MPEG üç hissədən ibarətdir: Audio, Video və System (sonuncu digər ikisinin birləşməsini və sinxronlaşmasını yerinə yetirir). MPEG-in bir neçə müxtəlif standartı mövcuddur. Onların arasında daha geniş istifadə olunanı bunlardır: MPEG-1, MPEG-2 və MPEG-4. Bu standartlar bir-birindən sıxma dərəcəsinə, habelə çıxışdakı görüntünün keyfiyyətinə görə fərqlənirlər.

MPEG-də bir neçə kadrdan vahid tam kadr düzəldilir. Digərləri isə alqoritmlərin köməyilə süni yaradılan kadrlardır. Faktiki olaraq yüksək sıxma dərəcəsinin hesabına informasiyanın yalnız bir hissəsi saxlanılır. Əsas kadr nə qədər seyrək şəkildə yerləşdirilərsə, görüntünün keyfiyyəti bir o qədər aşağı olar. Ona görə də filmin faylının ölçüsü ilə onun keyfiyyəti arasında kompromisin tapılması vacibdir.

MPEG faylında səs bu qaydada kodlaşdırılır: insan qulağının ayırd edə bilmədiyi bütün səslər atılır. Səs axınının bu cür emalının hesabına filmdəki səs cığırının ölçüsü azalır. Bundan başqa, MPEG-in bəzi növləri bir neçə səs cığırını dəstəkləyir, məsələn, filmə baxarkən onun ingilis versiyasından azərbaycan versiyasına, yaxud əksinə keçmək olar. Bu, müasir DVD (Digital Versatile Disk – "müxtəlif təyinatlı rəqəmsal disk") qurğularında olduqca geniş istifadə edilir.

Videomontaj proqramı işinin ekrandan fotosu. Montajın parametrlərinin seçilməsi.



ÖZÜ ÜÇÜN REJİSSOR

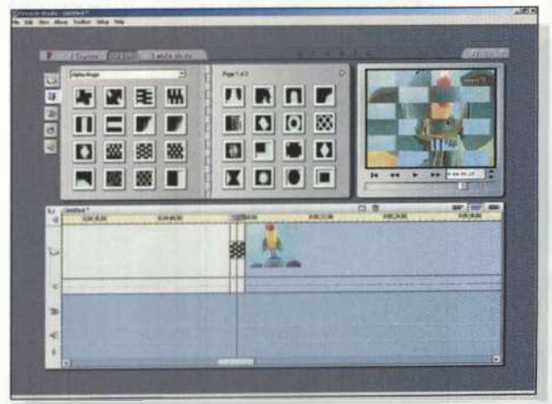
Əlbəttə, filmə baxmaq üçün kompüterdən istifadə etmək yaxşıdır, amma daha maraqlısı onun köməyilə şəxsi film yaratmaqdır. Bu məsələnin həlli üçün bəzi əlavə avadanlıqlar və səbir tələb olunur. Əvvəlcə videokamera ilə çəkilmiş materialı rəqəmsal formaya çevirmək və informasiya daşıyıcısında saxlamaq lazımdır. Çevirmə prosesi videogörüntünü dəstəkləyən xüsusi platadan istifadə etməklə həyata keçirilir. Əgər kompüter artıq TV-tüner platasına (monitorun ekranından teleproqramlara baxmaq mümkündürsə) malikdirsə, artıq kompüterdən heç nə tələb olunmayacaq. Videomaqnitofon və ya kamera kompüterə qoşulduqdan sonra videoaxının diskə yazılması üçün proqramlar işə düşür.

Ümumiyyətlə, adi televizorun ekranında görüntünün ölçüsü 576 sətir təşkil edir. Lakin televizorda təsvir hərəsi 288 sətirdən ibarət olan iki hissədən tərtib edilir. Birinci hissədə bütün tək sətirlər, ikincisində isə cüt sətirlər yerləşir. Televiziya ekranında bunu, adətən, sezmək olmur, amma video EHM-də işlənilərkən, xüsusilə, görüntülərin tez-tez dəyişdiyi səhnələrdə şəbəkə effektini müşahidə etmək olar. Şəbəkənin aradan qaldırılması üçün təsvirin bütöv şəkildə olmasını təmin edən xüsusi proqram-filtrlər tətbiq edilir. Bu proses Deinterlacing adlanır.

Kompüterin sürəti eyni vaxtda həm videonu, həm də səsi emal etməyə imkan verərsə, onda səsi görüntü ilə birlikdə (əksər hallarda, bu, məhz belə olur) yazmaq olar. Amma bunları ayrı-ayrılıqda da yazmaq mümkündür.

Filmin kompüterdə redaktə edilməsi üçün xüsusi bahalı studiya avadanlıqları tələb olunmur, bütün proses olduqca aydın şəkildə baş verir və WYSIWYG prinsipinə uyğun gəlir. Diskdə neçə filmin yerləşməsi isə hansı kodlaşdırma proqramının seçilməsindən asılıdır. Bunun üçün keyfiyyətlə ölçü arasında kompromis axtarmaq lazım gəlir (keyfiyyət nə qədər yüksək olarsa, fayllar bir o qədər çox yer tutur). Videonun saxlanması üçün keyfiyyəti artıran və faylın ölçüsünü azaldan bir çox müxtəlif formatlar mövcuddur. Yaxın gələcəkdə, heç şübhəsiz, İnternet vasitəsilə ideal keyfiyyətli videoya baxmaq imkanı yaranacaq.





Videomontaj proqramı işinin ekrandan fotosu.
Yuxarıda – montaj; aşağıda – xüsusi effektin seçilməsi.

Frame (ing.) – videoaxının ayrılmaz hissəsi – kadrdır.
Frameskip (ing.) – kadrın atılması, düşməsidir. Bu, o vaxt baş verir ki, kompüter kadrı işləyib çatdıra bilmir, məsələn, sürət çatmır, yaxud kadr çox mürəkkəb olur (səhnələrin tez-tez dəyişməsi).
Codec (ing.) – görüntünün kodlaşdırılması və əksinə, kodun açılması proqramıdır. Digər proqramların videonu müəyyən edilmiş formatda "oxuması və yazması" üçün kompüterdə yazılan kiçik proqramdır.

AUDİOMONTAJ

XIX əsrin sonunda radionun kəşf edilməsi nəticəsində kütləvi informasiyanın yeni sahəsi – radio verilişləri meydana gəldi. Radioproqramın hazırlanması üçün maqnit yazısı texnologiyasından istifadə edilməsi inqilabi addım idi. Birbaşa efir veriliş zamanı qonaqların hökmən studiyada olmasını tələb edirdi. Lakin studiyada əvvəlcədən veriliş iştirakçılarının səslərini maqnitofona yazmaq, sonra isə bu yazını səsləndirmək daha əlverişli idi. Verilişi "radio dinləyicilərinin xahişilə" dəfələrlə efirə təkrar vermək olardı. Ən başlıcası isə, onu montaj etməklə efirə xüsusi olaraq "hazırlayırdılar". Bütöv yazıdan bir hissəsini senzura mülahizəsinə görə, bir hissəsini pauzalara, eyni sözlərin təkrarlanmasına görə kəsirdilər.

"Cut and paste" (*ing.* "kəs və yapışdır") texnologiyası audiomontaj zamanı görülən işin vəziyyətini dəqiq əks etdirirdi. Analoq lentli maqnitofon, yapışqan və qayçı montaj studiyasının əsas alətləri idi. Montaj prosesi, həqiqətən, "yaradıcı" iş sayılırdı.

Səsdəki səhvləri tuturdular, qayçı ilə dəqiq işləyərək, parçanın əvvəlində və sonunda lenti kəsirdilər, gərəksiz hissələri atırdılar, sonra kəsilmiş parçaları səliqə ilə yapışdırırdılar. Əsas şərt maqnit lentini korlamamaq, əzməmək, dartıb uzatmamaq idi. 20 dəqiqəlik verilişin montaj edilməsinə bir neçə saat vaxt sərf edilirdi.

Montaj başa çatdırıldıqdan sonra ayrıca fon musiqisi seçilib götürülürdü və veriliş iki maqnitofondan üçüncüyə yazılırdı (nitq və musiqi). Bundan sonra veriliş efirə vermək üçün hazır hesab olunurdu.

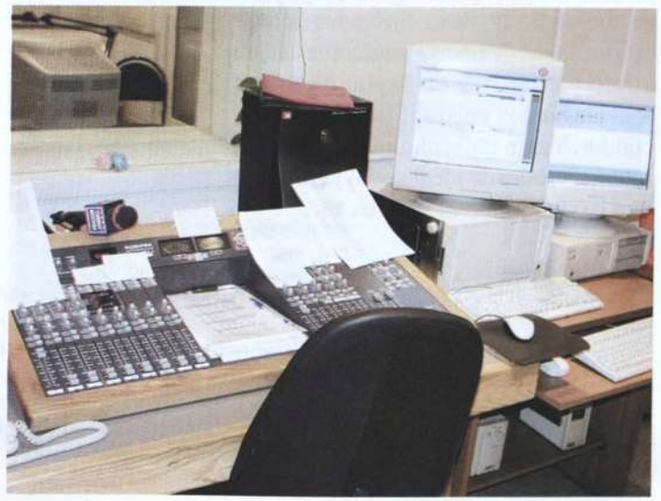
Maqnit materialı bahalı idi, ona görə də lentdən köhnə yazıları pozur, yenidən maqnitləşdirir və təkrar iş prosesinə buraxırdılar. Bəzən lent mütəmadi düzəlişlərə davam gətirmirdi və birbaşa efir vaxtı qırılır, yaxud növbəti yazma zamanı zəif hiss edilən çıqqıltı verirdi.

Bir neçə dəfə istifadə etmək üçün planlaşdırılan hazır material başqa lentə köçürülərək, arxivə yığılırdı. Arxivdə xüsusi saxlama şəraiti yaradılırdı: lentlərin çox qurumaması və korlanmaması üçün lazımi temperatur və rütubətlilik təmin olunurdu. Lentlərə müntəzəm qulluq göstərilirdi, gərginliyi götürmək və yapışmaması üçün onları mütəmadi olaraq əks istiqamətə sarıyırdılar. Lent köhnəldikdə yazılar yoxlanılır və hətta üzü köçürülürdü. Yazıların uzun müddət qalması üçün hər cür imkandan istifadə edirdilər. Arxivdə saxlanılan lentlərə hərbçilərin öz silahlarına yanaşdıqları kimi yanaşırdılar.

Moskva Radio Evində XX əsrin axırlarına aid ənənəvi montaj pultu.
▼

Radiostansiyanın müasir montaj pultu.
▼
▼





Amma analoq yazı üsulunun çox böyük çatışmazlığı mövcuddur. Hətta studiya aparatında təkrar yazarkən maqnitofondan kənar səslər qarışır, analoq siqnalın keçdiyi güçləndiricidə korlanma baş verirdi, həmçinin maqnit lenti səsküy yaradırdı. Hər dəfə yazının üzünü köçürərkən keyfiyyət hiss ediləcək dərəcədə aşağı düşürdü.

XX əsrin sonunda kompüterin köməyilə səsi peşəkar studiya maqnitofonlarında olduğundan heç də pis olmayan şəkildə saxlamağı, yazmağı və səsləndirməyi öyrəndilər. Vinil plastinkaları və maqnitofon kasetləri yerlərini rəqəmsal kompakt və mini disklərlə işləyən studiyalara verməli oldular.

Ən böyük inqilabi dəyişiklik montaj prosesində baş verdi. Analoq maqnitofonu, uzun lentləri, yapışqan və qayçını fərdi kompüter əvəz etməyə başladı. "Cut and paste" texnologiyası saxlanılsa da, rəqəmsal formaya keçirildi. Monitorun ekranında səs sırası səsin amplitudasının zamana görə qrafiki kimi təsvir edilirdi. Artıq səhvləri sezmək olurdu. Vaxt şkalası asanlıqla miqyaslaşdırılmağa başladı, fraqmenti saniyənin mində bir hissəsi dəqiqliyilə qeyd etmək mümkün oldu (lent üçün bu millimetr hissəsi qədər ola bilərdi). Əvvəlcə səsə ayrıca parça kimi qulaq asır və yalnız bundan sonra qərar qəbul edilirdi ki, bu hissəni kəsib atmaq və ya sərhədləri sürüşdürmək lazımdır. Nəinki bir neçə dəfə ölçüb, sonra kəsmək, hətta parça səhvən kəsilib atılmışsa belə, hər şeyi geri qaytarmaq mümkün oldu. Lazımi parça fonoqramın istənilən yerinə asanlıqla keçirilirdi. Montaj proqramları fon musiqisini verilişə qoşmağa imkan verirdi. Ümumiyyətlə, əvvəllər fiziki işlərə sərf edilən saatları artıq indi dəqiqələr əvəz edir.

Mini-disk (MD) Sony firması (Yaponiya) tərəfindən hazırlanıb. Bu rəqəmsal maqnitooptik daşıyıcı adi CD-də olduğu qədər (74 dəq.) informasiya tutumuna malikdir. Lakin CD-dən fərqli olaraq, istənilən yazı parçalarını dəfələrlə MD-yə yazmaq və pozmaq olar. Səsi 5 dəfə sıxmaq lazım gəldikdə MD-dən istifadə olunur.

Verilişin emalı zamanı tətbiq edilən ən sadə effektlər arasında səs tonunu azaldan və artıran effekti, müxtəlif salonların və stadionların əks-səda effektini göstərmək olar.

Audiomontaj proqramlarının imkanlarının, həqiqətən, həddi-hüdudu yoxdur. Montaj edilən veriliş səs rejissorunun kompüterinə daxil olur və oradan birbaşa efirə verilir. Yazıların saxlanması üçün daha əlverişli imkan yaranıb. Mini-disklərdəki və kompüterlərdəki verilənlər vaxt keçdikcə heç də keyfiyyətini itirmir. Belə ki, 30 dəq.-lik veriliş cəmi 50 Mbaytlıq disk tutumu tələb edir, yəni onun daimi saxlanması cüzi xərc tələb edir.

Köhnə texnologiyadan daha hə qaldı? Montaj zamanı insanın aydın olmayan yazını tamamilə orijinal, maraqlı, yaddaqalan verilişə çevirdiyi yaradıcılıq prosesi bədii redaktorun quru elmi mətni uşaq ensiklopediyası dilinə çevirməsi kimidir. Radioda bu zarafatı əbəs yerə səsləndirmirlər: "Siz bizə yalnız əlifba danışın, biz isə özümüz verilişi montaj edərik".

Yapışqan sonradan skotça çox oxşayan yapışqanlı lentlə əvəz edildi.

Radio verilişlərində səs MPEG-1 Layer-3 standartında 256 kbit/s axını ilə sıxılmaqla yazılır.

Studiya maqnitafonları.





HESABLAMALAR

KOMPÜTERDƏ HESABLAMALAR

Havanın hərarəti, təzyiqi, rütubəti, buludluğu, düşən yağıntının miqdarı, küləyin istiqaməti və gücü haqqında informasiyalar hava haqqında kifayət qədər dəqiq proqnoz tərtib etməyə imkan verir. Təəssüf ki, bu proqnoz onun tərtib edildiyi mövsüm artıq ötüb keçdikdən sonra hazır olacaqdır, yəni yay üçün havanın necə olacağını gələn payız xəbər verəcəklər.



Bu problemin həllində mühüm məqam məsələlərin həllində dəqiqliyin və sürətin artırılmasına, eləcə də yeni sinif məsələlərin həllinə imkan verən kompüterlərin məhsuldarlığının yüksəldilməsidir.

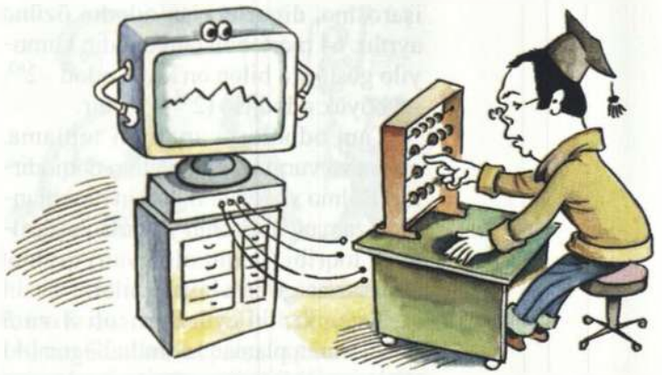
Kompüterdə hesablamalar hesab qurğusunda (HQ) həyata keçirilir. HQ informasiyanı başqa şəklə çevirir; kompüterin digər qurğularında (yaddaş, idarəetmə, giriş, çıxış) isə informasiyanın yalnız zaman və məkan üzrə yerdəyişməsi baş verir. HQ iki cür olur: universal və ixtsaslaşdırılmış. Universal HQ kompüterin əmrlər sistemində nəzərdə tutulmuş bütün hesablama əməliyyatlarını yerinə yetirir. Kompüterin əmrlər sisteminə adətən toplama, çıxma, müqayisə, vurma, bölmə və bəzən kvadrat kökün hesablanması əməliyyatları daxildir.

HMQ (hesab-məntiq qurğusu) – universal hesab qurğusunun ən sadə növüdür. O, tam ədədləri emal edir, toplama, çıxma, sürüşmə və məntiq əməliyyatlarını yerinə yetirir.

İxtisaslaşdırılmış HQ alqoritmik cəhətdən yaxın olan əməliyyatlar qrupunu yerinə yetirir. Beləliklə, toplama qurğusu – toplama, çıxma, müqayisə əməliyyatlarını; vurma qurğusu – vurma əməliyyatını; bölmə qurğusu – bölmə və kvadrat kökün hesablanması əməliyyatlarını yerinə yetirir.

Universal HQ kompüterdə olan birneçə ixtisaslaşdırılmış HQ-dan fərqli olaraq kiçik həcmə malikdir (yəni, müasir prosessorların hazırlandığı yarımkeçirici kristallarda az yer tutur). İxtisaslaşdırılmış HQ maşında bir neçə hesablama əməliyyatını eyni vaxtda yerinə yetirməyə imkan verir ki, bu da kompüterin sürətini və məhsuldarlığını artırmış olur.

Avadanlıqlar kristalda nə qədər çox olarsa, HQ bir o qədər mürəkkəb olur və onların işi sürətləndikcə mikroprosessorun məhsuldarlığı yüksəlir. Bundan başqa kompüterdə HQ-nın sayını artırmaqla hesablama prosesini paralelləşdirmək mümkündür. Əvvəllər bəzi əməliyyatlar (məsələn, bölmə və kvadrat kökün hesablanması) xüsusi proqramların köməyilə yerinə yetirilirdisə, artıq bu işlər aparat yolu ilə görülür ki, bu da proqram realizasiyasından daha sürətlidir.



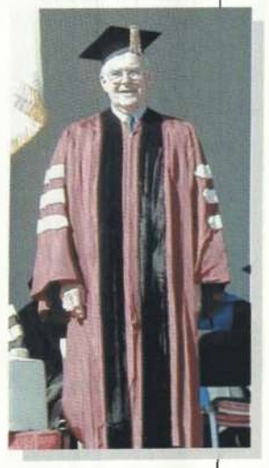
Hələ 1965-ci ildə Qordon Mur (Intel şirkətinin yaradıcılarından biri) görmüşdü ki, hər bir neçə ildən bir buraxılan yarımkeçirici yaddaşın yeni kristalının tutumu özündən əvvəlki yaddaş kristalının tutumundan təxminən 2 dəfə çoxdur. Bu qiymətləndirmə mikroprosessorun kristalındakı tranzistorların sayı və mikroprosessorun məhsuldarlığı üçün də doğrudur.



90-cı illərdə multimediya hesab qurğuları meydana çıxdı. Onlardan istifadə qrafiki informasiyaların emalı mümkünlüyünü xeyli genişləndirdi, kompüter tomoqrafiyası və virtual reallıq meydana gəldi.

SABİT VERGÜLLÜ ƏDƏDLƏRİN TƏSVİRİ

Kompüterdə emal edilən ədədlər *sabit* və *sürüşkən vergüllü* ədədlərə bölünür. Sabit vergüllü ədədlər ikili şəkildə göstərilmiş adi tam ədədlərdir. Tam hissəni kəsr hissədən ayıran vergül qeyd olunmuş, yəni sabit yerdə durur. Vergülün qoyulduğu yer ədədin kiçik mərtəbəsindən sağda olur. Proqramlaşdırma dillərində belə kəmiyyətlər tam ədədlər kimi qeyd olunurlar. Çox zaman tam ədəd 32 və ya 64 ikilik mərtəbəli (bit) yer tutur. Bu zaman bir mərtəbə ədədin



Qordon Mur.

◄ Tomoqraf.

işarəsinə, digərləri isə ədədin özünə ayrılır. 64 mərtəbəli tam ədədin köməyilə göstərilə bilən ən kiçik ədəd -2^{63}, ən böyük ədəd isə $(2^{63}-1)$-dir.

Tam ədədlərlə aparılan toplama, çıxma və vurma əməliyyatları dəqiqdirlər. Bölmə və kvadrat kökün hesablanması zamanı nəticənin yuvarlaqlaşdırılması təqribi (dəqiq olmayan) qiymət alınmasına gətirib çıxara bilər.

Tutaq ki, iki eyni dərəcəli A və B ədədlərini toplamaq lazımdır. Əgər 1+1 toplasaq, ikilik mərtəbədə saxlanılan cəm 0-a bərabər olacaqdır (baxmayaraq ki, gözlənilən nəticə 2 idi). Yekun qiymətin itməməsi üçün yalnız nəticədən deyil, ikilik ədəddə *dolub-daşma əlamətindən – keçiddən* də istifadə edirlər. Keçidin özü də ikilik mərtəbədir. Belə ki, dolub-daşma olmadıqda keçid 0-a, dolub-daşma olduqda isə 1-ə bərabərdir. Əgər cəmi S, keçidi C ilə işarə etsək, onda doğruluq cədvəlini aşağıdakı kimi tərtib etmək olar:

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Göründüyü kimi, toplananlardan yalnız biri 1-ə bərabərdirsə, S cəmi 1-ə bərabərdir. Cədvəli reallaşdıran məntiqi element *birmərtəbəli cəmləyici* adlanır. Bu elementin köməyilə iki birmərtəbəli ədədi toplayaraq cəmi və keçidi almaq olar, lakin çox sayda mərtəbələrə malik ədədləri toplamaq olmaz. Bunun üçün qonşu kiçik mərtəbədən keçidi nəzərə almalı olan *Cin* keçid girişi vacibdir.

Tam birmərtəbəli cəmləyici üçün aşağıda göstərilmiş doğruluq cədvəlini yazmaq mümkündür:

A	B	Cin	C	S
0	0	0	0	0
0	1	0	0	1
1	0	0	0	1
1	1	0	1	0
0	0	1	0	1
0	1	1	1	0
1	0	1	1	0
1	1	1	1	1

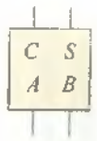
Bu cədvəlin ilk dörd sətri ($Cin = 0$) əvvəlki doğruluq cədvəlilə üst-üstə düşür. İki və ya üç vahid toplanırsa, keçid C var, tək sayda vahid toplanırsa (bir və ya üç), cəm $S = 1$ olacaqdır.

Tutaq ki, A və B – iki 4 mərtəbəli ikilik ədədlərdir, 0 – kiçik mərtəbənin, 3 isə yuxarı mərtəbənin nömrəsidir. Burada kiçik mərtəbədə yaranmış C_0 keçidi birinci mərtəbənin keçid girişinə verilir; C_1 keçidi ikinci mərtəbəyə daxil olur; C_2 yuxarı mərtəbəyə verilir, yəni keçidlərin giriş və çıxışları ardıcıl birləşiblər.

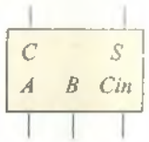
Keçidlər zəncirini təşkil edən bu cür cəmləyicini *ardıcıl* adlandırırlar. Bu cəmləyicinin ən sadə, lakin ən asta növüdür. Göstərilən sxem üzrə qurulmuş 64 mərtəbəli cəmləyicidə cəmin yuxarı mərtəbəsi bir mərtəbəli cəmləyicinin 64 dövründən sonra əldə edilir.

Cəmləyici, kompüterin sürətli işləməsində böyük rol oynayan vacib qovşaqlardan biridir. Odur ki, cəmləyicilərin işləmə sürətinin yüksək olması həmişə aktual məsələ olmuşdur. Halhazırda ardıcıl cəmləyicidən onlarla dəfə sürətli cəmləyicilər yaradılmışdır. Cəmləyicilər təkcə ədədlərin toplanmasında deyil, vurma, bölmə və hətta kvadrat kökün hesablanmasında da istifadə olunurlar.

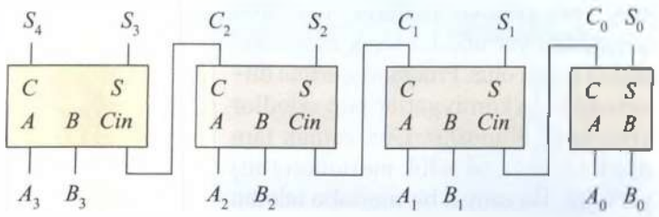
Birmərtəbəli cəmləyici.



Tam birmərtəbəli cəmləyici.



Ardıcıl cəmləyici.



Sabit vergüllü ədədləri çıxmaq üçün çıxılan əvvəlcə əlavə koda (əlavə kod dedikdə ilkin qiymətilə cəmi 0 verən ədəd başa düşülür) çevrilir, daha sonra azalanla cəmlənir.

Vurma əməliyyatını adi sütunlu vurma kimi yerinə yetirmək olar. Tutaq ki, E və T – vuruqlar, P isə hasildir. İlk əvvəl birinci E vuruğu T_0 vuruğunun kiçik mərtəbəsinə vurulur. Əgər $T_0 = 0$ olarsa, onda $E \cdot T_0 = 0$, əks halda $E \cdot T_0 = E$, yəni birinci vuruqla ikinci vuruğun mərtəbələrinin hasili ya sıfıra ya da birinci vuruğa bərabərdir. Bu cür hasili hissə-hissə hasil adlandırırlar. Sonra, birinci vuruq bir mərtəbə sola sürüşdürülür və ikinci vuruğun mərtəbəsinə (T_1) vurulur – növbəti hissə-hissə hasil yaranır. Bu cür proses ikinci vuruğun yuxarı mərtəbəsinə (T_{63}) qədər davam edir. Ümumi hasil əldə etmək üçün hissə-hissə hasilləri toplamaq lazımdır ki, bunu da artıq tanıdığımız cəmləyicilər vasitəsilə etmək olar.

Hasilin bu alqoritmi – ən sadə və kifayət qədər yavaşdır. Vurma prosesini sürətləndirməyə imkan verən çoxlu sayda alqoritmlər mövcuddur.

Bölmə – ən mürəkkəb və ən yavaş hesablama əməliyyatıdır. Bir addımda qismətin bir-neçə rəqəmini təyin etməyə imkan verən bölmə alqoritmləri mövcuddur.

Hal-hazırda bölmə əməliyyatı onu vurma ilə əvəz edən alqoritmlərin köməyilə həyata keçirilir. Bu halda bölünən bölənin tərs qiymətinə vurulur. Çox zaman tərs qiymət cədvəlin köməyilə təyin olunur. Bəzi prosessorlarda ümumiyyətlə, bölmə qurğusu yoxdur və bu əməliyyat xüsusi proqramlar vasitəsilə yerinə yetirilir.

Elmi hesablamalar üçün çox böyük və çox kiçik ədədlərdən istifadə xarakterikdir. Məsələn, kosmik obyektlərin kütlələri və aralarındakı məsafə çox böyükdür. Atomun daxilində elementar



hissəciklərin kütlələri və aralarındakı məsafə çox kiçikdir. Bu halda istifadə olunan ədədlərin geniş diapazonundan danışılır. Sabit vergüllü ədədlər isə dar diapazona malikdirlər.

İlk kompüterlər sabit vergüllü ədədlərlə işləyirdi. O zamanlar dar diapazon problemini həll etmək üçün proqramçılar miqyaslaşdırma tətbiq edirdilər. Əgər ədədlər çox kiçik idisə, proqramçılar onları eyni bir ədədə vururdu, əksinə, çox böyük idisə, onları eyni bir ədədə bölürdülər.

Miqyaslaşdırma məqsədilə proqramçı verilənləri, aralıq və son nəticələri analiz edirdi ki, bütün mərhələlərdə hesablama diapazonu kifayət etsin. Diapazonun sərhədindən kənara çıxma hesablamalarda kobud səhvlərə gətirib çıxarırdı. Məsələn, kompüter 10 mərtəbəli ədədlərlə işləyirsə, onda iki 6 mərtəbəli ədədin hasili 12 mərtəbəli olur, yəni hasilin 2 mərtəbəsi kompüterin mərtəbə cədvəlindən kənarda qalır. Bu zaman nəticə mənasız alınır. Miqyaslaşdırma prosesi proqramçılardan çoxlu güc və vaxt tələb edirdi.



Alan Klements xanımı ilə.

"Vurma və bölməni öyrənməyin heç olmasa bir həqiqi səbəbi vardı – bu əməliyyatların yerinə yetirilməsinin sonsuz sayda üsulları mövcuddur və deməli, yeni vurma qurğusu kəşf edilərsə, sonsuz sayda dissertasiyalar müdafiə oluna bilər".

Alan Klements (hesablama texnikası sahəsində amerika alimi). "Kompüter qurğularının prinsipləri". 1986-cı il.

SÜRÜŞKƏN VERGÜLLÜ ƏDƏDLƏRİN TƏSVİRİ

Sürüşkən vergüllü ədədlər kompüterdə belə təsvir olunurlar:

$$Q = (-1)^{İş} \cdot M \cdot 2^S,$$

burada, Q – ədədin qiyməti, $İş$ – ədədin işarəsi, M – ədədin mantissası, S – ədədin tərtibidir. Proqramlaşdırma dillərində sürüşkən vergüllü ədədlər adətən həqiqi ədədlər kimi qeyd edilirlər. Ədədin işarəsi bir bit tutur. Əgər, $İş = 0$ olarsa, o zaman $(-1)^{İş} = (-1)^0 = +1$ və Q müsbət işarəlidir. Əgər, $İş = 1$ olarsa, o zaman $(-1)^{İş} = -1$ və Q mənfi işarəyə malik olur.

M qiymətinə məhdudiyyət qoyulur:

$$1 \le M \le 2.$$

Bu bərabərsizliyi ödəyən mantissa *normallaşdırılmış* adlanır. S – tam ədəddir (mənfi və ya müsbət).

Ədədlərin bu cür təsviri mühəndis və alimlər tərəfindən istifadə olunan yazılara bənzəyir. Məsələn, mühəndis $a = 350\ 000\ 000\ 000$ əvəzinə $a = 3{,}5 \cdot 10^{11}$ yazacaq.

Sürüşkən vergüllü ədədlərdən istifadə kompüterdə ədədlərin dəqiqliyini və diapazonunu əhəmiyyətli dərəcədə artırmağa imkan verdi və proqramçını miqyaslaşdırma kimi yorucu prosesdən azad etdi.

Beləliklə, 64 formatlı sürüşkən vergüllü ədədlər 11 mərtəbəli tərtibə malikdirlər ki, bu da -2^{2047}-dən $+2^{2047}$-yə qədər olan ədədləri təqdim etməyə imkan verir. Sabit vergüllü iki qonşu ədədlər arasında məsafə 1-dir (axı, iki qonşu ədəd vahid qədər fərqlənirlər), lakin, 0-ın yaxınlığında yerləşən sürüşkən vergüllü iki qonşu ədəd arasında məsafə 2^{-2100}-ə bərabərdir.

Sürüşkən vergüllü iki ədədi toplayaq:

$$Q1 = (-1)^{İş1} \cdot M_1 \cdot 2^{S1},$$
$$Q2 = (-1)^{İş2} \cdot M_2 \cdot 2^{S2}.$$

Qüvvət şəklində verilmiş ədədləri toplamaq üçün onların qüvvət üstlərini (dərəcələrini) bərabərləşdirmək lazımdır. Tutaq ki, $S1 > S2$ və $S1-S2=SF$ (tərtiblər fərqi), o zaman aşağıdakı kimi yazmaq olar:

$$Q1 + Q2 = (-1)^{İş1} \cdot M_1 \cdot 2^{S1} + (-1)^{İş2} \cdot M_2 \cdot 2^{S2} =$$
$$= (-1)^{İş1} \cdot M_1 \cdot 2^{S1} + (-1)^{İş2} \cdot (M_2 \cdot 2^{-SF}) \cdot (2^{S2} \cdot 2^{SF}).$$

$Q2$ ədədini 2^{SF} ədədinə vurduq və böldük. $Q2$ dəyişmədi, yalnız onun tərkibi – mantissası və tərtibi dəyişdi.

YUVARLAQLAŞDIRMANIN SƏHVLƏRİ

Qeyd etmək lazımdır ki, sürüşkən vergüllü ədədlərlə aparılan hesab əməliyyatları sabit vergüllü ədədlərlə aparılan hesablamalara nisbətən xeyli mürəkkəbdir. Məsələn, sürüşkən vergüllü ədədlər üçün nəticə əməliyyatın yerinə yetirilmə ardıcıllığından asılıdır, yəni:

$$A + (B + C) \neq (A + B) + C.$$

Bu onunla əlaqədardır ki, sürüşkən vergüllü ədədlərlə aparılan hesablamalar dəqiq deyil, təqribidirlər. Sürüşkən vergüllü ədədlərlə aparılan əməliyyatlar zamanı mərtəbə cədvəlində yer çatmadığından kiçik mərtəbələr atılır və yuvarlaqlaşdırma aparılır. Məsələnin həlli zamanı yuvarlaqlaşdırma səhvləri toplanaraq tamamilə yanlış nəticənin alınmasına gətirib çıxara bilər.

Yüksək dəqiqlikdən istifadə hesablama alqoritmlərinin yoxlanılması üçün ən sadə metoddur. Belə ki, 32 mərtəbəli dəqiqlik əvəzinə 80 mərtəbəli tətbiq edilərsə, bu hər zaman lazımi nəticəni vermir. Məsələn, iki dəyişənli funksiyanı hesablayaq:

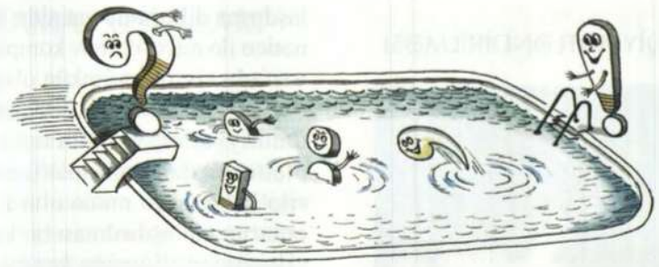
$$f(x, y) = 333{,}75y^6 + x^2(11x^2y^2 - y^6 - 121y^4 - 2) + 5{,}5y^8 + x/(2y),$$

Burada, $x = 77\ 617$ və $y = 33\ 096$. Aşağıdakı nəticələri 32, 64 və 80 mərtəbəli dəqiqliklə alırıq:

32 m	f = 1,172 603...
64 m	f = 1,172 603 940 053 1...
80 m	f = 1,172 603 940 053 178...

Hər üç nəticədə yuxarı mərtəbəli ilk yeddi ədəd üst-üstə düşür. Görünə bilər ki, mərtəbənin artması həqiqətən hesablamanın dəqiqliyini artırır. Əslində isə düzgün cavab tamam başqadır!

İnterval hesabı, hesabın sürüşkən vergüllü ədədlər üzərində əməliyyatlar zamanı meydana çıxan bu cür problemləri öyrənən bölməsidir. İnterval metodu cavabda ədəd əvəzinə, nəticənin yerləşdiyi intervalı verir. İntervalın çox böyük olması onu göstərir ki, seçilmiş alqoritm lazımı dəqiqliyi vermir.



Mantissanı 2^{SF}-ə vurmaq üçün vergülü SF mərtəbə sola keçirmək və ya mantissanı sağa sürüşdürmək lazımdır. Bu halda mantissanın SF qədər kiçik mərtəbələri mərtəbə cədvəlinin sərhədindən kənara çıxacaq və nəticədə hesablamalarda iştirak etməyəcəkdir. SF tərtiblər fərqinin hesablanması üçün böyük olmayan cəmləyicidən istifadə etmək olar. Mantissanın sağa sürüşdürülməsi prosessorun qovşağı tərəfindən yerinə yetirilir və yerdəyişdirici adlanır.

Digər tərəfdən $2^{S2} \cdot 2^{SF} = 2^{S2+SF} = 2^{S1}$, yəni ikinci ədədin tərtibi birinci ədədin tərtibinə bərabərdir və ədədləri toplamaq olar. Çevirməni davam etdirək:

$$Q1 + Q2 = [(-1)^{İş1} \cdot M_1 + (-1)^{İş2} \cdot (M_2 \cdot 2^{-SF})] \cdot 2^{S1}.$$

M_1 ilə (işarəni nəzərə almaqla) mərtəbələri SF qədər sağa sürüşdürülmüş M_2 mantissasını toplayaq. Mantissalar cəmləyici vasitəsilə toplanırlar. Nəticənin tərtibi iki tərtibdən ən böyüyünə, yəni $S1$-ə bərabər olacaq.

Sürüşkən vergüllü iki ədədin hasilinə baxaq:

$$Q1 = (-1)^{İş1} \cdot M_1 \cdot 2^{S1},$$
$$Q2 = (-1)^{İş2} \cdot M_2 \cdot 2^{S2},$$
$$Q1 \cdot Q2 = (-1)^{İş1} \cdot M_1 \cdot 2^{S1} \cdot (-1)^{İş1} \times M_2 \cdot 2^{S2} = (-1)^{İş1} \cdot (-1)^{İş2} \cdot M_1 \cdot M_2 \times 2^{S1} \cdot 2^{S2} = (-1)^{İş1 + İş2} \cdot M_1 \cdot M_2 \cdot 2^{S1} \cdot 2^{S2}.$$

Göründüyü kimi, sürüşkən vergüllü iki ədədin hasilini hesablamaq üçün mantissaları bir-birinə vurmaq lazımdır, hasilin tərtibi isə vuruqların tərtiblərinin cəminə bərabərdir. Hasilin işarəsi vuruqların işarələrilə təyin olunur: əgər işarələr eynidirsə, hasil müsbətdir, işarələr fərqlidirsə, hasil mənfidir.

Tutaq ki, $Q1$ qiymətini $Q2$-yə bölmək lazımdır:

$$Q1 / Q2 = [(-1)^{İş1} \cdot M_1 \cdot 2^{S1}] / [(-1)^{İş2} \times M_2 \cdot 2^{S2}] = (-1)^{İş1 - İş2} \cdot (M_1 / M_2) \cdot 2^{S1-S2}.$$

Sürüşkən vergüllü ədədləri bölmək üçün mantissaları bölmək, bölünənin tərtibindən isə bölənin tərtibini çıxmaq lazımdır. Əgər bölünənlə bölənin işarələri üst-üstə düşürsə, qismətin işarəsi müsbət, bölünənlə bölənin işarələri müxtəlifdirsə, işarə mənfidir.

İngilis dilli ölkələrdə sürüşkən vergül əvəzinə sürüşkən nöqtə deyirlər.

ANSI/IEEE 754 STANDARTI

Proqram təminatının qiyməti aparatın və ya mütəxəssislərin dediyi kimi "dəmirin" qiymətindən xeyli böyükdür. Bundan başqa, proqram təminatının yazılması xeyli vaxt aparır. Odur ki, yeni aparatı artıq mövcud olan proqrama uyğun layihələndirməyə çalışırlar. Proqram təminatı yüksək səviyyəli proqram-



IEEE ekspert qrupu.

HESABLAMALARDA XƏTANIN QİYMƏTLƏNDİRİLMƏSİ

1991-ci ildə Fars körfəzində, ABŞ-ın İraqa qarşı müharibəsi zamanı Amerikanın "Petriot" zenit raketləri batareyası iraqlıların buraxdığı, sovet istehsalı olan "Skad" raketini zərərləşdirə bilmədi. Raket amerikalı əsgərlərin olduğu kazarmaya düşərək 28 nəfərin həyatına son qoydu.



Amerikanın "Petriot" zenit raketi.

Faciənin səbəbi vaxtın hesablanmasında yol verilmiş xəta idi. Hesablamada kompüterin takt generatorunun göstərdiyi vaxtı (o saniyənin onda biri qədər ölçülür) 1/10 ədədinə vurmaq lazım idi, lakin bu onluq ədədi ikilik ədəd kimi dəqiq göstərmək mümkün deyildi. Verilən sabit ədədin saxlanmasında 24 mərtəbəli registrdən istifadə edilirdi. 1/10-in dəqiq qiymətilə onun ikilik koddakı qeyri dəqiq qiyməti arasındakı fərq ikilik kodda 0, 0 000 000 000 000 000 000 000 011 001 100... və ya onluq kodda 0, 000 000 095-ə bərabərdir. Kompüter 100 saata qədər işləmişdir, bu zaman ərzində vaxtın hesablanmasında 0,34 san qədər xəta yaranmışdır. "Skad" raketinin sürəti təqribən 1700 m/san idi, yəni həmin vaxtda o 500 m-dən çox yol qət etmişdi. Bu vaxt onun zenit raketləri tərəfindən tutula bilməməsi üçün kifayət idi.

Digər xətanın nəticəsi, faciəli olmasa da, çox baha başa gəldi. 1996-cı ildə Avropa kosmik agentliyinin "Arian-5" raketinin ilk buraxılışı oldu. Uçuş vaxtından 40 san sonra o verilmiş trayektoriyadan kənara çıxaraq partladı. Raket 10 il müddətinə yaradılmış və onun hazırlanmasına 7,5 mlrd dollar sərf edilmişdi.

Buna səbəb ətalət hesablama sisteminin proqram təminatında dəqiqliyin pozulması idi. Uçuş meydanına nəzərən raketin üfiqi sürəti 64 mərtəbəli sürüşkən vergüllü ədədlə ifadə olunurdu. Proqramda o 16 mərtəbəli sabit vergüllü ədədə çevrilirdi. Məlum oldu ki, bu ədəd 32768-dən (16 mərtəbədə göstərilməsi mümkün olan maksimal ədəddən) böyük olur. Çevirmə səhv cavabın alınmasına səbəb oldu və nəticədə raket partladı.



Fars körfəzində müharibə. Qaçqınlar. 1991.

laşdırma dillərində yaradılır ki, onu eyni nəticə ilə müxtəlif növ kompüterlər üçün istifadə etmək mümkün olsun.

Sürüşkən vergüllü ədədlərlə aparılan əməliyyatlar proqram təminatının bir platformadan digər platformaya keçirilə bilməsinə mane olurdu. Belə ki, onların reallaşdırılması bir kompüterlər ailəsindən digərinə keçdikdə kəskin dəyişirdi. Reallaşdırılmalar mantissa və tərtibin mərtəbələrinin sayına, onların kodlaşdırılma metodlarına, hesablama sistemlərinin əsaslarına, yuvarlaqlaşdırma metodlarına, dayanmalara və s. görə fərqlənirdilər.

Sürüşkən vergüllü hesablamalar sahəsində bu cür vəziyyət ona gətirib çıxardı ki, 1977-ci ildə ABŞ-ın elektrotexnika və radioelektronika üzrə mühəndislər institutunun (IEEE Computer Society) Kompüter cəmiyyətində sürüşkən vergüllü ədədlər üzərində ikilik hesablamalar aparmaq üçün standartın işlənməsi üzrə qrup yaradıldı. Bu qrupun üzvləri – mikroprosessor istehsalçıları və universitet alimləri idilər. Qrupun iclasları çox qızğın keçirdi. Bir vaxtlar deyirdilər ki, ABŞ-da baxılması maraqlı olan üç diqqətəlayiq yer var: Las-Veqas. Böyük kanyon və sürüşkən vergüllü ədədlər üzərində ikilik hesablamalar aparmaq üçün standartın işlənməsi üzrə yaradılmış qrupun iclasları.

Baxmayaraq ki, standart 1985-ci ildə qəbul olundu, artıq 1984-cü ildə onun layihəsi qabaqcıl firmalar (Intel, AMD, IBM və s.) tərəfindən reallaşdırılmışdı.

Mövcud olduğu tarix boyu bu faktiki olaraq beynəlxalq standart oldu və indi bütün aparat və proqram təminatı tədqiqatçıları onun tələblərinə əməl edirlər. Buna baxmayaraq 2001-ci ildə standarta yenidən baxmaq zərurəti meydana gəldi və bu istiqamətdə işlərə başlanıldı. Buna səbəb kompüter hesabında son 20 ildə əldə edilmiş nailiyyətlərin bu standartda öz əksini tapmaması idi.

HESABLAMA ÜSULLARI

Bir çox sahələrdə hesablamalar aparmaq tələb olunur və burada əsas rollardan birini hesablama riyaziyyatı oynayır. Təbiət elmləri adətən, bu və ya digər elmləri öyrənərək onun təsvirini verir, qanunları formalaşdırır. Təsvir ya keyfiyyət, ya da kəmiyyət xüsusiyyəti daşıyır. Hər hansı hadisəni və ya prosesi ədəd şəklində proqnozlaşdırmaq tələb olunursa (məsələn, havanın hərarətini, atmosfer təzyiqini və s. qabaqcadan demək), o halda kəmiyyət təsviri və ya riyazi model lazımdır.

DƏQİQ VƏ YA TƏQRİBİ...

Leopold Kroneker bir dəfə demişdir: "Tam ədədləri Allah yaradıb, qalanları isə insanların işidir". Tam ədədlərlə işləmək asan və xoşdur. Bu ədədlərlə hesablamaları nəzəri cəhətdən mütləq dəqiqliklə yerinə yetirmək mümkündür. Başqa məsələ – onluq kəsrlər kimi bizə daha yaxşı tanış olan *həqiqi* ədədlərdir. Sonuncular həm sonlu, həm də sonsuz – dövri və qeyri-dövri olur. Məsələn, orta məktəbin riyaziyyat kursundan məlumdur ki, π ədədi sonlu və ya sonsuz dövri onluq kəsr kimi ifadə olunmur, yəni *irrasionaldır*.

Fizika məsələlərinin həlli zamanı məhz həqiqi ədədlərə rast gəlinir. Cismin kütləsi, yükün miqdarı, sürət, təcil, ağırlıq qüvvəsi – bütün bunlar *fiziki kəmiyyətlərin* nümunələridir. Onlardan bəziləri, məsələn, kütlə və ya yük, *skalyar* olub, bir həqiqi ədədlə xarakterizə olunurlar. Digərləri isə, *vektorlar*, üç həqiqi ədədlərlə – vektorun koordinatları ilə verilirlər.

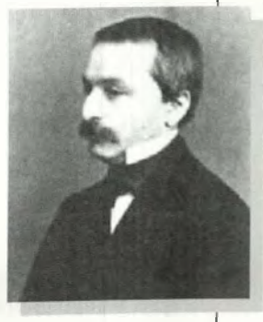
Hətta nəzəri olaraq, həqiqi ədədləri nə dəqiq yazmaq, nə də onlar üzərində dəqiq hesablamalar aparmaq mümkün olmur. Lakin bu hələ hamısı deyil. Praktikada fiziki məsələlərin həlli hər hansı ilkin verilənlərin və ya ən azı fundamental fiziki sabitlərin qiymətlərinin verilməsini nəzərdə tutur. Həm bu, həm də digər qiymətləri ölçmələrin köməyilə, yəni *təqribi* əldə edirlər.

> Xoşbəxtlikdən, bizdən heç vaxt mütləq dəqiq həll tələb olunmur. Buraxılan xəta, yəni axtarılan kəmiyyətin hesablanan qiymətinin onun həqiqi qiymətindən kənara çıxması, praktiki ehtiyaclarla təyin olunur. Məsələn, bizə otağın ölçüləri santimetr dəqiqliklə məlum ola bilər, lakin adi həyatda onun kub santimetr dəqiqliklə həcmi çətin ki, bizə lazım olsun.

ƏDƏDLƏRİN MAŞINDA TƏSVİRİ

Fiziki kəmiyyətlər həm çox böyük, həm də çox kiçik qiymətlər alırlar. Bununla belə həm böyük, həm də kiçik qiymətlərə bir misalda rast gəlinə bilər. Məs., vakuumda işığın sürəti 300 000 000 m/s qiymətinə yaxın, qravitasiya sabiti isə təqribən 0,000 000 000 066 7 Nm^2/kq^2 olur. Bu cür ədədlər üzərində işin rahatlığı üçün fiziklər çoxdan eksponensial yazı metodundan istifadə edirlər. 300 000 000 əvəzinə $3 \cdot 10^8$, 0,000 000 000 066 7 əvəzinə isə $6{,}67 \cdot 10^{-11}$ yazırlar. 10 rəqəminin qarşısında duran tam və ya kəsr ədədlər mantissa, 10-dan sonra yazılmış tam ədəd isə tərtib və ya eksponent adlanır. Ümumiyyətlə, yazılış o deməkdir ki, ədəd mantissa ilə əsasın müvafiq tərtibi hasilinə bərabərdir.

Həmin prinsip müasir kompüterlərdə həqiqi ədədlərin təsvirində istifadə edilir. Lakin burada onluq say sistemi əvəzinə ikilik say sistemindən istifadə olunur. Mantissa və tərtibin saxlanması üçün prosessorda müəyyən sayda mərtəbələr ayrılmışdır. Mantissanın tam hissəsini kəsr hissəsindən ayıran vergülün



Leopold Kroneker.



mövqeyi birdəfəlik müəyyən edilir (adətən, mantissanın ilk ikilik rəqəminin qarşısında). Onluq say sistemində olduğu kimi, bu yazılış o deməkdir ki, mantissanı tərtibə yüksəldilmiş əsasa vurmaq lazımdır. Amma indi əsas 2-yə bərabərdir.

Həqiqi ədədlərin eksponensial formada yazılışını bu cür də anlamaq olar: tərtib mərtəbələrin sayını təyin edir. Axtarılan ədədi əldə etmək üçün, mantissanın yazılışındakı vergülü mərtəbələrin sayı qədər sürüşdürmək lazımdır (tərtibin işarəsindən asılı olaraq sağa və ya sola). Ədədlərin təsvirinin bu cür formasını *sürüşkən vergüllü format*, həqiqi ədədlərin özlərini isə *sürüşkən vergüllü ədədlər* adlandırırlar.

Mantissada mərtəbələrin sayı ədədin dəqiqliyini, eksponentin mərtəbələrinin sayı isə ədədlərin diapazonunu təyin edir. Bir çox kompüterlər sürüşkən vergüllü ədədlərin iki formatını dəstəkləyir: qısa və uzun. Məsələn, qısa format 32 mərtəbə yer tutur. Onlardan 8 mərtəbə tərtibə, 23-ü mantissaya, 1-i isə işarəyə ayrılır. Hesab etmək olar ki, mantissanın yuxarı mərtəbəsi həmişə 1-ə bərabərdir. Bu halda onu yadda saxlamaq lazım deyil, ədədin dəqiqliyi 24 ikilik mərtəbə və ya təqribən 7 onluq mərtəbə təşkil edir. 8 mərtəbəli tərtib, ədədi 2^{-128}-2^{128} diapazonunda və ya təqribən 10^{-39}-dan 10^{39}-dək təsvir etməyə imkan verir.

Etiraf etmək lazımdır ki, əvvəllər 8-lik və hətta 16-lıq say sistemilə işləyən kompüterlər var idi.

Tutaq ki, uzun formatlı (64 mərtəbə) ədədlərdə mantissa üçün 52 mərtəbə, tərtib üçün isə 11 mərtəbə ayrılmışdır. Onda bu format 10^{-308}-dən 10^{308}-dək diapazonda olan ədədləri təsvir etməyə imkan verərdi.

Aydındır ki, maşında təsvir edilə bilən həqiqi ədədlər ədəd oxunu özünəməxsus tor ilə örtürlər, belə ki, sıfıra yaxınlaşdıqca onun oyuğu kiçilir. Bu o deməkdir ki, hər bir həqiqi maşın ədədi üçün onun iki qonşusu a_0 və a_1 ($a_0 < a < a_1$), var, a və onun qonşuları arasında isə heç bir ədəd yoxdur.

Ümumiyyətlə, həqiqi ədədlərin maşında təsvirini iki sabit kəmiyyət ilə xarakterizə etmək rahatdır. Birinci ε *maşın* – ən böyük ədəddir və vahidə əlavə edilərkən onu dəyişdirmir. Belə ədəd $1/2^l$ bərabərdir, l – mantissadakı mərtəbələrin sayıdır, qısa format üçün $1/2^{24} \approx 10^{-7}$-yə, uzun formatlar üçün isə $1/2^{53} \approx 10^{-16}$-yə bərabərdir. İkinci sabit θ maşın – maşında təsvir edilə bilən və sıfıra bərabər olmayan ən kiçik ədəddir. Onun qiyməti qısa və uzun formatlar üçün müvafiq olaraq $1/2^{128} \approx 10^{-39}$-a və $1/2^{1024} \approx 10^{-308}$-ə bərabərdir. Bu sabitləri bilməklə, hesablamalar zamanı xətaları nəzəri cəhətdən qiymətləndirmək olar.

YUVARLAQLAŞDIRMA XƏTALARININ ROLU

Düsturlarla hesablama zamanı dəqiqlik kəskin surətdə azala bilər. Odur ki, düzgün hesablama sxemini əldə etmək üçün çox zaman dolayı yol axtarmaq lazım gəlir. Bir çox sadə hallarda fiziki məsələnin həlli düstur ilə hesablanmaya gətirilir. Tutaq ki, məsələnin həlli zamanı e^x funksiyasının $x = 0$ nöqtəsi ətrafında Teylor sırasına ayrılışından istifadə etmək lazımdır:

$$e^x = \sum_{k=0}^{\infty} x^k / k!$$

Bu sıra x-in böyük olmayan qiymətlərində olduqca sürətlə yığılır. Praktikada növbəti toplananın əlavə olunması cəmi dəyişdirmirsə, hesablama prosesini dayandırırlar.

x-in mütləq qiymətcə böyük mənfi qiymətlərində problem meydana çıxır: toplananların qiyməti əvvəlcə böyük sürətlə artır, sonra isə azalmağa başlayır. Sıranın bəzi hədləri gözlənilən nəticədən kifayət qədər böyük ola bilər.

Hər bir toplananın hesablanmasında meydana gələn xəta onun tərtibinin ε maşın qiymətinə hasilinə bərabər olduğu üçün, nəticədə yekun xəta çox böyük olur. Bu problemi həll etmək üçün e^x funksiyasının xassələrindən istifadə etmək daha məqsədəuyğun olardı. Məsələn,

$$e^{-x} = 1 / e^x$$

$$e^{N+x} = e^N \cdot e^x$$

yaxud

$$e^{N \cdot x} = (e^x)^N,$$

eyniliyindən istifadə etməklə x-in böyük müsbət qiymətləri üçün sıranın yığılmasını yaxşılaşdırmaq (həm də hesablamanı sürətləndirmək) olar, N – tam ədəddir.

TƏNLİYİN KÖKÜNÜN TAPILMASI

Bir çox məsələlər

$$f(x) = 0$$

tənliyinin həllinə gətirilir. Burada $f(x)$ – qiyməti hesablanan funksiyadır. Məsələn,

$$x^2 - a = 0$$

tənliyinin həlli $\sqrt{a}$ -ya bərabərdir. Burada $f(x) = x^2 - a$, a – həqiqi ədəddir.

Tutaq elə bir parça məlumdur ki, onun uclarında $f(x)$ funksiyası müxtəlif işarəli qiymətlər alır. Əgər bizi hər hansı ixtiyari bir kökü tapmaq maraqlandırırsa, onda parçanı yarıya bölmə üsulundan istifadə edə bilərik. Bizim halda, əgər $a > 1$ olarsa, onda bu parça [0, a] olacaqdır. Tənliyin kökü bu parçada yerləşir. $f(x)$ funksiyasının qiymətini parçanın orta nöqtəsində hesablayırıq. Yeni parça kimi ilk parçanın o yarısı götürülür ki, onun uclarında funksiya müxtəlif işarəli qiymətlər alır. Beləliklə, yeni parçanın uzunluğu iki dəfə qısalır. Proses o vaxta kimi davam etdirilir ki, parçanın uzunluğu verilmiş xətadan kiçik olsun.

İNTERVAL HESABI

Hesablama zamanı xətaların qiymətləndirilməsinə yanaşmalardan biri interval hesabından istifadəyə əsaslanmışdır. Burada əsas fikir ondan ibarətdir ki, həqiqi ədədin təqribi qiyməti ədədlər cütü ilə əvəz olunur. Bu ədədlər cütünün əmələ gətirdiyi interval daxilində ədədin dəqiq qiyməti yerləşir. Ədədlər üzərində hesab əməliyyatları intervallar üzərində əməliyyatlarla əvəz edilir. Bu, konkret verilənlər üzərində xətanın necə böyüdüyünü görməyə və hesablamanın dəqiqliyinin vacibliyinə əmin olmağa imkan verir.

Kompüterdə interval hesabının komandaları olmadığından, metodun realizə olunmasından ötrü xüsusi alt proqramlardan istifadə etmək lazım gəlir. Bu isə hesablama vaxtını xeyli artırır. Odur ki, göstərilən yanaşma seriyalı hesablamaların aparılmasında tətbiq edilmir. Lakin o, ayrı-ayrı hesablamalarda xətaların rolunun analizində, eləcə də, teoremin isbatında xətanı dəqiq qiymətləndirmək lazımdırsa, o zaman riyazi sabitlərin qiymətlərinin hesablanmasında əhəmiyyətli olur.






```
alq Kökün hesablanması
baş həq A, B, e, C
| e: = 0.01
| çapet yş, "Daxil et A="; daxilet A
| çapet yş, "Daxil et B="; daxilet B
| db nə qədər ki B-A>2*e
| | C : = (A + B) / 2
| | əgər f (A) * f (C) < = 0
| | | onda B: = C
| | | əks halda A: = C
| | tamam
| ds
| C: = (A + B) / 2
| çapet yş, "Kök=", C,
|                  "dəqiqlik=", e
son
alq həq f (həq X)
baş
| qiym: = X**2-2
son
```

Tutaq ki, prosesin başlanğıcında parçanın uzunluğu L-ə bərabərdir. Onda ε dəqiqliklə cavabı almaq üçün prosesi $\log_2(L / \varepsilon)$ dəfə təkrarlamaq lazım gələcəkdir.

Parçanı yarıya bölmə metodu bir tənliyin kökünü tapmağa imkan verir, ancaq onu tənliklər sisteminin həllinə tətbiq etmək o qədər də asan məsələ deyil. *İterativ metodlar* bu çatışmazlıqdan xalidir. Əvvəlcə məsələ

$$G(x) = x$$

tənliyinə gətirilir.

Sonra hər-hansı üsulla kökə ilk yaxınlaşmanı x_0 təyin edirlər. Nəhayət, $x_{k+1} = G(x_k)$ sxemi üzrə yaxınlaşmanı dəqiqləşdirirlər. Əgər $G(x)$ funksiyası və başlanğıc yaxınlaşma x_0 düzgün seçilmişsə, onda bu proses yığılır və sonda lazım olan dəqiqliklə həlli alırıq. Yığılma şərti belə ifadə olunur: axtarılan həll və başlanğıc yaxınlaşma elə bir parçada yerləşməlidirlər ki, bu parça üzərində ixtiyari iki x_1, x_2 nöqtələri üçün

$$|G(x_2) - G(x_1)| < q\,|x_2 - x_1|$$

bərabərsizliyi doğru olsun. Burada q – vahiddən kiçik müsbət ədəd, $|x|$ isə x vektorunun moduludur.

Əgər ilkin yaxınlaşmadan kökə qədər olan parçada G funksiyasının törəməsinin modulu vahiddən kiçikdirsə, onda bu şərt doğrudur. Burada ümumi halda, x artıq bir ədəd deyil, ədədlər yığımı, məsələn, vektordur. Müvafiq olaraq $G(x)$ vektoru vektora çevirən funksiyadır.

Xətti tənliklər sisteminin həllində geniş istifadə olunmuş Zeydel metodu da iterativ metoddur. Dəqiq metodlardan fərqli olaraq (məsələn, Qauss metodu), Zeydel metodu təqribidir. Məsələnin həlli lazımi dəqiqlik əldə edilənə qədər addımların təkrarlanması yolu ilə həyata keçirilir. Tutaq ki

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i = 1, 2, \ldots, n.$$

sistemi verilib. İlkin yaxınlaşma x_i^0 ($i = 1, 2, \ldots, n$) verilir. Sonra həllin dəqiqləşdirilməsi üçün k sayda addımlar həyata keçirilir. Hər addımda yeni yaxınlaşmanın komponentləri x_i^k hesablanır. Bu hesablamalar üçün düsturlar belə alınır. Sistemin i-ci tənliyini:

$$\sum_{j=1}^{i-1} a_{ij} x_j^k + a_{ii} x_i^k + \sum_{j=i+1}^{n} a_{ij} x_j^{k-1} = b_i.$$

şəklində yazırıq. Onda x_i-nin yeni qiymətinin (yəni, k-cı addımda alınan) hesablanması üçün belə düstur alarıq:

$$x_i^k = (b_i - \sum_{j=1}^{i-1} a_{ij} x_j^k + \sum_{j=i+1}^{n} a_{ij} x^{k-1}) / a_{ii}.$$

Zeydel metodu heç də həmişə yığılmır. Sual oluna bilər ki, məsələnin həlli üçün dəqiq metodlar olduğu halda, nə

üçün iterativ metodlardan istifadə olunur. Məsələlərin həllində tez-tez xüsusi matrislərə, məsələn, seyrək matrislərə yaxud sıfır olmayan elementlərini asanlıqla hesablamaq mümkün olan matrislərə rast gəlinir.

Qauss metodu ilə matrisin çevrilməsi zamanı sıfra bərabər olmayan elementlərin sayı sürətlə artır və matrisi bütünlüklə yadda saxlamaq lazım gəlir.

Zeydel metodunda yalnız cəmlər hesablanır, odur ki, seyrək matris halında yaddaşa xeyli qənaət etmək olar. Bu zaman matris bütünlüklə deyil, yalnız onun sıfra bərabər olmayan elementləri və onların indeksləri yadda saxlanılır.

DİFERENSİAL TƏNLİKLƏRİN HƏLLİ

Elementar funksiyaların yaxınlaşması və tənlik həlli hər addımda rast gəlinən tipik riyazi məsələlərdir. Fiziki məsələlərə bilavasitə baxış bir qayda olaraq bu və ya digər dərəcədə mürəkkəb diferensial tənliyə və ya bu növ tənliklər sisteminə gətirib çıxarır.

İki yaydan asılmış cismi gözdən keçirək. X oxunu yayların dartılma xətti boyunca yerləşdirək. Əgər cismi X oxuna nəzərən dartsaq, cismə onu ilkin vəziyyətə qaytarmağa yönələn qüvvə təsir edəcək. Huk qanununa görə bu qüvvə aşağıdakı ifadəyə bərabər olacaq:

$$F = -kx.$$

Nyutonun ikinci qanunundan istifadə etməklə, $ma = -kx$ alarıq.

Yada salsaq ki, təcil sürətin törəməsi ($a = \upsilon'$), sürət isə koordinatın törəməsidir ($\upsilon = x'$), onda diferensial tənliklər sistemini alarıq:

$$\upsilon' = -kx / m,$$
$$x' = \upsilon.$$

İndi təklif olunan sistemin həlli üçün sadə metod qurmaq olar. Əgər zaman intervalı Δt kiçik olarsa, onda $x'(t)$ və $\upsilon'(t)$ törəmələrini aşağıdakı düsturlarla təqribi hesablamaq olar:

$$x'(t) \approx (x(t+\Delta t) - x(t))/\Delta t,$$
$$\upsilon'(t) \approx (\upsilon(t+\Delta t) - \upsilon(t))/\Delta t.$$

$t = 0$-dan $t = T$-yə qədər olan zaman intervalını Δt uzunluqlu kiçik parçalara bölək: $t_0 = 0$, $t_1 = \Delta t$, ..., $t_i = t_{i-1} + \Delta t$, ..., $t_N = T$. İndi başlanğıc t_0 zamanında cismin x_0 koordinatına υ_0 sürətini verməklə, t_1, ..., t_N zaman anlarında onların qiymətini ardıcıl şəkildə bu düsturlardan istifadə edərək hesablamaq olar:

$$x_{i+1} = x_i + \upsilon_i \Delta t,$$
$$\upsilon_{i+1} = \upsilon_i - (kx/m)\Delta t.$$

Beləliklə, parçada funksiyanın, yəni sonsuz sayda ədədlər təşkil edən obyektin axtarılması məsələsi bu parçanın ayrı-ayrı nöqtələrində x və υ funksiyalarının qiymətlər cədvəlinin tapılmasına gətirib çıxardı. "Sonsuz" məsələnin "sonlu" məsələyə bu cür gətirilməsi prosesi *diskretləşmə* adlanır. Alınmış həllin dəqiqliyi və onun N-dən asılılıq problemilə yaxınlaşmalar nəzəriyyəsi məşğul olur.

* * *

İstənilən ədədi üsul üçün aktual olan daha bir problem – effektivlik problemidir. Kompüter meydana gəlməmişdən əvvəl proqramın mürəkkəbliyini qiymətləndirmək üçün vurmaların sayını hesablayırdılar. Əslində isə hesablamalarda vurmanın payına düşən hissə heç də ən böyük deyil. Birincisi, müasir kompüterlərdə həqiqi ədədlərin vurulması və toplanması eyni və ya təxminən eyni vaxt tələb edir. Lakin, operativ yaddaşa müraciət vaxtı istənilən hesab əməliyyatına sərf olunan vaxtdan xeyli böyükdür. Bundan başqa, məsələnin kompüterlərdə həlli zamanı bu və ya digər üsulun seçilməsində alqoritmin aparatların xüsusiyyətlərinə uyğunlaşdırılması mühüm rol oynayır.



Karl Fridrix Qauss.

Elementlərinin çoxu sıfıra bərabər olan matrislər seyrək matrislərdir.

NYUTON METODU

Ümumi hal	Nyutonun misalı
Tutaq ki, $y = f(x)$ funksiyası verilib	Nyuton $y = x^3 - 2x - 5$ funksiyasını seçir
$f(x) = 0$ tənliyinin təqribi həlli tələb olunur	Nyuton $x^3 - 2x - 5 = 0$ tənliyini təqribi həll etməyə hazırlaşır
Fərz edək, bizə kökdən az fərqlənən x_0 qiyməti məlumdur	Nyutona məlumdur ki, $x = 2$ qiyməti kökdən az fərqlənir
$x = x_0 + p$ əvəzləməsi edək	Nyuton $x = 2 + p$ əvəzləməsi edir
$f(x) = 0$ tənliyində x-in yerinə $x_0 + p$ ifadəsini yazırıq və yeni $g(p) = 0$ tənliyini alırıq	Nyuton yeni tənlik alır: $p^3 + 6p^2 + 10p - 1 = 0$
Tənlikdə sərbəst həddi və p-yə görə birinci tərtibli həddi saxlayıb, tərtibi vahiddən böyük olan həddləri ataq: $Bp + A = 0$	Nyuton tənlikdən $p^3 + 6p^2$ hədlərini atır və $10p - 1 = 0$ tənliyini saxlayır
$Bp + A = 0$ tənliyindən kökün təqribi qiymətini p^0 tapırıq	Nyuton $10p - 1 = 0$ tənliyindən $p = 0{,}1$ tapır
İlk tənliyin təqribi kökü kimi $x_1 = x_0 + p_0$ götürürük	Nyuton bu addımı hesablamanın sonuna qədər kənara qoyur

NYUTON METODUNUN DƏQİQLİYİ

Nyuton metodunun bir addımı heç bir nəticə vermirsə, o zaman daha bir addım etmək olar:

$$x_2 = x_1 - f(x_1)/f'(x_1)$$

və s. Nyuton metodunun üstünlüyü ondadır ki, kökün başlanğıc təqribi qiyməti kökə kifayət qədər yaxındırsa, hər addımda cavabın doğru onluq işarələrinin sayı minimum iki dəfə artmış olacaq. Nyuton metodu ilə dəqiq kökü əvvəlcədən məlum olan $x^2 - 1 = 0$ tənliyini həll etmək istəsək, o zaman kökün təqribi qiyməti kimi $x_0 = 1{,}1$ götürülür. Bu təqribilik bir doğru işarəyə malikdir. Güman etmək olar ki, Nyutonun bir addımından sonra ən azı iki doğru işarə alınacaq, iki addımdan sonra isə – doğru işarələr dörddən az olmayacaq. $(x^2 - 1)' = 2x$ olduğu üçün, onda

$$x_1 = x_0 - f(x_0)/f'(x_0) = 1{,}1 - 0{,}21/2{,}2 \cdot 1{,}1 - 0{,}095 = 1{,}005,$$
$$x_2 = x_1 - f(x_1)/f'(x_1) = 1{,}005 - (1{,}0052 - 1)/2{,}010 \times 1{,}005 - 0{,}004985 = 1{,}00001.$$

Nəticə gözlənildiyindən yaxşı oldu: bir addımdan sonra üç, iki addımdan sonra isə beş doğru işarə.

A və B əmsallarını p-yə nəzərən olan tənlikdə $x = x_0 + p$ əvəzləməsi aparmadan və p üçün tənlik yazmadan hesablamaq olar. Bir halda ki, $f(x_0 + p) = g(p) = \ldots + Bp + A$, onda $p = 0$ olarsa, $A = f(x_0)$ alınar. Analoji olaraq, $f(x_0 + p) = \ldots + Bp + A$ bərabərliyinin hər iki hissəsini p dəyişəninə görə differensiallasaq və $p = 0$ qiymətini qoysaq, $B = f'(x_0)$ alarıq. Beləliklə, $f(x_0 + p) = 0$ tənliyi əvəzinə biz $f'(x_0)p + f(x_0) = 0$ tənliyini həll etmiş oluruq. Bu tənliyin kökü $p_0 = -f(x_0)/f'(x_0)$ düsturu ilə verilir. Deməli, $f(x) = 0$ tənliyinin kökünün yeni təqribi qiyməti kimi $x_1 = x_0 - f(x_0)/f'(x_0)$ götürmək olar.

Kökün təqribi x_0 qiymətindən daha dəqiq x_1 qiymətinə keçid *Nyuton metodunun addımı* adlanır.

$f'(x_0)p + f(x_0) = 0$ tənliyində $f'(x_0)$ əmsalı sıfra bərabər və ya ona yaxındırsa, Nyuton metodu işləmir. Eyni hal tənliyin iki kökü təqribi x_0 qiymətinə yaxın olduqda da baş verir.

Bu halda, Nyuton p-yə nəzərən tənlikdə yalnız sərbəst və xətti hədlərin deyil, həm də kvadrat hədlərin qalmasını təklif edir:

$$f(x_0 + p) = g(p) = \ldots + Cp^2 + Bp + A$$

(isbat etmək olar ki, $C = f''(x_0)/2$).

Sıranın 3 və daha yuxarı tərtibli hədlərini atsaq kvadrat tənlik alarıq. Bu tənlikdən x_0-a iki düzəliş və kökün iki yeni təqribi qiyməti tapılır. Bunların hər birinə adi Nyuton metodunu tətbiq etmək olar.

Nyuton differensial hesabının yaradıcılarından biri idi. Diferensial hesabının əsas ideyası ondan ibarətdir ki, düsturla və ya sonsuz sıra ilə verilən istənilən funksiyanın qrafikinə mikroskop altında baxsaq o, müəyyən düz xətdən fərqlənməyəcək. Bu düz xətt funksiyanın qrafikinə toxunan adlanır, onun bucaq əmsalı isə funksiyanın törəməsinə bərabərdir. Nyuton metodunun həndəsi mənası: $f(x) = 0$ tənliyini təqribi x_0 kökü ətrafında həll etmək üçün funksiyanın qrafikini bu qrafikə x_0 nöqtəsində toxunan düz xətlə əvəz etmək və toxunanın X oxu ilə kəsişməsini tapmaq lazımdır. Ona görə də, Nyuton metodunu bəzən *toxunanlar metodu* adlandırırlar. Nyuton metodu yalnız bir tənliyə deyil, tənliklər sisteminə də müvəffəqiyyətlə tətbiq oluna bilər.

Metodun mahiyyəti əvvəlki kimi qalır. Məsələn, iki məchullu iki tənlikli sistem üçün

$$f(x, y) = 0,$$
$$g(x, y) = 0$$

x_0 və y_0-ın təqribi qiymətlərini biliriksə, daha dəqiq həlli

$$x = x_0 + p,$$
$$y = y_0 + q$$

əvəzləməsi etdikdən və

$$f(x_0 + p, y_0 + q) = \ldots + Ap + Bq + C = 0,$$
$$g(x_0 + p, y_0 + q) = \ldots + Dp + Eq + F = 0.$$

yazdıqdan sonra tapmaq olar.

Son sistemdə, yazılmış hədlərdən başqa, bütün hədləri atsaq, o zaman alınmış iki məchullu iki xətti tənliklər sistemini həll etməklə, p və q təshihlərini tapmaq mümkün olacaq.

Bu metod həndəsi olaraq f və g funksiyalarının qrafiklərini onların toxunma müstəvilərilə əvəzlənməsi deməkdir.

> Yüksək tərtibli tənliklər tamamilə bu cür həll olunur ... metodun qaneedici olması əməlin düzgün seçilməsindən xəbər verir. Buna əsasən, lazım olduqda əməli yada salmaq asandır.
>
> *İsaak Nyuton.*

NYUTON NƏ YAZIR

Nyuton "Sonsuz hədli tənliklərin köməyilə analiz" (1664-1671-ci il) əsərində tənliklərin ədədi (təqribi) həll üsullarını belə izah edir.

"Tutaq ki, $x^3 - 2x - 5 = 0$ tənliyini həll etmək tələb olunur və axtarılan kökdən özünün onda birindən də az ədəd qədər fərqlənən 2 ədədi məlumdur. Onda fərz edirəm ki, $2 + p = x$ və bu ifadəni tənlikdə yerinə qoymaqla yeni tənlik alıram:

$$p^3 + 6p^2 + 10p - 1 = 0,$$

Bu tənliyin kökünü tapmaq lazım gəlir (ilk tənliyin x kökü (2+p)-yə bərabər olacaqdır, burada p – yeni tənliyin köküdür). Buradan ($p^3 + 6p^2$ kiçik olduğu üçün onu nəzərə almırıq) təqribən alırıq:

$$10p - 1 = 0; \text{ yaxud } p = 0{,}1.$$

Nəticədə mən 0,1 yazıram və fərz edirəm ki, $0{,}1 + q = p$; bu ifadəni də əvvəlki kimi tənlikdə yerinə qoysam

$$q^3 + 6{,}3q^2 + 11{,}23q + 0{,}061 = 0.$$

tənliyi alınar.

$$11{,}23q + 0{,}061 = 0$$

tənliyi demək olar ki, həqiqətə daha uyğundur. Beləliklə, q təqribən – 0,0054-ə bərabər olacaqdır.

– 0,0054 + $r = q$ fərz edərək, əvvəlki kimi tənlikdə yerinə yazıram və bu əməliyyatı lazım olan qədər davam etdirirəm...

Təqribən $r = -0{,}000\ 048\ 53$ alıram və axtarılan nəticəni əldə edirəm:

+ 2,000 000 00
+ 0,100 000 00
– 0,005 400 00
– 0,000 048 53

+ 2,09455147 = x".



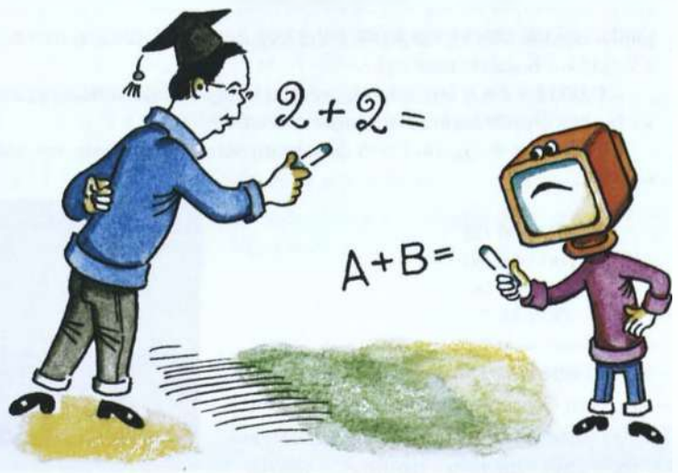
İsaak Nyuton.

KOMPÜTER CƏBRİ

Kompüter cəbri – riyaziyyatın cəbr və hesablama metodlarının qovuşuğunda olan bölməsidir. Buna görə də onun üçün dəqiq sərhədi təyin etmək çox çətindir. Kompüter cəbrinə ya həddən artıq cəbri problemləri, ya da həddən artıq hesablama problemlərini aid edirlər. Konkret məsələnin kompüter cəbrinə aid olub-olmaması çox zaman mütəxəssisin arzusundan asılı olur.

"Kompüter cəbri" termini "işarəli hesablama", "analitik hesablama", "analitik çevirmə" və s. terminlərin sinonimi kimi meydana gəlmişdir. Fransız dilində bu termin "formal hesablama" kimi ifadə edilir.

Hesablama metodlarından danışarkən nəzərə alınmalıdır ki, bütün hesablamalar həqiqi və ya kompleks ədədlərlə yerinə yetirilir. Həqiqətdə isə EHM–də hər növ proqram ancaq sonlu sayda rasional ədədlərlə işləyir, daha doğrusu, kompüterdə yalnız bu cür ədədlər təsvir olunurlar. Tam ədədlərin yazılışı üçün adətən 16 və ya 32 bit, həqiqi ədədlər üçün isə 32 və ya 64 bit ayrılır. Bu çoxluq hesab əməliyyatlarına nisbətən qapalı deyil və bu da müxtəlif dolub-daşmalarda, məsələn, kifayət qədər böyük ədədlərin vurulmasında və ya kiçik ədədlərə bölünməsində ifadə oluna bilər. Hesablama riyaziyyatının daha mühüm xüsusiyyəti ondan ibarətdir ki, tam və həqiqi ədədlər üzərində kompüter vasitəsilə yerinə yetirilən əməliyyatlar rasional ədədlər üzərində yerinə yetirilən əməliyyatlardan fərqlənir. Bundan başqa, kompüter əməliyyatları üçün hesab əməliyyatlarının əsas qanunları (qruplaşdırma, paylama) işləmir.



İşarə hesablamalarında obyektlər yığımı müxtəlifdir. Xüsusilə, onlarda ədəd anlayışı hesablama riyaziyyatına nisbətən kifayət qədər genişdir. Ona tam ədədlər, rasional ədədlər, cəbri ədədlər, çıxıq halqaları, sonlu meydanlar və s. daxildir. Tam ədədlərin mümkün olan qiymətlərinə praktiki olaraq məhdudiyyət qoyulmur: kompüter cəbrinin istənilən sistemi belə ədədlərlə işləyə bilər, məsələn 1000! (faktorial). Kompüter cəbrində həqiqi ədədlər daha az istifadə olunur, onlar ixtiyari dəqiqliklə, məsələn, minlərlə qiymətli onluq rəqəmlərlə, verilirlər. İstənilən halda sonlu olan (çünki, EHM-in əməli yaddaşı sonludur) rasional ədədlər çoxluğu daha tez-tez istifadə olunur, lakin istifadəçi heç vaxt bu məhdudiyyəti hiss etmir. Əgər vaxt məqbuldursa o, əməliyyatları praktiki olaraq istənilən rasional ədədlərlə yerinə yetirə bilər. Rasional ədədlər üzərində kompüter əməliyyatları uyğun riyazi əməliyyatlarla üst-üstə düşür. Beləliklə, hesablama metodlarının əsas problemlərdən biri – hesablama xətasının qiymətləndirilməsi – aradan qaldırılır.

Hal-hazırda çıxıq halqalarında hesablamalar geniş tətbiq olunur. Natural n ədədinin modulu üzrə çıxıq halqasını toplama, çıxma və vurma əməliyyatları ilə 0-dan $n-1$ qədər tam ədədlər çoxluğu kimi təsvir etmək olar. Belə əməliyyatın nəticəsi tam ədədlər üzərində apa-

rılan (əgər çıxmanın nəticəsi mənfidirsə, onda n əlavə edilir) uyğun əməliyyatın nəticəsinin n-ə bölünməsindən alınan qalıqdan ibarətdir. Real kriptoqrafik sistemlərdə n modulu üzrə çıxıqlar istifadə olunur, burada n – onluq yazısı bir neçə yüz rəqəm təşkil edən natural ədəddir. n sadə olduqda çıxıq halqaları xüsusi əhəmiyyətə malikdir. Bu halqalarda istənilən sıfır olmayan natural ədədə bölmə birqiymətli təyin olunur (onlar sonlu meydanlar adlanır).

Qeyri-xətti cəbri (əsasən kvadrat) tənliklərin həllində cəbri ədədlərə rast gəlinir. Cəbri ədədlər sadə radikal ifadələr, məsələn,

$$2+2\sqrt{3}-4\sqrt[5]{1}$$

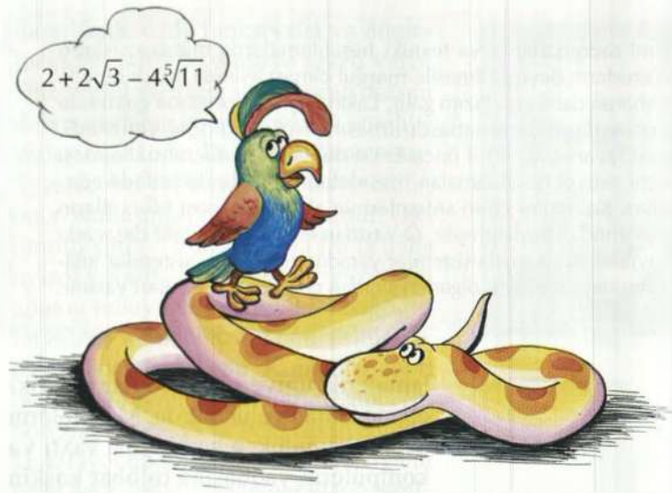
yaxud bir-birinə daxil edilmiş radikallar,

$$\sqrt[3]{2+3\sqrt[4]{2}}$$

ola bilər.

5 və ondan yuxarı dərəcəli tənliklər ümumi halda "radikallarla həll olunmur". Buna görə də kökü belə tənliyin kökü olan minimal çoxhədlini vermək lazım gəlir. Bəzən isə bu çoxhədlinin yeganə kökünün yerləşdiyi intervalı (düz xətt üzərində) yaxud oblastı (kompleks müstəvidə) göstərmək tələb olunur.

Riyaziyyat, fizika və digər elmlərin bir çox sahələrində tətbiq olunan analitik mexanikanın nəzəri metodları polinominal hesaba əsaslanmışdır. Bundan başqa, kompüter cəbrində matris halqaları (matrisin elementləri kifayət qədər ümumi şəkildə olan halqalara aiddir), üstlü, loqarifmik, triqonometrik funksiyaları və s. təyin etməyə imkan verən diferensial meydanlara (funksional meydanlara) baxılır. İnformasiyanın miqdarı hədsiz dərəcədə artdığından hesablamaların aralıq nəticələrini yazmaq üçün kifayət qədər böyük EHM yaddaşı tələb olunur.



KOMPÜTER CƏBRİNİN ALQORİTMLƏRİ

Kompüter cəbri hesablama riyaziyyatının alqoritmlərindən əhəmiyyətli dərəcədə fərqlənən alqoritmlərlə işləyir. Hesablama riyaziyyatının alqoritmlərinə aşağıdakı tələblər irəli sürülür: onlar məsələni verilmiş hesablama resurslarının köməyilə tələb olunan dəqiqliklə həll etməlidirlər. Kompüter cəbrində hesablama adətən yuvarlaqlaşdırılmadan aparılır. Yığılma analizinə az, lakin riyazi və ilk növbədə cəbri yığılmaya, mürəkkəb strukturların obyektlərinə isə kifayət qədər ətraflı diqqət yetirilir. Odur ki, kompüter cəbrində hesablama metodlarına nisbətən vaxta və simvol hesablamalarına nəzərən isə yaddaşdan istifadəyə məhdudiyyət daha çox əmək tələb edir.

Cəbri tədqiqatlarda kompüterdən istifadə mütəxəssislər qarşısında nəinki yeni məsələlər qoydu, həm də əvvəlki, xüsusilə, "sonlu sayda addımda" həll olunan məsələlərə yenidən baxmağa məcbur etdi. Nəticədə həll metodları bir-birlərindən müxtəlifliklərinə görə kəskin fərqlənirdilər və konkret hesab-

Elmi tədqiqatlarda və texniki hesablamalarda mütəxəssislərin düsturların dəyişdirilməsilə məşğul olması əslində ədədi hesaba nisbətən daha çox lazım gəlir. Lakin EHM-in meydana gəlməsilə axırıncıların avtomatlaşdırılmasına xüsusi diqqət yetirilirdi, baxmayaraq ki, 40 il öncə EHM-dən simvol diferensiallaşması kimi simvol hesablamaları məsələlərinin həllində istifadə edirdilər. Kompüter cəbri sistemlərinin aktiv işlənməsi 60-cı illərin axırlarından başlanmışdır. O vaxtdan etibarən müxtəlif dərəcədə yayılmış xeyli sayda sistemlər yaradılmışdır. Bəzi sistemlər inkişafını davam etdirir, digərləri sıradan çıxır, daima yeniləri yaranır.

lamalarda universal metodlar praktiki olaraq istifadə olunmurdu. Məsələlərin həcmi böyüdükcə hesablama vaxtı və kompüterin yaddaşına tələbat kəskin artırdı.

Nümunə kimi hər hansı mövqeli say sistemində iki ikirəqəmli ədədin hasilinə baxaq. Məsələn, onluq say sistemində:

$$(10a + b)(10c + d) = \\ = 100ac + 10(ad + bc) + bd$$

Dörd birrəqəmli ədədin vurulması, üç toplama, iki 10-un dərəcəsinə vurma əməliyyatlarını alırıq. Fərz edək ki, 10-un dərəcəsinə vurma çox tez, toplama yavaş, iki ədədin vurulması isə əhəmiyyətli dərəcədə yavaş (son dərəcə təbii mülahizədir) yerinə yetirilir.

Toplamaların sayını artırmaqla hasillərin sayını azaltmaq mümkündürmü?



Məlum oldu ki, mümkündür. Yalnız üç hasildən istifadə olunan belə alqoritm ilk dəfə 1956-cı ildə A.A.Karasuba tərəfindən təklif olunub. $a - b$ və $d - c$ fərqlərini bir-birinə vursaq, $ad + bc - bd - ac$ alarıq; digər iki vurma ac və bd hasillərini hesablamaq üçün istifadə olunur. Yalnız toplama və sürüşmə (2-nin dərəcəsinə vurma) əməliyyatlarından istifadə etməklə, bu üç qiyməti kombinasiyalaşdırsaq, axtarılan düsturu almaq asandır. Beləliklə,

$$ad + bc = (a - b)(d - c) + ac + bd.$$

Məsələ müxtəlif ümumiləşdirmələrə yol verir, məsələn: yalnız üç həqiqi hasildən istifadə etməklə iki kompleks ədədin vurulması; yalnız yeddi skalyar hasildən istifadə etməklə iki tərtibli iki matrisin vurulması.

Kompüter cəbrinin istənilən sisteminin əsasını çoxhədlilərlə hesablamalar təşkil edir. Son onillikdə aktiv şəkildə tədqiq olunan iki məsələyə baxaq. Onlardan birincisi – faktorizasiya məsələsi, yəni çoxhədlinin sadə vuruqlara ayrılmasıdır. Tutaq ki, tam əmsallı bir dəyişənli $f(x)$ çoxhədlisi verilmişdir, onun vuruqlara ayrılması tələb olunur. "Təmiz" riyaziyyat nöqteyi-nəzərindən bu məsələ tamamilə və qəti surətdə çoxdan həll olunmuşdur və tələb olunan ayrılışı "sonlu sayda addımda" tapmağa imkan verən alqoritmlər var. Bu alqoritmlərdən biri 1882-ci ildə Leopold Kroneker tərəfindən alınmışdır. Hal-hazırda həmin alqoritm onun adını daşıyır (baxmayaraq ki, Avstriya astronomu Şubert Kronekerdən 100 il əvvəl bu alqoritmi bilirdi).

Kroneker alqoritmi aşağıdakı faktlara əsaslanır:

• əgər n dərəcəli çoxhədlini trivial olmayan şəkildə vuruqlara ayırmaq olarsa, onda vuruqlardan heç olmasa birinin dərəcəsi $n/2$-dən çox olmamalıdır;

• m dərəcəli çoxhədlinin əmsalları birmənalı olaraq, onun $m + 1$ nöqtəsindəki qiymətlərilə təyin olunurlar;

• əgər $g(x)$ $f(x)$-ə bölünərsə, onda istənilən j tam ədədi üçün $g(j)$ də $f(j)$-yə bölünər;

• istənilən sıfır olmayan tam ədəd üçün yalnız sonlu sayda bölən var.

Təəssüflər olsun ki, bu metod çox böyük sayda variantların olmasını tələb edir və real məsələlərin həllində tətbiq oluna bilmir.

Yalnız XX yüzilliyin 60-cı illərində faktorizasiya alqoritmlərinin tədqiqində növbəti addım atıldı. Əmsalları sonlu meydandan olan coxhədlilərin vuruqlara ayrılması üçün effektiv alqoritm tapdılar. Bəzi klassik və yeni nəticələrlə uyğunlaşdırılmış bu metod yüzüncü dərəcəli (daha da yüksək) çoxhədliləri vuruqlara ayırmağa imkan verdi.

Digər məsələ sistemin bəzi standart formaya gətirilməsi yolu ilə cəbri tənliklər sisteminin (yəni çoxhədlilər) sadələşməsilə bağlıdır. Əgər sistem çoxlu sayda məchullardan asılıdırsa, lakin xətlidirsə, onda onu diaqonal şəklə gətirmək olar. Çoxlu sayda məchullardan asılı olan qeyri-xətti sistemlərin standart forması daha mürəkkəb şəklə (Qrebner bazisi adlanan) malikdir. O, elmin bir çox sahələrində tətbiq edilir, buna görə də, onun effektiv qurulma metodlarına xeyli iş həsr olunub.

Adətən, kompüter cəbri funksiyanın ədədi qiymətlərilə deyil, onların kifayət qədər ümumi şəklə gətirilmiş düsturlarla təsvirilə işləyir. Çox zaman elementar funksiyalar sinfi hesab əməliyyatları ilə yanaşı kökün tapılması, triqonometrik, loqarifmik, üstlü funksiyalar və digərlərindən istifadə etməyə icazə verir. Diferensial və inteqral hesabının əsasını diferensiallama və inteqrallama əməliyyatları təşkil edir. Diferensiallama əməliyyatı cəbri yolla asanlıqla təsvir edilir, onun reallaşması çətinlik yaratmır. Əks əməliyyatla – inteqrallama ilə iş daha mürəkkəbdir. Hesablama riyaziyyatı adətən müəyyən inteqrallarla bağlıdır; bu zaman hesablamanın nəticəsini müəyyən ədəd təşkil edir. Kompüter cəbri qeyri-müəyyən inteqralları hesablayır. Son zamanlara qədər belə hesab edilirdi ki, kifayət qədər geniş sinif funksiyalar üçün qeyri-müəyyən inteqralların tapılması üçün alqoritmlər mövcud deyil. Aktiv olaraq belə alqoritmlər yalnız XX əsrin sonlarında işlənmişdir.



Leopold Kroneker.

Hesablama riyaziyyatına və kompüter cəbrinə yanaşmada analoji müxtəliflik diferensial tənliklərin həllində müşahidə edilir. Hesablama riyaziyyatı həlli, qiyməti bəzi nöqtələrdə hesablanan funksiya kimi anlayır. Kompüter cəbri isə həlli düstur (çox zaman həmişə mümkün olmur) şəklində tapmağa çalışır. Bu sahəyə mühüm töhfə kimi iki tərtibli xətti diferensial tənliklərin 70-ci illərdə alınmış həll alqoritmlərini göstərmək olar.

Kompüter cəbrindən eyni zamanda diferensial tənliklərin həllərinin qüvvət sıraları şəklində tapılmasında, həllərin dayanıqlığının analizində, periodik həllərin axtarışında və diferensial tənliklərin digər nəzəri məsələlərində istifadə olunur. Kompüter cəbri diferensial tənliklərin ədədi həlli üçün sonlu fərqlər sxemlərinin qurulmasında geniş yayılmışdır.



İNTELLEKTUAL FƏALİYYƏTİN AVTOMATLAŞDIRILMASI

KOMPÜTER MODELLƏŞDİRİLMƏSİ

CAD

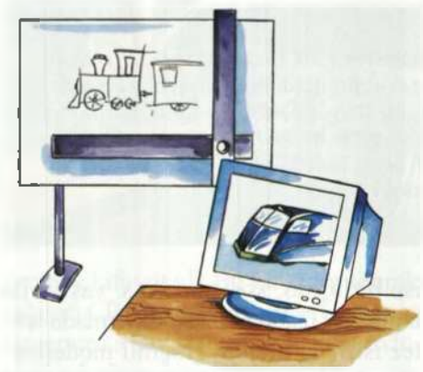
EHM-in köməyilə düzbucaqlıları, ellipsləri, paralel xətləri və müxtəlif mürəkkəb fiqurları asanlıqla çəkmək olar və bunun üçün çertyoj çəkənin sərrast əlinə ehtiyac yoxdur. Kompüterin ekranında çertyojun özünü də hazırlamaq asan olur. Belə ki, çertyoj çəkənin ixtiyarına tez-tez rast gəlinən elementlər, müxtəlif ölçülü şriftlər və stillər verilir. Hətta hazır çertyojların fraqmentlərini də götürmək olar (onları skanerdən də daxil etmək mümkündür). Çəkilmiş fiquru ixtiyari bucaq altında çevirmək və ya onu simmetrik əks etdirmək, böyütmək yaxud ölçüsünü kiçiltmək olar. EHM imkan verir ki, eyni tipli komandalar vahid makrokomanda kimi yazılsın, yeni komandalar xüsusi dildə böyük olmayan proqramlar kimi yazılıb sistemin funksiyalarını genişləndirsin. Əvvəllər modeli azca dəyişdikdə çertyoju bütünlüklə yenidən işləmək lazım gəlirdi, indi isə bu cür dəyişikliklər kompüterə çox tez daxil edilir və dəyişilmiş çertyoj tez çap edilə bilir. Bu tipli məsələlərin həlli üçün CAD (Computer Aided Design) və ya ALS – avtomatlaşdırılmış layihələndirmə sistemləri işlənmişdir.

Kompüterləri ilk dəfə avtomatlaşdırılmış layihələndirmədə və istehsalda Patrik Henreti tətbiq etmişdir. 1957-ci ildə o, proqramla idarə olunan PRONTO kommersiya sistemini işləmişdir.

Sonra 1960-cı ildə Ayren Sazerlend Massaçusets Texnologiya İnstitutunun Linkoln laboratoriyasında TX-2 kompüterində CAD sistemləri sənayesində ilk hesab edilən SKETCHPAD paketini yaratdı.

Elə həmin ildə McAuto (McDonnell Douglas Automation Company) şirkəti təsis olundu. Bu şirkət avtomatlaşdırılmış layihələndirmə sahəsinin tədqiq edilməsində önəmli rol oynadı. Müstəvi üzərində, sonra isə 3 ölçülü fəzada düz xətt və parçalardan ibarət olan şablonları çəkmək üçün ilk CAD proqramları ən sadə alqoritmlərdən istifadə edirdi. Sonrakı illər ərzində CAD sistemləri durmadan inkişaf etdi. Artıq 70-ci illərin sonunda 512 Kbayt operativ və 300 Mbayt disk yaddaşına malik 16 bitlik minikompüterin bazasında tipik CAD sistemi yaradıldı. Bu sistemin qiyməti təqribən 125 min dollar idi.

Mühəndislər həndəsi elementlərin (avtomatik ştrixləmə, ölçülərin göstərilməsi, çertyojun dəqiqliyi) avtomatlaşdırılmasının üstünlüyünü dərhal qiymətləndirdilər. Layihələndirilən obyektin adi çertyoju müstəvi üzərində həndəsi modelləşdirmənin ənənəvi üsulundan başqa bir şey deyildir. İnsan onu xətkeş, pərgar və transportirin köməyilə kağız və ya kalka üzərində yerinə yetirir. Bu kağız üzərinə addımbaaddım olmaqla bütün verilmiş və ya yeni meydana çıxan informasiyanı köçürürlər. Bu zaman müəyyən dəqiqliyə riayət edib, qurmaları maksimum mümkün olan miqyasda icra etmək lazımdır. Lakin kulman arxasında icra edilmiş çertyojda qurma dəqiqliyi nadir hallarda millimetrin onda bir hissəsindən az olur. CAD sistemlər isə nəinki qurmaları daha dəqiq icra etməyə imkan verir və lazımi hesablamaları avtomatik icra edir, həm də layihələndirilən obyektin parametrlərini hesablaya bilir (məsələn, ağırlıq mərkəzi, ətalət momenti, çəki, müxtəlif gərginliklər), tam sənədlər dəstini, modelin hesablamalarını və çertyojları almağa imkan verir. Əgər məmulat sifarişlə hazırlanmayıbsa və onu müstəqil şəkildə satmaq istəyirlərsə, onda onun EHM-in ekranında qurulmuş 3 ölçülü modelindən istifadə edərək, mümkün ola biləcək tələbi öyrənmək olar (məsələn, avtomobil istehsal etmək istəyirlərsə). Adətən, ekranda olan prototip istənilən materialdan hazırlanmış modeldən daha təbii görünür.



CAD sistemlərində adətən, müstəvi modelləşdirmənin iki tipi geniş yayılmışdır. Birinci halda əsas elementlər parçalar, qövslər, xətlər, əyrilər olur, modelləşdirmənin baza əməliyyatları kimi isə uzatma, kəsmə və elementlərin birləşməsi istifadə olunur. Bu tipə ən geniş yayılmış AutoCAD sistemi aiddir. Digər sistemlərdə isə qapalı konturlar əsas elementlər, birləşmə, tamamlama və kəsişmə əsas əməliyyatlardır.

3-ölçülü modelləşdirmədə də oxşar bölgü mövcuddur. Burada əsas obyektlər kimi müxtəlif səthlərlə məhdudlaşmış cisimlər götürülür, məmulatın təsviri

Kompüter modelləşdirməsi bir çox sahələrdə tətbiq olunur. Bu zaman müxtəlif layihələndirmə üsullarından istifadə olunur: həndəsi, texnoloji, erqonomik və s. Və bu üsullar sistemlərin daha dərin ixtisaslaşmasına gətirib çıxarmışdır, məsələn, CAD/CAM/CAE (Computer Aided Engineering). Tətbiq sahələrindən asılı olaraq layihələndirmə alətləri də fərqli ola bilər. Məsələn, 1991-ci ildə ArhiCAD sistemi yaradılmışdır ki, bu da arxitektor və dizaynerlərin alətidir.

Amerikanın "Boinq" korporasiyası 1978-ci ildə yeni (o zaman üçün) 757 və 767 təyyarələrinin modellərinin eyni zamanda layihələnməsinə başladı. Bu təyyarələrin çox fərqli olmasına baxmayaraq onların çoxlu oxşar hissələri vardı. Hazırlanma zamanı CAD-sistemlərdən geniş istifadə olundu. Bu da vaxta və vəsaitə qənaət etməyə imkan verdi.

Bu ALS paketi ilk dəfə 1982-ci ilin noyabrında Las-Veqasda COMDEX sərgisində nümayiş etdirildi. 2002-ci ildə artıq AutoCAD paketinin on yeddinci versiyası buraxıldı. Döyüş təyyarələrinin hazırlanması üzrə dünya lideri Suxoy konstruktor bürosu AutoCAD-dan istifadə edir.

isə birləşmə, kəsişmə və s. vasitəsilə alınır. Bu üsuldan maşınqayırmada tez-tez istifadə olunur. Həcmli modelləşdirmə, adətən, çertyojlardan daha rahat olur: məsələn, yığma çertyojda bolt bir neçə görüntü ilə, həcmli modelləşdirmədə isə bir görüntü ilə, yəni boltun modelilə təsvir oluna bilər.

Lakin həndəsi qurma yolu ilə hər şeyə nail olmaq mümkün olmur. Əgər obyektin, məsələn, təyyarənin tam modelini mütləq dəqiqliklə almaq lazımdırsa, onda nə etməli? Bu halda obyektlərin çoxpilləli dekompozisiyasından istifadə olunur. Məmulat – aqreqatlar sistemi, hər bir aqreqat – qovşaqlar sistemi və nəhayət hər bir qovşaq – detallar sistemi kimi təsvir olunur. Eyni prinsip həcmli modelləşdirmədə də istifadə edilir. Bu zaman elektrik motorunun layihələndirilməsində rotorlu diyircəkli yastığın dəqiq modelinə malik olmağa ehtiyac yoxdur. Onun diyircəklərinin oturma yerlərini göstərmək şərtilə, qabarit modelinin olması kifayət edir.

Tez-tez riyazi modelləşdirmə metodlarının köməyilə məmulatların ümumiləşdirilmiş modellərini, məsələn, daxili yanma mühərrikində porşenin modelini



qurmaq lazım gəlir (bu növ motorların hamısı eyni prinsiplərə əsaslanır və eynitipli detallardan ibarətdir: silindrlər, porşenlər, sürgü qolları və s.). İndi mühərrikin gücü və dövrlərinin sayı kimi ilkin verilənlərə görə detalların hesablanması metodikaları mövcuddur. Lakin konstruktor mühərriki yaradarkən davamlılıq, texnolojilik (ən sadə və qənaətli üsul ilə hazırlanma), təmirə yarama və s. kimi çoxlu kriteriyaları nəzərə almalıdır. Bu cür xarakteristikaların formalizə edilməsinin mürəkkəb olması ümumiləşdirilmiş modellərin qurulmasını məhdudlaşdırır (axı bir qayda olaraq konstruksiyaların tam varisliyi olmur).

CAM

Bir çox müəssisələr avtomatlaşdırma mərhələsində müşahidə edirlər ki, kompüter arxasında tez cizgi hazırlamaq məmulatların ümumi buraxılış müddətinin ixtisar olunmasına gətirib çıxarmır və çertyojların artan keyfiyyəti isə məmulatın özünün keyfiyyətinə o qədər də güclü təsir etmir. Bəzən hətta deyirlər ki, istehsal məsələlərini həll edərkən avtomatlaşdırılmış layihələndirmə sistemlərinin tətbiqi o qədər də effekt vermir.

Lakin ilkin həndəsi modelləşdirmə olmadan proqramla idarə olunan mürəkkəb dəzgahlardan istifadə etmək praktiki cəhətdən real deyildir. Belə ki, avtomobilin cinahının iki eyni modelini nəinki əl ilə frezerləmək həddən artıq mürəkkəb, hətta onun emalını proqramla idarə olunan dəzgahda proqramlaşdırmaq olduqca çətin işdir. Belə proqram on minlərlə komandadan ibarət olacaqdır.

Həndəsi modelləşdirmə ən çox o zaman effektiv olur ki, sistem özündə həm konstruktor, həm də texnoloji modelləşdirməni saxlasın. Beləliklə, indi həm modellər qurmağa, həm də proqramla idarə olunan dəzgahlar üçün proqram hazırlamağa imkan verən inteqrasiya

olunmuş CAD/CAM sistemləri haqqında danışmaq olar.

CAM sistemi (Computer Aided Manufacture – Kompüter Vasitəsilə İstehsal) texnologiyada həndəsi məsələlərin həllini avtomatlaşdırmağa imkan verir. Tez-tez o, kəsici alətin hərəkət trayektoriyasının hesablanmasını icra edir. Həndəsi modelləşdirmənin nəticəsi həm çertyojlar yığımı, həm də proqramla idarə olunan dəzgahlar üçün xüsusi formatda olan proqram olacaqdır.

Bu mərhələdə alınmış proqramı redaktə etmək və sonra daxil edilmiş dəyişiklikləri yoxlamaq üçün proqramı yenidən CAD sisteminə qaytarmaq olar.

İlk proqramla idarə olunan dəzgahlar elə qurulmuşdu ki, iş prosesində dəzgahların mühərriklərini və alətləri idarə etmək üçün kontrollerləri var idi. Belə sistemlərə proqramlar perfolentlərdən daxil edilirdi, proqramlaşdırma isə komandaların əlavə və ya silinməsindən, ya da onların parametrlərinin dəyişdirilməsindən ibarət olurdu. Bu cür sistemin hər birinin unikal idarəedici dili var idi. Hərf və rəqəm ardıcıllığından ibarət olan komandalar dəzgahın iş rejimini təyin edir, alətin yerdəyişməsini, kəsmənin dərinliyini, hərəkət sürətini və s. təsbit edirdi. Sonralar kontrolleri kompüterlə əvəz etdilər ki, bunun da displeyi, klaviaturası və dəzgahı idarə edən xüsusi proqram təminatı var idi.

CAD/CAM sistemlərinin tətbiqi nəinki məmulatın yaradılmış 3D modelinə görə proqramı generasiya etməyə, həm də istehsal prosesinə qədər işin nəticəsini ekranda görməyə, səhvləri aşkarlamağa imkan verdi. CAD/CAM sistemləri modul prinsipi əsasında qurulmuşlar. Buna görə də qoyulmuş məsələyə uyğun olaraq istehsal prosesinin ayrı-ayrı həlqələrini layihələndirmək mümkün oldu. Beləliklə, istehsal prosesinin tam bir dövrünü qurmaq üçün müvafiq proqramla idarə olunan dəzgahları



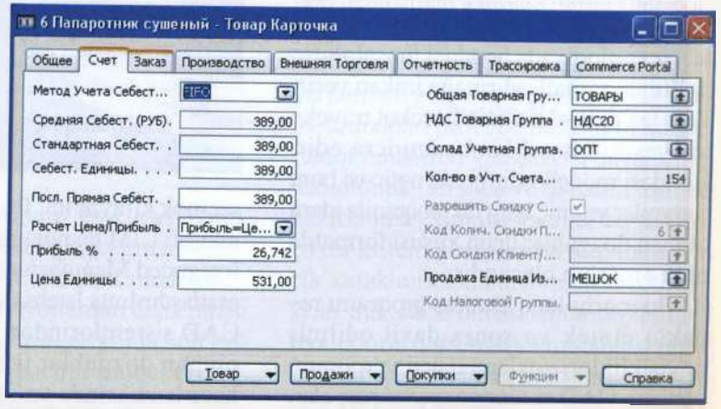
RPİ dəzgah.

seçmək kifayət idi. Bu ideyanın sonrakı inkişafı CIM texnologiyasına (Computer Integrated Manufacturing – Çevik Avtomatlaşdırılmış İstehsalat) gətirib çıxardı. CAD sistemlərindən proqramla idarə olunan dəzgahlar üçün proqramların hazırlanmasında və istehsal prosesinin qurulmasında istifadə edilir. Bu cür sistemlər CIM-in sazdanmasını təmin edən verilənlər bazasını, həmçinin bütün dəzgahlar üçün texnoloji proseslərin hamısını özündə ehtiva edir. İstehsal kiçik seriyalarla təşkil edildikdə, CIM o zaman daha faydalı olur. İnteqral sxemlər istehsal edən mini fabriklər CIM-lərdən ona görə istifadə edir ki, onları tez və rahat sazlamaq, həmçinin keyfiyyətə nəzarəti birbaşa təmin etmək mümkün olur.

Artıq lap yaxın zamanlarda sənaye texnologiyaları o qədər inkişaf edə bilər ki, kütləvi istehsal zərurəti aradan qalxar. Belə ki, tək-tək istehsal edilən məhsullar seriya ilə istehsal edilənlərdən baha olmaya bilər və hər yerdə avtomatik mini zavodlar işləyə bilər. Məsələn, alıcı mobil telefonun bir çox parametrlərini sifariş edə bilər: rəngini, korpusunu, funksiyalarını və s. Bu doğrudan da elmi-texniki inqilab olacaq!



İNFORMASİYA SİSTEMLƏRİNİN QURULMASI



Əgər müəssisə böyük deyilsə müasir müəssisənin idarə edilməsi rəhbərlikdən nəinki müəyyən vərdişlərin olmasını, həm də işlərin vəziyyəti haqda dolğun və adekvat informasiyanın olmasını tələb edir. Mühasibat qədimdən maliyyənin dövriyyəsi haqda informasiyanı toplayır və təqdim edir. Bəzən səhv olaraq elə güman edirlər ki, mühasibat uçotu fiskal orqanlar üçün, yəni vergilərin düzgün hesablanması üçün lazımdır. Əslində isə mühasibatdan gələn informasiya əsasında firmanın rəhbərliyi idarəedici qərarlar qəbul edir. Hətta elementar analiz aparmaq əsasında firmada işlərin vəziyyəti haqda müəyyən qənaətə gəlmək olar: mənfəət və ya zərər varmı, anbarlarda realizə olunmamış nə qədər mal qalıb, firmaya olan və ya firmanın başqa müəssisələrə borcu nə qədərdir və s. Lakin, yalnız müəssisənin balansına görə onun hər bir tərəfini görmək mümkün olmur, məsələn, anbarlarda qalan malların çeşidi. Bundan əlavə, anbarlarda malların yığılıb qalma səbəbləri də üzdə olan məsələ deyildir. Bunun üçün maliyyənin, istehsal prosesinin, satış bazarlarının ətraflı analizi tələb olunur. Həm də bu işləri görmək üçün çox vaxt tələb olunur və qoyulan suallara cavab alınana kimi planlaşdırılmada, istehsalda və maliyyədə növbəti səhvlər buraxıla bilər.

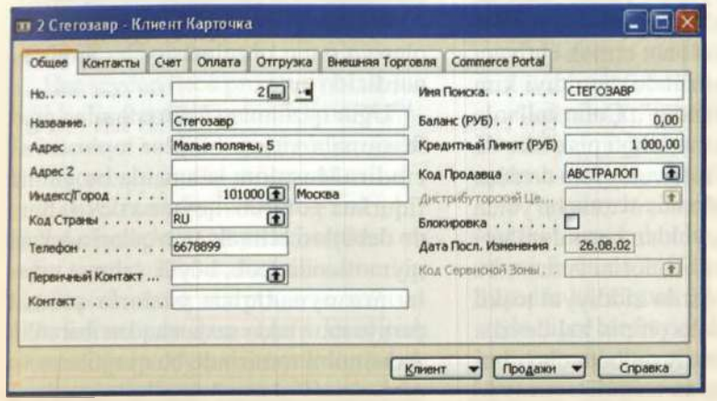
Buna görə də kompüter əsrində informasiya sistemləri olmadan keçinmək olmaz, çünki, onlar rəhbərə qərarları analiz və qəbul etməyə imkan verən informasiyanı çatdırır və müəyyən mənada avtomatlaşdırılmış idarəetmə sistemləri rolunu oynayırlar. Rəhbərə görə belə sistem tam olmalıdır, yəni firmanın direktoru sadəcə olaraq, bir düymə basaraq lazım olan bütün informasiyanı və mümkün ola bilən bütün inkişaf istiqamətlərini ala bilsin. Amma deməliyik ki, bu cür ideal sistemlər hələ ki, yoxdur. Buna görə də informasiya sistemlərini müəssisənin sifarişilə müasir bir verilənlər bazası əsasında "sıfırdan" başlayaraq hazırlayırlar. Lakin verilənlər bazası, mühasibat kimi hələ bütöv bir sistem deyil və proqram təminatının fərdi hazırlanması məsrəf və keyfiyyət baxımından daha çox uçot və idarəetmənin spesifik məsələləri üçün özünü doğruldur, məsələn, müəyyən unikal təbii resursların reyestrinin uçotu. Pərakəndə satış müəssisələrinin, məsələn, minlərlə dükanların, və ya texnikanın təminatlı təmir məntəqələrinin işi və orada cərəyan edən proseslər yaxşı öyrənilmiş və təsvir edilmişdir. Belə ki, orta kommersiya firmalarının maliyyə bloklarının fiskal hissəsi praktik olaraq eynidir: onların hamısı vergi müfəttişliyinə eyni cür hesabatlar verir, onların hesab planları üst-üstə düşür və s. Və yaxud firmanın anbarı varsa, onda praktiki olaraq bütün anbarlar üçün məhsulların uçotu, tədarükü və göndərilməsi eynidir. Qəbul olunan mal uçota alınmalı, istənilən anda anbarın vəziyyəti haqda, icra olunmuş yükləmələr haqda məlumat almaq imkanı olmalıdır.

Ona görə də konkret fəaliyyətə yönəldilmiş müəssisələr üçün müəyyən universal informasiya sistemlərindən istifadə etmək məqsədə uyğundur. Məsələn, əgər müəssisə iki əməkdaşdan ibarətdirsə və sifarişçinin yanına getməklə ona məsləhət verirsə, onda belə firmaya yəqin ki, informasiya sistemində yalnız maliyyə hesabatı aparmaq kifayətdir. Yəni cari maliyyə vəziyyəti haqqında təsəvvürə malik olmaq, mənfəət və zərər haqda informasiya almaq, onları əvvəlki dövr ilə müqayisə etmək, mənfəəti planlaşdırmaq və s. kifayətdir.

Lakin müştərilərin uçotu üçün bazanı bütün cari, hətta potensial müştərilər üçün də aparmaq məqsədə uyğundur. Bu bazaya təkcə müştərilər haqda məlumat deyil, həm də işlər üçün hesablar salınmalı, onların vaxtında ödənilməsinin yoxlanması və borcu olanlara vaxtında sorğuların göndərilməsi haqqında məlumatar salınmalıdır. Yəni belə funksiyalar müstəqil deyil, maliyyə blokunda olan informasiyanın əsasında reallaşdırılmalıdır, məsələn, bilmək lazımdır ki, bu sifarişçi ona görə "yaxşıdır" ki, o hesabları vaxtlı-vaxtında ödəyir.

Eyni informasiyanı, məsələn, müəssisə malların satışı ilə məşğul olarsa belə, saxlamaq lazımdır.

Amma bu zaman yeni bir modul olaraq – mal göndərəndən alınmış malların həcmini yoxlayan və firmanın maliyyəsi və anbarı ilə bağlı olan bir modula da malik olmaq vacibdir. Müəssisənin apardığı bütün işləri, ələlxüsus da onlar uzunmüddətli müqavilələrə əsaslanırsa, hesaba almaq lazımdır. Burada hər qrup işin gəlirlə işləməsini gözləmək, məsrəf olunmuş və ya olunacaq hər bir resursu uçota almaq və planlaşdırmaq zəruridir. Resursa həm materiallar (misal üçün inşaat firmasına baxılırsa), həm də



Əgər siz cəmdə kommunal xərcləri üçün 0 manat 0 qəpik almışsınızsa, onda müvafiq idarəyə zəng etməyə tələsin. Çünki, ola bilsin ki, bu hesabın ödənilməməsi nəticəsində avtomatlaşdırılmış informasiya sistemi sizin isti suyunuzu kəssin.

işçilər (məsələn inşaatçılar briqadası) aid edilə bilər. Əgər işin başlanğıcında onun büdcəsi və planlaşdırılan mənfəəti hesablanacaqsa, yerinə yetirilmə prosesində isə rəhbər işin cari vəziyyətini, onun məsrəflərini və maliyyə vəziyyətini izləyə bilərsə, onda bu müəyyən situasiyalarda həlledici faktora çevrilə bilər (məsələn, iş sərfəli olmadıqda müqavilədən imtina edilməsi və yaxud planlaşdırılan resurslar başqa bir tikintidə məşğul olarsa). Bütün belə suallara layihələndirilən sistem cavab verəcəkdir.

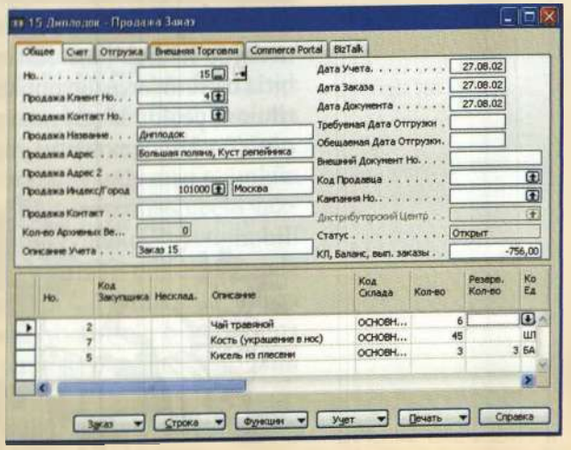
İstehsal firmaları üçün təbii ehtiyac duyulan vəzifələr istehsalın idarə edilməsi sistemi, keyfiyyətin yoxlanması, uçot, bölgü və istehsal güclərinin planlaşdırılmasıdır. Yaddan çıxarmamalıyıq ki, istehsal ziyanla işləməməlidir. Məsələn, təmir ilə məşğul olan firma, yəni xidmət firması təbii ki, dövri xarakter daşıyan bu prosesi nəzərə almaq və idarə etmək üçün müvafiq idarəedici bloka malik olmalıdır. Belə ki, avtoservisdə müştəriləri müntəzəm olaraq texniki xidmətə çağırmaq lazımdır.

Bu cür sistemlərin müasir tərkib hissələrindən biri də marketinq modullarıdır ki, onların köməyilə bu və ya digər kontakt və təqdimetmə mərasimlərinin effektivliyini nəzərə alıb qiymətləndirmək olur.

Oxşar sistemlər modullardan ibarət olaraq yığılır və konkret müəssisə üçün köklənir.

Bu cür sistemlərin layihələndirilməsi və tətbiqi asan məsələ olmayıb, bir qayda olaraq bir neçə mərhələdə icra edilir. Adətən, istənilən tətbiqdə analitiklər, sifarişçi firmanın işini yaxşı təsəvvür edən mütəxəssislər, habelə sistemin sazlanmasını yekunlaşdıran proqramçılar iştirak edir. İş prosesində analitiklər tez-tez müəssisədə baş verən proseslərin modellərini qurmaq üçün xüsusi metodlardan istifadə edirlər. Bu texnologiyalar kifayət qədər yaxşı işlənmişdir və ideal vəziyyətdə belə layihələndirmənin nəticəsi firmaya sistemin avtomatik köklənmiş variantı olacaqdır. Amma qeyd etməliyik ki, möcüzələr baş vermir və hər bir konkret halda özəl xüsusiyyətlər meydana çıxır. Ona görə də parametrlərin köklənməsindən yan keçmək mümkün olmur. Çox vaxt müəssisədəki prosesləri sistemdə qəbul edilmiş standarta uyğunlaşdırmaq lazım gəlir. Bəzən isə əksinə, sistemi yenidən elə qurmaq lazım gəlir ki, o müəssisədə adət halını almış və yaxşı yoluna qoyulmuş istehsal proseslərini pozmasın. Daha doğrusu, burada müəssisəni sistemə, sistemin isə müəssisəyə uyğunlaşdırılmasını həyata keçirmək lazımdır.

Sistemin tətbiq effektivliyi ən vacib mərhələ olan yekunlaşdırma və personalın öyrədilməsi, habelə, sınaq istismarından sənayeyə keçid mərhələlərindən asılıdır. Qeyd etməliyik ki, 100 tətbiqdən haradasa yalnız 60-ı uğurlu alınır. Önəmlisi odur ki, informasiya sistemlərinin layihələndirmə prosesi əvvəldən düşünülmüş texnologiya ilə müşayiət olunsun, yəni tətbiq bir istehsalat prosesi olub, proqramlaşdırma şamanlığından yan keçsin.



EKSPERT SİSTEMLƏRİ

Müəyyən bir işi yerinə yetirən adama, tez-tez bu işdə böyük uğurlar qazanmış başqa birisilə məsləhət etmək ehtiyacı yaranır. Atalar sözündə deyildiyi kim "Ağıl ağıldan üstündür". Çətin hallarda bir neçə bilicinin məsləhəti pis olmazdı. Lakin bir çoxları belə vəziyyətə də düşə bilər: mühərriki sönmüş avtomobil yolun kənarında qalıb, yoldan keçənlər "yeganə doğru" məsləhətlərini verir və bu məsləhətlər bir-birilə ziddiyyət təşkil edir. Lakin bu hələ ən pis hal deyil... Digər bir oxşar vəziyyət belə ola bilər: ucqar kənd xəstəxanasında və yaxud açıq dənizdə gəmidə həkim ona məlum olmayan bir xəstəliyin simptomları ilə qarşılaşıb. Bu halda təcili tədbirlər görmək lazımdır, belə ki, söhbət xəstənin həyatını xilas etməkdən gedir və nə etmək lazım gəldiyi də məlum deyil.

Bu və onlarla buna bənzər hallarda ekspert sistemləri (ES) kömək edə bilər. Ekspert sistemləri intellektual proqram sistemləri sinfinə daxildir. Bu sistemlər elə məsələləri həll etməyə qadirdir ki, onların həlli üçün ya alqoritmlər yoxdur, ya da ki, hazır metod və alqoritmləri tətbiq etmək üçün zəruri olan informasiya əlçatan deyildir. İxtiyari ekspert sistemi konkret professional fəaliyyət sahəsində meydana gələn məsələlərin həlli üçün nəzərdə tutulub (xəstələrin müalicəsi, birja bazarında vəziyyətin qiymətləndirilməsi, geoloji kəşfiyyatın nəticələrinə görə faydalı qazıntıların yerinin müəyyən edilməsi, şahmat oynamaq və s.). Aydındır ki, universal ES çox mürəkkəb olar və onun qurulması şübhəli görünərdi.



Uğur qazanmaq üçün istənilən professionala xüsusi biliklər toplusu vacibdir. Məsələn, şahmatda bu yalnız fiqurları götürüb-qoymaq deyil, həm də debütləri bilmək, mövqeləri adekvat qiymətləndirmək, böyük şahmat ustalarının oynadıqları yüzlərlə şahmat partiyasını yadda saxlamaqdan ibarətdir. Avtomobil təmirində bu qurğuların və əsas nasazlıqların öyrənilməsi, onların vaxtlı-vaxtında aşkara çıxarılması və aradan qaldırılması deməkdir. Həkim insanın anatomiyasını və fiziologiyasını mükəmməl mənimsəməli, ən çox yayılmış xəstəlikləri və onların simptomlarını, habelə müalicə üsullarını bilməlidir. Hər hansı bir sinif məsələnin həlli zamanı ES-nin işləməsi üçün zəruri və kafi olan biliklər toplusu predmet oblastı adlanır (qeyd edək ki, bilikləri və verilənləri fərqləndirmək lazımdır). Bir predmet sahəsindən olan biliklər bir-birilə bağlıdır. Bu biliklər müxtəlif obyektlərin, proses və hadisələrin təsviri, həmçinin onlar arasındakı münasibətlər ola bilər. Bütün bu məlumatlar verilənlər bazasında (VB) saxlanılır. Verilənlərin strukturu adətən sadə olur (bir qayda olaraq cədvəl şəklində). Verilənlərdən fərqli olaraq biliklər daha mürəkkəb struktura malikdirlər və xüsusi qaydada saxlanılırlar.

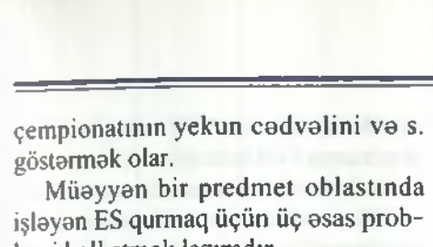
Bilikləri prosedur və deklarativ olaraq iki tipə ayırmaq olar. Prosedur biliklər fəaliyyət ardıcıllığını təsvir edir və buna alqoritmlər, fövqəladə hallar zamanı xidməti təlimatlar və s. daxildir. Bütün yerdə qalan biliklər deklarativ biliklərdir. Onlara misal olaraq bu məqalənin mətnini, Evklid aksiomalarının formulirovkasını, futbol üzrə dünya

çempionatının yekun cədvəlini və s. göstərmək olar.

Müəyyən bir predmet oblastında işləyən ES qurmaq üçün üç əsas problemi həll etmək lazımdır.

Hər şeydən öncə predmet oblastının biliklərini formalizə etmək, yəni *biliklərin təsviri modelini* seçmək lazımdır. Bir neçə belə model mövcuddur.

Şəbəkə modelində fərz edilir ki, bilikləri obyektlər (anlayışlar) toplusu və onlar arasındakı əlaqələr (münasibətlər) kimi təsvir etmək olar. O, *semantik şəbəkə* formasında əyani qrafik kimi şərh oluna bilər (anlayış və obyektlər şəbəkənin təpələri, onlar arasındakı əlaqələr isə onları birləşdirən xətlərdir). Məsələn, "Çovdar səpilmiş sahədə uca şam ağacı bitir. Onun altında isə mamırla örtülmüş qaya parçası var" mətni aşağıdakı kimi təsvir edilə bilər.



Şəbəkə modelinin qüsuru ondan ibarətdir ki, burada biliklər müxtəlif üsullarla təsvir oluna bilir, amma əlaqələr çox zaman qeyri bircins olur. Bu da biliklərin analizini çətinləşdirir və qəlizləşdirir.

Şəbəkənin quruluşunun sadələşdirilməsi arzusu freym anlayışının yaranmasına gətirib çıxardı. Bunu 1975-ci ildə suni intellekt sahəsində görkəmli alim olan Marvin Minski təklif etdi. Freym obyekti, anlayışı, hadisəni, prosesi və s. təsvir edən minimal strukturdur, ada malikdir və ardıcıl slotlardan təşkil olunmuşdur. Məsələn, "Görüş" adlı freymi bu cur təyin etmək olar:

Görüş = <Oyun> <Komanda 1>
<Komanda 2> <Hesab>

Bu cür freym prototip adlanır və onun slotları dəyişən qiymətlər ala bilər. Slotlar müəyyən qiymətlər aldıqda freymin nüsxəsi alınır:

Görüş = <Futbol> <Real>
<Barselona> <2:2>.

Aydındır ki, istənilən slotun itməsi freymin tamlığını pozur, bəzən də mənasının itməsinə gətirib çıxarır.

Freymlərdən həmçinin şəbəkənin təpələri arasında olan əlaqələri təsvir etmək üçün də isifadə olunur. Çox mühümdür ki, slotun qiyməti öz növbəsində digər bir freym ola bilər. Bu, biliklərin mürəkkəb iyerarxik quruluşa malik olan strukturlarını qurmağa imkan verir.

Şəbəkə modelindən fərqli olaraq produksion modeldə biliklərin təsvir olunması produksiya sistemlərinə əsaslanır (onun ideyası 1943-cü ildə amerika riyaziyyatçısı Emil Post tərəfindən verilmişdir). Ən sadə produksiya ad və nüvədən ibarətdir:

<Produksiyanın adı> : <Nüvə>.

"Əgər A, onda B" şəklində olan nüvə A və B faktları, fəaliyyəti, hadisələri arasında olan əlaqəni təsvir edir (burada A səbəb, B isə nəticədir). Məsələn, "Əgər meşəni doğrayırlarsa, onda yonqar tökülür" və ya "Əgər duman çökübsə, onda faranın işıqları yandırılmalıdır". Bəzən nüvədən əvvəl bir şərt də gəlir; onda produksiyanın sağ tərəfi aşağıdakı kimi olacaq: "C şərtinin ödənməsi daxilində: əgər A, onda B". Başqa sözlə şərt produksiyanın tətbiq oluna bilməsi üçün situasiya yaradır. Mənzildə olub faraları yandırmaq mümkün deyil, ona görə də son misalı belə yazmaq məqsədə uyğundur: "Avtomobili sürəndə, əgər duman çökübsə, onda faraları yandırın". Produksiya başqa elementlərdən ibarət ola bilər, amma ad və nüvə onun məcburi hissələri kimi qalmalıdır.

Özünün üstünlüklərinə görə produksiya modeli geniş yayılmışdır.

Slot – freymin bir hissəsidir. Onun tərkibində ad və kəmiyyət olur (bu başqa freym də ola bilər). Tərkibindən slotu çıxarılmış freym öz tamlığını və mənasını itirmiş olur.

Onun universallığı bundan ibarətdir ki, o, həm prosedur, həm də deklarativ bilikləri təsvir etməyə, yeni bilikləri əlavə etməyə, habelə produksiya sistemlərini paralel emal etməyə imkan verir. Əsas qüsuru ondan ibarətdir ki, yeni bilikləri əlavə etdikdə, onların uzlaşmasını yoxlamaq çətin olur.

Biliklərin müxtəlif təsvir formaları əsasında proqramlaşdırma dilləri yaradılıb. Bu dillər ES-də geniş tətbiq olunur. Məsələn, məlum PROLOG dili produksiya modelinə, FRL isə freym modelinə əsaslanır (FRL – Frame Representation Language).

ES-nin işlənilməsilə bağlı digər mühüm problem biliklər bazasının (BB) yaradılmasıdır Daha doğrusu ES-nin fəaliyyəti üçün zəruri və kafi olan biliklərin alınması və seçilməsi ES-nin yaradılmasında ən mühüm problemdir. ES-də biliklər sistemi dolğun və ziddiyyətsiz olmalıdır. BB-nı formalizə olunmuş və formalizə olunmamış kimi iki hissəyə bölmək olar. Birincilər ya sorğu kitablarına daxil edilir, ya da təlimatlar, cədvəl, düstur, alqoritmlər kimi yazılır və asanlıqla BB-na daxil edilir. Lakin bütün bunlar professional fəaliyyət üçün kifayət deyil. Misal üçün, yaxşı şahmat oynamaq üçün yalnız oyun qaydalarını bilmək qətiyyən kifayət deyil.

İkinci növ bilikləri hazır şəkildə almaq mümkün deyil. Onların toplusu kitablarda yoxdur, müvafiq düstur və alqoritmlər mövcud deyil, amma buna rəğmən bu bilikləri müvafiq mütəxəssislərin təcrübəsinə, bacarıq və intuisiyasına əsaslanaraq müəyyən qaydalar kimi formula



etmək mümkün olur. ES məhz formalizə olunmamış biliklərin köməyilə məsələləri həll etmək üçün nəzərdə tutulmuşdur.

ES qurarkən əsas mərhələlərdən biri də ekspertlərdən zəruri olan biliklərin alınmasıdır. Burada ekspert dedikdə müəyyən predmet oblastında dərin biliklərə malik olan mütəxəssis başa düşülür. Məhz onlar ES-nin malik olacağı əsas bilikləri təyin edirlər.

Ekspertlə ünsiyyət məsələsi özü özlüyündə sadə məsələ deyil. Psixoloji xarakterli məsələdən başqa (kontaktı necə yaratmalı, nəyin gərək olduğunu necə aşkarlamalı, verilən məlumatların yəqinliyini necə müəyyən etməli) daha bir məsələ meydana çıxır. Çox zaman elə olur ki, mütəxəssis bu və ya digər qərarı nə üçün verdiyini özü belə izah edə bilmir. Məsələn, qrosmeyster uduşla nəticələnmiş şahmat partiyasını araşdırarkən özü uzun müddət qalibiyyətə aparan gedişi götür-qoy edir. Və analiz göstərir ki, doğrudan da seçilmiş gediş ən yaxşı gediş imiş. Hər bir saniyə qiymətli olduğundan, aydındır ki, partiya oynanılarkən belə analiz mümkün deyildir. Deməli, qrosmeyster bu zaman intuitiv hərəkət edirdi və özünün böyük təcrübəsinə, erudisiyasına, mövqedən aldığı ümumi təəssürata əsaslanıbmış. Eləcə də sərkərdə döyüş meydanında özünün son ehtiyatını sol cinaha atırsa, o, çətin ki, bunu nə üçün məhz o zaman etdiyini, bundan bir saat tez etmədiyini izah edə bilsin. Bu cür biliklərin BB-na daxil edilməsi üçün qərarın verilməsini əsaslandırmaq lazımdır.

VB-nın köməyilə konkret predmet, proses və ya situasiya haqqında əvvəldən oraya daxil edilmiş informasiyanı almaq olar. Kompüter proqramları işlədikdə verilənlər emal obyekti kimi çıxış edirlər və onlar passiv olurlar. Proqram onları tapır, dəyişir və yenidən yaddaşda müəyyən yerə yazır. Biliklər bazası VB-dan onunla fərqlənir ki, daxili məntiqi nəticə mexanizminə malikdir.

Buna görə də ES işlədikdə biliklər özlərini fəal aparırlar: bu və ya digər biliklərin dəyişməsi onları emal edən proqramlara müraciəti aktivləşdirə bilər. Daxili prosedurlar nəinki mövcud olan bilikləri dəyişdirir, hətta onlardan yeni biliklər də əldə etməyə imkan verir. Beləliklə, BB həm ekspertlərdən alınan biliklərlə, həm də iş prosesində ES–nin özünün qurduğu biliklərlə zənginləşir.

Əsas odur ki, ES-də yalnız klassik məntiqə əsaslanan məntiqi nəticə mexanizmi reallaşmır. Onlarda həm də doğruyabənzər (qeyri-səlis məntiqə əsaslanan qeyri-səlis nəticə) və oxşarlığa və assosiasiyalara əsaslanan mühakimələrə yer verilir ki, bunlar da insan tərəfindən, adətən, qeyri-müəyyənlik şəraitində məsələlərin həlli zamanı tətbiq olunur.

Bu və ya bəzi köməkçi məsələləri həll edən vasitələr külliyatı ES-lərini təşkil edir. Onun adi sxemi aşağıdakı şəkildə göstərilmişdir.

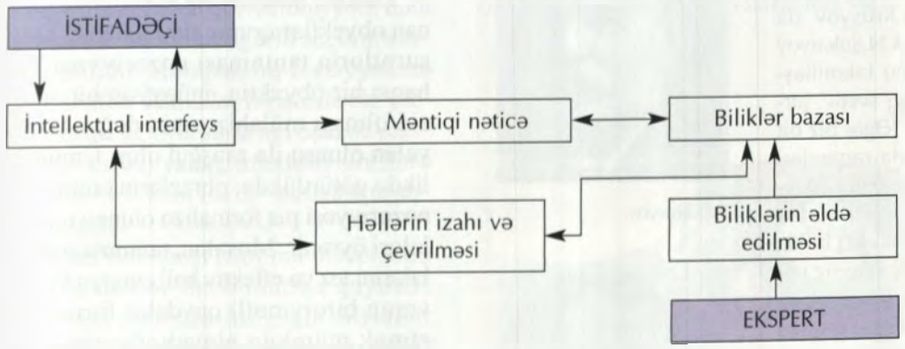
Məntiqi nəticə bloku bir və ya bir neçə həll hazırlayır. Özü də ES bu həlləri istifadəçiyə məcburi qaydada diqtə etmir, yalnız mümkün həll variantlarını



müəyyən üstüntutma və ya ehtimal əsasında təklif edir. İş dialoq formasında aparılır və buna görə də arzuolunandır ki, ES özünün fəaliyyət və nəticələrini "izah etmək" qabiliyyətinə malik olsun. Axı bəzən, alınmış variantı istifadəçi uyğun gəlməyən və ya heç də ən yaxşı olmayan variant kimi dəyərləndirə bilər. Bunun da səbəbi həldə yox, onun kifayət qədər əsaslandırılmamasındadır.

Hal-hazırda insanlara müxtəlif sahələrdə kömək edən bir çox ES vardır – havanın proqnozunu verən sistemlərdən tutmuş, verilmiş keyfiyyətlərə malik olan yeni üzvi maddələr sintez edən sistemlərə kimi.

Şübhə yoxdur ki, bu sistemlərin tətbiq sahələri getdikcə genişlənəcək, imkanları isə artacaqdır.



OBRAZLARIN TANINMASI

"Obrazların tanınması" termini altında elmi-tədqiqat işlərinin tətbiqi riyaziyyat bölməsinə aid istiqaməti başa düşülür. Surətlərin tanınması sahəsinin mütəxəssisləri hesablama texnikasının köməyilə elə məsələlər həll edirlər ki, onlar insanın intellektual fəaliyyətilə birbaşa əlaqəlidirlər. Burada əsas üsul hipotez və fərziyyələrin yolu ilə obrazların tanınması prosesinin riyazi modellərinin

"DÜŞÜNCƏ MAŞINI"

Şəxsi mülahizə və qərarları sınamaq və onların doğruluğunu yoxlamaq ehtiyacı həmişə insanlara xas olmuşdur. Fransız filosofu Rene Dekart (1596-1650) zənn edirdi ki, insanın qazandığı bilikləri riyaziyyatda teoremlər kimi ciddi qaydada çıxarmaq olar. Mülahizələri hesab formasında təsvir etmək mümkündürmü? Bu problem barəsində Qotfrid Vilhelm Leybnis də düşünmüşdür. Doğrudur, bundan 150 il sonra, XIX əsrin ortalarında Çarlz Bebbic özünün analitik maşın ideyasını təklif edəndə onun köməkçisi xanım Lavleys buna şübhə ilə yanaşmışdı. O yazırdı ki, analitik maşın nə isə yeni bir şey yaratmağa və ya analitik asılılıqları və həqiqəti tapmağa iddia etmir. Amma qeyd etməliyik ki, bu xanımın təsəvvürləri analitik maşınların çox məhdud xarakteristikalarına əsaslanırdı.



Rene Dekart.



Uilyam Stenli Cevons.

Buna baxmayaraq hələ Bebicin yaşadığı dövrdə, 1870-ci ildə ingilis məntiqçisi Uilyam Stenli Cevons (1835-1882) ilk dəfə olaraq sadə məntiqi çıxarışları mexanikləşdirə bilən ilk maşını yaratdı (əslində bu maşın məntiq cəbrinin funksiyalarını hesablaya bilirdi). Sonralar bunu rus alimi, professor P.D.Xruşyov da (1849-1909) etdi, professor A.N.Şukaryov (1864-1936) isə bu qurğunu təkmilləşdirdi. 1914-cü ildə "Vokruq sveta" jurnalında A.N.Sokov yazırdı: "Əgər biz bir rıçaqın köməyilə milyonlarla rəqəmləri toplayıb, çıxıb, vura biliriksə, onda aşkardır ki, elə bir zaman gələcəkdir ki, biz məntiqi maşına da sahib olacağıq bu da müvafiq düyməni basmaqla, səhvsiz nəticə və hökmlər çıxarsın. Bu insanın külli miqdarda vaxtına qənaət edər və həyatın ruhunu təşkil edən yaradıcılıq, fərziyyə, fantaziya və ilham üçün geniş imkanlar açardı".



P.D.Xruşyov.



A.N.Şukaryov.

Lakin qeyd etməliyik ki, əsl "düşünən maşının" yaranmasından yalnız kompüter ixtira olunandan sonra danışmaq mümkün oldu.

qurulmasından ibarətdir. Misal olaraq insanın onun foto şəklinə görə tanınması, melodiyanın tanınması, xəstəliklərin diaqnostikası və s. kimi identifikasiya məsələlərini göstərmək olar.

Adətən, tanınma obyektləri (foto şəkil, melodiya, xəstəlik) ümumi termin olan "obraz" sözü ilə ifadə olunur. İnsan tərəfindən obyektin tanınması prosesi obrazların tanınması nəzəriyyəsində (yaxud sadəcə olaraq tanınma nəzəriyyəsində) obyektin bu və ya digər oxşar obyektlər külliyatına aid edilməsi kimi izah olunur.

Belə ki, insanın foto şəklə görə tanınması prosesi obyektin (konkret insanın foto şəklinin) bu insanın bütün mümkün ola bilən fotoşəkilləri çoxluğuna aid edilməsi kimi başa düşülür. Melodiyanın tanınması isə obyektin (melodiyanın) bu melodiyanın bütün mümkün ola bilən ifaları çoxluğuna aid olması kimi başa düşülür. Xəstəliyin diaqnozunun qoyulması xəstədə müşahidə olunan simptomların konkret xəstəliyin bütün mümkün ola bilən simptomları çoxluğuna aid edilməsi kimi başa düşülür.

Ümumi əlamətlərlə xarakterizə olunan obyektlər qrupu sinif adlanır. Yəni surətlərin tanınması nəzəriyyəsi hər hansı bir obyektin müəyyən bir sinfə aid olması mülahizəsinin doğru və ya yalan olması ilə məşğul olur. Ümumilikdə götürdükdə, obrazların tanınması nəzəriyyəsi pis formalizə olunan məsələləri öyrənir. Məsələn, tanınma məsələlərini tez və effektiv həll etməyə imkan verən birqiymətli qaydalar formalizə etmək mümkün olmadıqda meydana çıxan məsələləri pis formalizə olunan məsələlərə aid etmək olar.

Obrazların tanınması nəzəriyyəsində fərz olunur ki, istənilən obyekt sonlu xarakteristikalar yığımına malikdir. Bu xarakteristikaların qiyməti həmin obyekti təyin edir. Başqa sözlə desək, bu xarakteristikalar obyekti təsvir edir.

Buna görə də xarakteristikalar – əlamətlər, əlamətlər yığımı – əlamətlər fəzası, əlamətlərin sayı isə – əlamətlər fəzasının ölçüsüdür. Məsələn, fotoşəkillərin əlamətlər fəzası onun rəqəmsal açılışının (ifadəsinin) sətrlər külliyatı hesab edilir. Burada hər sətr bir əlamətdir. Akustik rəqslərin Furye sırasına ayrılışındakı əmsallar toplusu melodiyanın əlamətlər fəzasını təşkil edir. Hər bir Furye əmsalı əlamətdir. Aydındır ki, yuxarıda göstərilmiş misallar üçün əlamətlər fəzasının ölçüsü çox böyük ola bilər. Məsələn, əksər məsələlərdə az sayda əlamətlər seçmək mümkük olmur ki, onlar obyekti bu və ya digər sinfə aid etməyə imkan versin.

Obrazların tanınması nəzəriyyəsində istifadə olunan modellərin hamısının əsasını kompaktlıq fərziyyəsi təşkil edir. Bu fərziyyəyə görə əlamətlər fəzası elə təşkil olunmalıdır ki, oxşar obyektlər bir-birinə yaxın, oxşar olmayan obyektlər isə bir-birindən uzaq yerləşsinlər. Obrazların tanınmasının bütün riyazi modelləri aşkar və ya qeyri-aşkar şəkildə bu fərziyyəyə əsaslanır. Konkret identifikasiya məsələsinin müvəffəqiyyətli həlli əlamətlər fəzasının uğurlu seçilməsindən asılıdır. Kompaktlıq fərziyyəsinə görə siniflər əlamətlər fəzasında elə yerləşməlidir ki, eyni sinfə aid olan obyektlər bir-birinə yaxın, müxtəlif siniflərə aid olan obyektlər isə bir-birindən uzaqda olsunlar.

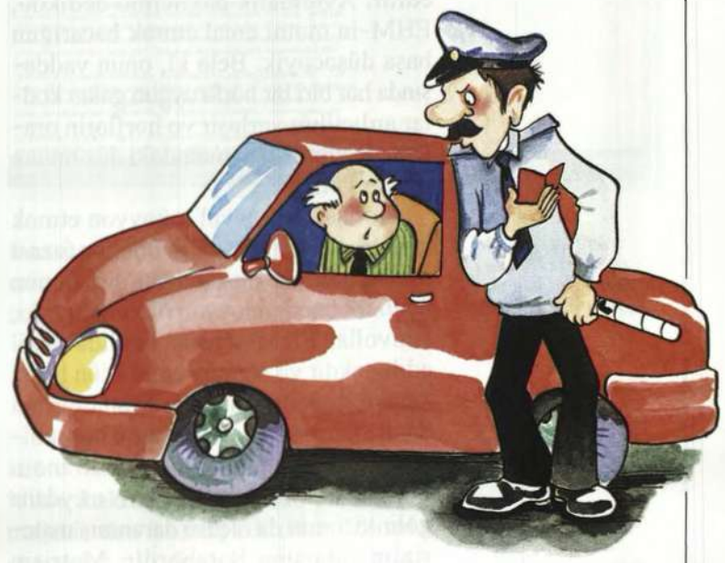
İnsan hər hansı tanınma məsələsini həll etməzdən öncə mütləq qaydada öyrənmə mərhələsini keçir. Misallar üzərində ona göstərirlər ki, hansı obyekt hansı sinfə aiddir, bütün sinifləri sadalayırlar və yoxlayırlar ki, o, bu məlumatları yaxşımı qavrayıb. Modellərdə də eyni yanaşma tətbiq olunur.

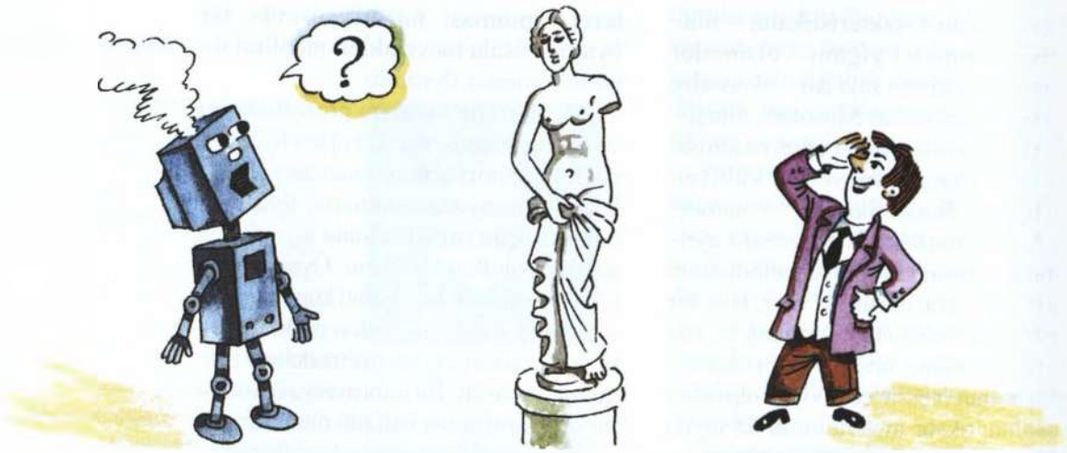
Başqa sözlə desək, tanınma prosesinə başlamazdan əvvəl modeli elə öyrətmək lazımdır ki, o, əlamətlər fəzasının bölgüsünü aparmış olsun. Obrazların tanınması nəzəriyyəsində iki öyrətmə üsulu mövcuddur: müəllim ilə və müəllimsiz öyrətmə.

Müəllim ilə öyrətmə prosesində bu və ya digər sinfə mənsub olan obyektlər çoxluğu apriori olaraq (öncədən) verilir. Bu informasiyadan əlamətlər fəzasının bölgüsü üçün istifadə olunur ki, onlar da müvafiq siniflərlə bağlanır. Öyrətmənin, yəni tanınmanın keyfiyyəti konkret sinfə mənsubiyyəti əvvəlcədən məlum olan obyektlər üzərində tanıma modelinin işilə qiymətləndirilir. Bu informasiya öyrətmə modeli tərəfindən istifadə olunmur.

İkinci üsulda da yenə obyektlər çoxluğu verilir, amma bu halda onların hər hansı bir sinfə mənsubluğu təyin olunmamışdır. Hətta, ümumi halda, siniflərin sayı da məlum olmaya bilər. Bu halda əlamətlər fəzası kompaktlıq fərziyyəsini və məsələnin özəlliyini özündə əks etdirən kriteriya əsasında siniflərə bölünür.

Obrazların tanınması nəzəriyyəsində istifadə olunan üsullar əsaslı şəkildə ilkin məlumatın özəlliyinə söykənir. Buna görə də indi elə elmi istiqamətlər formalaşıb ki, onlar vizual, musiqili obyektlərin, tibbi diaqnostikanın və s.





tanınması məsələlərini uğurla həll edir. Eyni zamanda tanımanın optimal keyfiyyətinə nail olmağa imkan verən ümumi nəzəriyyə də inkişaf etdirilir. Bu nəzəriyyə obyektlərin özəlliyinə əsaslanmır və universal sistemlər qurmağa imkan verir.

Əyanilik üçün EHM-ə mətnlərin avtomatik daxil edilməsi məsələsinə müəllimlə öyrətmə üsulunun köməyilə tanınma məsələsi kimi baxaq.

Tutaq ki, EHM-ə makina mətni daxil edilir. Avtomatik daxiletmə dedikdə, EHM-in mətni emal etmək bacarığını başa düşəcəyik. Belə ki, onun yaddaşında hər biri bir hərfə uyğun gələn kodlar ardıcıllığı yerləşir və hərflərin oradakı düzümü onun mətndəki düzümünə uyğun gəlir.

Hər şeydən əvvəl müəyyən etmək lazımdır ki, obyekt və əlamətlər fəzası olaraq nələr təyin olunmalıdır. Bunun üçün ilk öncə müəyyən olunmalıdır ki, simvollar EHM-ə hansı şəkildə daxil ediləcəkdir və sonrakı emal üçün hansı xarakteristikalar gərəkdir. Mətnlərin adi daxil edilməsi qaydası onların hər səhifəsinin daranmasıdır. Nəticədə mətn əvəzinə EHM-də ədədi matris meydana gəlir ki, onun da ölçüsü daranmış materialın sahəsinə bərabərdir. Matrisin elementləri daranmanın müvafiq nöqtəsində parlaqlığın dərəcəsinə mütənasib ədədlərdir. Başqa sözlə, səhifə darananda EHM-də orijinalı təkrarlayan müxtəlif parlaqlı nöqtələrdən təşkil olunmuş şəkil alınır.

Makina mətnində hərflərin bir-birindən boşluq ilə ayrılması xüsusi əhəmiyyətə malikdir. Bu EHM-ə xüsusi proqramın köməyilə daranmış səhifənin bütün görüntüsünü (təsvirini) hər biri bir hərfə uyğun gələn, böyük olmayan hissələrə bölməyə imkan verir.

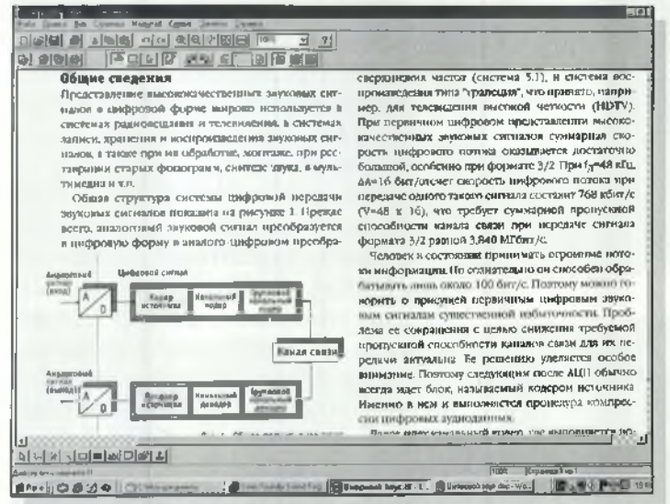
Proqram ilkin matrisin hissələrini mətndə hərflərin yerləşmə ardıcıllığı kimi, yəni soldan sağa və yuxarıdan aşağıya düzür.

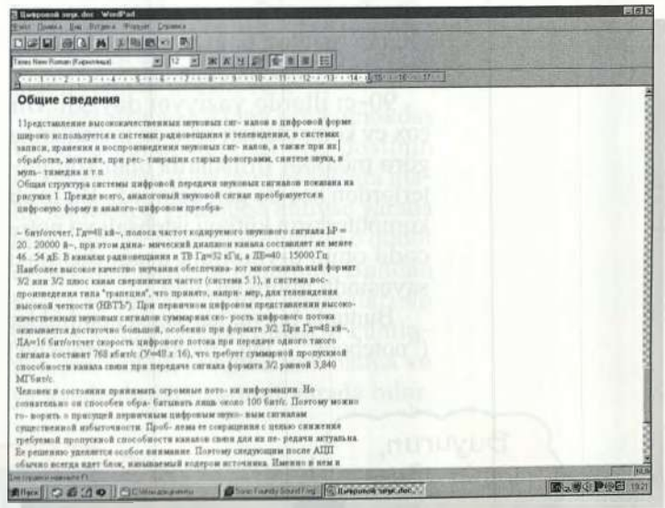
Aydındır ki, obyekt kimi ilkin matrisin kiçik fraqmentləri seçilə bilər. Müxtəlif kəmiyyətlər (məsələn, statistik momentlərin qiymətləri, ortoqonal ayrılışların momentləri və s. bu kimi hesablana bilən kəmiyyətlər) bu cür matris-obyektlərin xarakteristikaları ola bilər. Obyektlərin bu sonlu xarakteristikaları toplusu məhz əlamətlər fəzasını təşkil edir. Beləliklə, hər bir obyektə əlamətlər fəzasında bir nöqtə uyğun gəlir, fəzanın ölçüsü isə obyektin xarakteristikalarının sayı ilə müəyyən olunur. Qeyd etmək zəruridir ki, çap və daranma xətalarına görə eyni bir hərfə əlamətlər fəzasında müxtəlif nöqtələr uyğun gələcəkdir.

Güman etmək olar ki, əlifbanın hər bir hərfinə əlamətlər fəzasında özünəməxsus sahə uyğun gəlir və bu sahələr o qədər də üst-üstə düşmür. Deməli, əlamətlər fəzasında 32 müxtəlif sinif (azərbaycan əlifbasına uyğun olaraq) var. Onda azərbaycan makina yazısının avtomatik EHM-ə daxil edilməsi məsələsi hər bir hərfin 32 sinifdən birinə aid edilməsi məsələsi kimi formula edilə bilər. Daxil edilən mətnin hərflərinin avtomatik tanınmasına qədər (yəni 32 sinifdən birinə aid edilməsinə) EHM-ə bütün hərflərin etalon nümunələri daxil edilməli və onların hansı sinfə aid olduğu da göstərilməlidir. Belə hazırlıq EHM-ə əlamətlər fəzasında hər biri bir hərfə uyğun gələn sinif qurmağa imkan verir.

Siniflərin qurulması proseduru elə EHM-in müəllim ilə öyrədilməsinə uyğun gəlir. Bu cür öyrətməni müxtəlif üsullarla aparmaq olar. Məsələn, yuxarıda göstərilmiş məsələni minimal sayda hipermüstəvidən istifadə etməklə, əlamətlər fəzasını 32 kəsişməyən sahəyə bölməklə həll etmək olar. Seçilmiş metodun keyfiyyətini standart yolla, hərflərin test ardıcılığının tanınmasında səhvlərin hesablanması yolu ilə qiymətləndirmək olar.

Beləliklə, obrazların tanınması kimi formalaşan məsələlərin həlli müəyyən xassələrə malik əlamətlər fəzasının olması haqqında aprior (evristik) fərziyələrə əsaslanır. Bu xassələr əlamətlər fəzasını elə bölməyə imkan verir ki, test misalında EHM minimal səhvə yol vermiş olur.





Mətnin tanınması: yuxarıda daranmış mətn; aşağıda mətnin tanınmasının nəticəsi.

EV OFİSİ

80-ci illərdə müxtəlif təşkilatlar, müəssisələr, tədris, ev üçün yaradılmış kompüterlərdə, axırıncılar (ev üçün nəzərdə tutulmuş kompüterlər) çox zəif imkanlara və minimal proqram dəstinə malik idilər. Lakin kiçik biznesdə məhz belə maşınlardan istifadə edirdilər. Məsələ orasındadır ki, fərdi kompüterlərə sis-

temli xidmət onların yalnız quraşdırılması və təmirini deyil, həm də proqram təminatı ilə işi asanlaşdırmaq üçün məsləhəti də nəzərdə tuturdu. Bu isə çox baha olub, yalnız böyük firmalar üçün mümkün idi. Əgər ayrı-ayrı şəxslərin və kiçik sahibkarların EHM və onun proqram təminatını almağa vəsaitləri çatırdısa (çətinliklə olsa da), professional baxımdan onu müşayiət etmək onların maliyyə imkanları xaricində idi. Buna görə də əvvəllər belə müştərilər kompüterlərdən yüksək etibarlılıq, proqramlardan isə quraşdırılma və istifadə sadəliyi tələb edirdilər. Ev ofisində istismar olunan proqram təminatının dəyəri kompüterin çox da ucuz olmayan dəyərindən dəfələrlə çox idi. Ona görə də kompüterləşmiş ofisin avadanlığına sərf olunan məbləğ həm ev, həm də kiçik firmalar üçün çox baha idi.

90-cı illərdə vəziyyət dəyişdi. Bir çox ev kompüterləri artıq öz imkanlarına görə mötəbər firmalarda olan kompüterlərdən heç də geri qalmırdı. Bu fərdi kompüterlərin kütləvi istehsalı və nəticədə onun qiymətinin aşağı düşməsi sayəsində mümkün oldu.

Bununla bir vaxtda satışa noutbuklar ("notebook" – *ing.* "bloknot" deməkdir) çıxarıldı. Bunlar bir yerdən başqa bir yerə asanlıqla aparıla bilən, müstəvi səthə malik maye kristallı ekranı olan kompüterlər idi. Belə maşınlar kiçik biznes üçün çox rahat idi. Belə ki, onun sahibi bütün sənədləri özü ilə daşıya bilər və hətta işi digər ofisdə, mehmanxana, ev və başqa yerlərdə davam etdirə bilərdi. Doğrudur noutbuk masaüstü kompüterlərə nisbətən baha idi, lakin o, tez-tez lazım gələn səfərlər üçün əvəzolunmaz idi.

Yeni güclü maşınlar üçün müvafiq proqram təminatı da lazım idi. Nə xüsusi istifadəçiləri, nə də iş adamlarını çox məhdud imkanlara malik proqramlar dəsti qane etmirdi. Buna baxmayaraq, lazımi proqramların satın alınması yenə də onlar üçün çətin idi. Bunu nəzərə alan bəzi firmalar onlara "limited edişn" (*ing.* "limited edition" – "məhdud seriyalar" deməkdir) təklif etməyə başladılar ki, bunlar da evdə və kiçik müəssisədə işləmək üçün tam yararlı idilər.

XX əsrin sonunda SOHO termini meydana gəldi (*ing.* "Small Office/Home Office" – "Kiçik Ofis/Ev Ofisi"). Bu müvafiq komponovkalı, proqram və avadanlıqları olan, ev və kiçik biznes üçün nəzərdə tutulmuş kompüter idi.

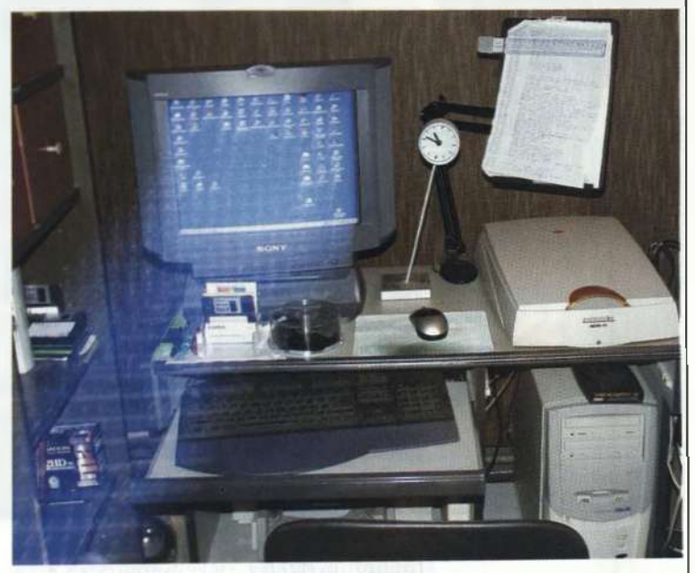


Müasir SOHO Microsoft Windows tipli pəncərə sisteminə əsaslanır. Onun ofis dəsti mükəmməl mətn redaktorundan, elektron cədvəllərdən, verilənlər bazası və poçt proqramından ibarətdir. Poçt proqramı sadə planlaşdırıcı, qeyd kitabçası və təqvim funksiyalarını özündə saxlayır (təbii ki, Microsoft firmasının müvafiq dəstinin adı Office adlanır və onun belə komponentləri var: Word mətn redaktoru, Excel elektron cədvəli, Access verilənlər bazası və Outlook dəsti).

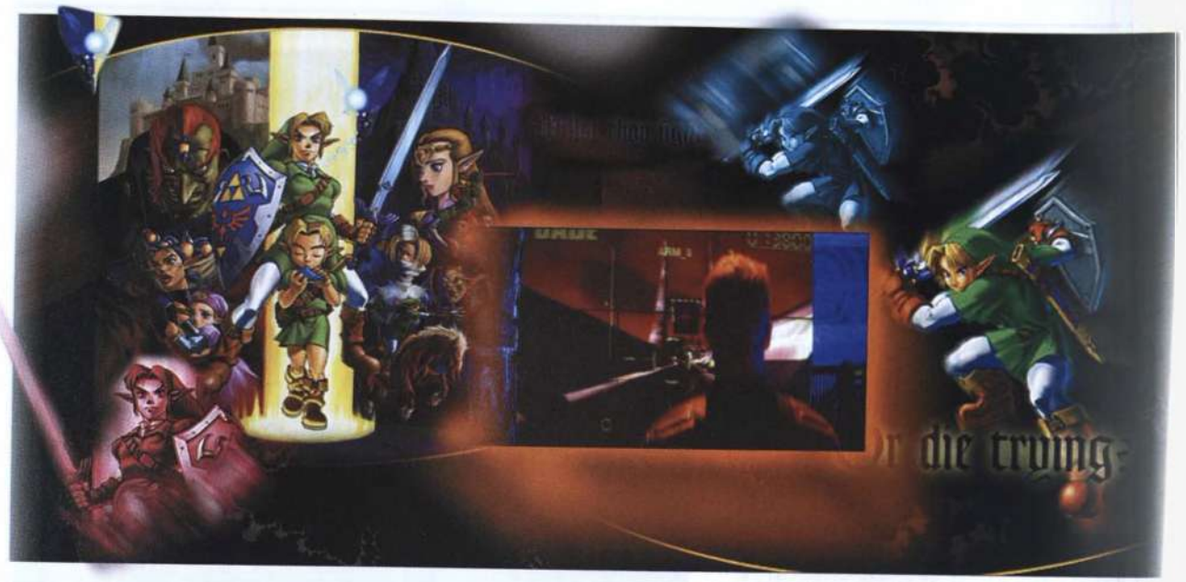
Microsoft Outlook tipli sistemlər faktiki olaraq orqanayzer (təşkilatçı) rolunu oynayır. Onların köməyilə İnternet vasitəsilə poçtu alıb və göndərmək, təqvim və qarşıda duran işləri tərtib etmək, tərəfdaşların koordinatını qeyd etməklə, fərdi qeyd kitabçasını yaratmaq, cari işlərin siyahısını tərtib etmək və onların icrasına nəzarət etmək, müxtəlif qeydlər aparmaq və s. mümkündür.

Təqdimatlar hazırlayan dəst (Power Point) mətni, şəkli və fotoları kombinə etmək və sonra onları lazımi ardıcıllıqla düzməklə, təqdimatı ya kompüterin ekranına, ya da noutbuka qoşulmuş proyektorun köməyilə adi ekrana verə bilir.

Ev və ofis üçün proqramlar dəsti bir qayda olaraq şəxsi veb səhifələr hazırlamaq üçün proqramları da özündə saxlayır. Veb səhifə ona biznesi və özü haqqında şəxsi informasiyanı geniş kütləyə təqdim etmək imkanı verir.



Əgər firmada ikidən artıq əməkdaş varsa, o halda standart ofis dəstinin imkanlarından daha artıq imkanlara malik idarəetmə sisteminə ehtiyac yarana bilər. Bu cür sistemlər evlərdə o qədər də tez-tez istifadə olunmur. Bundan başqa bəzən sahibkarın maraqları və biznes xüsusiyyətləri əlavə, o cümlədən çox bahalı, proqram təminatına və ya avadanlıqlara zərurət yarada bilər.



OYUNLAR

KOMPÜTER OYUNLARI

Kompüter oyunlarının meydana gəlməsi kompüterlərin eksperimental və demək olar ki, məxfi sferadan (axı hərb sahələrində raket və mərmilərin trayektoriyası kompüterlərdə hesablanırdı) elmi və praktiki sahəyə keçid dövrünə təsadüf edir. Bu keçid XX əsrin 60-cı illərinin sonunda baş verdi.

Kompüter bu və ya digər formada istifadəçinin "dostu" olan interfeysə çevrildi. Lampa və sirli indikatorların əvəzinə əlifba-rəqəmli displeylər meydana gəldi. Əlbəttə, heç bir qrafikadan söhbət belə gedə bilməzdi... Lakin kompüterdə insanlar işləyir və insana xas olan heç bir xüsusiyyət onlara yad deyil. Və beləliklə, bir gözəl axşam, ağır iş günündən sonra, bir gənc proqramçı qərara gəldi ki, kiçik bir proqram yazsın və o, çox da mürəkkəb olmayan oyunu, məsələn, "Öküzlər və inəklər" oyununu proqramlaşdırdı... Təbii ki, belə fikir yalnız tək onun ağlına gəlməmişdi... Beləliklə, əyləncə üçün proqramlar tez-tez meydana gəldi və onlar hətta kompüterlə birgə təqdim olunan proqramlar dəstinə daxil edilməyə başladı.

Kompüter qrafikasının yaranması və əsl ev kompüterlərinin meydana gəlməsilə oyun sənayesi yüksəlməyə başladı. Oyunlar, pirat surətlərini saymamaq şərtilə, minlərlə nüsxə buraxılmağa başladı. Təxminən on il müddətinə Sinclair Research firmasının ZX-Spectrum ev kompüteri üçün 6 mindən artıq (!) müxtəlif növ oyun yazılmışdı.

İndi oyun sənayesi fərdi kompüter sənayesinin dayaq nöqtələrindən birinə çevrilmişdir və doğrudan da, evdə kompüter oyundan başqa daha nə üçün gərəkli ola bilər?



Hər bir zarafatda həqiqət var, kompüter oyunları olmasaydı, dünya "kompüter inqilabının" şahidi olmazdı.

Oyun sənayesində aşağıdakıları fərqləndirirlər:

• *Mətn oyunları.* Bunlara "Star Trek", "Adventure", "Rogue" kimi oyunlar aiddir. Oyunçu məqsədə çatmaq üçün müəyyən işlər görməlidir: qalaktikanı fəth etməli, xəzinəni tapmalı, əjdahanı öldürməli və s. Bu cür oyunlar üçün əlifba-rəqəmli displey kifayətdir. Burada idarəetmə, məsələn, "on addım irəliləməli" tipli sadə əmrlərin daxil edilməsilə icra edilir. Bu əmrlərə kompüterin cavab reaksiyası belə ola bilər: "Qaranlıq mağaradasınız və sizin yanınızda pis niyyətləri olan nəhəng əjdaha var . Hərəkətiniz nədən ibarət olacaqdır?" Oyunu davam etdirmək üçün oyunçu yeni əmrlər daxil etməlidir.

• *Oyunlar üçün əlavə qurğular.* 1980-ci ildə ATARİ firmasının menecerlərindən biri anladı ki, artıq adi oyuncaqlarla heç kimi təəccübləndirmək mümkün deyil və yeni istiqamətləri mənimsəmək gərəkdir. Elektronika o dövrdə artıq geniş imkanlara malik idi. Beləliklə, oyunlar üçün əlavə qurğular meydana gəldi – bu kiçik ölçülü qutulardan ibarət olub, adi televizorlara qoşulan və müasir baxımdan sadə oyunları oynamağa imkan verirdi. Kompüter qrafikasının yenilik olduğu o dövrdə personajların televizor ekranlarında sxematik təsviri alıcıları heyran qoyurdu. Sonralar əlavə qurğuların gücü artmağa başladı və oyunlar, yəni proqramlar əvvəlcə kartriclərə, sonra isə kompakt disklərə köçürüldü. Artiq bu gün oyunlar üçün əlavə qurğular öz imkanlarına görə ən yaxşı ev kompüterlərini üstələyir. Bu qurğularda ən müasir qrafiki prosessorlar istifadə olunur. Onlar qrafikləri tez emal etmək üçün xüsusi olaraq yaradılmışdır. Digər tərəfdən, analoji imkanlarına görə, oyunlar üçün əlavə qurğular fərdi kompüterlərlə müqayisədə kifayət qədər ucuzdur və alıcı dəbdən düşmüş köhnə aparatın əvəzinə yenisini almaq imkanına malikdir.

• *Ev kompüterləri.* Əlavə qurğularda oynanılan oyunlar nə qədər populyar olsa da, istifadəçi təkcə oyunlarla kifayətlənmək istəmirdi, o, həm də hansısa sadə məsələləri də həll etmək istəyirdi (məsələn, proqramlar yazmaq, ev kitabxanasının kataloqunu aparmaq və s.). Ev kompüterləri adi televizorlara qoşulurdu, böyük olmayan daxili yaddaşa malik idi və informasiyanın yazılması üçün kasetli maqnitofona qoşulurdu. Onlardan ən məşhurları – ZX-Spectrum (Böyük Britaniya), Commodore 64 (ABŞ və Qərbi Avropa), MSX (Yaponiya) idi.

İndi isə oyunlar, adətən, PC və yaxud Macintosh platformaları üçün buraxılır. Hər il satışda müxtəlif zövqlərə uyğun gələn yeni-yeni oyunlar meydana gəlir. Bu yeni oyunlarda üç ölçülü qrafikanın, səs və süni intellektin ən son nailiyyətlərindən istifadə olunur.



Əgər sizin oyuncaqlarla dolu disklərinizin sayı uşaqlarınızın müvafiq disklərindən azdırsa, onda siz bu dünyaya bir qədər tez gəlmisiz.





Kompüter oyunlarını aşağıdakı janrlara bölmək olar.

• *Arkad oyunları.* Bu oyunların məqsədi gəlmələri məhv etmək, bütün xəzinələri yığmaq və ya hansısa divlərdən qurtulub qaçmaqdan ibarətdir. Onlar qrafikanın və səs effektlərinin böyük çeşidliliyilə fərqlənirlər. Üstəlik bu zaman oyunçu bir o qədər də təmkinli deyilsə, onda nəticədə parça-parça olmuş klaviatura və sınmış coystiklərsiz (hərəkəti idarə edən xüsusi qurğü) keçinmək mümkün olmayacaqdır.

• *Strategiya.* Bu janr özündən əvvəlkindən heç də az ehtiraslı deyildir. Adətən, oyunçu hərbi sərkərdə olur və onun da komandanlığına hansısa qoşunlar verilir. Burada düşmən ordusunu darmadağın etmək üçün, adətən, sürətli reaksiyadan əlavə, həm də müəyyən əqli qabiliyyət tələb olunur.

• *Simulyatolar.* Bu janrın oyunlarında istifadəçi özünü pilot və ya kosmik gəminin kapitanı qismində sınaya bilər. Müasir aviasimulyatorlar o qədər mürəkkəbdir ki, adətən, uçuş aparatının bütün idarəetmə detallarını əks etdirmək üçün klaviaturadakı düymələr kifayət etmir.

• *Məntiqi oyunlar.* Bu oyunlara – tapmacalar, kart oyunları, şahmat və dama oyunları aiddir. Bu oyunlarda oyunçu üçün "baş sındırası" məqamlar çoxdur.

• *Rol oyunları.* Bu oyunlar "Zirzəmi və divlər" tipli oyunların meydana gəlməsilə intişar tapıb ki, bunun da önəmli cəhəti hər bir oyunçunun hansısa bir fərqli xüsusiyyətə malik olmasındadır (zərbənin gücü, sehrin gücü, həyat gücü və s.). Oyunçunun qrupda rolu bu göstəricilərin balansından asılıdır. Məsələn, sehrkar ən böyük intellektə malikdir, döyüşçü güclü, oğru isə diribaşdır və s. Qrup müəyyən tapşırığı yerinə yetirməlidir: divə qalib gəlməli, xəzinəni tapmalı, şahzadə xanımı xilas etməlidir. Hər bir hərəkət uduşla başa çatır. Oyunu oyunçuların hər bir hərəkətini izləyən və yeni hadisələri elan edən "mağaralar ustası" idarə edir.

Oyunun elektron versiyasında "usta" rolunda bir qayda olaraq kompüter çıxış edir. Zamana görə oyun dövrlərə bölünür. Bir dövr ərzində qrup yalnız müəyyən sayda hərəkətləri icra edə bilər. Bu hal stolüstü oyunlardakı vəziyyəti xatırladır. Belə ki, oyun lövhəsi üzərində hər bir hərəkət atılan zərin rəqəmlərinə





uyğun gəlir. İnternetin inkişafı ilə rol oyunları daha da populyarlaşdı. Artıq oyunçunun tərəf müqabilinin harada olması önəmli deyil – istər Çində olsun, istərsə də Norveçdə.

Oyunların uşaq və yeniyetmələrin kövrək şüurlarına təsiri birmənalı qiymətləndirilmir. Kimsə bu oyunlar vasitəsilə məntiqi düşünmə qabiliyyətini inkişaf etdirir, kimsə də əksinə, divlərə atəş açmağa aludə olur və onun ətrafında olan real dünyanı unudur. Bu zaman mübahisəsiz odur ki, kompüterə maraq oyanır.

Oyunlar çox önəmli öyrətmə vasitəsidir. Qavrama üçün vacib olan informasiyanı oyun formasında həm beşyaşlı uşaqlara (hesabı və ya yazını öyrətmək üçün), həm də kifayət qədər yaşlı adamlara ötürmək olar (məsələn, işgüzar oyunlar və ya oyun formasında olan testlər vasitəsilə).

Əgər kompüter sənayesindən söhbət gedirsə, onda qeyd etməliyik ki, oyunlar yeni kompüterlərin istehsalını, üç ölçülü qrafika və səs sahəsində təkmilləşdirmələri stimullaşdıran ən güclü amillərdən biridir.

Oyunlar kompüter tarixinin bir hissəsidir. Və məlumdur ki, tarix spiralvari inkişaf edir. Bizi qarşıda oyunlar üçün yeni əlavə qurğuların meydana gəlməsi, kompüterlərin "ev üçün" və "o qədər də ev üçün olmayan" bölgüləri, çoxlu sayda yeni oyunlar və bunlarla bağlı texnologiyalar və avadanlıqlar gözləyir.

"TETRİS" FENOMENİ

Rus proqramçısı Aleksandr Pojitnov demək olar ki, əjdaha yaratmışdır. Və bu əjdahanın da adı "tetrisdir". Bu məşhur oyun bütün kompüter platformalarını, cib qurğularını və mobil telefonları fəth etmişdir. Bütün dahiyanəliklərdə olduğu kimi, onun da qaydaları çox sadədir. Yuxarıdan həndəsi fiqurlar düşür, onları da elə fırlatmaq və yerləşdirmək lazımdır ki, bu fiqurlar arasında boş yer qalmasın. Bu oyunun nəticəsində bütün dünya üzrə kompüterlər meqavatlarla enerji işlətmiş, göz həkimləri yeni xəstələrin axını sahəsində daha artıq mənfəət əldə etmişdir və hələ də "tetris" bizim monitorların ekranlarını fəth etməkdə davam edir. Belə görünür ki, tetris proqramı bu vaxta qədər yaradılmış bütün kompüter oyunlarından ən yaxşısıdır.

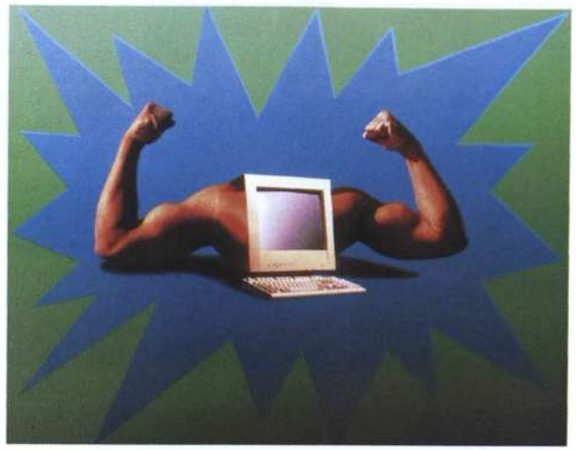
TRENAJORLAR

Kompüter texnologiyaları təkidlə gündəlik həyatımıza daxil olmaqdadır. Kompüterlər hətta ilk baxışdan elektronika və informatika ilə heç bir əlaqəsi olmayan sahələrdə də tətbiq olunur.

Məsələn, ilk baxışda müasir kompüterlər ilə hər bir şeyi idmançının reaksiyası, dözümlülüyü və gücünün həll etdiyi idman arasında nə əlaqə ola bilər? Sən demə, burada birbaşa əlaqə varmış!

Belə ki, kompüterdə nəinki idmançı üçün ən yaxşı pəhrizi hesablamaq və ya futbol meydançasında rəqibi keçmək üçün əlverişli kombinasiyanı qurmaq olar, həm də yolda "Formula-1" yarışı üçün hərəkəti modelləşdirmək olar.

Son on ildə kompüterlərdən idmançıların məşqlərində istifadə edilməsi adi hala çevrilmidir. Məsələn, Xuan Pablo Montoya "Formula-1" yarış-

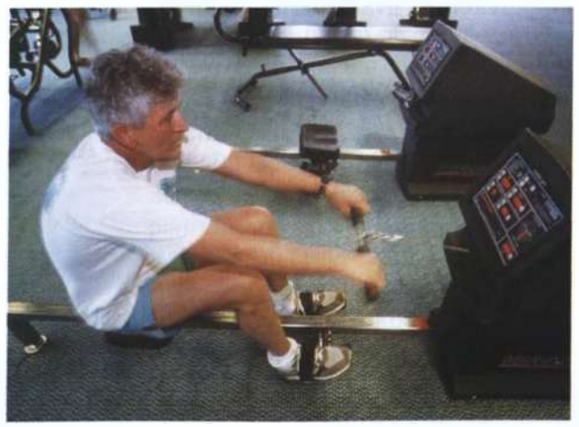


larının debütünə hazırlaşarkən "Grand Prix" oyunu vasitəsilə yarışın yolu ilə tanış olmuşdu.

Kompüterlərin trenajorlar kimi geniş istifadə edilməsi üçölçülü qrafika texnologiyasının inkişafı, kompüterlərin məhsuldarlığının artması və güclü proqram təminatı vasitələrinin yaradılması hesabına yeni yazılan proqramların maya dəyərinin aşağı düşməsilə əlaqədardır.

Trenajorun əsas vəzifəsi – simulyasiya, yəni real həyat vəziyyətinin imitasiya edilməsidir. Bu, məsələn, avtomobil yarışında hərəkət, xizəksürmədə dik yüksəklikdən eniş, yaxud dənizdə yaxta ilə hərəkətin imitasiyası ola bilər.

Landşaftın yaxud yolun konturlarını çəkən proqram bu və ya digər coğrafi



nöqtənin reallığını maksimal dərəcədə dəqiq əks etdirməlidir. Bu zaman küləyin sürəti, işıqlanma, səsin təsiri kimi amillər nəzərə alınmalıdır. Yaxşı olardı ki, məşq edənə yarışlarda rast gəlinən iki-üç ekstremal vəziyyət göstərilsin. Belə hallar onun reaksiyasına qiymətləndirmək və nəticədə ona lazımi tövsiyələr vermək üçün gərəkli olur.

Əlbəttə, nə qədər mürəkkəb və universal olsa da, yalnız proqram ilə kifayətlənmək olmaz. Deməli, ətraf mühitin daha tam və dolğun simulyasiyası üçün təkcə proqram təminatına yox, həm də texniki vasitələrə ehtiyac vardır.

Gəlin baxaq: avtoyürüşü simulyasiya edən proqramdan əlavə avtotrenajoru yaratmaq üçün daha nələr lazımdır. Axı real həyatda yalnız düymələri basmaqla uzağa getmək olmaz.

ƏKS ƏLAQƏLİ SÜKAN

XX əsrin 90-cı illərində oyun avtomatlarında yürüş yolu ilə hərəkət edən maşını idarə etmək mümkün idi. "Daxiletmə qurğusu" olaraq sükandan istifadə edilirdi. O əsl sükana oxşayırdı, amma heç bir güc tətbiq edilmədən o tərəf bu tərəfə fırlanırdı və ona görə də bu cür "maşın sürmədən" xüsusi bir effekt hasil olmurdu. İllər keçdi. Oyun manipulyatorları ilə məşğul olan konstruktorlar qəribə şey hazırladılar – bu əks əlaqəli manipulyator idi. Sükan, coystik, GamePad (videooyunlar üçün iki əllə tutulan cihaz: bir əlin barmaqları altında "atəş" düymələri, digərlərində isə hərəkəti idarə etmək üçün tutacaq) oyunun şərtlərindən asılı olaraq özlərini müvafiq qaydada aparırdılar.

Məsələn, oyunçu dik virajdan çıxanda sükan nəzərə çarpacaq müqavimətlə dönürdü. Daha doğrusu, əsl real vəziyyət yaşanırdı.

Əks əlaqəli qurğunun daxilində kifayət qədər mürəkkəb komponentlər var

idi: servomotorlar, vibrasiya ötürücüsü, elektromaqnitlər. Bütün bunları əsas proqramın işlədiyi EHM-dən siqnalları qəbul edən bir və ya bir neçə mikrosxem – kontrollerlər idarə edir. Kontroller müəyyən komponentləri ya qoşur, ya da ayırır və beləliklə, sükanın ağırlığını hiss etdirir və ya maşının döngələrdə özünü necə aparmasını sınaqdan keçirməyə imkan verir.

Bundan başqa əsl maşında çoxlu sayda qurğular, məsələn, qaz və əyləc pedalları, sürət ötürücüsü var. Onları da trenajorda simulyasiya etmək mümkündür.

Prinsipcə bu cür avadanlıq tamamilə kifayətdir ki, istifadəçi özünü meteorun əsl pilotu kimi hiss etsin (ola bilsin ki, bura bir qədər də böyük monitor əlavə etmək lazım gəlsin). Çoxkanallı səs və əyləclərin ciyiltisi, habelə mühərrikin səsi təsviri tamamlayır.

Bu cür trenajoru evdə də qurmaq olar, amma professional trenajorlarda, adətən, irəlini, arxanı və yanları müşahidə etmək üçün bir neçə monitordan istifadə olunur. Bundan əlavə "pilotun" kreslosu bütün incəlikləri və nüansları ilə real pilotun kreslosunu təkrar edir.

Mümkündür ki, "Formula-1" yarışlarını keçirmək üçün gələcəkdə bahalı



Mətbuatın xəbərlərinə görə 13 yaşlı oğlan uşağında kömürsındıran çəkiclə işləyən fəhləyə xas olan professional xəstəliyin əlamətlərini müşahidə etmişlər. Və bunu belə izah etmişlər ki, bu yeniyetmə uzun müddət (gündə altı saat olmaqla) əks əlaqəyə malik olan coystiklə müvafiq video oyunla məşğul olmuşdur.

yollara ehtiyac qalmasın. Bu fikir fantastika deyil. Fərz edək ki, zalda avarçəkmə yarışlarında idmançılar öz yerlərində oturaraq, avarlarla suyu bir hovuzdan başqa bir hovuza axıdır və bu zaman tamaşaçı və hakimlər monitorların ekranlarında bu "yarışın" nəticələrini müşahidə edirlər. İnanın ki, bununla həmin yarışların baxımlılığı və gərginliyi azalmır.

SÜNİ GERÇƏKLİKLƏR

"...Dünən, yalnız dünən evdən milyonlarla işıq ili məsafəsində olan lənətə gəlmiş planeti kəşf etdim. Son taqətim məni tərk edirdi, özümü söyürdüm ki, bu planeti tədqiq etməyə gedərkən özümlə robotu götürməmişəm. Yerin ulduz donanmasında qəbul edilmiş qaydaya görə yeni planetdə robotsuz mənim heç bircə addım da atmağa ixtiyarım yox idi... Möhkəm isti idi, təngənəfəs halda mən tozlu, isti günəşin yandırıb tökdüyü səhra ilə qaçırdım. Yapışqan kimi tər az qala gözümə düşəcəkdi, mən qarışmış saçlarımı alnımdam yana etdim. Əllərim və ayaqlarım qançır olmuşdu. Qulaqbatırıcı bağırtı damarlarda qanımı dondurmuş və bədənimi iflic edən qorxu fikrimi cəm etməyə imkan vermirdi. Yalnız iradəmi qeyri-adi toplamaqla ayaqlarımı zorla sürüyürdüm.

Mənim gəmimə iki yüz metrə qədər məsafə qalmışdı ki, orada da təhlükəsiz sığınacaq və istənilən əjdahanı məhv edə bilən silah var idi. Amma yerin ayağımın altında titrəməsindən bildim ki, daha mənim xilas olmaq şansım qalmayıb... Və mən qaçdım.



Bütün vücudumu vəhşi, hər şeyi dəlib keçən ağrı bürüdü, elə bil kürəyimə qaynar qazan bağlamışdılar və o, qıc olmuşdu, ayaqlarım titrəyirdi və huşumu itirərkən mən anladım ki, bu dəfə də əjdahaya məğlub oldum...

Bu yuxu deyildi. Ağrı səngiyəndə, tövşüməkdə davam edərək gözlərim qarşısında belə bir yazı gördüm: "Oyun qurtardı". Dərindən nəfəs alıb, dəbilqəni çıxardım və təəssüflə kombinezonu soyunmağa başladım. Bədənim gizildəyir və ağrıyırdı, nəfəsim hələ də qeyri-sabit idi və mən qapını açıb Virtual reallığı tərk etdim..."

"Virtuallıq" termini müxtəlif elmlərdə və bilik sahələrində əvvəllər də istifadə olunmuşdu, məsələn fizikada bəzi obyektlərin xəyali olduğunu göstərirdi. O, həmçinin kompüterin köməyilə reallaşdırılan üçölçülü makro modellərdə də tətbiq olunur.

Qəbul olunmuş stereotipə görə virtual reallıq nə isə kompüter oyunları və fantastik filmlərlə bağlı olan bir şeydir. Bu qismən belə olsa da, alimlər belə hesab edirlər ki, virtual reallığa real həyatda da yer tapıla bilər. Belə ki, bu çox böyük ölçülü verilənləri emal etməyə və elmi araşdırmaların nəticələrini "görünən" etməyə imkan verir.

Modelləşdirilmiş obyekti müxtəlif mühitlərə yerləşdirmək olar, onun hərəkətini nəinki xəyali fəzada imitasiya edib, izləmək, həm də onun işləməsini nümayiş etdirmək də olar. Kompüter modelləşdirməsi mürəkkəb məmulatı real müəssisədə deyil, virtual mühitdə yaratmağa və qiymətləndirməyə imkan verir. Bu ələlxüsus da bahalı, mürəkkəb və universal komplekslər üçün aktualdır.

İndiki zamanda virtual modelləşdirmə texnologiyaları daha geniş miqyasda tətbiq edilir. Obyektin virtual modelinin yaradılması onun sonrakı istehsalı, virtual prototip (təhlükəsizlik, funksionallıq və s. xassələri) mərhələsində qiymətləndirilməsi və onun hazırlanmasının optimallaşdırılması üçün nəzərdə tutulmuşdur. Yalnız qaneedici nəticələr alındıqdan sonra fiziki obyektin hazırlanması haqda qərar qəbul edilir. Bu ona görə zəruridir ki, elmi və mühəndis işləmələrinə daha az vaxt və vəsait sərf edilsin. Onda laboratoriyadan istehsal sexinə qədər olan yol qısalar və məmulatın xarici görünüşünün aspektləri hazır məhsul üzərində deyil, kompüter modelləri üzərində təyin edilər.

Virtual reallığın iki əsas tipi mövcuddur: səhnə və ekran (*ing.* scenic və screen). Səhnə tipində – insan sanki virtual mühitin içərisində yerləşir və onun obyektlərindən biri olur. Ekran tipində isə insan virtual aləmdən təcrid edilmişdir və onu kənardan müşahidə edir (məsələn, kompüterin arxasında oturan insan kimi).

Ekran sistemləri virtual mühitə qismən daxil olmağı təmin edir, yəni onun bir hissəsi kompüterdə modelləşdirilir, digər bir hissəsi isə fiziki olaraq, müxtəlif pultlar, işçi yerləri, kabinələr və s. şəklində imitasiya edilir.

İngiliscə "virtual reality" söz birləşməsi "faktiki gerçəklik" kimi tərcümə olunur. Lakin bu gün istifadə etdiyimiz "virtual reallıq" termini süni yaradılan və həqiqətə oxşayan reallığa aid edilir.

Səhnə tipli virtual reallıq praktiki olaraq virtual mühitə tam daxil olmağa imkan verir. İnsanın ətraf mühit ilə təması beş əsas hiss orqanının köməyilə baş verir: görmə, eşitmə, lamisə (toxunma hissi), iybilmə və dad. Bəzi və yaxud bütün hiss orqanları üçün informasiya mənbəyini dəyişdirdikdə, başqa bir dünyada olma təəssüratı yaranır. Bu tipli reallığın formalaşdırılması üçün xüsusi texniki vasitələr gərək olur: müxtəlif növ təsvir sistemləri, dəbilqələr və eynəklər (onlar qeyri-məhdud və dəyişən görmə sahəsini təmin edirlər), müxtəlif növ ötürücülər (onlar bədənin vəziyyətini və mövcud olmayan obyektlərə toxunulmaqla duyulan əlaqələri yoxlayır). Bu zaman önəmli elementlər kimi fəza səslərinin səsləndirilməsi sistemləri çıxış edir. Beləliklə, virtual kosmik aparatın səthində insan maşınların səsini, öz nəfəsini, habelə həmkarlarının canlı danışığını eşidəcək və s.

Belə bir fikir yaranıb ki, dəbilqə üstəgəl əlcəklər virtual reallığın elə zəruri və kafi atributlarıdır ki, onların köməyilə kompüterdə yaradılmış süni aləmə daxil olmaq olar. Əslində isə virtual aləmi reallaşdırmaq həddindən artıq çətindir. Ondan da çətini bizim hiss orqanlarımız, məsələn, görmə üçün informasiyanı dəyişdirməkdən ibarətdir. Biz dünyanı gözün torlu qişası vasitəsilə görürük. Alimlərin hesablamalarına görə tor qişasının ayırdetmə qabiliyyəti 8000×8000 nöqtədən ibarətdir. Bu cür şəkli kompüterdə almaq üçün rənglərin dərinliyi 24 bit olduqda, ən azı 394 Mbayt video yaddaş, şinin buraxma qabiliyyəti isə bir saniyədə 30 kadr olmalıdır. Və bu yalnız təsvir üçündür, bundan əlavə eşitməni, lamisəni "aldatmaq" üçün bundan da az olmayan resurslar tələb olunur. Bundan başqa virtual görüntünü yaratmaq üçün çoxlu sayda problemlər mövcuddur:

• Şəkli birbaşa tor qişasına vermək lazımdır. Bunun üçün nəinki kəllənin, həm də göz bəbəyinin (ola bilsin ki, göz büllurunun fokus nöqtəsində) yerdəyişməsini izləmək lazımdır. Belə mexanizm sadəcə stereoskopik yox, əsl fəza təsviri yaratmağa imkan verir.

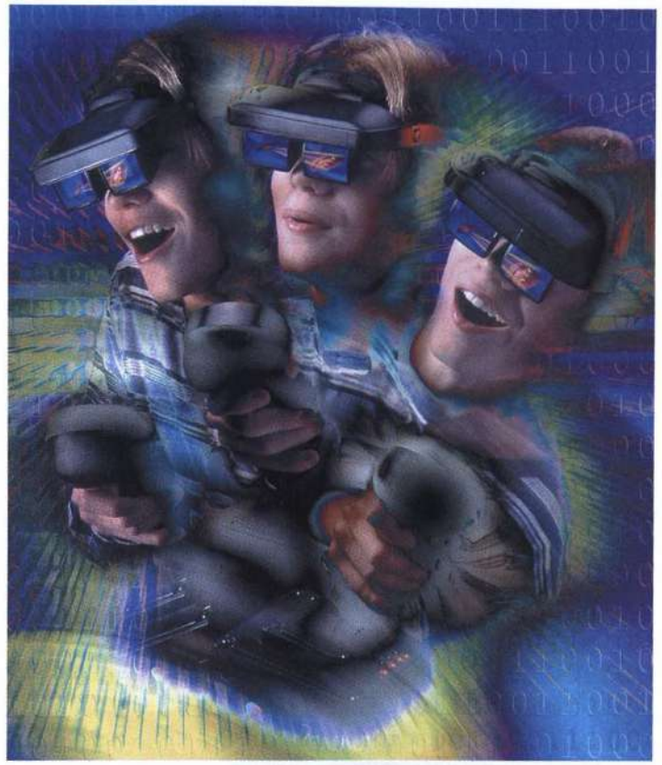
• Qurulmalı olan təsvir yarımsferik, özü də dəyişən ayırdetmə qabiliyyətinə malik olmalıdır.

• 24 bitlik rəng ilə ulduzların ölgün və günəşin parlaq rəngini modelləşdirmək olmaz.

• 3 ölçülü modeli yadda saxlayıb emal etmək üçün böyük operativ yaddaş və böyük hesablama gücünə malik prosessor tələb olunur.

Virtual dünyanın təsvirlərinin ən sadə ötürülmə qaydası növbə ilə sağ və sol göz üçün təsvirləri formalaşdırmaqdan ibarətdir. Bunun üçün ekranda lazımi şəkil meydana gələndə onu elə bir qurğu ilə sinxronlaşdırmaq lazımdır ki, o, növbə ilə gah bir, gah da digər gözü örtmüş olsun. Tezliklər gözlərin yorulmaması üçün 85-100 Hs, streotəsvirlər üçün isə bundan iki dəfə çox olmalıdır (150-300 Hs).







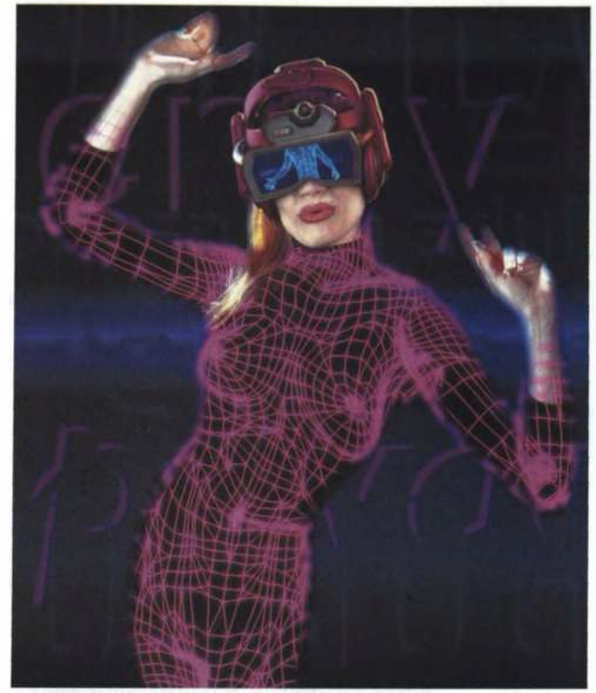
Digər üsul əvvəlkindən də bahadır. Bu üsulun əsas məqsədi rəngli matrislərdə yaxud kiçik monitorda, dəbilqədə yaxud virtual reallığın eynəklərində hər bir göz üçün görüntülərin qurulmasıdır. Hal-hazırda lamisənin imitasiyası kifayət qədər uğurla inkişaf etdirilir. Xüsusi siçan, kursoru düymənin üzərində sürüşdürməklə onun qabarıqlığını, pəncərənin ölçüsü dəyişdikdə onun müqavimətini və qrafiki redaktorda təsvirin relyefini hiss etməyə imkan verir. Force Feedback (güc vasitəli əks əlaqə) texnologiyasını dəstəkləyən coystiklər yürüş simulyatorunda oyun zamanı yolda olan çala-çuxura müvafiq olaraq "silkələnməyə" və böyük sürətlə döngələri keçdikdə "çətinlik çəkməyə" vadar edir. "Atəş" oyunlarında isə atəş açdıqda "geritəpmə" və ya güllə sizə dəydikdə isə müvafiq "silkələnmə" effekti alacaqsınız. Əgər görməni, səsi, lamisəni heç də pis imitasiya etmirlərsə (halbuki, burada da mükəmməlliyə çatmağa xeyli var), başqa hisslərlə vəziyyət daha qəlizdir. Hələ ki, nə "iy", nə də "dad" kartları yoxdur. İdeal virtual reallığa o zaman çatacağıq ki, birbaşa onurğa və baş beynində olan görmə, iy, lamisə, eşitmə sonluqlarına "qoşulmaq" mümkün olsun.

Buna baxmayaraq, artıq indi virtual reallığın bir çox nailiyyətlərindən geniş istifadə olunur: yeni qurğuların layihələrinin işlənməsi, insan orqanizminin tibbi tədqiqi, mənzillərin interyerinin modelləşdirilməsi və bir-birilə bağlı olan obyektlərin vizual təsvirinin qurulması və s. Bu çox uzaqda deyil: cərrah kabinetin ortasında duraraq, insan üzərində deyil, onun virtual modeli üzərində çox mürəkkəb bir əməliyyatı icra edir. Həkim bədən və toxumaları əla hiss edir. Və bütün işi robot icra edir, onu isə öz növbəsində əks əlaqə qurğusu vasitəsilə kompüter idarə edir. Kompüteri isə öz növbəsində həkim

idarə edir. Təbiidir ki, təbabətdə bu cür qurğular çox dəqiq və real hissləri təmin etməlidirlər.

Cərrah bədənin istənilən hissəsinin təsvirini böyüdə bilər, ona lazımi rakursdan baxa bilər və bıçaq tutan əlinin təsadüfən titrəməsindən qorxmayaraq onunla işləyə bilər. Mikrorobotların iştirakı ilə o, əməliyyatı bilavasitə insanın daxilində, məsələn, ürəyində apara bilər. Bundan əlavə, belə bir əməliyyat həkim ilə xəstə birbirindən uzaq məsafədə olduqda keçirilə bilər, məsələn, İnternet vasitəsilə. Tutaq ki, həkim Azərbaycanda, xəstə isə harada isə Şimal qütbündə ekspedisiyadadır. Təbiidir ki, belə bir fəaliyyət üçün mütləq etibarlı əlaqə gərəkdir. Virtual reallığın nailiyyətlərindən yalnız təbabətdə deyil, insana təsir edən ziyanlı şüalanmanın mövcud olduğu yerlərdə, aqressiv mühitdə, kosmosda və s. istifadə edilə bilər. Bu nailiyyətlərdən trenajorimitatorlarda (məsələn, avtomobillərdə, təyyarələrdə, kosmik aparatlarda) və əlbəttə ki, əyləncə sənayesində müvəffəqiyyətlə istifadə edilir.

Virtual reallıq öz qanunları və obyektləri olan süni yaradılmış bir dünyadır. Uydurulmuş aləmlər insan təxəyyülünün nəticəsi olub, kompüterin köməyilə yaradılırlar. Buna nümunə olaraq, kitabları və kino filmləri göstərə bilərik. Virtual reallıq çoxsaylı fantast yazıçılar və alimlər tərəfindən təsvir edilmişdir. Onların qəhrəmanlarından bəziləri virtual dünyalar yaradan yaradıcılardır, digərləri virtual reallıqlar vasitəsilə paranormal qabiliyyətlər qazanırlar, başqaları isə virtual fəzada yaşayırlar. "Çəməndbiçən" filmində baş qəhrəman inkişafda öz həmyaşıdlarından geri qalır, lakin virtual reallığın köməyilə o, dahinin intellektinə və eyni zamanda manyakın qəddarlığına sahib olur. "Matris" filmində kompüter virtual reallığı elə modelləşdirir ki, seyrçi real və virtual



dünyanın arasında çaşıb qalır. "Qəribə Toma" filmində baş qəhrəman sadəcə olaraq virtual dünyada yaşayır. Öz xəstəliyinin müəyyən xüsusiyyətlərinə görə o, heç vaxt evdən çıxmır və ətraf aləmlə ünsiyyətini yalnız İnternet vasitəsilə qurur. Və görəsən bu gələcəyin modeli deyilmi?

"...Mən təəssüf hissilə sıralanmış kabinələrə baxdım və onlardan birinə hansısa oyunçu daxil olurdu... Mən



yaxında olan bir avtomata yaxınlaşdım ki, kredit kartımı yoxlayım. Avtomat mənim kredit kartımı uddu, işıqları ilə göz vurdu, səs çıxardı, displeyə hesabımdakı pul qalığının miqdarını çıxardı və kartı geri itələdi. Mən düşündüm ki, bu pul haçana qədər mənə çatar və mən nə vaxt yenidən öz hesabımı artıra biləcəm... Ayaqlarım titrəyənə qədər oynamaq istəyirdim... Hesablamalarımı qurtarıb, şübhələrlə məni özünə çəkən kabinələrə bir də baxdım. Dərindən nəfəs alıb, bədbin fikirləri başımdan ataraq, Virtual reallığın kabinələrinə tərəf inamla addımladım... Sifətimdə bəxtiyar bir təbəssüm var idi..."

OYUNLARDAN BİLİKLƏRƏ DOĞRU

Daha çox insanı özünə cəlb etməklə, kompüter oyunları get-gedə özünəməxsus düşkünlüyə çevrilir. Oyunları nəinki uşaqlar, kədərli də olsa, böyüklər də oynayır. Kompüter oyunlarının meydana gəlməsilə alimlər onlara çox vaxt sərf etməyə başladılar. Elmi tədqiqat institutunun elmi əməkdaşlarından biri, iki uşaq atası, işdən sonra evə getməyib, bütün istirahət günlərini fasiləsiz olaraq, gecələr yatmadan, kompüterdə oynamaqla keçirmişdi. Şahidlər danışırlar ki, bazar ertəsi günü qapını açdıqda onların həmkarının üzündə üç günlük saqqal var idi, amma onun gözləri xoşbəxtlikdən parlayırdı.

Əlbəttə, oyunları qadağan etmək olmaz. İndi fərdi kompüterlər geniş yayılmışdır və yəqin, elə bir uşaq tapılmaz ki, kompüter oyunları oynamaq istəməsin.



"Mikroprosessor vasitələri və sistemləri" jurnalının XX əsrin 90-cı illərində keçirdiyi kompüter axşamlarından birində belə bir şüar səsləndi: "Oynaya-oynaya öyrən!". Alimlər bu qənaətə gəldilər ki, trenajorlar, kompüter dərslikləri, müxtəlif fənlər üzrə praktikumlara oyun elementləri əlavə edilsə, onda uşaqlar materialı daha yaxşı qavrayar və yadda saxlayarlar.

Amma o dövrün kompüterləri bu arzuları həyata keçirmək üçün kifayət qədər təchiz olunmamışdır. Bunun üçün xeyli təkmilləşmələr tələb olunurdu. Kompüterlərdə ən azı iki inqilabi dəyişiklik baş verməli idi.

Birinci inqilabi addım multimediyanın (Multimedia) meydana gəlməsi oldu. Bununla bizə səs, musiqi və televiziya təsvirindən pis olmayan təsvir gəldi.

İkinci inqilabi dəyişiklik CD-lərin (Compact Disk) meydana gəlməsilə baş verdi. Bunlar böyük tutuma malik olub, baha olmayan yaddaş qurğuları idi. Bir belə diskə 300 min səhifə mətn (qeyd edək ki, on ədəd belə CD-yə dünyanın bütün riyaziyyatçılarının bir il ərzində aldıqları nəticələri yazmaq olar), 80 min rəqəmsal səs yazısı, minlərlə yüksək keyfiyyətli şəkillər, televiziya filmləri və s. yazıla bilər.

Adi CD mahiyyətcə, yeni bir şey deyildi, axı əvvəllər də məktəb dərslərində televiziyadan istifadə olunurdu, dərsliklər isə rəngli fotoşəkillərlə zənginləşdirilirdi.

Amma hipermətnli ensiklopediyaların kvest – oyunlara (macəralarla müşayiət olunan oyunlar) bənzərliyi var. Onlar nəinki öyrənilən material haqda əraflı sorğu kitabına malikdirlər, həm də oradakı informasiyalar rəngli fotoşəkillərlə və filmlərin fraqmentlərilə tamamlanır və istifadəçiyə imkan verir ki, xüsusi 3 ölçülü eynəklər vasitəsilə həcmli təsvirləri seyr edə bilsinlər. Bunun vasitəsilə şagird oyuna cəlb olunur və bu zaman zoopark və ya muzeydə gəzir, qəzəblənmiş tiranozavrdan xilas olur, zamandan-zamana səyahət edir.

Yoxlama sualları da həmçinin oyun formasında təklif edilir: bu konkret əşyaların axtarışı və ya səslərinə görə heyvanların tapılması ola bilər.

Buna baxmayaraq ensiklopediya elə ensiklopediya olaraq qalır. Kataloqlu kitabxana zalının analoqu predmeti adına görə, məsələn, öküzlər haqqında bütün informasiyanı tapmağa imkan verir. Maraqlı olan materialın fotoşəkillərə, məqalələrə və ya filmlərin fraqmentlərinə görə tapılması da axtarış prosesini asanlaşdırır. Əgər mövzunun öyrənilməsi üçün hər hansı bir əlavə material lazım olsa, şagird dəfələrlə belə ensiklopediyaya müraciət edə bilər.

Lap körpələr üçün canlı kitablar yaradılır ki, onlar da uşaqları maraqlı oyun aləminə aparır. Uşaq hər şeyə siçan vasitəsilə toxunmaqla, maraqlı ölkəyə müəmmalı səyahət etmək imkanı qazanır. Bu zaman predmetlər canlanır, dovşan danışır, televizor yandırılıb-söndürülür və ya düyməni basmaqla bir



proqramdan digərinə keçir, telefonla zəng etmək mümkün olur, rəsm lövhəsi təklif edliir ki, ürəyin istəyən qədər çək və ya heyvanları rənglə, saat isə dil açıb deyə bilir ki, vaxt neçədir... Beləliklə, zəhlə tökmədən öyrənmə prosesi baş verir. Siz saata görə vaxtı təyin edə, hansı musiqi alətinin necə səsləndiyini öyrənə, dovşanla nəzakətlə danışa bilərsiniz və s. Məsələn, "Vinni Pux və hər şey" kimi klassik əsərlər üçün canlı kitablar mövcuddur. Və əlbəttə, bu hekayəyə ən məzəli ayı balası ilə, Dovşan ilə və Puxun bütün dostları ilə bağlı oyunlar daxil edilib.

Kompüter təhsildə böyük irəliləyiş etmək imkanı yaratdı. Materialın unikal təqdim edilmə imkanı və müstəqil oyun məşğələlərinin bolluğu uşaqlara predmeti hərtərəfli qavramağa imkan verir. Lakin əsas odur ki, (daha bəzi teleserial-larda olduğu kimi yox), oyunda Herakl İkinci Dünya müharibəsində iştirak etməsin.

KOMPÜTER





KOMPÜTERƏ QƏDƏRKİ DÖVRÜN TARİXİ

HESABLAMA MAŞININA GEDƏN YOL

Sadə bir sual – ilk kompüter hansı olmuşdur? Bu suala kifayət qədər qəribə cavab almaq olar: "IBM PC XT". Amma kompüterlərin tarixi xeyli zəngindir. İnsan dühası ilə yaradılmış ilk kompüter əlbəttə ki, IBM PC XT deyildir. Bu yalnız IBM firmasının istehsal etdiyi ilk fərdi kompüterdir.

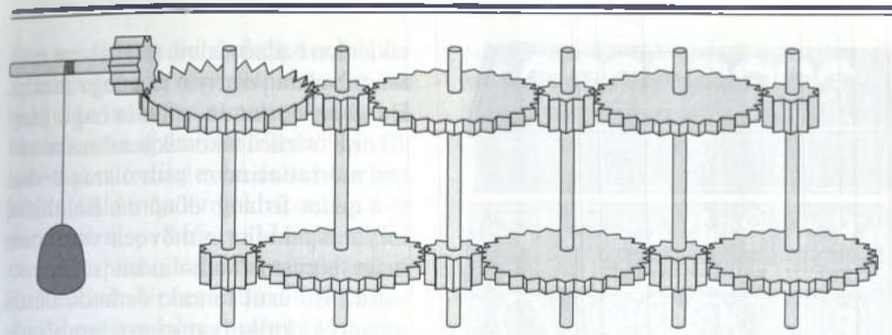


Leonardo da Vinçi.

İlk öncə kompüter (*ing.* computer – "hesablayıcı", "hesaba alan") eynitipli mürəkkəb hesablamaları aparmaq üçün nəzərdə tutulmuşdu. Müasir hesablama texnikasının tarixi yarım əsrdən bir qədər çox olsa da, ilk mexaniki kompüter kimi abak (o, bəzən, adi çubuqlar ilə assosiasiya olunur) çubuqları qəbul edilir. Onun haqqında ilk məlumatlar hələ yeni eradan əvvələ gedib çıxır. Bu "kompüter" yaddaş əvəzinə daşlara malik idi. Bu daşları hərəkət etdirməklə, insan hesablamaları – toplama və çıxmanı həyata keçirirdi. Mərtəbəli toplamanı, məsələn, sütunlu toplama kimi, sayğacda hesablamaq daha rahat olduğu üçün onlar öz hesablama "güclərini" indiyə qədər qoruyub saxlaya biliblər.

XX əsrdə böyük alim, parlaq ixtiraçı və rəngkar olan Leonardo da Vinçinin əlyazmaları tapıldı. Orada mexaniki hesablama maşınından söhbət gedir. Onun iş prinsipi odometrin (avtomobildə gedilmiş yolun uzunluğunu ölçən cihaz) iş prinsipini xatırladır. Əgər kilometrlərin təkərciyini artım istiqamətində fırlatsaq, onda 9-dan sonra 0 gələcək, amma onluqların təkərciyi 1 vahid artacaq, yəni sayğac 10 göstərəcək.

Belə "mürəkkəb" cihazın ilk arifmometrlər ilə ümumi cəhətləri çoxdur. Arifmometrlərdə ədəd 10 dişə malik olan diskin vəziyyətilə (fırlanması) kodlaşdırılırdı.



Leonardo da Vinçinin kalkulyator mexanizmi.

Bu cür əlaqəli disklərin köməyilə cəmləyici qurğu düzəltmək olar.

1642-43-cü illərdə fransız filosofu və riyaziyyatçısı Blez Paskal (1623-1662) ilk mexaniki hesablayıcı qurğunu ixtira etdi. Bu qurğu onluq say sistemində ədədləri toplamağa imkan verirdi.

Mexaniki cəmləyici xüsusi təkərcikli disklərdə ədədləri toplayırdı. Paskalın maşınında beşrəqəmli ədədin onluq rəqəmləri disklərin dönməsi vasitəsilə verilirdi və onlarda ədədi bölgülər var idi. Nəticələr pəncərəciklərdə oxunurdu. Toplama zamanı vahidin növbəti mərtəbəyə keçməsini nəzərə almaq üçün disklərdə uzun diş var idi. Disklər mexaniki qaydada aralıq çarxlar vasitəsilə birbirilə bağlı idilər: yüzlük, onluq və təkliklərin diskləri bir tərəfə, artım tərəfinə fırlanırdı. Dilçək mexanizmi vasitəsilə xüsusi rıçağın müəyyən bucaq qədər kənara çəkilməsi birrəqəmli ədədi "daxil etməyə" və onu kalkulyatorda saxlanılan ədədlə toplamağa imkan verirdi. Belə ötürücü hər bir diskdə var idi və bu çoxrəqəmli ədədləri toplamağa imkan verirdi. Paskalın birinci kalkulyatorunda 5 rəqəm var idi, sonra isə o, bunun sayını 8-ə çatdırdı.

Əgər sayğaclar yalnız informasiyanı yadda saxlayan qurğular idisə, Paskalın cəmləyici maşını mexaniki hesablayıcı idi və o, vahidləri avtomatik olaraq digər onluq mövqeyə keçirə bilirdi.

Mexaniki cəmləyici yalnız toplaya bilirdi və çıxma üçün isə "tamamlayıcının köməyilə çıxma" üsulundan istifadə edilirdi. Məsələn, beşrəqəmli Paskal maşınında 4125 ədədindən 737 ədədini çıxmaq üçün 4125 ədədinə 737 ədədini 100000-ə tamamlayan ədədi əlavə etmək lazım idi (ən kiçik mərtəbəli rəqəmdən başqa bütün rəqəmlər 9-a tamamlama ilə alınır). Bu ədəd 99263 olur.

$$4125 - 737 = 3388,$$
$$4125 + 99263 = 103388.$$

5 mövqeli maşında 6 mövqe olmadığı üçün biz nəticədə 103388 deyil, 3388 alırıq.

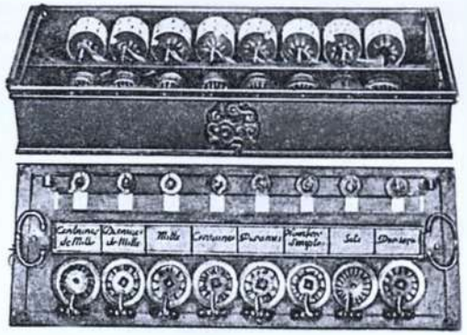
Daha bir maşın alman filosofu, riyaziyyatçısı və fiziki Qotfrid Vilhelm Leybnis (1646-1716) tərəfindən ixtira edilmişdir.

1673-cü ildə ildə Leybnis yalnız ədədləri toplayan deyil, həm də vuran "kompüteri" ixtira etdi. Leybnisin



"Volksvagen" avtomobilinin odometri.

Paskalın hesablayıcı maşını.



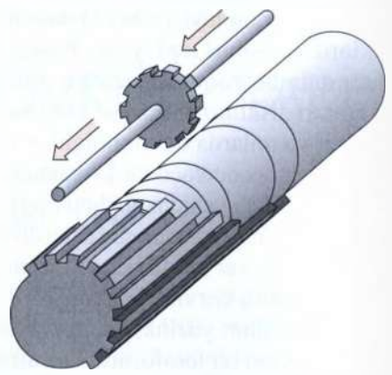
Leybnisin hesablayıcı maşını.



hesablayıcı maşını toplamanı eynilə Paskalın maşınında olduğu kimi icra edirdi, lakin onun konstruksiyasına hərəkət edən hissə (mütəhərrik kiçik karet) və dəstək daxil edilmişdi. Dəstəyin köməyilə xüsusi təkər və ya aparatın daxilində yerləşdirilmiş baraban (sonrakı təkmilləşdirilmiş variantlarda) fırlanırdı.

Maşında hər bir mərtəbə özünəməxsus mexanizmə malik idi. Bu da digər qonşu mərtəbələrin mexanizmlərilə bağlı idi. Leybnis *addımlı barabanlardan* – doqquz müxtəlif uzunluqlu dişciklərdən istifadə edirdi (mərtəbəyə mütənasib olaraq dişciyin uzunluğu artırdı). Baraban dönəndə, onunla bağlı olan 10 dişli ötürücü təkərcik barabana nəzərən mərtəbəsindən asılı olaraq 0-dan 9-a qədər fırlanıb dönürdü. Beləliklə, Leybnis ədədlərin mövqeli vurulması üçün "sürüşmə" əməlindən istifadə etmişdi. Bu üsul sonrakı əsrlərdə bütün mexaniki kalkulyatorların təməlində dayandı.

Əlbəttə, bu günün tələblərilə yanaşsaq bu maşınları əsl kompüter (və hətta kalkulyator belə) adlandırmaq olmaz. Mexaniki kompüterlərin tarixi – ədədi çarxlar və ötürücülərin tarixidir.



▸ Leybnisin hesablayıcı maşınının addım barabanı.

ABAK VƏ SAYĞACLAR

Abak insanın tətbiq etdiyi birinci hesablama alətidir. Abakda hesablama aparmaq üçün xüsusi sahə vardı. Bu sahədə mərtəbələr üzrə qruplaşdırılmış *hesablama elementlərini* hərəkət etdirirdilər. Məhz bu ideya ilk baxışdan fərqli görünən müxtəlif alətləri – yunan və roma abaklarını, çin və rus sayğaclarını, habelə xətlərdə icra olunan hesablamaları birləşdirir. Çox güman ki, əvvəllər abak rolunu (fərz edirlər ki, yunanca bu sözün kökü "toz" deməkdir) üstü toz və ya qumla örtülmüş lövhə yerinə yetirirdi. Tozla örtülmüş lövhə üzərində cizgilər çəkmək və daşların yerini dəyişmək olurdu. Abak hələ Qədim Misirdə və ehtimal ki, Babildə də məlum idi, onu finikiyalı tacirlər Yunanıstana aparıb çıxarmışdılar.

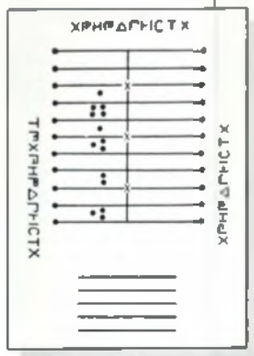
Qədim yunan tarixçisi Herodot (e.ə. 484-425-ci illər) göstərir ki, "ellinlər öz hərflərini soldan sağa, misirlilər isə sağdan sola yazır və sayırlar". Bu abak haqqında ilk yazılı xatırlatmadır (şübhə yoxdur ki, burada söhbət məhz abakdan gedirdi, çünki başqa hesablama üsulları o zaman mövcud deyildi). Amma abakın ilk qədim təsvirini yunan ustalarının e.ə. III əsrdə hazırladıqları güldanda görmək olar. Onun üzərində

fars şahı I Dara və abakda hesablama aparan xəzinədar təsvir olunmuşdur. Bu qurğunun məhz lap əvvəllərdən də pul hesablarının aparılması üçün istifadə olunmasını təsdiqləyən daha bir sübut bizim günlərə qədər gəlib çatan qədim yunan abakı – 846-cı ildə qazıntılar zamanı tapılmış salamin tava daşıdır. Bu, üzərində bir neçə sütun təşkil edən paralel xətlər çəkilmiş mərmər lövhədir. Sol sütunda böyük pul vahidləri – drahmlar və talantlar (yunanlar hesablamaları adlı ədədlər üzərində aparırdı) hesablanırdı. Burada ikilik-beşlik hesablama sistemindən istifadə olunurdu: mərtəbələrin qiyməti 1, 5, 10, 50, 100, 500, 1000, 5000 dirhəm, 1 və 5 talant olurdu. Sağ tərəfdə isə xırda pul vahidləri – xalk və obollar hesablanırdı. Hər sonrakı mərtəbənin qiyməti iki dəfə artıq olurdu: 1, 2, 4 və 8 xalk (bu 1 obola bərabər idi). Mərtəbələrin dəyəri onlarla yanaşı duran yunan hərflərilə işarə edilirdi. Təsadüfi deyil ki, "Tarix" kitabının müəllifi Polibiy (e.ə. II əsr) saray əyanlarını hesab lövhəsindəki daşlara bənzədir. Bu daşların dəyərini (bir xalk yaxud tam talant) hesabdar müəyyənləşdirirdi. Belə lövhə üzərində daşları əlavə etmək, götürmək və mərtəbədən mərtəbəyə sürüşdürməklə, toplama və çıxmanı asanlıqla yerinə yetirmək mümkün idi.

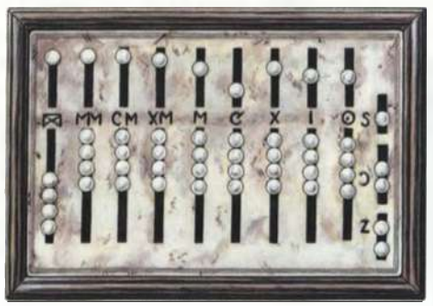
Qədim Romada abakı "calculi" – "daşlar" adlandırsalar da, əslində daşlardan istifadə etmirdilər (buradan da latın sözü olan calculare – "hesablamaq" sözü, bu sözdən isə "kalkulyator" sözü əmələ gəlmişdir). Abak dəyişilərək əsl hesablama alətinə çevrilmişdir. Romalılar onu bürüncdən, rəngli şüşədən və ya fil sümüyündən iki sıra yarığı olan lövhə formasında düzəldirdilər. Bu yarıqlar üzərində daşlar hərəkət etdirilirdi (aşağı sırada dörd, yuxarı sırada isə bir daş var idi). Yuxarı sıranın daşının dəyəri aşağı sıranınkından 5 dəfə çox idi (məsələn, 80 ədədi yuxarı sıranın bir və aşağı sıranın onluqları göstərən üç daşı ilə təsvir olunurdu: 50+3 · 10 kimi). Yarıqların bir neçə sırası 1/12, 1/24, 1/48 və 1/72 kəsrlərinin misillərini təsvir etmək üçün istifadə olunurdu. Roma abakında işləmək yunan abakında işləməkdən daha rahat idi.

Roma imperiyasının süqutundan sonra, yalnız X əsrdə, elm və mədəniyyətin bir neçə əsrlik tənəzzülündən sonra abak yenidən Avropada yayılmağa başladı. Onun dirçəlməsi erkən orta əsrlərin ən parlaq və savadlı adamlarından biri olan Oriaklı Herbertin (940-1003) adı ilə bağlıdır. O, 999-cu ildə Roma papası oldu və II Silvestr adını götürdü. Yada salaq ki, M.A.Bulqakov "Usta və Marqarita" əsərində Patriarx göllərindəki səhnədə Volandın dilindən bunları söyləyir: "Burada, dövlət kitabxanasında onuncu əsrin cadugəri Avrilaklı Herbertin orijinal əlyazmaları tapılıb. Və tələb olunur ki, mən onları araşdırım. Mən dünyada bu sahədə yeganə mütəxəssisəm". Çox güman ki, o, riyazi məzmunlu əlyazmaları nəzərdə tuturdu, çünki Herbertin ən məşhur əsəri "Abakın qaydaları haqda" adlanır.

Herbertin abakı üç-üç qruplaşdırılmış 27 sütundan və kəsrləri təsvir etmək üçün 3 əlavə sütundan ibarət idi. Daş əvəzinə üstünə 1-dən 9-a qədər rəqəm yazılmış jetonlardan istifadə olunurdu (əvvəllər bunlar ərəb rəqəmləri deyildi). Herbert nəinki abakı təkmilləş-



Yunan abakı.



Roma abakı.

Herbertin abakı.

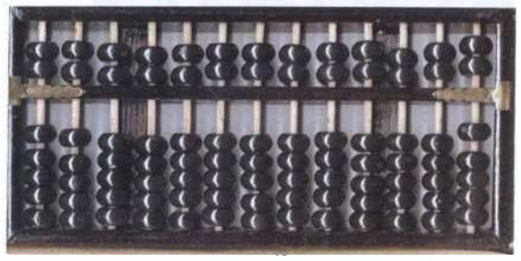


dirdi, həm də mürəkkəb hesablamalar aparmağa imkan verən tam qaydalar sistemi yaratdı.

Abakdan hamı, o cümlədən tacirlər, sərraflar və sənətkarlar da istifadə edirdi. Bir sıra görkəmli alimlər abakçılar (abasistlər) adlanırdılar. Abasistləri Oriaklı Herbertin şərəfinə herbekistlər də adlandırırdılar. Herbertin ixtirası hətta altı əsr keçməsinə baxmayaraq, praktki hesablamalar aparmaq üçün əsas alət olaraq qalırdı. Blez Paskal təsadüfi olaraq yazmırdı: onun hesab maşını "qələm və jetonların köməyilə" yorucu hesablamalar aparmaqdan imtina etməyə gətirib çıxaracaq. Digər hesablama maşınının yaradıcısı Leybnis kağız üzərində hesablamaqdansa, jetonlar vasitəsilə hesablamaları üstün tuturdu. Qərbi Avropada hesab müəllimlərini, hətta məşhur riyaziyyatçı Nikkolo Tartalyunu (1499–1557) da, abak magistrləri adlandırırdılar.

XII–XIII əsrlərdə abak xətlər üzərində hesab formasını aldı. Onda xüsusi cızılmış cədvəllər və jetonlardan istifadə edilirdi. Jetonları həm cizgilər üzərində, həm də onların arasında yerləşdirmək olardı. XVIII əsrə qədər bəzi Avropa ölkələrində xətlər üzərində hesablamalar öz mövqeyini qoruyub saxlayırdı və yalnız sonralar öz yerini kağız üzərində hesablamalara verdi. Abak e.ə. IV əsrə qədər Çində də məlum idi. O zamanlar xüsusi lövhə üzərində yerləşdirilən hesablama çubuqlarından istifadə edilirdi. Tədricən onları rəngbərəng fişkalar əvəz etdi, lövhə isə başqa forma aldı və beləliklə, X əsrdə çin sayğacı olan suan-pan meydana gəldi. Onlarda lövhənin yerini çubuğa düzülmüş daşlar (hər çubuqda yeddi daş olmaqla) və çubuqlara bərkidilmiş çərçivə tutdu. O, bərabər olmayan iki hissəyə bölünmüşdü: birində beş daş (vahidlər), digərində isə iki daş (beşliklər) var idi. XVI əsrdə suan-pan Çindən Yaponiyaya keçdi. Burada onu "soroban" adlandırırdılar. Sorobanın da çərçivəsi iki hissədən ibarət idi, amma onlar bir və dörd daşlardan təşkil olunmuşdu.

Qədim çin sayğacı.



Sayğaclar Yaponiya və Rusiyada təxminən eyni vaxtda meydana gəldiyi üçün belə fikir var ki, guya onlar Rusiyaya Çindən gətirilmişdi. Ancaq bu belə deyil. Rus sayğacları Çin sayğaclarından çox fərqlənirdi. İndi tədqiqatçıların əksəriyyəti güman edir ki, onlar Rusiyada müstəqil şəkildə ixtira olunmuşdu.

Ələlxüsus da ona görə ki, bu ideya boş yerdə meydana gəlməmişdi. Çünki Rusiyada ta qədim zamanlardan "daşlarla hesab" geniş yayılmışdı. Bu isə avropalıların xətlər üzərində hesablamalarına yaxın idi. Jetonlar əvəzinə adətən meyvə çəyirdəklərindən istifadə edilirdi (1634–1636-cı illərdə Rusiyaya səyahət etmiş alman alimi və səyyahı Adam Olearіy qeyd edirdi ki, mirzələr bu məqsədlə öz yanlarında gavalı çəyirdəklərilə dolu olan torbalar gəzdirirdilər).

XVI əsrdə rus sayğaclarının ilkin variantı olan ağacdan düzəldilmiş sayğac meydana gəldi (100 ildən sonra isə "sayğac" sözü meydana çıxdı). Onların quruluşu kifayət qədər mürəkkəb idi – hər sırada on çəyirdək olmaqla, dörd hesablama sahəsi vardır. Moskvada, Tarix muzeyində saxlanan, bizə qədər gəlib çatmış ən qədim sayğac XVII əsrin ortalarına aid edilir.

O, bir-birilə bərkidilmiş iki çərçivədən, daha doğrusu, hər biri üç hesablama sahəsinə bölünmüş iki yeşikdən ibarətdir. Hər yeşikdə on çəyirdəkli sıralardan ibarət olan bir sahə və birdən dördə qədər çəyirdəyi olan sıralardan təşkil olunmuş iki sahə var.

Artıq XVIII əsrin əvvəli üçün rus sayğacları indiki – müasir şəklinə düşdü. Onun sonrakı təkmilləşmələri – ölçülərinin dəyişdirilməsi, məftilin əyriliyi, çərçivənin forması ondan istifadənin rahatlığını təmin etmək üçün aparılmışdı. Sayğacları daha güclü hesablama aləti etmək üçün çox cəhdlər olmuşdu (məsələn, onları müxtəlif ədədi cədvəllərlə birləşdirmək yolu ilə). Sonuncu dəfə belə cəhd 1921-ci ildə olmuşdu. Demək olar ki, 300 il ərzində Rusiyada sayğaclar çox geniş yayılmışdı. Onlara ölümcül zərbəni yalnız ucuz elektron cib kalkulyatorları vurdu.

KİM İDİ BİRİNCİ?

Hesablama maşınının ilk dəfə Paskal tərəfindən ixtira edilməsi üç yüz ildən artıq zaman ərzində mübahisəsiz qəbul edilmişdi. Lakin tarixçilərə Tübingen universitetinin (hal-hazırda bu torpaqlar Almaniyaya məxsusdur) professoru Vilhelm Şikkardın məşhur riyaziyyatçı və astronom İohan Keplerə ünvanladığı məktub məlum idi. Bu məktubda Vilhelm Şikkard hesablamaları avtomatik yerinə yetirən maşın ixtira etməsi haqqında məlumat verirdi. Bunu təsdiq edən başqa dəlillər olmadığı üçün bu məlumata uzun müddət şübhə ilə yanaşmışlar.

O vaxtlar Keplerin bütün əlyazmaları imperator II Yekaterina tərəfindən alınmış və Rusiya Elmlər Akademiyasının arxivində saxlanılırdı. İkinci Dünya müharibəsindən sonra, 1957-ci ildə bu sənədlərin fotonüsxəsini öyrənən tədqiqatçı Frans Hammer (Ştutqardda yerləşən Kepler elmi mərkəzinin direktoru) hansısa mexanizmin çertyojunun eskizini aşkar etdi. Bu mexanizm hesablama maşınına çox oxşayırdı. Məşəqqətli və məqsədyönlü axtarışlardan sonra tədqiqatçı Şikkardın Keplerə ünvanladığı 25 fevral 1624-cü il tarixli məktubunda onun tərəfindən yaradılmış hesablama maşınının xarici görünüşünün ətraflı təsvirini tapdı.

Təsvirdən bu nəticəyə gəlmək olar. Altımərtəbəli onluq hesablayıcı on dişli və bir dişli təkərciklərdən ibarət olub, toplama, çıxma, həmçinin cədvəl vurmasını və bölməsini yerinə yetirmək üçün nəzərdə tutulmuşdur. Şikkarda mərtəbələr arasında keçid problemini həll etmək müyəssər oldu. Maşın onluq və yüzlükləri toplayıb sola keçirə bildi. Çıxma zamanı təkərciklər geriyə fırlandığı üçün, onlıqların ötürülməsi mexanizmi reversiv idi.

Elə həmin məktubunda Şikkard məktubdaşına xəbər verirdi ki, o, mexanik Vilhelm Pfister ilə birgə iki ədəd maşın (bunları o, hesab saatları adlandırır) quraşdırmışdı, lakin onlar böyük yanğın nəticəsində yanmışdılar. Professor yazırdı ki, bu itkidən çox mütəəssir olub və yeni maşını qurmaq üçün isə vaxtının olmamasından şikayətlənirdi (yeri gəlmişkən yanmış maşınlardan biri Keplerə hədiyyə üçün nəzərdə tutulmuşdu). Görünür ki, Şikkard hesablama maşını üzərində işə bir də qayıtmamışdı. O, 1665-ci ildə vəba epidemiyası zamanı vəfat etmişdir.

Elmi ictimaiyyətə Şikkardın hesablama maşınını görmək müyəssər olmasa da, onun mövcudluğu hər halda şübhə altına alınmamalıdır. Birincisi, ona görə ki, Kepleri aldatmaq Şikkardın nəyinə gərək idi (o vaxtlar elmi etikanın tələblərinə çox ciddi riayət olunurdu), ikincisi isə, hesablama maşınının sonralar tapılmış digər çertyojları və mexanik üçün yazılmış təlimatlar onun xeyrinə olan arqumentlərdir. Bundan başqa, XX əsrin 60-cı illərində Tübingen universitetinin alimlərinə bu çertyojlara görə öz böyük həmyerlilərinin maşınının modelini qurmaq müyəssər oldu.

Bundan sonra elə güman edilirdi ki, tarixçilər cəsarətlə hesablama maşınının yeni banisinin adını təqdim edə bilərlər. Lakin 1967-ci ildə Leonardo da Vinçinin əvvəllər məlum olmayan əlyazmaları tapıldı. Bu əsərlərdə on dişli təkərciklər əsasında qurulmuş 13 mərtəbəli cəmləyici qurğunun eskizi var idi. Beləliklə, məlum olur ki, dahi Leonardo hesablama maşınının ideyasını Şikkarddan ən azı 100 il əvvəl düşünmüşdü. Əlyazmanın tarixini dəqiq təyin etmək mümkün olmasa da, məlumdur ki, gündəlik hələ 1492-ci ildə, Amerikanın kəşfindən əvvəl yazılmağa başlanılmışdı.

Hesablama texnikasının inkişafının ilk dövrü hələ axıradək yazılmamışdır. Heç kim bilmir ki, tarixçilər və arxivçilər daha hansı maraqlı faktlar aşkar edəcəkdir.

BLEZ PASKAL

Dördüncü tərtib qəribə əyrilərdən olan "Paskal ilbizi" riyaziyyatda çoxdan məlumdur. Bu əyrinin adında Etyen Paskalın (1588-1651) adı əbədiləşmişdir. Onun ailəsində 19 iyun 1623-cü ildə Blez adlı oğlan dünyaya gəlmişdi. Etyen Paskal hakimlər zümrəsinə mənsub varlı, savadlı adam idi, pis riyaziyyatçı deyildi və hətta məşhur fransız alimi Pyer Ferma ilə məktublaşırdı. Arvadının ölümündən sonra o, bütün ömrünü uşaqların tərbiyəsinə həsr etmişdi. Blez hələ lap erkən yaşlarından dahilik əlamətlərini biruzə verirdi. O, nadir yaddaşa malik idi, dörd yaşında artıq oxumağı və yazmağı bacarırdı. On yaşında isə anlamaq istəmişdi ki, nə üçün zərbə zamanı saxsı boşqab səs verir və nəticədə özünün ilk elmi işini, "Səs haqda traktat"ını yazmışdı. İki ildən sonra bu oğlan uşağı müstəqil olaraq üçbucağın daxili bucaqlarının cəmi haqda teoremi isbat etdi, daha bir il sonra isə onu Parisin ən böyük riyaziyyatçıları cərgəsinə bərabərhüquqlu üzv kimi qəbul etdilər.

1640-cı ildə Etyen Paskal Ruan şəhərində "polis, ədliyyə və maliyyə intendantı" kimi bir vəzifəyə təyin olundu. Bu vəzifə o cümlədən bütün əyalət üzrə vergilərin yığımına nəzarəti də nəzərdə tuturdu. Hesab olunur ki, gənc Blezdə hesablama maşını haqqında ideya onun atasına kömək etmək arzusundan doğmuşdu. Onun atası kağız üzərində və "qələm və jetonların köməyilə" hesablama lövhəsində mürəkkəb hesablamalar aparırdı. Çox güman ki, onu dərin düşüncələrə dahi filosof Rene Dekartın belə bir fikri sövq etmişdi: "bəzi əqli proseslər öz mahiyyətlərinə görə mexaniki proseslərdən fərqlənmir". Sonralar Paskal yazırdı: "Hesab maşınlarının fəaliyyəti onları düşünən varlıqların fəaliyyətinə daha çox oxşayır, nəinki canlıların. Ancaq maşınların iradəsi yoxdur, canlıların isə var". Belə mühakimə yürütmək olar ki, ilk dəfə o süni və insan intellekti problemini qoymuşdu.



Blez Paskal.

1640-cı ilin sonlarında Blez gələcək maşının əsas ideyasını, mərtəbələrin avtomatik sürüşdürülməsi ideyasını formalaşdırdı: "Hər bir təkərcik on hesabi rəqəm qədər hərəkət edərək, özündən sonrakı təkərciyi bir rəqəm sürüşməyə vadar edir". 22 may 1649-cu ildə hesab maşınına görə əldə etdiyi şah imtiyazı (müasir patentin proobrazı idi) məhz belə səslənirdi. Bu imtiyaz Blez Paskalın maşın kəşf etdiyini təsdiq edirdi və ona bu cür maşınları istehsal etmək və satmaq hüququ verirdi.

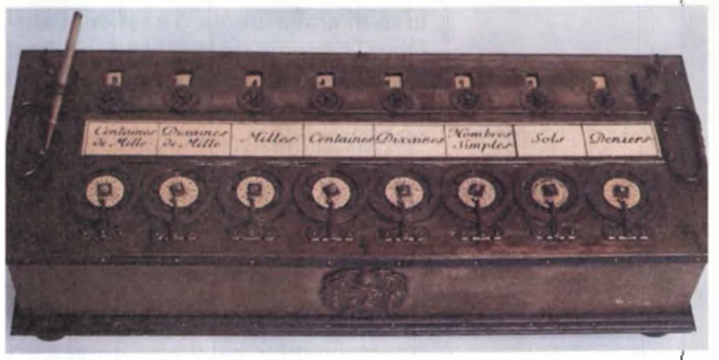
İlk model artıq bir neçə aydan sonra hazır oldu, lakin o işləmədi. Onda Paskal ikinci modelin hazırlanması ilə məşğul oldu. Onun üzərində fasilələrlə 1642-ci ilə qədər işləmişdi. Maşının ilk nümunəsinin bundan üç il sonra hazırlanmasına baxmayaraq, məhz həmin il hesablama maşınının ixtira tarixi hesab edilir.

Paskal sonralar yazırdı ki, o, "nə vaxtını, nə əməyini, nə də vəsaitlərini əsirgəmirdi". Ustalara rəhbərlik edir, tez-tez özü tokar dəzgahında detalları yonur, maşın üçün uyğun gələn materialları – abnos ağacını, misi, fil sümüyünü seçirdi. Paskal müxtəlif konstruksiyaları sınaqdan keçirirdi. Onun maşınları yalnız ölçülərinə görə deyil, həm də mərtəbələrinin sayına görə fərqli idi.

Müəllif öz əsərinin kommersiya uğuruna ümid edirdi. O, əsl reklam kampaniyası açdı. Elan çap etdirdi ki, bütün arzu edənlər onun hesablama maşınından istifadə etsin (faktiki olaraq, dostu, riyaziyyatçı Robervalın evində müəyyən saatlarda açdığı qəbul tarixdə ilk hesablama mərkəzi oldu). Maraqlıdır ki, Paskal elanda məmulatın davamlığını xüsusi qeyd edirdi. O, maşını sərt sınaqdan keçirdi, onu 1100 km-dən artıq məsafəyə karet ilə apardı (başqa sözlə, kompüterlərin testləşdirilməsi ideyası çox böyük tarixə malikdir!).

Paskal öz maşınını krallığın ən tanınmış adamlarının salonlarında nümayiş etdirdi, ümumi baxış üçünsə onu Lüksemburq sarayına qoydu. Hətta maşının bir nüsxəsini o, 1652-ci ildə İsveç kraliçası Xristinaya hədiyyə göndərdi. Amma o, əsl istehsalı təşkil edə bilmədi. 1645-ci ildən 1653-cü ilə qədər cəmi 50 hesablama maşını hazırlamışdı (onlardan səkkizi bizim günlərə qədər gəlib çatmışdır). Onlardan bir neçəsini o hətta sata da bildi. Məsələn, şahzadə Kondeyə o, maşını 100 livrə satdı. Amma maşın ona bu məbləğdən dörd dəfə baha başa gəlmişdi (qeyd edək ki, o biri maşınların maya dəyəri 600 livrə çatırdı). Bu maşınları hesablama məqsədilə deyil, daha çox maraqlı oyuncaq kimi satın alırdılar. Paskal hələ bir müddət də maşını təkmilləşdirməyə davam etdi, hətta kvadrat kök hesablayan maşın da yaratmaq istədi, amma 1653-cü ildən sonra o, bir də bu işə qayıtmadı.

Ola bilər ki, o, ya qarşısına qoyduğu məsələni həll etmiş sayırdı, ya da ki, onun sonrakı perspektivini görmürdü. Cəmiyyət hələ onun ixtirasını qəbul etməyə hazır deyildi. Digər tərəfdən, 1654-cü ilin 15 noyabr tarixində alimin bütün sonrakı həyatını dəyişən hadisə baş verdi. Həmin gün Paskal yalnız möcüzə sayəsində sağ qaldı: onun kareti az qala Sena çayı üzərindəki körpüdən düşəcəkdi. Paskal çox güclü mənəvi böhran keçirdi və tanrının iradəsini hiss edib, yaradan qarşısındakı borcunu xatırladı. 1655-ci ildən o artıq kübar həyatdan uzaqlaşıb, Por-Royaldakı monastıra çəkilərək, yarımmonax həyatı keçirməyə başladı. O, hərdənbir riyazi məşğuliyyətinə qayıtsa da, onun günləri əsasən fəlsəfi və ilahiyyat əsərləri yazmaqla, ehtiyacı olanlara yardımlar etməklə keçirdi. Blez Paskal 19 avqust 1662-ci ildə Parisdə vəfat etdi. Bu zaman onun cəmi 39 yaşı var idi.



Səkkizmövqeli Paskal maşını.

Bəşəriyyət tarixinə Blez Paskal ən böyük dahilərdən biri kimi daxil olmuşdur. O, riyazi analiz və ehtimal nəzəriyyəsini yaradanlardan biri olmuşdur. O, görkəmli fizik, klassik hidrostatikanın yaradıcısı olmaqla bərabər, həm də, öz ixtiralarını həyata keçirməyə can atan gözəl mexanik və ixtiraçı idi. Hidravlik pres ideyası ona məxsus idi. Tarixdə ilk olaraq şəhər nəqliyyatının işini təşkil etməyi o təklif etmişdir. Böyük karetlər (omnibuslar) sərnişinləri müəyyən edilmiş marşrutlar üzrə daşımalı və özü də bu çox ucuz olmalı idi. Hamıya məlum

olan əl arabasını da o kəşf etmişdir! Onun ilk hesablama maşınının yaradıcısı olmasını isə hətta "kibernetikanın atası" Norbert Viner də qeyd etmişdi.

Paskalın yazıçı kimi nəsri də nümunəvi səviyyədə qəbul olunurdu (həmvətənlilərindən biri demişdi: "onun qələmindən süzülən hər bir sətir qiymətli bir daş-qaşdır"). Paskalın "İdeyalar" əsəri özünün dərinliyilə Stendal və Lev Tolstoyu, Bodler və Turgenevi heyran etmişdi...

Tərsimi həndəsədə "Paskal teoremi", riyaziyyatda "Paskal üçbucağı", hidrostatikada "Paskal qanunu" əsrləri adlayıb gəlmiş terminlərdir. Bizim günlərimizdə isə alimin adını təzyiq vahidi kimi qəbul etmişlər və onun adı ən populyar proqramlaşdırma dillərindən birinə verilmişdir.

QOTFRİD VİLHELM LEYBNİS

Leybnis, sən dahi və uzaqgörən kitabların yaradıcısısan!
Sən qədim peyğəmbərlər kimi dünyadan ucada dayanırdın.
Sənin əsrin sənə mat qalsa da, sənin peyğəmbərliyini anlaya bilmirdi
Və yaltaqcasına tərif ilə ağılsız danlaqları biri-birinə qatışdırırdı.

V.Bryusov

Qotfrid Vilhelm Leybnis 1646-cı il iyulun 1-də Leypsiq şəhərində anadan olmuşdur. Onun atası etika professoru, babası isə Leypsiq universitetinin hüquq professoru idi. Uşağın yeddi yaşı tamam olmamış, o, atasını itirir. Qotfrid uşaq yaşlarından çox məşğul olurdu: artıq səkkiz yaşında yunan və latın dillərini müstəqil öyrənmişdi, on beş yaşında isə gimnaziyanı bitirdi.



Qotfrid Vilhelm Leybnis.

1661-ci ildə Leybnis Leypsiq universitetinin hüquq fakultəsinin tələbəsi oldu. O, hüquq elmlərindən başqa həm də Leypsiq, Yena və Altdorf universitetlərində fəlsəfə və riyaziyyatı da öyrənirdi. 1666-cı ildə 20 yaşlı tələbə eyni zamanda iki dissertasiya müdafiə etdi: "Kombinatorika sənəti haqda" (Leypsiq universitetində) və "Dolaşıq işlər haqda" (Altdorf universitetində).

1668-ci ildə o, Mayns şəhərinin kurfürstünün yanında işə başladı. 1672-ci ildən 1676-cı ilədək o, Parisdə diplomatik xidmətdə olmuş və bu ona riyaziyyat və təbiətşünaslıqla məşğul olmağa mane olmamışdı. Burada Leybnis bir sıra alimlərlə tanış oldu. Bunlardan biri də məşhur holland ixtiraçısı və fiziki Xristian Hyüygens (1629-1695) idi. Görəndə ki, Hyüygensə nə qədər çoxlu və mürəkkəb hesablamalar aparmaq lazım gəlir, cavan oğlan onun əməyini mexaniki maşının köməyilə yüngülləşdirməyi qərara aldı. Əvvəl o, böyük Paskalın maşınını təkmilləşdirmək istədi. Buna oxşar hesab alətinin ilk təs-

virini o, hələ 1670-ci ildə vermişdi. İki il sonra o, hesab maşınının yeni təsvirini təklif etdi. Onun əsasında bir nümunə hazırlanıb, 1673-cü ildə Londonun kral cəmiyyətinin iclasında nümayiş etdirildi. Alimin sözlərinə görə o, bütün hesab əməllərini, xüsusilə vurmanı mexaniki yolla tez və etibarlı yerinə yetirmək üçün arifmometr icad etmişdi. Amma etiraf etdi ki, o, mükəmməl deyil və onu yaxşılaşdırmağa söz verdi. 1676-cı ildə Leybnis Londona hesablama maşınının yeni variantını gətirdi, amma bu model praktiki hesablamalar üçün az yararlı oldu. Leybnis dəfələrlə öz arifmometrinə qayıtdı. Təkmilləşdirilmiş maşın yalnız 1694-cü ildə Hannoverdə göründü.

Leybnis öz ixtirası ilə fəxr edirdi. O yazırdı: "Mənə elə bir hesab maşını qurmaq nəsib olub ki, o Paskal maşınından müqayisəolunmaz dərəcədə fərqlənir. Mənim maşınım toplama və çıxmanı ardıcıl icra etmədən, böyük ədədlər üzərində vurma və bölmə əməliyyatını tez yerinə yetirməyə imkan verir". Maşın üzərində iş ona 24 min talerə başa gəldi. Müqayisə üçün qeyd edək ki, nazirin illik əmək haqqısı 2 min talerə bərabər idi. İxtiranın mütərəqqiliyinə baxmayaraq, hesablama maşını geniş yayılmadı. Buna səbəb XVII əsrin sonu və XVIII əsrin əvvəllərində bu cür bahalı və mürəkkəb texnikaya tələbin olmaması idi.

1694-cü ildə Leybnis konstruksiyanın nümunələrindən birini I Pyotra hədiyyə etmək qərarına gəldi. Amma maşın xarab çıxdığından və mexanik onu qısa müddətdə təmir edə bilmədiyindən bu istək baş tutmadı. Başqa bir əfsanəyə görə maşın hansı yollasa I Pyotrun əlinə düşübmüş. O, Çin imperatorunu Avropanın texniki nailiyyətlərilə heyrətləndirmək üçün maşını ona hədiyyə etmişdi.

1676-cı ildə Leybnis Almaniyaya qayıtdı və Hannover hersoqlarının yanında qulluğa girdi. Onların müxtəlif tapşırıqlarını yerinə yetirərək, o, əvvəlcə sarayın kitabxanaçısı, sonra isə hersoqun istorioqrafı və gizli hüquq müşaviri vəzifəsində işlədi.

Lakin Leybnisin fəaliyyəti onun rəsmi vəzifə borclarından xeyli kənara çıxırdı. O, tibbi tədqiqatlar üçün kimyəvi təcrübələr aparmağa da vaxt tapırdı, müxtəlif qurğular da icad edirdi (məsələn, şaxtalardan suyu çəkən nasoslar üçün o, külək mühərriki icad etmişdi). Geologiya, psixologiya, linqvistikaya, xüsusən də fəlsəfə, fizika, mexanika və riyaziyyata önəmli töhvələr vermişdi.

Diqqətəlayiq faktdır ki, I Pyotrun xahişilə o, Rusiyada təhsilin və dövlət idarəçiliyinin inkişafı layihələrini işləyib hazırlamışdı. Leybnis rus çarına yazırdı: "Mən o kəslərə aid deyiləm ki, onlar yalnız öz vətənlərini və ya millətlərini sevirlər, mənim düşüncələrim bütün bəşəriyyətin səadətinə yönəlmişdir... Və mənim üçün ruslara çox yaxşılıq etmək, almanlara az yaxşılıq etməkdən daha xoşdur...".

Bütün həyatı boyu Leybnis aktiv yazışma aparmışdı. Özündən sonra o, minlərlə ünvanlara fransız, alman və

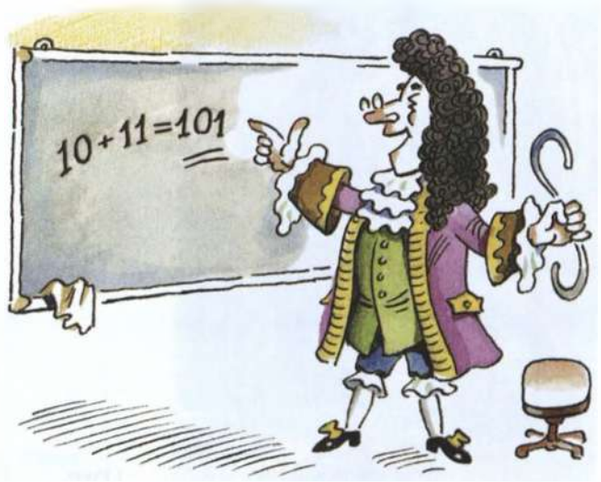


I Pyotr.

İssak Nyuton.



latın dillərində yazdığı 15 mindən çox məktub miras qoymuşdu. Leybnis Nyuton ilə də yazışma aparmışdı. Lakin yazışma yalnız bir il davam etmiş və bu böyük alimlərin əməkdaşlığına gətirib çıxarmamışdı ("Diferensial və inteqral hesabının yaradılması. Nyuton və Leybnis" məqaləsinə, "Uşaqlar üçün ensiklopediya" seriyasının "Riyaziyyat" cildində bax).



Leybnisin müasirlərini onun fantastik erudisiyası, fövqəladə yaddaşı, heyrətamiz iş qabiliyyəti, geniş dünyagörüşü və fəaliyyət diapazonu, istənilən problemdə mahiyyəti qavramaq qabiliyyəti heyrətləndirirdi. Leybnis müasirlərindən heç kimə xas olmayan son dərəcə lakonikliyə və stil dəqiqliyinə, yaradıcılıq enerjisinə və ümumiləşdirmə qabiliyyətinə malik idi.

Ümumiləşdirməyə olan ehtiyac onu universal idrak metodunu axtarmağa sövq edirdi. Leybnisə görə dünya Yaradanın Ağlı ilə yaradılmış və "mümkün olan dünyalardan ən yaxşısıdır" və o, onu Yaradanın dəyişə bilməyəcəyi qanunlarla idarə olunur. Leybnis hesab edirdi ki, dünya insanın dərrakəsilə öyrənilə bilər və nəticədə ümumi "əvvəlcədən müəyyən edilmiş harmoniya" bərqərar olar. Bunun üçün dünyanın vahid idrak metodu lazımdır.

Leybnis məntiqi qeyri-müəyyənliklərlə dolu olan söz aləmindən obyektlər və mühakimələr arasındakı münasibətlərin tamamilə dəqiq təyin olunduğu riyaziyyat aləminə keçidin yolunu göstərdi. O, məntiqdə riyazi simvolikanın tətbiq olunmasını təklif etdi və ilk dəfə məntiqdə ikilik say sisteminin tətbiqinin mümkünlüyü ideyasını söylədi. Bu sonralar avtomatik hesablama maşınlarında istifadə olunmağa başladı.

Qotfrid Leybnisin xidməti həm də ondadır ki, o, elmi praktikada öz təsdiqini tapmış coxlu riyazi terminlər daxil etmişdir: "funksiya", "diferensial", "diferensial hesabı", "diferensial tənlik", "alqoritm", "absis", "ordinat", "koordinat". Bunlardan başqa o, diferensial və inteqralın işarələrini, məntiqi simvolikanı daxil etmişdir.

Leybnis 14 noyabr 1716-cı ildə yoxsulluq içərisində ölmüşdür. Elm aləmində tez-tez olduğu kimi bu böyük alimin xidmətləri də ona sərvət gətirməmişdi.

ÇARLZ BEBBİC

"Mən özümə hesabat verirəm ki, mənim mülahizələrim nə isə həddən ziyadə utopik görünə bilər və onlar hafizələrdə Laput filosoflarını yada salarlar..." – deyən böyük ingilis alimi Çarlz Bebbic kədərlə etiraf edirdi.

(*Laputa Conatan Sviftin uydurduğu uçan ada idi. Orada saxta elmi mülahizələr irəli sürən və real həyatdan tamamilə ayrılmış müdriklər yaşayırdılar*).

Çarlz Bebbic 26 dekabr 1791-ci il tarixdə İngiltərənin cənub qərbində, Totnes qraflığının kiçik Devonşir şəhərində "Pred, Makvord və Bebbic" bank firmasının sahiblərindən biri olan Benjamin Bebbicin ailəsində anadan olmuşdu. Atası ona ölümündən sonra böyük miras qoymuşdu. Uşaqlıqdan Bebbic müxtəlif mexanizmlərlə maraqlanırdı. Yeni bir oyuncağı hədiyyə alanda o, mütləq soruşardı: "Onun içində nə var?". Əgər cavab onu qane etmirdisə, o, hökmən oyuncağı söküb, hissələrə parçalayardı. Zəif səhhətinə görə o, 11 yaşına kimi evdə anası Yelizaveta Tipin himayəsində oxumuşdu. Sonra isə onu İngiltərənin ən yaxşı özəl məktəblərindən birinə qoymuşdular. Buranın zəngin kitabxanası onu heyran etmişdi. Başqa kitablarla yanaşı orada riyaziyyata da aid gözəl kitablar var idi...

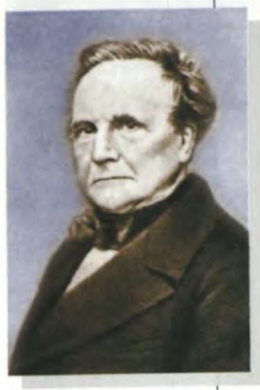
Çarlz Bebbic 1810-cu ildə Kembric universitetinin Triniti kollecinə daxil oldu və ciddi riyazi istedad nümayiş etdirdi. Burada aydın oldu ki, Bebbic riyaziyyatı həmyaşıdlarından daha yaxşı bilir. Az sonra Çarlz biliyinə görə öz müəllimlərini də ötüb keçdi. O belə nəticəyə gəldi ki, Britaniya kontinental ölkələrdən öz riyazi hazırlıq səviyyəsinə görə xeyli geri qalır. Bu geriliyi aradan qaldırmaq üçün o, öz dostları Con Herşel, Corc Pikok və digər gənc riyaziyyatçılarla birgə Analitik cəmiyyət təsis etdilər. Bu cəmiyyət aktiv fəaliyyətlə əvvəl Kembricdə, sonra isə digər universitetlərdə faktiki olaraq riyazi təhsilin yenidən qurulmasına səbəb oldu. Bu başlanğıc yaxşı nəticələrə gətirdi. Ona görə Bebbicin ölümündən sonra da İngiltərədə yeni riyazi məktəbin formalaşması onun adı ilə bağlanırdı.

1813-cü ildə "Analitik cəmiyyətin əsərlərində" Bebbic özünün ilk elmi "Sonsuz hasillər haqqında" məqaləsini çap etdirdi. Elə həmin ildə o, Triniti kolecindən Müqəddəs Pyotr kollecinə keçdi və 1814-cü ildə universiteti bitirib, bakalavr dərəcəsini aldı. Onun ikinci – "Funksional hesabının xülasəsi" əsəri ümumi şəkilli funksional tənliklərin öyrənilməsinə həsr edilmişdi.

O, bu əsərini London kral cəmiyyətinin iki iclasında (15 iyun 1815-ci il və 14 mart 1816-cı il) məruzə etdi və 1816-cı ildə Bebbici həmin cəmiyyətə üzv seçdilər.

1814-cü ildə universiteti bitirdikdən sonra Çarlz Bebbic asudə centlmen-filosof həyatı sürməyə başladı. Böyük Britaniyanın kübar cəmiyyəti onu qəbul edirdi, o Çarlz Dikkenslə yaxından dostluq edirdi. 1815-ci ildə o, Şropşir nəslindən olan Corciya Vitmur ilə evləndi. Bu nigahdan onların səkkiz uşağı doğuldu, təəssüf ki, bunlardan beşi uşaq ikən vəfat etdi.





Çarlz Bebbic.

Ç.Bebbicin yaxın dostları arasında tanınmış yazıçı Çarlz Dikkens (1812-1870) də var idi.



1817-1820-ci illərdə o, riyaziyyat sahəsində daha bir neçə məqalə çap etdirdi ki, onların da əksəriyyəti funksional analiz sahəsində idi. XX əsrin 60-cı illərində Britaniya muzeyində Bebbicin çap olunmamış daha bir əsəri – "Analizin fəlsəfəsi" tapıldı. Bu əsər 1821-ci ildə yazılmışdı.

Sonralar Bebbic riyaziyyatla az məşğul olmuşdu və onun işləri, bir qayda olaraq, tətbiqi xarakterli məsələlərlə bağlı idi.

1818-ci ildə Bebbic Plimutda "hava zəngində" su altına dalmada iştirak etdi. Sonralar, 1826-cı ildə özünün "Sualtı zəng" məqaləsində böyük olmayan sualtı qayığın çertyojunu vermişdi.

Bir dəfə "Don Juan" operasını dinləyərkən darıxdı və bir neçə dəqiqədən sonra zaldan çıxıb, səhnənin idarə edilməsi sisteminə baxmağa getdi.

Başqa bir əhvalat da maraqlı idi. Alfred Tennisonun "Günahın görünməsi" poemasının bir fraqmenti:

Hər dəqiqə insan ölür,
Ancaq hər dəqiqə
yeni insan da doğulur

Bebbici müəllifə belə bir məzmunda məktub yazmağa vadar etmişdi: "Sizə bildirməliyəm ki, bu hesablamalarınızda dünya əhalisinin sayının daim tarazlıq halında olması fikri əsas götürülmüşdür. Amma bir fakt yaxşı məlumdur ki, dünya əhalisi daim artır. Buna görə də mən Sizə məsləhət görməyə məcburam ki, Sizin gözəl poemanızın növbəti



nəşrində hesablama səhvi aşağıdakı kimi düzəldilsin:

Hər dəqiqə insan ölür,
Lakin hər dəqiqə
bir tam yüzdə on altı insan doğulur.

Əlavə edim ki, dəqiq ədəd 1,167-dir, amma qafiyə qaydalarına görə bir tam yüzdə on altı kimi yazmaq məqbul hesab edilə bilər".

Bebbic dəfələrlə Fransada olmuşdu. Orada məşhur astronom və fizik Pyer Laplas, fizik və riyaziyyatçı Jan Batist Furye ilə tanış olmuşdu. Lakin Bebbici daha çox təsirləndirən az tanınmış fransız alimi – riyaziyyatçı Q.Proni olmuşdu. Onun qəmli tarixçəsi XIX əsrin sonlarından başlamışdı. O vaxt imperator Napoleon radikal ictimai islahatlar aparmaq məqsədilə Proniyə (o zaman o, siyahıya alma bürosuna başçılıq edirdi) metrik ölçü sisteminə keçmək məqsədilə yeni loqarifmik və triqonometrik cədvəllərin yaradılmasını tapşırmışdı.

Proni icraçıları üç ixtisas qrupuna bölərək, əməyin bölgüsü ideyasını hesablama prosesinə keçirdi. Proninin xidməti ondan ibarət oldu ki, mürəkkəb hesablamaları sadə hesablamalara gətirən alqoritmik yanaşmanı kəşf etdi. Bu sadə hesablamalar icraçılardan yaradıcı yanaşmanı tələb etmirdi. Lakin işlər yekunlaşandan sonra, 17 cilddən ibarət olan bu əlyazmaları çap etmək üçün Fransada vəsait tapılmadı. Böyük zəhmət hesabına başa gəlmiş iş lazımsız bir şey oldu.

Bebbic Proninin işlərinin acınacaqlı vəziyyətini görmüşdü, lakin bu ona əsas ideyanı duymağa mane olmadı. Bu ideyanın əsas qayəsi mürəkkəb hesablamaların mexaniki yolla yerinə yetirilməsi mümkünlüyü idi. Alimin bu ideyası onu azsavadlı icraçıları mexaniki qurğularla əvəz etmək fikrinə gətirib çıxardı. Bu ideya həm də onu ilk diferensial maşın yaratmağa sövq etdi.

1822-ci ildə Bebbic fərq maşınının təsvirini yazıb qurtardı. Bu maşın 18 rə-

qəmə qədər dəqiqliklə hesablama aparmağa imkan verirdi. Maşının çertyoju baş nazirin masasının üstünə qoyuldu. 1823-cü ildə onun qurulması üçün ilk maliyyə yardımı ayrıldı. Qurma işi on il davam etdi. Bu illər ərzində maşının konstruksiyası daha mürəkkəbləşdi, yaradıcılıq uğurları Bebbic və Cozef Klement (işlərə rəhbərlik etmək üçün təyin olunmuş istedadlı mühəndis və dövlət məmuru idi) arasında baş verən qızğın toqquşmalarla növbələşirdi. Fərq maşınının birinci versiyasının hansı səbəbdən sona çatdırılmamasını indi söyləmək çətindir. Görkəmli ingilis alimləri Bebbicə qarşı çıxış edərək, belə bir maşının qurulmasının mümkünlüyü ideyasını rədd edirdilər. 1833-cü ildə işin maliyyələşdirilməsi dayandırıldı, tamamlanmamış maşın isə dövlət mülkiyyətinə keçdi və sonradan lazımsız şey kimi kənara atıldı. Ola bilər ki, uğursuzluğa səbəb Bebbicin hərtərəfli və dağınıq fikrə malik olması idi. O gah, dəmir yolları ilə maraqlanır (o, bir neçə mühüm, o cümlədən paravoz üçün spidometr ixtira etmişdi), gah da öz maşınının komponentlərinin kütləvi istehsalı texnologiyası ilə maraqlanırdı.

Bu və ya başqa səbəbdən ilk fərq maşını ilə iş yarımçıq qaldı. Lakin bu maşın müəllifi yeni ixtiralara sövq etdi.

Sonralar Bebbic ikinci fərq maşınını qurmaq cəhdində bulunsa da, təşkilati problemlər onun arzusunun bu dəfə də gerçəkləşməsinə mane oldu.

1834-cü ildə alim universal hesablama maşını yaratmaq qərarına gəldi və bunu analitik maşın adlandırdı. Bu layihədə o, ilk dəfə hesablama prosesini proqramın köməyilə idarə etmək fikrinə gəldi. O, elə bir mexaniki qurğu düzəltmək istəyirdi ki, bu qurğu yalnız hesablamaqla kifayətlənməsin, həm də tutulmuş proqramdan və hesablamaların aralıq nəticələrindən asılı olaraq öz işini idarə edə bilsin. Bu kəşf öz dövrünü 100 il qabaqlamışdı! Bu cür maşının imkanları alimin özünü də heyrətləndirdi. 1835-ci ilin mayında alim yazırdı: "Alti ay ərzində mən birinci maşından fərqlənən maşının layihəsini tərtib etdim. Onun malik olacağı hesablama gücünün qarşısında mən özüm də heyrət edirəm və hələ bir il əvvəl mən buna inana bilməzdim". 1840-cı ildə Bebbic öz maşını haqqında mühazirələr oxumaq üçün Türinə (İtaliya) dəvət olunur. Mühazirələr böyük uğur qazanır və dinləyicilərdən biri, gənc mühəndis Luici Menabrea onların konspektini Fransada nəşr etdirir.



Baron de Proni.

Bebbicin 80-dən artıq elmi əsər yazmasına baxmayaraq, fərq və analitik maşınların iş prinsiplərinin ətraflı təsviri onun qələminin məhsulu deyildir. O, maşınların yaradılması ilə o qədər məşğul idi ki, onların təsvirinə vaxtı qalmırdı. Fərq maşını Lardner, analitik maşın isə Luici Menabrea tərəfindən təsvir olunmuşdur.

Əfsus ki, analitik maşın qurulmadı. 1851-ci ildə Bebbic yazırdı: "Analitik maşınla bağlı olan bütün işləmələr öz hesabıma icra olunmuşdu. Mən bir sıra təcrübələr apardım və elə bir həddə çatdım ki, bundan o yana imkanlarım tükəndi. Buna görə də məcburam ki, sonrakı işdən imtina edim".

O, analitik maşının qurulması naminə vəsait əldə etmək üçün istənilən maceraya hazır idi.

Bebbic ilk layihəni xanım Ada Lavleys ilə birgə hazırladı. Bu xanımı hesablama maşını ideyası ilə "xəstələndirmişdi". Bebbic pul qazanmaq məqsədilə üç cildlik roman yazmaq fikrinə düşsə də, bu fikirdən yan keçdi. Daha sonra yeni bir layihəyə uydu – xaç və sıfır oyunu üçün avtomatın qurulması. Lakin bu ideya da həyata keçmədi. Bebbic öz analitik maşını üzərində həyatının son günlərinə qədər işləmişdi. Alim 1871-ci ildə 80 illiyinə iki ay qalmış vəfat etdi.



Ada Avqusta Lavleys.

Çarlz Bebbic yalnız riyaziyyatçı deyil, həm də filosof, iqtisadçı və hətta siyasi



Henri Provost Bebbic.

iqtisadçı idi. Hələ 1832-ci ildə o, "Maşınların və istehsalın iqtisadiyyatı" kitabını yazmışdı (o, öz elmi və texniki məqsədləri üçün bütün Avropanı gəzib dolaşmışdı).

1872-ci ildə Britaniyanın elmə yardım assosiasiyasının xüsusi komitəsi qeyd edirdi: "Biz hesab edirik ki, bu cür cihazların olması adi hesabi əməlləri yerinə yetirərkən əməyə qənaət etməyə imkan verməklə yanaşı, həm də insanın imkanlarının son həddinə yaxın, gələcəkdə praktik cəhətdən mümkün olan, çox şeyləri etməyə imkan verəcəkdir".

İxtiraçının oğlu general Henri Bebbic (1824-1918) 1874-cü ildə istefaya çıxaraq bir neçə il atasının mirasının öyrənilməsinə sərf etdi və 1880-ci ildə atasının çetryojlarına əsasən analitik maşının mərkəzi qovşağını qurmağa başladı. 1888-ci ildə o, π ədədinin 1-dən 32-yə qədər natural ədədlərə hasilini 29 rəqəm dəqiqliyilə hesabladı!

Henri Bebbic 1896-cı ilə qədər atasının maşını üzərində işi uğurla davam etdirmişdi. Onillik fasilədən sonra iş davam etdirilmiş və hesablamanın nəticələrini çap etməyə imkan verən analitik maşının nümunəsi araya gəlmişdi.

Bebbicin maşını işlək maşın oldusa da, ixtiraçı bunu görə bilmədi.

BEBBİCİN MAŞINLARI

FƏRQ MAŞINI

XIX əsrin əvvəlində fransız alimləri maraqlı hesablama üsulu təklif etdilər. Məsələ kiçik hissələrə bölünürdü. Hər bir kiçik hissə yalnız sadə əməllərdən ibarət idi və bunlar da riyaziyyatda yalnız hesabi əməllərini bilən və onları demək olar ki, mexaniki icra edən çoxsaylı adamlara həvalə edilirdi. Kembric universitetinin dekanı Çarlz Bebbic (ona bir vaxtlar İsaak Nyuton başçılıq etmişdi) anladı ki, bu monoton riyazi hesablamalar təkrar olunan əməliyyatlar yığımından ibarətdir. Bu əməliyyatlar əsasən müxtəlif riyazi cədvəllərin tərtib edilməsində istifadə olunur. Bebbic cədvəllərdə baş verən hesablama və çap səhvlərini aradan qaldırmaq üçün bu cür əməliyyatları maşında aparmaq qərarına gəldi. O, avtomatik mexaniki hesablama maşınını layihələndirməyə başladı. Bu maşının iş prinsipi n-1 dərəcəli çoxhədlilərin belə bir xassəsinə əsaslanırdı: n tərtibli sonlu fərqlər sıfra bərabərdir. Sonlu fərqlərin bu xassəsini aşağıdakı misal üzərində göstərmək olar. Fərz edək ki, ikinci tərtibli çoxhədlini hesablamaq lazımdır:

$$y = x^2 + 3x - 4.$$

Bunun üçün beş sütunlu cədvəl qurulur. Birinci sütunda arqumentin, ikincidə çoxhədlinin, qalanlarda isə sonlu fərqlərin qiymətləri yerləşir:

$\Delta^1_x = Y_x - Y_{x-1}$ (birinci fərq),
$\Delta^2_x = \Delta^1_x - \Delta^1_{x-1}$ (ikinci fərq),
$\Delta^3_x = \Delta^2_x - \Delta^2_{x-1}$ (üçüncü fərq).

X	Y	Δ^1	Δ^2	Δ^3
0	-4	4	2	0
1	0	6	2	0
2	6	8	2	0
3	14	10	2	0
4	24	12	2	
5	34	14		
6	50			
7	**66**			

Birinci fərqi almaq üçün funksiyanın qiymətindən onun əvvəlki qiyməti çıxılır. Belə ki, birinci sətirdə

$$\Delta^1_1 = Y_1 - Y_0 = 0 - (-4) = 4.$$

İkinci sətir üçün

$$\Delta^1_2 = Y_2 - Y_1 = 6 - 0 = 6.$$

İkinci fərqi almaq üçün birinci fərqdən əvvəlki fərqin qiyməti çıxılır. Onda birinci sətir üçün alırıq ki,

$$\Delta^2_1 = \Delta^1_1 - \Delta^1_0 = 6 - 4 = 2$$

Eyni prinsip ilə də üçüncü fərqi qururuq. Cədvəldən göründüyü kimi, üçüncü fərqlər sıfıra bərabərdir, ikinci fərqlər isə sabitdir. Əgər fərqlərin qiymətlərini və funksiyanın əvvəlki qiymətini toplasaq, onda çoxhədlinin növbəti qiymətini almaq olar. $x = 7$ olduqda

$$Y_7 = Y_6 + \Delta^1_5 + \Delta^2_4 + \Delta^3_3.$$

Beləliklə, bütün üçüncü tərtib fərqlər sıfıra bərabər olduğu üçün ($\Delta^3 = 0$):

$$Y_7 = Y_6 + \Delta^1_5 + \Delta^2_4 = 50 + 14 + 2 = 66.$$

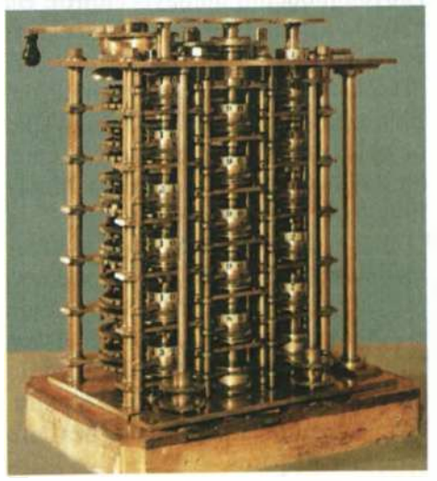
Yəni funksiyanın yeni qiymətini onun əvvəlki hesablanmış qiymətlərini toplamaqla almaq olar.

Cədvəllərin tərtib edilmə prosesini avtomatlaşdırmaq üçün qurulan maşını Bebbic fərq maşını adlandırdı. O, yumor hissilə qurduğu maşının iş prinsipi haqda suala altı işarənin köməyilə cavab verirdi: $\Delta^n U_x = 0$. Amma bu cavabı nadir adamlar başa düşürdü (bu ifadə onu göstərir ki, n-1 dərəcəli U_x çoxhədlisi üçün n tərtibli fərqlər sıfıra bərabərdir). Fərq maşınının əsas təyinatını Bebbic cədvəllərin tərtib edilməsində görürdü, amma o, həm də artıq mövcud olan cədvəlləri də yoxlaya bilirdi. Bu məqsədlə əməliyyatlar əks ardıcıllıqla – çoxhədlinin qiymətlərindən sonlu fərqlərə doğru icra edilməli idi. Əgər çoxhədlinin qiymətini hesablayarkən səhv buraxılırdısa, onda yüksək tərtibli fərqlərin sabit qiymətləri əvəzinə müxtəlif ədədlər alınırdı. Lakin Bebbic öz şərhlərində göstərirdi ki, bütün səhifənin təzədən hesablanması və nəticələrin yoxlanılması bəzən əks hesablamalar aparmaqdan daha sadədir.

Bebbicin maşınının ən vacib elementi onluq ədədləri yadda saxlayan qurğu – *registr* idi. 18 mərtəbəli ədədləri emal etmək nəzərdə tutulduğundan, bir ədədin registri 18 dişli çarxdan ibarət idi. Dərəcəsi n olan çoxhədlini hesablamaq üçün $n + 1$ registr lazım idi (bir registr funksiyanın əvvəlki qiymətini və n registr isə sonlu fərqləri yadda saxlamaq üçün). Maşın alti dərəcəli çoxhədlini hesablamaq üçün nəzərdə tutulduğundan, onda yeddi 18 mərtəbəli registr var idi.

Maşında əsas əməl toplama olduğu üçün, təbii ki, cəmləyici, daha dəqiq desək, hər bir registr ilə birləşdirilmiş qurğular da var idi. Konstruktiv olaraq maşın üç sıra dişli çarxlardan ibarət idi: birinci sıra – registrlərin dişli çarxları, ikinci sıra – cəmləməni yerinə yetirmək üçün oxlara bərkidilmiş dişli hesablama təkərləri və üçüncü sıra – növbəti toplama əməlini yerinə yetirmək üçün *istiqamətverici dişləri* olan oxlar idi. Bebbic onluqların sürüşdürülməsi əməliyyatını iki hissəyə ayırdı. Sonralar bu bütün mexaniki kalkulyatorlarda istifadə olundu. Maşının mexaniki blokunun ölçüləri belə idi: uzunluğu – 3 m, eni – 1,5 m; dişli hesablama təkərinin diametri isə təxminən 13 sm idi.

Maşında hesablamalara paralel olaraq nəticələri çap edən qurğudan istifadə etmək nəzərdə tutulmuşdu.



Çarlz Bebbicin fərq maşını.

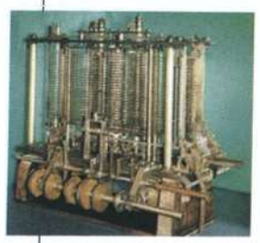


Bebbicin fərq maşını, əvvəlki bütün layihələrdən fərqli olaraq, əvvəlcədən verilmiş əməli icra etməklə yanaşı (məsələn, çoxhədliləri sonlu fərq üsulu ilə hesablamaq), həm də xüsusi seçilmiş düsturlar əsasında müəyyən funksiyaların qiymətini hesablaya bilirdi.

Fərq maşınını qurmaq böyük problem idi: yalnız yeni konstruksiya və qovşaqlar qurulmamalı, həm də onların hazırlanması üçün alətlər də lazım idi. Metalın emal dəqiqliyi təxmini idi və həm də lazımi səviyyədə ixtisaslı ustalar çatışmırdı.

1822-ci ildə Bebbic "Astronomik və riyazi cədvəllərin hesablanmasında maşınların tətbiqi haqda qeydlər" məqaləsini çap etdirdi. Burada o, riyazi funksiyaların hesablanması və alınmış nəticələrin çap edilməsi üçün maşının təsvirini verdi və elə həmin il də onun işləyən modelini nümayiş etdirdi. Bu maşının üç ədəd beş mərtəbəli registri var idi. O, iki dərəcəli çoxhədliləri hesablamaq üçün nəzərdə tutulmuşdu. Bu maşında Bebbic kvadratlar cədvəlini hesabladı. London kral cəmiyyəti onun ixtirasını yüksək qiymətləndirdi. Müəllif bundan çox ruhlanmışdı.

Çarlz Bebbicin analitik maşını.



1823-cü ildə Bebbic Britaniya hökumətinin maliyyə dəstəyini alıb, əsl fərq maşınını qurmağa başladı. O, hesab edirdi ki, maşının qurulmasına üç il vaxt sərf olunacaq. Lakin bu işi heç on ilə də başa çatdırmaq mümkün olmadı. Maşın nəzərdə tutulduğundan beş dəfə artıq vəsait apardı.

1833-cü ildə maşının qurulmuş hissələrinin sınağı göstərdi ki, hesablamalar düzgün aparılmışdı. Maşın əməlləri planlaşdırılmış dəqiqlik və sürətlə yerinə yetirirdi. Lakin hökumət bu sahədə pioner olan alimə maliyyə yardımından imtina etdi. Başa çatdırılmamış fərq maşını və onun bütün çertyojlarını 1843-cü ildə Londonda Kral kollecinin muzeyinə qoydular. İndi Kembricdə saxlanılan model məhz bu maşının hissələrindən düzəldilmişdir.

ANALİTİK MAŞIN

1833-cü il sınaqlarından sonra Bebbic prinsipcə yeni maşın yaratmağı planlaşdırdı. Bu maşın proqramın köməyilə, yəni əvvəlcədən qurulmuş plan əsasında müxtəlif əməlləri yerinə yetirməli idi. İlk çertyojlar 1834-cü ildə meydana gəldi və onu analitik maşın adlandırdılar.

Bebbic fərq maşınının quruluşunu bir qədər dəyişdirmək qərarına gəldi: çevrə üzrə oxları elə yerləşdirməli ki, nəticə sütunu axırıncı fərq sütununun yanında yerləşsin və onunla asanlıqla bağlansın. Bu təkmilləşmə hesablanma ardıcıllığına təsir etməklə, hesablama prosesini idarə etməyə imkan verirdi. Tezliklə alim bütün hesab əməllərini aparmağa imkan verən və bu hesablamaları idarə edən müstəqil qurğunun ideyasını formalaşdırdı.

Analitik maşın xalis mexaniki qurğu kimi düşünülmüşdü. Lakin alim hesablamaları əl ilə deyil, kənar enerji mənbəyinin köməyilə, xüsusi halda, buxar mühərrikini tətbiq etməklə icra etmək istəyirdi.

1842-ci ildən 1848-ci ilə qədər olan dövr yeni maşının yaradılmasına həsr edildi. Müəllif hökumət tərəfindən heç bir maliyyə dəstəyi olmadan öz işini davam etdirirdi. 1849-cu ildə Bebbic analitik maşının sxemini təqdim etdi. Maşın üç əsas blokdan ibarət idi:

• Anbar (Store) – registrlərdə ədədləri yadda saxlamaq üçün mexaniki təkərlərdən ibarət yaddaş. O, 50 rəqəmli min dənə onluq ədədi yadda saxlamaq üçün nəzərdə tutulurdu. Əgər sonrakı hesablamalarda registrdə yadda saxlanan ədəd lazım deyildisə, ədədi çap etmək və azad olan registri başqa hesablamalar üçün istifadə etmək olardı. Əgər hesablamalarda lazım olan bütün ədədləri yadda saxlamaq üçün yaddaş kifayət etmirdisə, onda onları perfokartlara yazırdılar (jakkard kartlarının analoqu) və bu zaman bu kartların miqdarı heç nə ilə məhdudlaşmırdı ("Avtomatlaşdırma və perfokartlar" məqaləsinə bax).

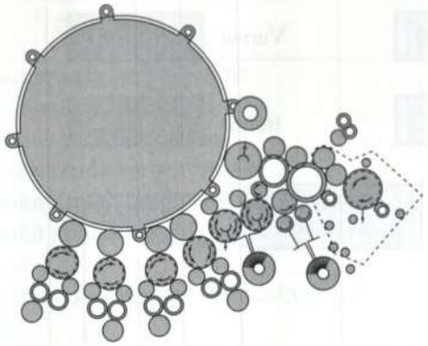
• Fabrik (Mill) – hesab əməliyyatlarını yerinə yetirmək üçün blok. Bebbic öz maşınının məhsuldarlığını belə qiymətləndirirdi: 2 dənə 50 rəqəmli ədədin vurulması və 100 rəqəmli ədədin 50 rəqəmli ədədə bölünməsi dəqiqədə 1 əməliyyat olmaqla, 50 rəqəmli ədədlərin toplanması (və çıxılması) isə saniyədə 1 əməliyyat olmaqla yerinə yetirilirdi.

• Hesablama prosesini idarə etmək, yaddaşdan ədədləri seçmək, hesablamaları yerinə yetirmək və nəticələri çap etmək üçün qurğu (müəllif bu qurğuya ad qoymamışdı).

Maşında istifadə olunan perfokartlar iki qrupa bölünürdü: əməliyyat perfokartları – hesab əməliyyatlarının yerinə yetirilməsi üçün; idarəedici perfokartlar – ədədlərin regisrtlərdən hesab qurğusuna yüklənməsi və geriyə yaddaşa atılması, həmçinin giriş-çıxış əməliyyatlarının yerinə yetirilməsi üçün.

Bebbicin əməkdaşı olan Ada Avqusta Lavleys maşında ədədlərin ötürülməsini idarə etmək üçün istifadə olunan perfokartları dəyişənlər adlandırmağı təklif etdi:

• "sıfırıncı kart" ədədləri registrdən hesab qurğusuna yükləyir və bu zaman registrin məzmunu pozulur, yəni sıfırıncı mövqeyə qoyulur;



Analitik maşının mexanizminin təxmini sxemi.

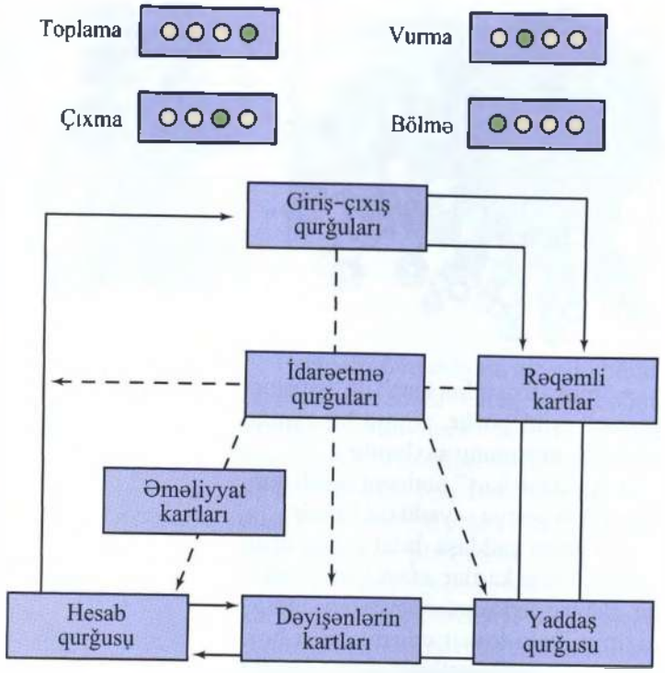
• "mühafizə edən kart" da "sıfırıncı kartın" işini görür, amma bu zaman registrin məzmunu saxlanılır;

• "çatdıran kart" nəticəni hesab qurğusundan geriyə – yaddaşa ötürür.

Ədədləri yaddaşa daxil etmək üçün tətbiq olunan kartlar *ədədi* kartlar adlanır. Bebbic perfokartlardan oxuma mexanizmini belə təsvir edirdi: "Biri boş, digəri isə perfokartlarla dolu olan iki qutu çoxüzlü prizmanın qarşısında və arxasında yerləşib. Bu prizma fasilələrlə fırlanır və hər dəfə kiçik məsafədə irəliləyir, sonra dərhal geri dönür. Kart prizmanın üzərindən yalnız mexanizmin hər gedişindən sonra məkik kimi keçir (Jakkardın dəzgahında olduğu kimi). Keçən kartlar, kartların yığılması üçün olan boş qutuya çatana qədər aşağı düşür və orada bir-birinin üzərinə yığılır".

İxtiraçının oğlu Henri Bebbic analitik maşında belə bir misalın hesablanmasını nümunə göstərirdi: "Əgər $(ab + c)d$ düsturunu nümunə kimi götürsək, onda





Perfokartlarda hesab əməliyyatlarının kodlaşdırılması.

bütün növ kartlardan istifadə edilməlidir. Onların aşağıdakı yüklənmə ardıcıllığından istifadə ediləcək: *a*, *b*, *c* və *d* kimi ilkin ədədlər üçün dörd kart bir yerə yığılacaq və çarxın (və ya prizmanın) üzərinə qoyulacaq. Bu ədədlər maşının anbar adlanan hissəsində sütunlarda (dişli mexanizmə) yerləşdiriləcək. Hər bir alınmış ədəd yadda saxlanılır və istifadə üçün hazır vəziyyətdədir... Üç əməliyyat kartı və on dörd idarəedici kart tətbiq olunur və birgə yığılmış hər bir kart yığımı öz çarxının (və ya prizmanın) üzərində yerləşdirilir...".

Cədvəldə analitik maşında

$$y = (ab + c)d$$

düsturunu hesablamaq üçün əməliyyatlar ardıcıllığı göstərilmişdir.

Ədəd yaddaşın bir registrini tutur (onluq rəqəmli təkərlərdən ibarət sütun). Təkərlərin çevrələri boyunca 0-dan 9-a qədər rəqəmlər yazılıb. Hər bir təkər başqasından asılı olmayaraq fırlanır. İdarəedici kartlar seçilmiş sütunların təkərlərini elə qaldırır ki, onları dişli reykalar ilə ilişdirsin. Bu zaman hər bir təkər ədədin mərtəbəsi qədərində dişlərin sayına bərabər miqdarda fırlanır. Nəticədə perfokartın üzərində göstərilmiş ədəd yaddaşa yazılır. İşarənin yadda saxlanması üçün sütunda olan ən üstdəki təkərdən istifadə olunur. Əsas valın bir dövrəsində ədədi sütunda saxlamaq və ya onu yaddaşdan Anbara ötürmək olar.

Hesab qurğusunda ədədləri toplayarkən bir sütunda ədəd sıfıra kimi azalır və bu kəmiyyət qədər digər sütunu da fırladır. 4 ədədini 8 ədədinin üzərinə gəldikdə nəticə 2 olur və təkərin fırlanması zamanı ardıcıl olaraq 9, 0 və 1 rəqəmləri gəlib keçir. 9 ədədi 0 ədədinə keçdikdə sonra yerinə yetiriləcək sürüşmə rıçaqı yerindən tərpənir.

Bebbicin danılmaz xidməti ondan ibarətdir ki, o, hesablamanın gedişinin proqramla idarə olunmasını daxil etmişdir. Əgər iki ədədi müqayisə edib böyüyünü seçmək lazımdırsa (müqayisə əməliyyatı), onda bir ədəddən digərini çıxmaq lazımdır.

Nəticədə, əgər azalan çıxılandan kiçikdirsə, onda sürüşmə baş verməlidir. Bu zaman rıçaq ən son vəziyyətə gəlir və bu fərqin mənfi olması deməkdir. Rıçaqın bu vəziyyəti iş prosesinə əvvəlcədən hazırlanmış perfokartlar dəstini daxil etməyə imkan verir.

Xanım Lavleys bir və ya bir neçə "işlənmiş" kartların qəbuledici qutudan geriyə – ilkin qutuya qaytarılma üsulunu təklif etmişdir ki, bu da onların sonradan təzədən oxunması və əməliyyatların yerinə yetirilməsinə imkan verir. Beləliklə, proqramın ayrı-ayrı hissələrini təkrarlamaq, yəni proqram dövrləri təşkil etmək mümkün oldu.

Təəssüf ki, nə fərq maşınları, nə də analitik maşınlar müəllifin yaşadığı dövrdə qurulmadı. Bunun səbəblərindən biri kimi, mütəfəkkir Bebbicin öz dövrünü qabaqlaması göstərilir: texnikanın və istehsalın o zamankı səviyyəsilə çertyojları dəqiqliklə və təfərrüatı ilə icra etmək mümkün deyildi.

İdarəedici kartlar	Əməliyyat kartları	Əməliyyat
1	–	*a*-nı Anbarın 1-ci sütununda yerləşdirmək
2	–	*b*-ni Anbarın 2-ci sütununda yerləşdirmək
3	–	*c*-ni Anbarın 3-cü sütununda yerləşdirmək
4	–	*d*-ni Anbarın 4-cü sütununda yerləşdirmək
5	–	*a*-nı Anbardan Fabrikə (hesab qurğusuna) göndərmək
6	–	*b*-ni Anbardan Fabrikə göndərmək
–	1	*a*-nı *b*-yə vurmaq, $ab = p$
7	–	*p*-ni Anbarın 5-ci sütununa qoymaq və sonrakı istifadə üçün yadda saxlamaq
8	–	*p*-ni Fabrikə göndərmək
9	–	*c*-ni Fabrikə göndərmək
–	2	*p* və *c*-ni toplamaq, $p + c = q$
10	–	q-nü Anbarın 6-cı sütununa qoymaq
11	–	*d*-ni Fabrikə göndərmək
12	–	*q*-nü Fabrikə göndərmək
–	3	*d*-ni *q*-yə vurmaq, $dq = p_2$
13	–	p_2-ni Anbarın 7-ci sütununa qoymaq
14	–	p_2-ni çap etmək.

Analitik maşında müasir kompüterlərə xas olan bütün əsas elementlər nəzərdə tutulmuşdu: Anbar – *yaddaş*, Fabrik – *prosessorun hesab qurğusu*, idarəedici qurğu – *prosessorun idarəetmə qurğusu*. Maşının arxitekturu prakтik baxımdan müasir EHM-in arxitekturuna uyğun gəlir, analitik maşının yerinə yetirdiyi əmrlər isə, əsasən, müasir prosessorların bütün əmrlərini, o cümlədən də proqramın yerinə yetirilmə ardıcıllığının dəyişdirilməsi, şərti keçid və dövr əmrlərini özündə saxlayır. Ona görə də Bebbicin maşını ilk əsl kompüter adlandırıla bilər.

HERMAN HOLLERİT

1860-cı il fevralın 29-da Amerikanın Buffalo şəhərində alman immiqrantları Holleritlərin ailəsində Herman adlı oğlan uşağı doğuldu. Oğlan məktəbə gedəndə məlum oldu ki, o, disqrafiya adlı xəstəlikdən əziyyət çəkir və çox çətinliklə yazır (bununla belə, o, heç də pis rəsm çəkmirdi). Baxmayaraq ki, riyaziyyat və təbiət elmlərini mənimsəmək Herman üçün elə də çətin deyildi, qrammatika dərslərində çəkdiyi əziyyətlər onu 14 yaşında məktəbi atmağa vadar etdi. Evdə gərgin çalışması sayəsində o, bir il sonra Kolumbiya universitetinin Dağ-mədən məktəbinə qəbul olundu. Müəllimlərdən biri – professor U.Traubric, istedadlı gəncə diqqət yetirdi və məktəbi qurtardıqdan sonra onu 1879-cu ildə özünün başçılıq etdiyi əhalinin siyahıya alınması üzrə Milli büroya işə dəvət etdi. Bu hadisə Herman Holleritin həyatında həlledici məqam oldu.

Büro əhalinin siyahıya alınmasında statistik informasiyanın yığılması və emalı ilə məşğul olurdu. Əhalinin siyahıya alınması ABŞ-da qədim ənənəyə malik idi. Onlardan birincisi hələ Corc Vaşinqtonun zamanında baş tutmuşdu.



Herman Hollerit.

RUSİYADA ƏHALİNİN İLK SİYAHIYA ALINMASI

1895-ci il iyunun 5-də imperator II Nikolay əhalinin birinci ümumi siyahıya alınması haqqında əsasnamə təsdiq etdi, orada yazılmışdı ki, "bu Rusiya üçün tamamilə yeni, bu vaxta qədər görünməmiş bir işdir. Siyahıyaalma ondan ibarət olacaqdır ki, hökümətin bütün ölkə ərazisində, həmçinin ayrıca götürülmüş hər bir qəzada, nahiyədə, kənddə və malikanədə yaşayan əhalinin dəqiq sayını bilməsi üçün, ölkənin bütün sakinləri tək-tək sayılacaq".

Siyahıyaalma 1897-ci ilin 9 fevralına təyin olundu. Ölkənin hər bir sakini haqqında aşağıdakı məlumatları toplamaq nəzərdə tutulurdu: adı, ailə vəziyyəti, təsərrüfat başçısına münasibəti, cinsi, yaşı, zümrəsi, dini, doğulduğu yer, ana dili, savadı, məşğuliyyəti, fiziki qüsurları.



1895-ci ilin avqustunda Berndə (İsveçrə) Beynəlxalq Statistika institutunun sessiyasında Mərkəzi statistika komitəsinin (MSK) direktoru N.A.Troynitskiy amerikalı mühəndis Herman Holleritlə tanış oldu. Amerikalı mühəndis bu sessiyada əhalinin siyahıya alınmasında hesablama maşınlarından istifadə olunmasına dair məruzə ilə çıxış edirdi. Əməkdaşlığa maraq qarşılıqlı oldu. Hollerit öz maşınlarının satış bazarının genişlənməsində maraqlı idi, Troynitskiy isə gözəl başa düşürdü ki, əhalinin siyahıya alınması haqda məlumatları əl üsulu ilə emal etmək mümkün olmayacaq.

1896-cı il noyabrın əvvəlində Hollerit rus hökumətindən siyahıyaalmada iştirak etmək üçün dəvət aldı və artıq dekabrın 15-də Sankt-Peterburqa gəldi. Müqavilənin şərtləri haqda çox tez razılığa gəlindi. Müqaviləyə əsasən, Holleritin firması Rusiyaya əvvəllər başqa ölkələrdə istifadə edilmiş 35 köhnə maşın icarəyə verdi. Bu maşınları 1900-cü ilin aprelinə, ABŞ-da növbəti siyahıyaalma başlanana qədər qaytarmaq lazım idi. Bundan əlavə, Rusiya 70 yeni tabulyator və çeşidləyici maşınlar dəsti, həmçinin 500 perforator da aldı.

Texnikanın quraşdırılmasını Holleritin özünün rəhbərliyi altında Peterburqda hazırlamaq qərarına gəldilər. Elə buradaca vəsaitə qənaət etmək məqsədilə onun çertyojları əsasında ayrıca bəndləri və detalları da hazırladılar. Herman Holleritin qonorarı 67571 dollar təşkil edirdi ki, bunun da 59500 dolları yeni maşınların pulu idi.

Siyahıyaalma təyin olunmuş vaxtda başlandı. Onu yüz minlərlə hesablayıcı aparırdı: yerli çinovniklər, qulluqçu və ziyalılar siyahıyaalma vərəqlərini doldururdular. Məsələn, Melixovda siyahıyaalma məntəqəsinə A.P.Çexov başçılıq edirdi. O zarafatla özünü yerli hesablayıcıların "bölük komandiri" "bosmanı" adlandırırdı.

Yanvarın 11-də Anton Pavloviç yazırdı: "Hesablayıcılara çox pis mürəkkəb qabları, pivə zavodunun yarlığına bənzəyən yöndəmsiz ayırma nişanları və siyahıyaalma vərəqləri yerləşməyən portfellər vermişdilər və elə təsəvvür yaranırdı ki, qılınc qınına yerləşmir. Rüsvayçılıqdır. Səhər tezdən daxmaları dolaşıram və öyrəşmədiyimə görə başım qapının üst direyinə dəyir və tərs kimi ağrıdan çatlayır...". Yanvarın 30-da isə o yazırdı: "Siyahıyaalma bitdi. Bu iş məni olduqca təngə gətirdi. Belə ki, barmaqlarım ağrıyana qədər saymaq, yazmaq və 15 hesablayıcıya mühazirə oxumaq lazım idi. Hesablayıcılar çox mükəmməl, gülməli dərəcəyə çatacaq qədər dəqiqliklə işləyirdilər. Amma qəzalarda siyahıyaalmaya məsul yerli idarə rəisləri özlərini çox pis aparırdılar. Onlar heç bir şey etmirdilər, bu işləri anlamırdılar və ən gərgin dəqiqələrdə özlərini xəstəliyə vururdular".

Lakin MSK üçün ağır vaxtlar hələ irəlidə idi. Doldurulmuş siyahıyaalma vərəqələri emal edilmək üçün paytaxta gətirilirdi. Göndərilən vərəqlər çox pis vəziyyətdə idi, bir çox əməkdaşlar sadəcə olaraq nə və necə edəcəklərini bilmirdilər... İş o qədər ləng gedirdi ki, 1897-ci ildə Holleriti yenidən Rusiyaya çağırdılar. Onun məsləhətləri vəziyyəti bir qədər düzəltsə də, 1902-ci ilin yanvarında yenə də emal planını xeyli ixtisar etmək lazım gəldi. Hesablamalar təxminən səkkiz il davam etdi – imperiya üzrə siyahıyaalmanın ümumi yekun nəticələri yalnız 1905-ci ilin sonunda nəşr edildi.

Rusiyada əhalinin ilk siyahıyaalınması tarixdə işlərin təşkil olunmaması nümunəsi kimi qaldı. O hadisələrin son şahidi olan Herman Holleritin tabulyatoru isə 1952-ci ildən Moskvadakı Politexnik muzeyinin kolleksiyasında saxlanılır.

XIX əsrdə siyahıyaalma hər on ildən bir keçirilirdi. Əhali durmadan artırdı və inkişaf edən Amerika iqtisadiyyatı əhalinin tərkibi haqda müxtəlif məlumatlara böyük ehtiyac duyurdu. Məsələn, 1880-ci ildə əhalinin sayı 50 milyon idi və hər nəfərə 210 qrafadan ibarət olan kart açılmalı və doldurulmalı idi! Hesablamalar və nəticələrin emalı yeddi il yarıma qədər uzandı və demək olar ki, təxminən növbəti siyahıyaalmaya qədər davam etdi. Bu, işlərin dalana dirənməsi demək idi. Təcili olaraq işin yeni təşkili üsulları tələb olunurdu.

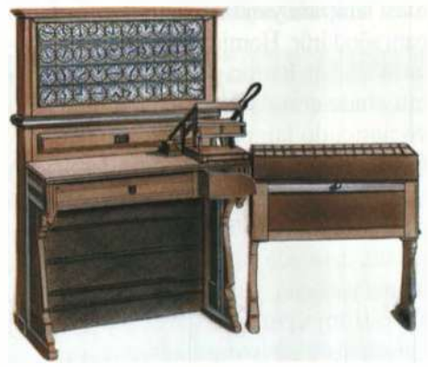
Yekun məlumatları tərtib edən departament bürosuna rəhbərlik edən doktor Con Billinqs Holleritin hesablayıcılarının işini mexanikləşdirmək fikrini irəli sürdü. O, informasiyanın yazılması üçün perfokartlardan istifadə edilməsini təklif etdi. Doğrudur, əvvəlcə Hollerit barabana dolanmış və metal stol üzərində sürüşən kağız lentdən istifadə etmək istəyirdi. Onu yuxarıdan zəif bərkidilmiş mismar sırasından ibarət metal zolaq sıxırdı, mismar lentin deşiyinə düşəndə elektrik dövrəsi hesablama mexanizmini hərəkətə gətirərək qapanırdı. Tezliklə aşkar oldu ki, kağız tez-tez cırılır, informasiya axıra qədər oxunmur və lazımi informasiyanı axtaranda lenti təkrar yenidən sarımaq çox vaxt aparır. Buna görə də lentdən perfokartın xeyrinə imtina edildi. Sonralar Hollerit yazırdı ki, son qərara o, qatarda konduktorun işini müşahidə edən zaman gəlmişdi. Konduktor biletdə müəyyən olunmuş yerlərdə deşiklər açmaqla sərnişinin cinsini, saçının və gözünün rəngini qeyd edirdi. Nəticədə bilet sahibinin xarici görünüşünü təsvir edən perfokarta oxşayan bir şey alınırdı.

İlk tabulyator 1887-ci ildə Baltimorun statistik bürosunda sınaqdan çıxarıldı. Nəticələr xeyli ümidverici oldu və iki ildən sonra daha bir sınaq baş tutdu – Sent-Luisdə əhalinin qismən siyahıyaalınması keçirildi. Əl ilə hesablamalara nisbətən vaxta qənaət iki dəfə artıq, bəzi iş sahələrində isə on dəfə artıq oldu! Hökumət Holleritlə avadanlıq göndərilməsi haqda müqavilə bağladı və artıq 1890-cı ilin iyulunda dünyada ilk dəfə maşınların tətbiqilə əhalinin siyahıyaalınması keçirildi. 62 milyon karta qeyd olunmuş məlumatların emalı iki ilə yaxın vaxt apardı, qənaət isə 5 milyon dollar təşkil etdi – bu çox böyük bir məbləğ idi, çünki ABŞ-ın dövlət büdcəsi o zamanlar cəmi 100 milyon dollar təşkil edirdi. Holleritin sistemi nəinki yüksək sürət təmin edirdi, həm də müxtəlif parametrlərə görə statistik məlumatları müqayisə etməyə imkan verirdi. Hollerit öz maşınını təkmilləşdirdi. O, rahat klavişli perforator işləyib hazırladı, perfokartların qəbul edilməsi və çeşidlənməsi prosedurlarını avtomatlaşdırdı. Ümumilikdə, ixtiraçı 30-dan artıq patent aldı.

Holleritin xidmətləri bütün dünyada qəbul edildi. Filadelfiyadakı Franklin İnstitutu onu medalla təltif etdi, ona 1893-cü ildə Paris sərgisində qızıl medal təqdim etdilər. İxtiraçının Alma-materi olan Dağ-mədən məktəbi, ali təhsilinin olmamasına baxmayaraq ona fəlsəfə doktoru elmi dərəcəsi verdi, Avropa və Amerkanın elmi cəmiyyətləri isə onu özlərinin fəxri üzvü seçdilər. Holleritin tabulyatoru Avstro-Macarıstanda, Kanadada, Norveçdə, İngiltərədə və başqa ölkələrdə istifadə olundu.



Con Billinqs.



Holleritin maşını.

IBM-in ofisi. 1948-ci il.





Tomas Uotson.

1896-cı ildə Hollerit Tabulating Machine Company (TMC) kompaniyasını təsis etdi. Onun maşınları hər yerdə tətbiq olunurdu: dəmir yolu və sənaye müəssisələrində, böyük ticarət firmalarında və sığorta kompaniyalarında. Onların köməyilə əmək haqqını hesablayır, anbar uçotunu aparır və bir çox başqa məsələləri həll edirdilər. 1900-cü ildə ABŞ Dövlət Departamenti siyahıyaalma üçün yenidən Holleritin sistemini seçdi. Lakin dövlət çinovnikləri tezliklə Holleritin sistemindən imtina etmək qərarına gəldilər. Əhalinin siyahıya alınması üzrə Milli büronun əməkdaşı, mühəndis Ceyms Pauers yeni maşınlar yaratdı. Onlar tam mexaniki olub, kifayət qədər mürəkkəb konstruksiyaya malik idilər. Hollerit elə həmin il dövlət sifarişlərini itirərək sahibkarlıq fəaliyyətindən uzaqlaşdı. O, firmasını sataraq, onu alan Computer Tabulating Recording Company (CTRC)-də məsləhətçi kimi qaldı. Artıq 1921-ci ildə CTRC-in məsləhətçisi Herman Hollerit istefaya çıxanda firma gələcəyə ümidlə baxırdı. Daha üç il sonra isə firmanın direktoru Tomas Uotson firmanın adını dəyişdirərək IBM (International Business Machines) qoydu.

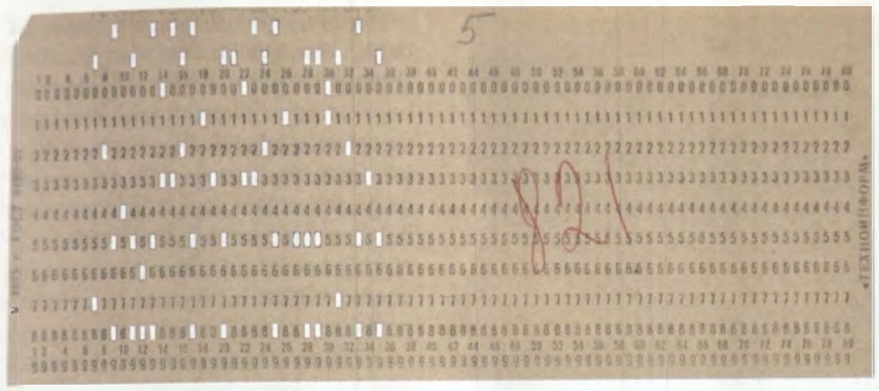
Herman Hollerit 1929-cu il noyabrın 17-də Vaşinqtonda vəfat etdi. IBM kimi möhtəşəm imperiyanın yaradıcısı Tomas Uotson olsa da, Holleritin çoxsaylı mükafat və patentləri kompaniyanın muzeyində ən layiqli yerləri tutur.

AVTOMATLAŞDIRMA VƏ PERFOKARTLAR

Proqramla idarə olunan ən sadə maşınlardan biri – adi elektrik lampasıdır. Elektrik açarının köməyilə onu həm yandırmaq, həm də söndürmək olar. Əgər elektrik açarını söksək, onda görərik ki, orada kontaktlar var ki, onların qapanması lampanı yandırır, açılması isə lampanı söndürür. Həmin aktı qapalı kontaktların arasına karton parçası qoymaqla da müşahidə etmək olar. O, dövrəni açacaq və nəticədə lampa sönəcək. Əgər kartonda deşik açılsa (məsələn, deşikaçan vasitəsilə), onda lampa yanacaq. Lampa üçün proqram tərtib edib, onu karton parçasının üzərində yazmaq olar: deşik varsa, lampa yanacaq, yoxdursa – yanmayacaq. Oxşar ideya perfokartlarda (deşiyi olan karton kartlar) istifadə olunmuşdur.



▸ "Jakkard dəzgahı" üçün perfokartların hazırlanması.



1970-ci illərin perfokartı (əl ilə yazılmış nömrə dəstdə kartı digərlərilə səhv salmamaq üçün yazılırdı).

XVIII əsrdə Fransada toxucu dəzgahlarının konstruktorları perfokartlardan istifadə edərkən elə etməyə cəhd göstərirdilər ki, masura (dəzgahın əsas detalı) proqrama uyğun olaraq avtomatik işləsin. Bu işdə Lion toxucusunun oğlu Jozef Mari Jakkar uğur qazandı. O, 1801-ci ildə perfokartlarla idarə olunan avtomatik toxucu dəzgahı yaratdı.

Kartda deşiyin olması və ya olmaması masuranın hər gedişində sapı yuxarı qalxmağa və ya aşağı düşməyə məcbur edirdi. Beləliklə, köndələn sap perfokartın üzərindəki proqramdan asılı olaraq hər bir uzununa sapın bu və ya digər tərəfinə keçə bilirdi. Yüzlərlə belə saplar qəribə naxışlar yaratmaq üçün istifadə olunurdu. Nəinki dəzgaha, həm də toxucu dəzgahlarında toxunan və haqlı olaraq daha mürəkkəb və dolaşıq toxumalara aid edilən "jakkard" toxumasına Jozef Mari Jakkarın adı verildi.

Bu dəzgah verilmiş proqramla avtomatik işləyən ilk kütləvi sənaye qurğusu idi. Onu insanın yaratdığı ən mükəmməl maşınlardan biri hesab edirlər. 1801-ci ildə bu dəzgahın iri naxışlı parçalar üçün nəzərdə tutulmuş modeli Paris sərgisində medal ilə mükafatlandırıldı, Jakkarın özü isə Paris muzeyinə (İncəsənət və Peşələr Konservatoriyasına) işə dəvət olundu. Artıq yalnız Fransada 10 mindən artıq belə dəzgah işləyirdi. O, 1808-ci ildə bu gün də məşhur olan naxışlı toxuculuq üçün daha mükəmməl dəzgah yaradır. Lakin bu ixtiranın həm də qəmgin bir tarixi vardır. Bu dəzgahların Lion şəhərində Jakkarın emalatxanasında qurulması 1803-cü ildə Lion toxucularının üsyanına səbəb olmuşdur. Parça istehsalının həcminin artması və bunun nəticəsində onların ucuzlaşması və asanlıqla alına bilməsi sosial partlayışa gətirib çıxardı.

Perfokartlar nəinki toxuculuq işində, həm də statistikada dönüş yaratdı. Statistika daim böyük həcmli informasiyaların emalı problemlərilə qarşılaşır. Əhalinin siyahıya alınması üzrə Amerika Milli bürosunun əməkdaşı Herman Hollerit ilk dəfə praktiki məsələləri həll etmək üçün "kompüter" tətbiq etdi.

Holleritin maşınına aşağıdakılar daxil idi:

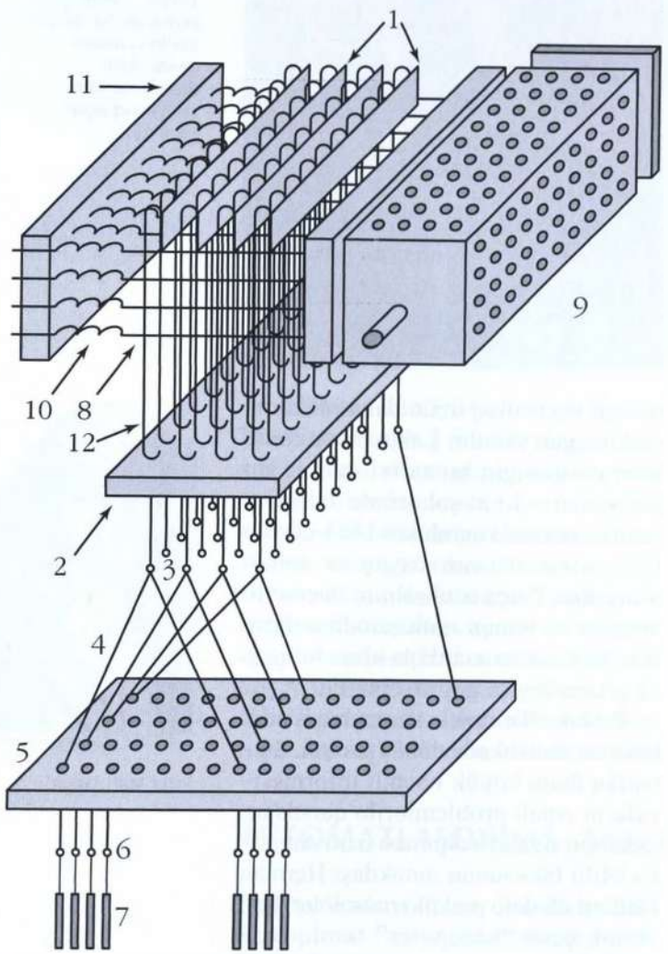
• Klavişli perforator, eyni zamanda bir neçə kartda dəqiqədə, təxminən 100 deşik açırdı (perforasiya edirdi).



Holleritin maşını.



Jozef Mari Jakkar.



• Çeşidləyici maşın. Bu maşın üst qapağı olan qutular dəstindən ibarət idi. (burada kartlar müəyyən konveyer ilə hərəkət edirdi; kartların bir tərəfində yaylar üzərində hesablayıcı oxlar, digər tərəfində isə civə ilə dolu qab var idi; ox perfokartdakı deşiyə düşəndə yayın hesabına o biri tərəfdə olan civəyə toxunurdu və müvafiq qutunun üst qapağını açaraq elektrik dövrəsini qapayırdı perfokart da ora düşürdü);

• Tabulyator. O da müvafiq qaydada işləyirdi, lakin elektrik dövrəsinin qapanması müvafiq sayğacın bir vahid artmasına gətirib çıxarırdı.

Perfokartlar ölçülərinə görə dollar kağızı boyda idi və perforasiya üçün (12 sırada 20 mövqe olmaqla) 240 mövqeyi vardı. İndi belə deyə bilərik ki, Holleritin maşınında perfokartların köməyilə informasiyanın kodlaşdırılmasından istifadə edilmişdir.

Jakkar dəzgahının sxemi.

1 – ayaqcıqlar
2 – çərçivə lövhəsi
3 – çərçivə qaytanları
4 – arkat qaytanları
5 – ayırıcı lövhə
6 – üzcüklər
7 – yükcüklər
8 – iynələr
9 – perforasiya olunmuş prizma
10 – yay
11 – lövhə
12 – qarmaqlar

"ELMİ" KALKULYATORLAR



Karl de Polmer.

1820-ci ildə Fransada Karl Ksavye Tomas de Kolmer (1785-1970) Leybnis maşınının ideyalarından istifadə edərək ilk mexaniki kompüter-kalkulyatoru – *arifmometri* yaratdı. Maşın toplama və çıxma əməliyyatları ilə yanaşı, vurma və bölmə əməliyyatlarını da yerinə yetirə bilirdi. Tomas de Kolmer 1820-ci ilin 18 noyabrında öz ixtirasına görə Fransada 1420 saylı patent aldı.

Arifmometr əsl kommersiya uğuru qazandı. Sonrakı 90 il ərzində min beş yüzdən artıq maşın istehsal olundu. 1878-ci ildə Almaniyada Artur Burqhard bu cür maşınların istehsalını təşkil etdi.

Rusiyada 1874-cü ildən icveçli Vilqod Odner arifmometrin yaradılması üzərində işləmişdir, 1890-cı ildə isə artıq zavodda stolüstü hesablama maşınları istehsal etməyə başladılar (onların modifikasiyası olan "Feliks" maşınları Rusiyada XX əsrin 80-ci illərinə qədər istehsal olunurdu). Cihazın əsas detalı – dişli təkər Odnerin adını daşıyır).

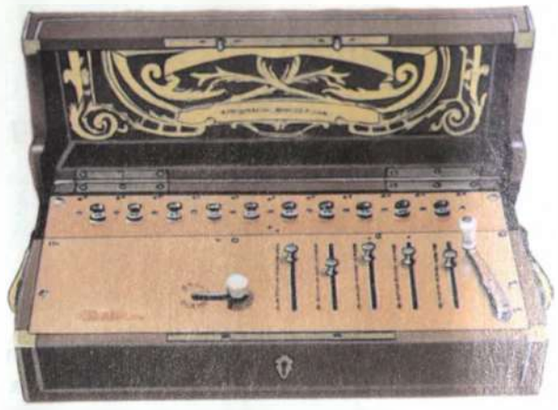
XX əsrin əvvəlində Odnerin arifmometri bütün dünyada istehsal edilirdi, yalnız Rusiyada 1914-cü ilə qədər 20 mindən çox arifmometr işləyirdi.

İsveçdə 1913-cü ildən MADAS (*ing.* "multiplication", "automatic" "division", "addition" and "Subtraction" – "vurma", "bölmə", "toplama" və "çıxma") istehsal edilir. Vurma və bölməni tam avtomatik icra edən arifmometrlər 1927-ci ildən istehsal edilməyə başlandı.

"Arifmometr" termini elə dörd hesab əməliyyatını yerinə yetirə bilən mexaniki qurğu deməkdir. Mexaniki kalkulyator təkmilləşdirilərək ona bir sıra funksiyalar əlavə olundu: aralıq nəticələrin yadda saxlanması, onlar üzərində növbəti əməliyyatlar, nəticələrin çapı və s. Bu, elmi tədqiqatlar nəticəsində deyil, stolüstü hesablama maşınlarına kommersiya tələbatının artması nəticəsində meydana çıxdı. Elmi nəticə ondan ibarətdir ki, müasir kompüterlərdə də sadə (bəzi maşınlarda isə mürəkkəb) hesab əməliyyatlarını yerinə yetirən hesab qurğusu (və ya qurğuları) var. Hesab qurğusunun əməliyyat apara bildiyi ədədin mərtəbələrinin maksimal sayı *maşın sözünün ölçüsü* adlanır.

XIX əsrin sonlarındakı tərəqqi riyazi fizikanın inkişafını tələb edirdi. Dəmir yollarının layihələndirilməsində və paroxodların, parça fabriklərinin və körpülərin tikintisində elə maşınlar lazım idi ki, onlar çoxlu sayda təkrarlanan hesablamaları yerinə yetirə bilsinlər. Lakin belə maşınların istehsalına hələ çox var idi.

1880-ci ildə amerikalı ixtiraçı Tomas Alva Edison (1847-1931) təhlükəsiz elektrik lampalarının istehsalına başladı. 1883-cü ildə o, vakuum balonuna platin elektrod daxil etdi, ona cərəyan buraxdı və müşahidə etdi ki, elektrod ilə kömür sapı arasından cərəyan keçir. Termoelektron emissiyası adı ilə məlum olan bu hadisə "Edison effekti" adlandırıldı.



Kolmerin arifmometri.

1896-cı ildə General Electric kompaniyası (Almaniya) elektrikdə yeni standart işləyib hazırladı: elektrik cihazlarının qidalanması üçün dəyişən cərəyandan istifadə edilməsi. Hesablamaları aparmaq və nəticələri kağızda çap etmək üçün kalkulyatorlar elektrik motorları ilə təchiz edildi. Kompüterlərdə elektrik cərəyanı ilə işləyən komponentlərdən daha çox istifadə edilməyə başlandı.

1904-cü ildə ingilis fiziki Con Ambroz Fleminq (1849-1945) Edisonun ixtirası əsasında diod-cərəyanı yalnız bir istiqamətdə buraxan elektron cihaz yaratdı. İki ildən sonra amerikalı mühəndis Li de Forest (1873-1961) triodu kəşf etdi.

XIX əsr və bütün ondan əvvəlki tarix göstərdi ki, insanın iştirakı olmadan, avtonom rejimdə, əvvəlcədən hazırlanmış proqram əsasında işləyən avtomatik hesablama qurğusu yaradılmalıdır. Lakin ilk kompüterləri mexaniki hesab etmək olar, çünki onlarda yaddaş qurğusu kimi mexaniki qurğulardan istifadə edilirdi.

Hələ 1848-ci ildə ingilis riyaziyyatçısı Corc Bul məntiqin postulatlarını (qaydalarını) təsvir etdi ki, burada da yalnız iki qiymət ala bilən (0 və ya 1, "hə" və ya "yox") cəbri elementlər üzərində əməliyyatlar aparılırdı. Sonralar bu cəbr Bul cəbri adlandırıldı. Faktiki olaraq



Vilqod Odner.



"Feliks" arifmometri.

onun cəbrinin sayəsində kompüterin məntiqi sxemini qurmaq mümkün oldu.

Beləliklə, XX əsrdə kompüterlərdə informasiyanın ikilik kodlaşdırılması tətbiq olunmağa başladı. Burada minimal yaddaş həcmi (1 bit) iki vəziyyəti – 0 və 1-i özündə əks etdirən çevirici idi ki, bu da mühəndislər tərəfindən asanlıqla "reallaşdırılırdı". Lakin çeviricinin elektron "versiyası" ilə bağlı məsələ qalxdı.

1918-ci ildə rus alimi Mixail Aleksandroviç Bonç-Bruyeviç və bundan bir az sonra, 1919-cu ildə ingilis alimləri V.İkkiz və F.Cordan bir-birindən asılı olmadan elektron rele yaratdılar. Bu, iki vəziyyətdən (0 və ya 1) yalnız birində ola bilən qurğu idi. Hal-hazırda *trigger* kimi tanınan rele EHM-in yaddaşının əsasını təşkil etdi.

XX əsrə yaxın EHM yaradılması üçün zəruri olan hər bir şey hazır idi.

KONRAD SUZE

Konrad Suze 1910-cu ilin 22 iyununda Berlində anadan olmuşdu. O, 1935-ci ildə Berlinin Texniki Universitetində inşaatçı-mühəndis diplomu aldı və davamlılıq nəzəriyyəsi üzrə mütəxəssis oldu. Suze 1934-cü ildə, hələ tələbə ikən hesablama qurğularının yaradılması üzərində işləməyə başlamışdı. Buna görə də onu mübaliğəsiz olaraq Avropa qitəsində müasir hesablama texnikasının pioneri adlandırmaq olar.

Suze analitik, elmi təfəkkür tərzilə fərqlənirdi. O, mürəkkəb mühəndis hesablamaları üçün mövcud olan hesablayıcı qurğulardan daha güclüsünü yaratmaq istəyirdi. Hesablama problemlərini düşünərək o, bir sıra nəticələrə gəldi. Birincisi, hesablayıcı proqram vasitəsilə idarə olunmalıdır, həm də yaxşı olardı ki, ikilik say sistemindən və sürüşkən vergüllü hesabdan istifadə olunsun. İkincisi, tam avtomatlaşmış hesab qurğusunun, böyük yaddaş həcminin və iki dayanıqlı vəziyyətə malik elementlərin olması vacibdir.



Edisonun lampası.

Beləliklə Suze faktiki olaraq Çarlz Bebbicin analitik maşınının əsas ideyalarını təkrar etdi (lakin o, bunu bir neçə ildən sonra bildi). Yalnız bir məsələdə Suzenin ideyaları onun dahi sələfinin ideyalarından geri qalırdı: o, şərti keçid əmrini nəzərə almamışdı.

Suze həm də başa düşdü ki, hesablama universal anlayışdır və bu sadəcə olaraq verilənlərin emalıdır ki, onları ikili mərtəbələrin kombinasiyası şəklində təsvir etmək mümkündür (indi biz onları bit adlandırırıq, Suze isə "hə / yox vəziyyəti" deyirdi). Nəzəriyyə praktika ilə yoxlanmalı idi və Suze yaddaş qurğusunun yaradılmasından başladı. Paskal dövründən hesablayıcılarda ədədlər çevrəsinə on rəqəm yazılmış disklər dəsti vasitəsilə təsvir olunurdu. İkilik sistemin seçilməsi yalnız iki: "açıqdır" və "bağlıdır" vəziyyəti olan elementlərdən, məsələn, reledən istifadə etməyə imkan verdi. Lakin Suze əvvəlcə mexaniki hellər üzərində dayandı. Onun yaratdığı yaddaş müəyyən istiqamətdə yerini dəyişə bilən metal lövhələr dəstindən ibarət idi. Emal olunan kəmiyyətlərin qiymətləri və yerinə yetirilən əməliyyat lövhələrin yerdəyişməsilə verilirdi. Bu, alınan nəticələrə uyğun olaraq digər lövhələrin yerdəyişməsinə

gətirib çıxarırdı. Nəticə yaddaşın bitlər dəstində yadda saxlanırdı və öz növbəsində, sonrakı hesablamalarda istifadə oluna bilərdi. 1936-cı ildə yaddaş qurğusuna patent alındı. Bu qurğuda ədədlərlə yanaşı istənilən informasiyanı, o cümlədən, ikilik şəkildə yazılmış proqramları da saxlamaq olardı.

Yaddaş kifayət qədər yığcam idi və min sözə kimi genişləndirilə bilərdi (belə həcmli rele yaddaşı 40 min rele tələb edərdi və bütöv bir otağı tutardı).

1936-1938-ci illərdə ilk dəfə tamamilə mexaniki maşın olan Z1 maşını quruldu. O, üzərində proqramın 3 ünvanlı əmrləri yazılmış perfolentlər vasitəsilə idarə olunurdu. Yaddaş 24 bitlik 16 ədəd həcmində idi. Maşın etibarlı deyildi və buna görə də Suze yeni element bazasına keçməyi qərara aldı. Bütün mexaniki elementləri elektromexaniki relelər ilə əvəz etdi. Amma əvvəlcə o, qeyd olunmuş nöqtəli 16 mərtəbəli ikilik ədədlərlə işləyən, böyük olmayan Z2 maşınını yaratdı. Onun üzərində iş 1939-cu ildə başa çatdı. 200 köhnə telefon relesindən hazırlanan hesab qurğusu maşında mexaniki yaddaşla birləşdirildi.

Bu təcrübə Suzeyə öz gücünə inam hissi yaratdı. O hiss etdi ki, mexaniki elementlərdən istifadə etmədən hesablayıcı maşın yarada bilər.

1939-cu ildə Z3 maşını üzərində iş dayansa da (Suzeni hərbi xidmətə çağırdılar), o, axşamlar və istirahət günlərində gərgin çalışaraq 1941-ci ildə bu işi bitirdi. Bu, proqram vasitəsilə idarə olunan elektromexaniki universal maşın idi və tarixdə ilk dəfə olaraq Çarlz Bebbicin analitik maşın ideyasının gerçəkləşməsi demək idi!

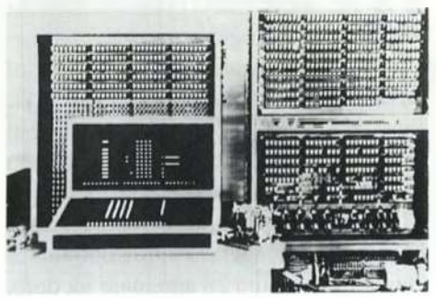
Əvvəlki modellərdə olduğu kimi burada da ikilik say sistemindən istifadə olunmuşdu. Əmrlər bir ünvanlı idi (bu bir çox hallarda hesablamanın sürətini artırırdı, çünki yaddaşa artıq yazılar



Konrad Suze.

yazılmırdı) və onlar perfolentdən (perfolent rolunu kinolent oynayırdı) daxil edilirdi. İlkin verilənlər klaviaturadan daxil edilirdi, hesablamanın nəticələri isə xüsusi tabloda işıqlanırdı. Sürüşkən nöqtəli ədədlər üzərində əməliyyatlar aparat vasitəsilə yerinə yetirilirdi (cəmi doqquz hesab əməliyyatı var idi: toplama, çıxma, bölmə, kvadrat kökün hesablanması, həmçinin $^1/_2$, 2, $^1/_{10}$, 10 və -1 ədədlərinə vurma). Toplama 0,3 saniyə, vurma isə 4 saniyədən 5 saniyəyə qədər vaxt aparırdı. Hesab qurğusu 600 reledən ibarət idi. Hərəsi 22 bit olan (1 bit işarəyə, 14 bit mantisaya, 7 bit də tərtibə) 64 ədəd həcmində yaddaş üçün daha 1800 rele tələb olunurdu.

Z3 maşınından təyyarələrin konstruksiyalarının davamlılığı ilə bağlı çox



Z3 maşını.

Suzenin Plankalkul dilində şahmat oyunu üçün yazdığı proqramın altproqramlarından biri.

```
P 148   | R(V)  =>  R 148                                              (1)
     V  |  0        0
     A  |  5        0

        | x [(x∈V)∧(x = L0)]  => Z                                     (2)
     V  |      0               0
     K  |          1
     A  | 4    5   3           4

        | (Ex) [(x∈V)∧R 17 (Z,x)∧(x = 0)∨ x                            (3)
     V  |      0          0
     K  |                 0 0   1     1.3
     A  | 4    4  5       2 2   3     0

        |      ∧ Ey [(y∈V)∧y∧ R 128 (v, y,x)                            (4)
     V  |           0              0
     K  |                 1.3            0 0
     A  |      4     5    0          5 2 2
```

mürəkkəb hesablamaları aparmaq üçün istifadə olundu.

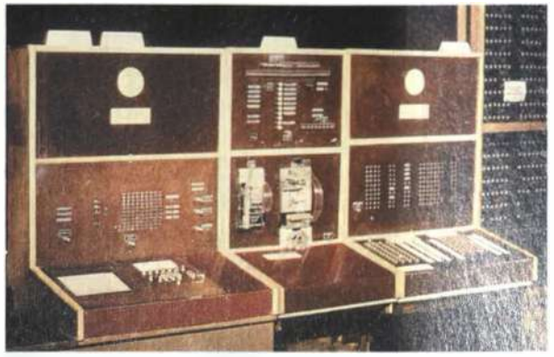
Bu maşın Berlinin bombalanması zamanı məhv oldu, lakin onun 20 il sonra düzəldilmiş surəti indi Münhen muzeylərindən birində nümayiş etdirilir.

1942-ci ildə Suze daha güclü Z4 maşınını qurmağa girişdi. Doğrudur, onun 1024 sözlük yaddaşı mexaniki idi, amma ədədlərin uzunluğu 32 bitə qədər artmışdı. Habelə giriş-çıxış sistemini təkmilləşdirmək və proqramçıya çevik proqramlar yazmaq üçün daha böyük imkanlar vermək nəzərdə tutulurdu. Təəssüf ki, bu ideyaları reallaşdırmaq ona müəssər olmadı.

Z4 maşınının taleyi onun sələflərinin taleyindən daha uğurlu oldu. Müharibənin axırında maşını bombardmanlardan xilas etmək üçün Suze Berlini tərk etdi. Bir yerdən başqa yerə gedərkən o, imkan tapıb işini davam etdirirdi. Almaniyanın təslim olmasına bir neçə həftə qalmış Konrad Suze qədim universitet şəhərciyi olan Gettingendə sığınacaq tapdı. Orada 1945-ci ilin 28 aprelində ilk dəfə, işləyən maşın yerli alimlərə nümayiş etdirildi. Bundan bir neçə gün sonra Suze maşını Qarts dağındakı yeraltı zavodlara (burada faşistlər "cəza silahı" üzərində işləri tamamlamaq üçün qızğın iş aparırdılar) göndərməkdən imtina edərək, Fau-1 və Fau-2 reaktiv mərmilərinin ixtiraçısı Verner fon Braunun əməkdaşlarının köməyilə onu naməlum bir yerə aparır. Yanmış, dağıdılmış ölkə ərazisindəki "səyahət" onu ölkənin cənubuna, dağ kəndi olan Xinterşteynə aparıb çıxardı. Məhz burada, köhnə tövlədə Z4 maşınının qurulması başa çatdı. Elə buradaca Suze tarixdə ilk proqramlaşdırma dili olan Plankalkul dilini yaratdı.

1950-ci ilə qədər Z4 maşını praktiki olaraq Avropada işləyən yeganə kompüter idi. O, 1950-ci ildən 1954-cü ilə qədər Sürix şəhərindəki (İsveçrə) Texniki Universitetdə uğurla istismar olundu. Daha sonralar tarixi əhəmiyyətini nəzərə alaraq Z4 maşınını Münhendə muzeyə qoydular.

Suze kompüterlərin inkişafı və tətbiqinin bir çox istiqamətlərində pioner idi. Çox güman ki, dünyada ilk dəfə məhz o, texnoloji proseslərin idarə edilməsi və onlara nəzarət üçün hesablama qurğusu yaratmışdır. Müharibə vaxtı Suze (mühəndis və konstruksiyaların davamlılığı sahəsində mütəxəssis kimi) idarə olunan reaktiv mərmilərin istehsalı işlərinə cəlb edilmişdi. Hazırlanan mərmilərin kifayət qədər dəqiq olmaması onların aerodinamik xüsusiyyətlərini pisləşdirirdi və buna görə də istehsal xəttinin sonunda hər bir məmulat 80 ötürücünün köməyilə diqqətlə yoxlanırdı. Bundan sonra məmulatın layihə normalarından kənarlaşmasını və zəruri olan düzəlişləri hesablamaq üçün kifayət qədər mürəkkəb hesablamalar aparılırdı. Əvvəlcə Suze əməliyyatları qeyd olunmuş ardıcıllıqla yerinə yetirən, 500 reledən ibarət ixtisaslaşmış hesablayıcı yaratdı. Maşın 12 adamı əvəz edirdi və arası kəsilmədən iki il ərzində – 1942-ci ildən 1944-cü ilə qədər hər gün iki növbə işlədi. Lakin bu zaman texnikin iştirakı tələb olunurdu. O, ötürücülərin göstəricilərini oxuyur, yazır və onları hesablayıcılara daxil edən operatora ötürürdü. Suze bu prosesdə insanın iştirakının tamamilə aradan qaldırılmasına nail oldu. Verilənlər ötürücülərdən avtomatik olaraq oxunurdu və indi bizim analoq-rəqəm çeviricisi adlandıra biləcəyimiz qurğu vasitəsilə hesablayıcıya ötürülürdü. Beləliklə, Suze hesablayıcını qapalı texnoloji prosesə daxil edə bildi.



Z4 maşını

Deməli, tam avtomatlaşdırılmış nəhəng müasir istehsal qurğularının, həmçinin məişət qurğularının içərisində yerləşdirilən miniatür mikroprosessorların kökü Konrad Suzenin rele qurğusuna gedib çıxır.

Hələ müharibə başlanana qədər Suze dostu Helmut Şreyer ilə birgə 2 min elektron lampadan ibarət olan tamamilə elektron maşının layihəsini hazırlamışdı. O dövrdə elə hesab olunurdu ki, onların ideyası reallaşdırıla bilməz və rəsmi instansiyalar da bunu dəstəkləmirdi. Amma 1945-ci ilə yaxın onlar on mərtəbəli ədədlərlə işləyən 100 lampadan ibarət olan hesab qurğusunun maketini hazırladılar (Müharibədən sonra 18 min lampadan ibarət ENIAC kompüterinin meydana çıxması məlum olduqda Suze sarsıldı – bu cür mürəkkəb elektron qurğunun yaradılmasının mümkünlüyü onu heyrətə salmışdı). Suzenin bu işi inkişaf etdirilə bilmədi, çünki məğlub Almaniyada elektronika sahəsində tədqiqatlar qadağan edilmişdi.

Hesablama texnikası tarixində Konrad Suzenin xüsusi yeri var. Bir neçə kö-



məkçisilə işləyən bir adam nəinki digər ölkələrdə böyük elmi kollektivlərin gördüyü işləri görürdü, hətta bir çox hallarda onları qabaqlayırdı. Mexaniki və rele maşınlarından başlayaraq o, elektron qurğuların rolunu dərk etməyə qədər gəlib çatdı. Suze proqramlaşdırma nəzəriyyəsi və praktikası sahəsində, süni intellekt sistemləri və idarə olunan hesablama maşınlarının yaradılması sahələrində ilk cığır açan alim oldu... Elm tarixində tez-tez elə hallara rast gəlmək olur ki, alimlərin zamanı qabaqlayan ideyaları müasirləri tərəfindən tətbiq olunmur. Lakin Suzenin ideyaları aktual idi və istifadə oluna bilərdi. Amma mövcud şərait buna mane oldu. Baxmayaraq ki, Suze hələ sağlığında ideyalarına görə nüfuz qazanmışdı (çoxlu beynəlxalq mükafatlara layiq görülən Suze 1995-ci ildə vəfat etmişdir). Hesablama texnikası öz yolu ilə inkişaf etdi. Bəlkə də Suzenin işləri o dövrdə yaxşı məlum olsaydı, bu yol başqa cür olardı.

ANALOQ MAŞINLARI

Bebbicin maşınlarından fərqli olaraq analoq maşınlarında ədədlər hər-hansı fiziki kəmiyyətlərlə verilirdi. Deyirlər ki, analoq maşınları fasiləsiz (analoq) formada təsvir olunmuş informasiyanı emal edir.

Analoq maşınlarının inkişafı, təbii axarla, sadə həlledici qurğulardan mürəkkəbə doğru getmişdir. Hələ e.ə. IV minillikdə sonralar Babilistan dövlətinin meydana gəldiyi yerlərdə, torpaqölçmə işləri aparılarkən və kartlar tərtib edilərkən analoq hesablamalarının prinsiplərindən istifadə olunurdu.

E.ə. 80-ci ildə yunanlar Günəş sisteminin geosentrik modelindən istifadə edərək planetari yaratdılar. Beləliklə də, Günəşin və planetlərin vəziyyətini təyin etmək mümkün oldu. Məntiqi məsələləri həll etmək üçün nəzərdə tutulmuş mexaniki qurğu haqqında əfsanə mövcuddur və guya müəllifi də filosof Platon olmuşdur.



Buna baxmayaraq loqarifmik xətkeş ilk analoq hesablama qurğusu kimi qəbul olunur. Loqarifmik şkalanı ingilis riyaziyyatçısı Edmund Günter kəşf etmişdir. O, bunlardan parçaların toplanması və çıxılması üsulu ilə ədədləri vurmaq və bölmək üçün istifadə edirdi. 1654-cü ildə Robert Bissaker (Fransa) ilk loqarifmik xətkeşi icad etdi.

Xətkeşdə parçalar ədədlərin loqarifmlərini özündə əks etdirir. Buna görə də vurma zamanı parçalar toplanır ($\ln ab = \ln a + \ln b$), bölmə zamanı çıxılır ($\ln {}^{a}/_{b} = \ln a - \ln b$), kvadrat kökün hesablanması zamanı isə yarıya bölünürdü ($\ln \sqrt{a} = {}^{1}/_{2} \ln a$). Əgər xətkeşin şkalaları eyni ölçülü olsaydı, onda onun vasitəsilə yalnız toplama və çıxma yerinə yetirilərdi.

Qrafik və *nomoqramları* analoq hesablama qurğularının başqa növü hesab etmək olar. Qrafiklər ilk dəfə 1791-ci ildə bir neçə dəyişənli funksiyanın qiymətlərini hesablamaq üçün naviqasiya təlimatı kimi istifadə olunmuşdur.

O zamandan bəri müxtəlif məsələlərin həlli üçün çoxsaylı analoq cihazları ixtira edilmişdir. XIX əsrdə adi tənliklərin həlli üçün su ilə dolu olan birləşmiş qablardan istifadə olunurdu. 1814-cü ildə ingilis mühəndisi C.German analoq qurğusu – qapalı əyrilə məhdudlaşmış müstəvi fiqurların sahəsini hesablayan *planimetr* icad etdi. Diyirlənən təkərli birləşdirici qurğu yaradan İsveçrə alimi Yakob Amsler 1854-cü ildə planimetri təkmilləşdirdi. İngilis fiziki Ceyms Tomson isə *friksion inteqrator* adlanan cihaz icad etdi.

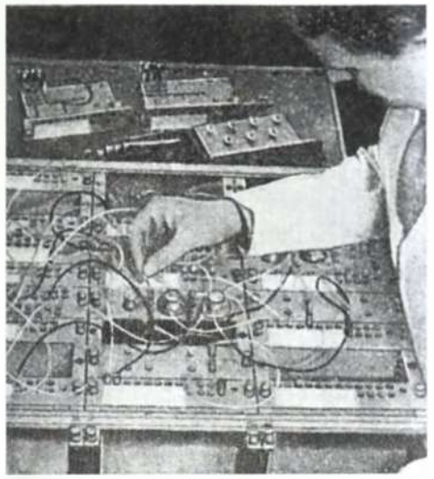
1876-cı ildə onun doğma qardaşı, fizik Uilyam Tomson baron Kelvin friksion inteqratoru tətbiq etməklə müxtəlif limanlarda dəniz suyunun qabarma vaxtını və hündürlüyünü təyin edən cihaz ixtira etdi. Hesablamalar Günəş və Ayın vəziyyətinə əsaslanırdı, onları təsvir edən mexanizmlər isə elektromotor vasitəsilə hərəkətə gətirilirdi. Kelvin həmçinin xətti tənliklər sistemini həll edən maşının müəllifidir. Bununla bərabər, o, bir neçə inteqratorun birləşdirilməsi yolu ilə diferensial tənliklərin həllinin mümkünlüyünü göstərmişdi, lakin texnikanın aşağı səviyyəli olması bu ideyanı reallaşdırmağa imkan vermədi.

Diferensial tənliklərin həlli üçün ilk mexaniki hesablama maşınını 1904-cü ildə rus mexaniki və riyaziyyatçısı A.H.Krılov yaratdı.

O, 1879-cu ildə mexaniki analoq inteqrallayıcı qurğunu – inteqrafı (qrafik üsulla verilmiş ixtiyari funksiyanın inteqrallanması üçün qurğu) icad etmiş polyak riyaziyyatçısı Bruno Abdank-Abakanoviçin ideyalarına əsaslanırdı.

1909-cu ildə Artur Rayt toplama və çıxma əməlliyyatlarını yerinə yetirmək üçün, elektrik dövrəsində müqavimətlərin ardıcıl və paralel birləşdirilməsi qanunu əsasında işləyən cihaz düzəltdi ($R = R_1 + R_2$ – ardıcıl birləşmə və ${}^{1}/_{R} = {}^{1}/_{R_1} + {}^{1}/_{R_2}$ – paralel birləşmə üçün).

İkinci dünya müharibəsi dövründə analoq maşınlarının işlənib hazırlanması xüsusi intensivliklə aparılırdı. Həmin illərdə artilleriya atəşinin idarə edilməsi və bombaların tuşlanması üçün qurğular yaradıldı. Analoq qurğuları ordunun tələblərinə tamamilə cavab verirdi, belə ki, verilənləri daxil etdikdən sonra dərhal cavabın alınması mümkün idi.



"Eliot" firmasının maşınının kommutasiya panelində yeni məsələnin qurulması.

Analoq maşınlarının dəqiqliyinin son həddi 0,1%-ə çatırdı ki, bu bir tərəfdən qarşıya qoyulmuş məsələlərin çoxunu müvəffəqiyyətlə həll etməyə imkan verirdi, digər tərəfdən isə aradan qaldırıla bilməyən maneə yaradırdı. Belə ki, maşının özünün hissələrinin böyük dəqiqliklə hazırlanmasını tələb edirdi.

Analoq maşınlarında çox vaxt ədədlər gərginliyin qiymətilə kodlaşdırılır. Adi potensiometrin (dəyişən müqavimətin) köməyilə, cihazın dəstəyini ya saat əqrəbinin hərəkəti və ya onun əksi istiqamətində uyğun kəmiyyət qədər çevirməklə toplama və çıxma əməlləri yerinə yetirilir. Potensiometrin çıxışındakı gərginlik kəmiyyəti onun girişindəki gərginlik kəmiyyəti və dəstəyin dönmə bucağı ilə düz mütənasibdir ($U_{çıxış} = U_{giriş} f(\varphi)$, burada φ – dəstəyin dönmə bucağı, f – potensiometrin dəyişən müqavimətinin onun tam müqavimətinə olan nisbətinin dəyişmə funksiyasıdır), başqa sözlə potensiometrdən vurma (və ya bölmə) cihazı kimi istifadə etmək olar: $z = xf(y)$. İxtisaslaşmış sinus-kosinus potensiometrləri eyni

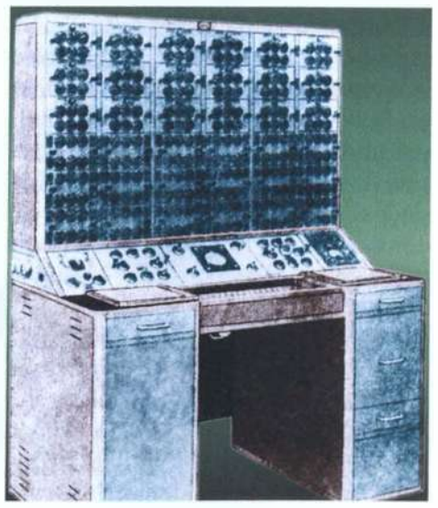


Nomoqramlar – xüsusi çertyojlar olub müəyyən tip məsələlərin, məsələn, kvadrat tənliyə gətirilmiş məsələlərin həlli üçün nəzərdə tutulur.

zamanda iki funksiyanı – $z = x \cos y$ və $z = x \sin y$ funksiyalarını reallaşdırmağa imkan verir .

Digər vurma üsulunda iki çarxdan istifadə edilir ki, burada da vuruqlara çarxlarda olan cərəyan, hasilə isə çarxlar arasındakı cərəyan uyğun gəlir. Gücün ölçülməsi cihazı – vatmetr iki deyil, üç kəmiyyətin vurulmasını tələb edir: $p = UI \cos \varphi$, burada φ bucağı U və I arasındakı faza bucağıdır. Vatmetrin bir çarxından I cərəyanı, digərindən isə U gərginliyinə mütənasib olan cərəyan buraxırlar.

Ev üçün elektrik sayğacı isə daha mürəkkəb iş görür. O, $UI \cos\varphi$ ifadəsini zamana görə inteqrallayır, yəni hər bir zaman anında bu kəmiyyəti onun davam etdiyi kiçik müddətə vurur və alınmış bütün cəmləri toplayır. Bu zaman diskin sürəti istifadə olunan gücə, dövrlərin miqdarı isə – zamana görə inteqrala bərabər olur. Ev sayğacı elektromexaniki cihazdır. Bundan fərqli olaraq elektron analoq maşınlarında inteqrallama adətən kondensatorda elektrik yükünün yığılması yolu ilə baş verir. Diferensial tənliklərin həlli üçün nəzərdə tutulmuş *diferensial analizatorlarda* əsas proses zamana görə inteqrallama əməliyyatıdır.



"Şot Brazers" firmasının universal analoq maşını.



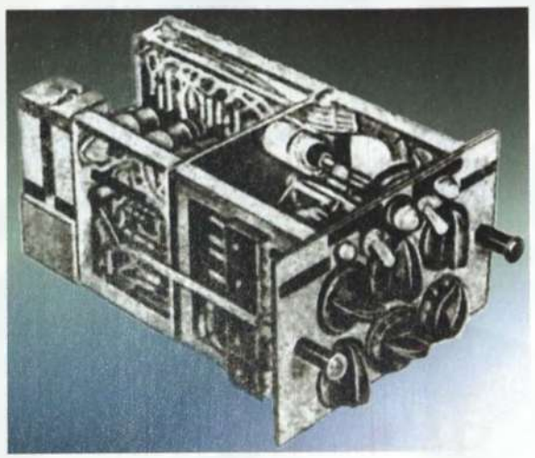
Analoq maşını məsələni ümumi halda həll etməyə qadir deyil ki, sonra yeni ilkin verilənlər daxil etməklə ümumi həldən xüsusi həllər almaq mümkün olsun.

İşləmə prinsipinə görə analoq maşınları yalnız konkret ədədlərlə işləyə bilər və ancaq xüsusi həllər verə bilər.

Sabit cərəyan gücləndiriciləri elektron analoq maşınlarının vacib elementləridir. Toplama, çıxma, inteqrallama, diferensiallama, zaman şkalasının dəyişməsi və s. əməliyyatlar bu gücləndiricilərə əsaslanır. Böyük maşınlarda bəzən 500-dən artıq gücləndiricidən istifadə edilirdi. Bu cür cihazlar maşında asanlıqla yerləşdirilə bilən xüsusi oyuqlarda quraşdırılır. Bu, gücləndiricilərin əsas oyuqlarını, kondensatorları və müqavimətləri tətbiq edərək istənilən sxemləri yığmağa imkan verir. Birləşmə elementlər sxemində olan bütün naqillərin çıxarıldığı kommutasiya panelində (bu, telefon stansiyasının panelinə oxşayır) baş verir. Oyuqlararası birləşmə tez çıxarılıb taxıla bilən şteqerli naqillər vasitəsilə icra edilir. Məsələnin həllini sxemin qida mənbəyini qoşmaqla dəqiq əqrəbli cihazda, qeyd edən özüyazanda və yaxud elektron şüa borusunda almaq olar.

XX əsrin ortalarında artıq diferensial analizatorlardan başqa modelləşdirmədə istifadə olunan digər geniş sinif analoq maşınları da mövcud idi. Maşının ayrı-ayrı elementləri bilərəkdən sxemdə elə birləşdirilir ki, sistemin modelini təsvir





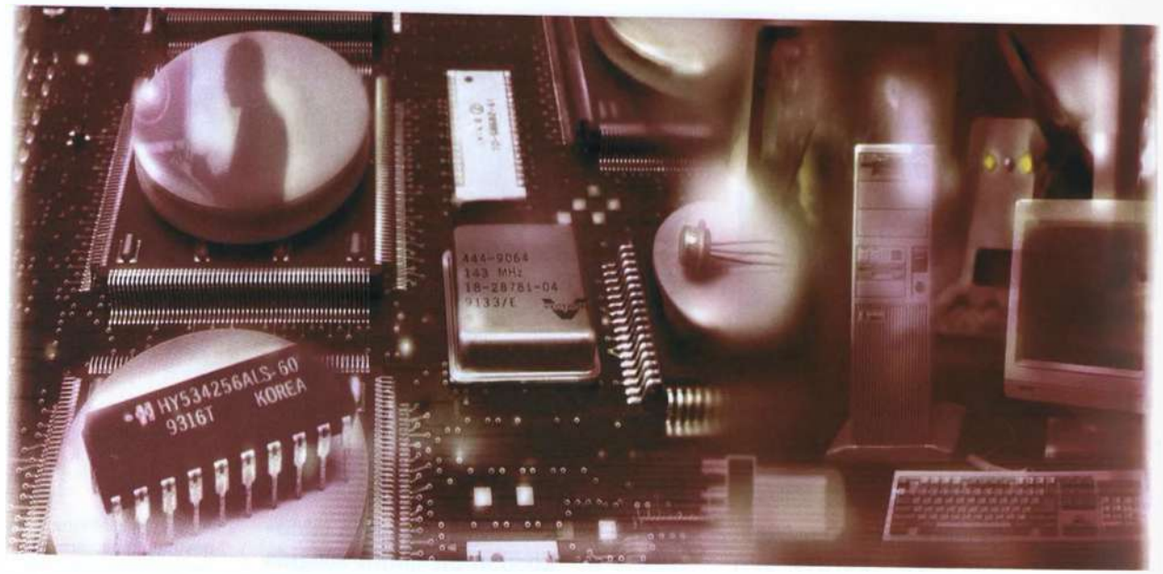
Analoq maşınlarına qoyulan sabit cərəyan gücləndiricisinin standart bloku.

etmiş olsun. Maşının sistemdə analoqun fiziki davranışını təsvir edən ayrı-ayrı hər bir hesablayıcı elementi bir riyazi tənliyi həll edir. Analoq maşınlarının bu xüsusiyyəti alimlərə riyazi düsturların arxasında həqiqi fiziki sistemi görməyə imkan verir. Analoq maşınları aviasiyada aerodinamik sistemlərin və təyyarənin idarə edilməsi sistemlərinin modelləşdirilməsində aktiv istifadə edilirdi.

Analoq maşınları bu gün də istifadə olunur. Bir çox avtomobillərdə iki tip – rəqəmli və analoq qurğularını özündə birləşdirən cihaz vardır. Bu, əqrəb vasitəsilə avtomobilin hərəkət sürətini göstərən analoq qurğusu, hamıya məlum olan spidometrdir. Spidometrin tablosunda, mərkəzdən bir qədər aşağıda odometrin pəncərəsi yerləşir ki, bu da keçilmiş yolun rəqəmli sayğacıdır.

XX əsrin sonunda bəzi avtomobil istehsalçıları cihazlar panelini tamamilə rəqəmli panellə əvəz etməyə cəhd göstərdilər. Amma insanlar analoq şkalalarını görməyə adət ediblər, hətta rəqəmli saatlar da əqrəbli siferblatı olan saatları sıradan çıxara bilmədi və çox güman ki, heç vaxt da çıxara bilməyəcək.

Çox güman ki, analoq maşınlarının ikinci dirçəlişi hələ qabaqdadır, axı kvant kompüterləri də elə bu sinif maşınlara aiddir.



XX ƏSRDƏ KOMPÜTERİN TƏKAMÜLÜ

BİRİNCİ NƏSİL KOMPÜTERLƏR

MARK I

XX əsrin 30-cu illərinin sonunda perfokartları kodlaşdırılmış informasiya daşıyıcışı kimi istifadə edən çoxsaylı qurğular kifayət qədər etibarlı idi və o zamanın parametrlərinə görə yüksək oxuma sürətinə malik idi (dəqiqədə bir neçə perfokart). Alimlər rəqəmli maşınların – kompüterlərin yaradılmasına başlayanda onlardan informasiyanın saxlanması üçün və giriş-çıxış qurğuları qismində istifadə etməyə başladılar. Belə maşınlardan biri IBM firmasının laboratoriyasında Hovard Aykenin rəhbərliyilə bir qrup alim tərəfindən hazırlandı və onu Mark I adlandırdılar. Onu kalkulyatorlarda istifadə edilən standart elektromexaniki hissələrdən yığdılar. Mark I nəinki 23 mərtəbəli onluq ədədlər ilə dörd əsas hesab əməliyyatını yerinə yetirirdi, həm də xüsusi daxili alqoritmlərin köməyilə triqonometrik və loqarifmik funksiyaları hesablaya bilirdi. Maşın yalnız irəliyə tərəf hərəkət edən perfolentdən "proqramlaşdırılırdı" (perfolentdə də informasiya perfokartda olduğu kimi müəyyən yerlərdə deşiklərin açılması ilə kodlaşdırılırdı). Buna görə də dövrlərin yerinə yetirilməsi və idarənin geriyə ötürülməsi mümkün deyildi.

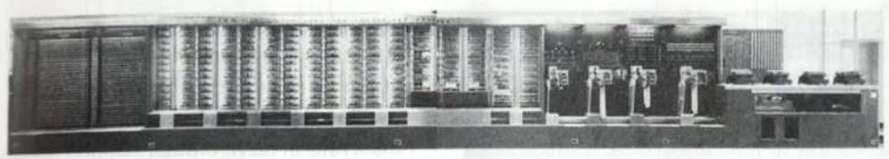
Hovard Ayken və Qreys Hopper.



Nəticə ya perforatora, ya da adi elektrik yazı maşınına verilirdi. Mark I hesab əməliyyatlarını yerinə yetirmək üçün fırlanan disk-sayğaclardan (arifmometrdə olduğu kimi), bəzi funksiyalar üçün isə *elektromaqnit relelərdən* istifadə edirdi. Beləliklə bu maşını "releli" kompüter kimi təsnif etmək olardı. Lakin onun ölçüləri çox böyük idi.

Mark I uzunluğu 20 m olan divar boyunca yerləşirdi və təxminən 5 ton ağırlığı var idi. Müasir parametrlərə görə maşın ləng (vurma üçün ona 3 saniyədən 5 saniyəyədək vaxt tələb olunurdu), amma tam avtomatik işləyirdi və dövrlərin qurulmadığı (onları əl ilə əməliyyatlar ardıcıllığına bölmək lazım gəlirdi) uzun hesablamalar üçün istifadə oluna bilərdi. Belə ki, $n!=n(n-2)...1$ hesablanması $n-1$ sayda vurma sətrindən ibarət olurdu və bu $n!$-ın dövr vasitəsilə bir neçə sətrdə hesablanmasından fərqlənirdi:

```
m := 1
faktorial := 1
db nə qədər ki m < n + 1
| faktorial := faktorial * m
| m := m + 1
ds
```

Mark I elektromexaniki hesablama maşını.

ENIAC

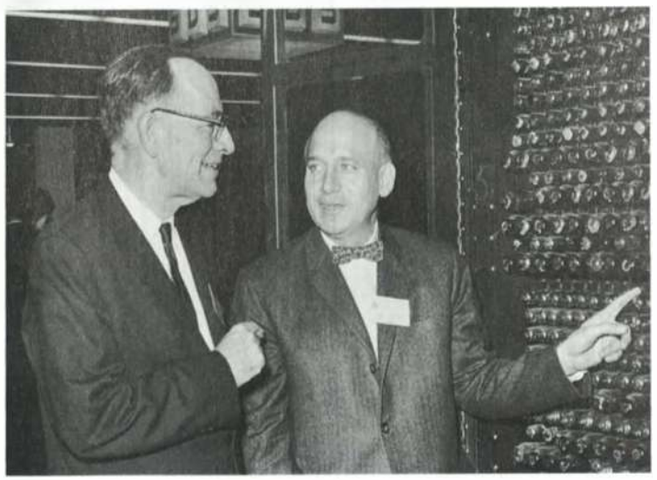
İkinci dünya müharibəsi alimləri avtomatik hesablama maşınları üzərində tədqiqatları fasiləsiz davam etdirməyə vadar edirdi. Hərbi texnika sürətli riyazi hesablamalar tələb edirdi. Məsələn, zenit atəşinin idarə edilməsi zamanı tuşlama sistemi üçün buna ehtiyac duyulurdu. Mexaniki kalkulyatorlar lazımi hesablama sürətini təmin edə bilmirdi və buna görə də hərbçilər müvafiq tədqiqatların aparılmasını və tezliklə elektron hesablama maşınlarının yaradılmasını təkid edirdilər. 1942-ci ildə Prosper Ekert və Con Mouçli öz məsləkdaşları ilə birgə Pensilvaniya (ABŞ) ştatındakı universitetin Elektrik Tədqiqatları Məktəbində sürətli EHM yaratmağı qərara aldılar və onu ENIAC (bu söz ingilis dilindən "elektron ədədi inteqrator və hesablayıcı" kimi tərcümə olunur) adlandırdılar.

Maşın sözünün ölçüsünün cəmi 10 dənə onluq rəqəmdən ibarət olmasına baxmayaraq (Mark I-də 23), ENIAC bir saniyədə 300 vurma əməliyyatı və təxminən 5000 toplama və çıxma əməliyyatı yerinə yetirirdi! Bu cür məhsuldarlığa, vurma cədvəllərinin nəticələrinin hazır cədvəllər şəklində yaddaşda saxlanması nəticəsində nail olundu. ENIAC "ləng" relelərin əvəzinə 18 min elektron vakuum lampalarından, kodlaşdırılmış informasiyanın daxil edilməsi və çapı üçün isə perfokartlardan istifadə edirdi.

Bir çox müasir hesablama maşınlarında olduğu kimi, ENIAC da bir neçə blok qurğudan ibarət idi: bir blok toplayıb çıxır, digəri vurur, üçüncüsü bölür və hətta kvadrat kökü də hesablaya bilirdi və s. Bundan əlavə 20 dənə onluq registr-sayğac var idi ki, onlar da toplamada və nəticələrin yadda saxlanmasında istifadə olunurdu. Ədədlərin registrlərdən oxunması və ora yazılması 0,2 milli saniyəyə yerinə yetirilirdi.

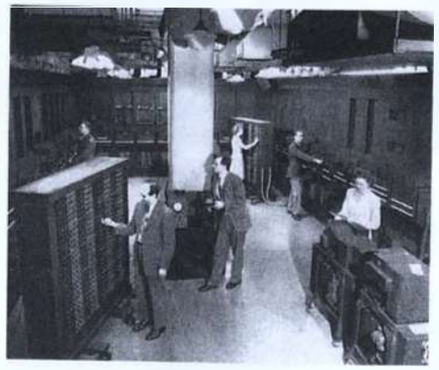
Onu nəzərə alsaq ki, yaddaşda yerləşən ədədlərin oxunması üçün sərf olunan vaxt registrlərdə yerləşən ədədlərin oxunmasına sərf olunan vaxtdan bir neçə tərtib çox idi, aralıq nəticələri

Con Mouçli və Prosper Ekert.



ENIAC indiki məktəbliləri öz imkanları ilə olmasa da, ölçülərilə heyrətə sala bilərdi. O, 150 m² sahədə yerləşirdi və 180 kVt-dan artıq enerji işlədirdi.

ENIAC kompüteri.



registrlərdə saxlamaq olurdı və bu, hesablama vaxtını azaldırdı. Məsələn, əgər 123+102+45 cəmini hesablamaq lazımdırsa, onda 123 və 102 yaddaşdan oxunur, birinci iki toplananın nəticəsi registrdə yadda saxlanırdı. 225+45 cəmi hesablanarkən yalnız 45 ədədini yaddaşdan götürmək lazım gəlirdi.

ENIAC maşınında hesablama proqramları mexaniki çeviricilərin və ştekerli elastik kabellərin köməyilə əllə daxil edilirdi. Həmin ştekerlər lazımi oyuqlara (ştekerbirləşmələrinə) yerləşdirilirdi ki, bu da XX əsrin əvvəllərindəki telefon stansiyalarını xatırladırdı. Faktiki olaraq bu maşında proqramlar yazılmırdı, "sancılırdı".

Hesablama proqramının dəyişdirilməsi heç də az səylər tələb etmirdi. ENIAC maşınının yaradılması hələ başa çatmamış, vaxtilə Mouçli-Ekert komandasının işində iştirak edən görkəmli amerika riyaziyyatçısı Con fon Neyman onunla maraqlanmağa başladı. O, bütün hesablama komandalarının və bütün funksiyaların daxil olduğu standart dəstli kabel birləşmələri bloku yaratmağı təklif edərək maşında əhəmiyyətli təkmilləşmələr apardı. Yaddaşın seçilmiş sahəsində saxlanan proqram hesablama prosesini idarə etməyə başladı. İkilik ədədlər dəstindən ibarət olan həmin proqram mütəxəssis olmayanlar üçün anlaşıqlı deyildi və buna görə də *maşın proqramı* adlandırıldı. Onun komandalarının hər biri müəyyən funksiyaya, yəni birləşmə blokunun müəyyən kabel birləşməsinə uyğun idi. Artıq proqramın yüklənməsi üçün yeni kabel birləşmələri qurmaq və ya köhnələri sökmək lazım gəlmirdi. Yalnız yeni proqramı yaddaşda yerləşdirmək lazım idi. Beləliklə, fon Neyman müasir hesablama maşınlarının fəaliyyətinin əsasını təşkil edən prinsipi ifadə etdi: EHM-in yaddaşında yalnız emal olunan ədədlər deyil, həm də proqramın özü saxlanacaq; onların hər ikisi də çox qiymətli ikilik ədədlər kimi saxlanılır. Həll olunmuş problemlərin geniş miqyası və ENIAC maşınının hər tərəfli tətbiq imkanları birinci nəsil kompüterlərin əsasını qoydu.

Sonralar, 1971-ci ildə ENIAC-ın ilk rəqəmli hesablama maşını olması barədə mübahisələr yarandı. 30-cu illərdə Con Atanasovun rəhbərliyilə qurulan daha sadə hesablayıcı eynilə vakuum lampalarında elektron çeviricilərin yaradılması prinsipinə əsaslanırdı. 1973-cü ildə bu məsələ ilə əlaqədar məhkəmə dinlənməsi keçirildi və qərara alındı ki, rəqəmli elektron maşınlarının əsas ideyalarına patent hüquqları Con Atanasova məxsusdur. Buna baxmayaraq ENIAC maşınını ilk yüksək sürətli elektron rəqəmli kompüter adlandırmaq olar. Bu maşınlar 1946-cı ildən 1955-ci ilə kimi müvəffəqiyyətlə işləmişdir.



Elektron lampalar.

IAS

İkinci dünya müharibəsindən sonra Con fon Neyman xüsusi kompüterin yaradılmasına başladı və bu kompüter IAS adlandırıldı (*ing.* "Institute for Advanced Studies" – "Qabaqcıl Tədqiqatlar İnstitutu"). Bu maşın ilk dəfə 1952-ci ildə Prinstonda (ABŞ) nümayiş etdirildi.

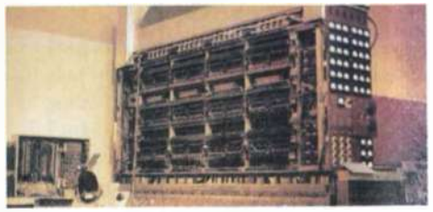
ENIAC maşınında maşın sözlərinin onluq mərtəbələrdə təsvir olunmasından fərqli olaraq bu maşında onlar ikilik mərtəbəli olaraq ölçülməyə başladılar. Bunun "günahkarları" mühəndislər idi, çünki onlara hesablama maşınının yaddaşını elektrik lampasında olduğu kimi ("yanır" və ya "yanmır"), cəmi iki vəziyyətə malik olan elementlərdən layihələndirmək rahat idi. IAS-da maşın sözü 40 bit təşkil edirdi. Məhz bu miqdarda informasiya yaddaş və prosessor arasında bir gedişə ötürülə bilərdi.

Yaddaş 4096 belə sözdən ibarət idi. Fon Neymanın ideyasına görə yaddaşa yazılmış söz ya prosessora ünvanlanan komanda (*göstəriş*), ya da verilənlər (məsələn, hesablamaq üçün) ola bilərdi. Göstəriş, bir qayda olaraq, komandanın ədədi kodundan (8 bit) və yaddaş xanasının ünvanından (12 bit) ibarət idi. Bütöv komanda IAS-ın maşın sözündən (20 bit) iki dəfə qısa idi.

Komanda	Ünvan
8 bit	12 bit

Beləliklə, IAS maşınının komandalarının sayı 2^8=256-dan çox deyil, ünvanlanan yaddaşın həcmi (onu riyazi ünvan fəzası da adlandırırlar) isə 2^{12} = 4096-dır və bu, yaddaşın real fiziki ölçüsünə uyğun gəlir, yəni IAS maşınının komandasında məzmunu əməliyyatda istifadə olunacaq yaddaş xanalarından birini göstərmək olar.

Toplama zamanı toplananlardan biri toplama əməliyyatı üçün ayrılmış xüsusi registrdə – akkumulyatorda, digəri isə hardasa yaddaşda yerləşdirilir. Fərz edək



IAS kompüterləri.

ki, akkumulyatorda yerləşən 28 ədədini və yaddaşın 342-ci xanasında yerləşən 33 ədədini toplamaq lazımdır. Onda 20 bitlik əmrdə "toplama" əməliyyatının ikilik kodu və ikinci toplananın 342-ci xanada yerləşən və ikilik təsviri 000101010110 olan ünvanı yazılacaq. Yaddaşda yerləşən iki ədədi toplamaq üçün əvvəlcə birinci ədədi registrə "yükləmək" (elə bil ki sıfır ilə toplama əməliyyatı aparılır), sonra isə artıq toplama komandasını yerinə yetirmək lazımdır, yəni iki komanda tələb olunur.

Komanda	Ünvan
Toplama	342

Yaddaş	Ünvan
	341
33	342
	343
	...

Prosessorun registrləri verilənləri və komandaların nəticələrini yadda saxlamaq üçün istifadə olunurdu. Komandanın arqument kimi icrası zamanı prosessor avtomatik olaraq tələb olunan registri alırdı. Başqa sözlə, registrin "ünvanı" komandanın özü tərəfindən təyin olunurdu.

IAS maşınının prosessoru *verilənləri emal edən blokdan* (hesab əməliyyatları və s.) və həmçinin proqramın yerinə yetirilməsini həyata keçirən *idarəedici qurğudan* ibarət idi. Prosessorun həm də *AC, MQ, DR, IBR, PC, IR* kimi "qəribə" adları olan tez işləyən registrləri var idi. Onlar göstərişlərin, ünvan və verilənlərin müvəqqəti saxlanması üçün nəzərdə tutulmuşdu. Maşının bütün qurğularının işini sinxronlaşdırmaq üçün elektron saatlardan istifadə edilirdi. Onlar *maşın taktları* adlanan bərabər zaman fasiləsindən sonra elektrik siqnalları verirdilər. Bir takt ərzində, məsələn, 40-bitlik sözü yaddaşdan 40 bitlik *DR* registrinə oxumaq (və ya yaddaşa yaz-

```
db
  əgər növbəti göstəriş göstəriş buferində (IBR) deyilsə
    onda yaddaşın (PC) ünvanından (DR) ünvanına oxu
    göstərişi (IR) göstəriş buferinə daxil et
    növbəti göstərişi (IBR) göstəriş registrinə daxil et
    (PC) komandalar sayğacını 1 vahid artır
  son
  komandanı icra et
ds
```

DR registrini həm də MBR (memory buffer register – "yaddaşın bufer registri") adlandırırlar, *AR* registri isə *MAR* (memory address register – "yaddaş ünvanının registri") adlanır.

maq) olardı, bu zaman *AR* registri maşın sözünün 12 bitlik ünvanına malik olurdu.

AC (akkumulyator) və *MQ* (vuruq) registrləri verilənləri və əməliyyatların nəticələrini müvəqqəti saxlamaq üçün istifadə olunurdu. Bu registrlər toplama, vurma və bölmə əməliyyatlarında istifadə edilirdi.

Bir maşın sözündə iki göstəriş yerləşdiyi üçün, yerinə yetirilmə zamanı onlar yaddaşdan eyni vaxtda yüklənirdi. Birinci göstəriş *IR* göstəriş registrinə yerləşməklə yerinə yetirilirdi. İkincisi isə "göstərişlər buferinə" – *IBR* registrinə yerləşdirilirdi və çox güman ki, növbəti addımda yerinə yetirilirdi. *PC* komandalar sayğacı (*ing.* "program counter" – "proqram sayğacı") proqramın növbəti göstərişinin ünvanını yaddaşda saxlayırdı.

IAS maşınında proqramın yerinə yetirilməsi kompüterin dövri olaraq iki addımı – hazırlıq addımını və əsas addımı yerinə yetirməsindən ibarət idi. Hazırlıq addımında göstərişin yüklənməsi, əsas addımda isə onun yerinə yetirilməsi baş verirdi.

IBR göstəriş buferində növbəti göstəriş olmadıqda IAS maşını *PC* komandalar sayğacını hər dəfə artırırdı. Hazırlıq mərhələsi bitdikdən sonra əsas addım gəlirdi və maşının bütün növbəti fəaliyyəti sonuncu göstərişin nəticələrindən asılı olurdu. Proqramda "başqa yerə" keçid göstərişinə rast gəlmək olardı ki, bu da komandalar sayğacının qiymətinin dəyişməsi demək idi.

Proqramın göstərişləri yaddaşda hansı ardıcıllıqla saxlanırdısa, sonradan həmin ardıcıllıqla da yerinə yetirilirdi. Beləliklə, IAS maşınında bütün növbəti rəqəmli maşınlara təsir edəcək aşağıdakı əsas prinsiplər istifadə olunurdu:

- hesab əməliyyatlarının yerinə yetirilməsi üçün hesab qurğusunun olması;
- proqramın və verilənlərin ümumi yaddaşda yerləşdirilməsi;
- proqramın icra edilməsinin standart dövrü;
- proqramın elementlərinin yaddaşda ardıcıl yerləşdirilməsi;
- registrlər (kiçik həcmli, sürətli yaddaş).

IAS maşını effektiv işləyirdi – vurma əməliyyatını 100 mik.s-yə, yaddaşa müraciəti (yaddaşdan oxuma və yaddaşa yazma) isə 50 mik.s-yə yerinə yetirirdi.

Con fon Neyman 1954-cü ildə alqoritmik dilin əsaslarını təklif etdi ki, o da çox sonralar populyar proqramlaşdırma dillərində öz tətbiqini tapdı. Dahi alim IBM firmasının məsləhətçisi kimi hesablama maşınlarının yaradılması işlərini davam etdirirdi. Onun IAS kompüterini birinci nəsil kompüterlərin əsas nümayəndəsi adlandırmaq olar.



Con fon Neyman.

CON VİNSENT ATANASOV

1971-ci ildə ABŞ Federal məhkəməsində elektron hesablama maşınına müəlliflik hüququnun müəyyən edilməsi üçün təhqiqata başlandı. Cavabdeh HONEYWELL firması – ENIAC maşınının patentinin sahibi idi. Bu prores iki il davam etdi və 1973-cü ildə Con Vinsent Atanasovun qələbəsilə başa çatdı.

Con Vinsent Atanasov 1903-cü ilin 4 oktyabrında ABŞ-ın şimalında, Nyu-York ətrafında bolqar emiqrantlarının ailəsində anadan olmuşdur (Atanasov soyadı imiqrasiya orqanları tərəfindən Atanasoff kimi dəyişdirilmişdi).

Riyaziyyata olan məhəbbəti ona məktəbdə riyaziyyatı tədris edən anası

Ayva Lyusena Pardidən keçmişdi. Atası isə oğluna elektrik işlərini öyrətmişdi: doqquz yaşlı oğlan uşağı evdə zədələnmiş elektrik məftillərini asanlıqla tapıb düzəldirdi.

1921-ci ildə Atanasov Florida ştatındakı universitetə daxil oldu və 1925-ci ildə oranı bitirib bakalavr dərəcəsi aldı. Onu bir sıra təhsil müəssisələrinə, hətta Harvarda müəllimlik etməyə dəvət etdilər, amma o, Ayova ştatının universitetini seçdi.

1929-cu ildə Atanasov Medisonda doktoranturaya daxil oldu və 1930-cu ildə helium atomlarının elektron quruluşuna həsr etdiyi doktorluq dissertasiyasını müdafiə etdi. Bu iş böyük həcmli hesablamalarla bağlı idi və bu onu hesablamaları avtomatlaşdırmaq üçün elektron qurğu yaratmağa sövq etdi. Tezliklə o, Ayova ştatının universitetində fizika və riyaziyyat üzrə professor-asistent vəzifəsini aldı. Con Atanasov buraya kompüter yaratmaq arzusu ilə qayıtmışdı. Orada o, elektron vakuum lampaları ilə sınaqlar keçirdi və riyazi hesablamalar aparmaq üçün o dövrdə məlum olan cihazları öyrəndi. Atanasov onları şərti olaraq analoq və "məxsusi hesablama maşınlarına", yəni rəqəmli maşınlara böldü ("rəqəmli" termini sonralar meydana gəldi). 1936-cı ildə Atanasov fizik Qlen Merfilə birgə həndəsi səthlərin analizi üçün kiçik analoq kalkulyatorunun modelini qurdu.

1939-cu ilin dekabrında Atanasov və onun asistenti Klifford Bərri xətti tənliklərin həlli üçün maşın nümunəsi



Rəqəmli maşın yaratmaq haqqında fikirlər Atanasovu heç bir dəqiqə də rahat buraxmırdı. Bir dəfə o, növbəti uğursuzluqdan sonra avtomobilə atılıb qəzəbli halda 300 km yol qət etmiş, sonra birdən adi yolüstü kafelərdən birinin qarşısında maşını saxlamışdı. Orada sakitləşəndən sonra çoxdan axtardığı həlli tapmışdı. Bu, rəqəmli maşının konsepsiyasının son variantı idi.

qurdular. Onlar bu maşını ABC (Atanasoff Berry Computer) adlandırdılar. Maşın 300 kq-dan ağır idi və o aşağıdakı tərkib hissələrindən ibarət idi:

- 30 ədəd 50 rəqəmli ikilik yaddaş sözündən ibarət olan, hər birinə müstəqil müraciət edilə bilən iki baraban. Yaddaş qurğusu enerjini avtomatik bərpa edən kondensatorlardan istifadə edirdi (indi buna DRAM – "dinamik yaddaş" deyilir);
- 300 elektron lampadan yığılmış və 60 hers takt tezliyilə ədədlərin mərtəbələr üzrə toplanması və çıxılmasını təmin edən (təxminən bir saniyədə bir toplama əməli) idarəetmə bloku;
- ikinci yaddaş kimi istifadə olunan, əl ilə qoyulan perfokartlar (perfokartlardakı deşiklər adi qaydada deşilmirdi, amma saniyədə 1500 bitə yaxın sürətlə, avtomobil şamlarında olduğu kimi, yandırılırdı. Bu texnologiya o qədər uğurlu idi ki, daxiletmə zamanı 100 min ədəddən yalnız birində səhv olurdu);
- İxtiranın patentləşdirilməsi cəhdləri nəticəsiz qaldı. ABC kompüterini nümayiş etdirəndən sonra Atanasov Ayova ştatı universitetindən yaratdığı kompüterə görə 650 dollar aldı. 1941-ci ilin yanvarında "Des Moines Tribune" qəzetində kiçik bir yazı çıxdı ki, "Ayova ştatı universitetində fizika professoru, doktor Con Atanasov iş prinsipinə görə insan beyninə bütün digər maşınlardan daha yaxın olan EHM yaradır". Amma o zaman buna yalnız mütəxəssislər diqqət yetirdilər (ENIAC maşınını quranda onun yaradıcıları Atanasovun ideyalarından istifadə etmişdilər). Lakin



Con Vinsent Atanasov.



Klifford Bərri.

◄ ABC kompüterinin bir hissəsi.

1996-cı ilin avqustunda Harri Slic və Con Erikson Atanasovun ideyalarından istifadə edərək ABC-2 kompüterini qurdular ki, bu da 5 məchullu 5 xətti tənlikdən ibarət tənliklər sisteminin həlli üçün nəzərdə tutulmuşdu. Onu yığmaq üçün onlar 500 vakuum lampasından, onlarla metr elektrik kabelindən və IBM firmasının köhnə perforatorundan istifadə etdilər.

İkinci dünya müharibəsinə görə bu maşın üzərində iş dayandırıldı.

Müharibə vaxtı Atanasov ABŞ Hərbi-dəniz qüvvələrinin Vaşinqton ştatındakı laboratoriyasının akustika bölməsinə rəhbərlik edirdi. Eyni zamanda o, Sakit okeanda atom bombasının sınaqdan çıxarılması layihəsinin də iştirakçısı olmuşdur. Atanasov 1948-ci ildə Ayova ştatının universitetinə döndükdə gördü ki, onun kompüterini söküblər, onun bütün çertyojları və diaqramları isə itib. Yalnız maşının bəzi hissələrini xilas etmək mümkün oldu.

Con Vinsent Atanasov 1995-ci ilin 15 iyununda ABC maşınına patent almadan vəfat etmişdir.

Atanasovun orijinal ideyaları universal EHM-in əsasını təşkil edən mühəndis həllərini qabaqlamışdı. XX əsrin 40-cı illərinin elektro-mexaniki maşınlarından fərqli olaraq Atanasovun kompüteri tamamilə elektron elementlərdən yığılmışdı.

İKİNCİ NƏSİL KOMPÜTERLƏR

Kompüterlərin ikinci nəsli təxminən 1955-ci ildən 1964-cü ilə qədər istehsal edilib. 1948-ci ildə AT&T Bell Laboratories (Bell Labs) amerika kompaniyasının mütəxəssisləri tranzistoru ixtira etdilər, onun müəllifləri Con Bardin, Uolter Bratteyn və Uilyam Şokli isə bu ixtira üçün patent aldılar. Bu ixtira dünyanı növbəti dəfə dəyişdirdi. Tranzistor EHM-in konstruksiyasından vakuum lampalarını tamamilə sıxışdırıb çıxartdı. O, on dəfələrlə az yer tuturdu, az miqdarda istilik ayırırdı, daha az enerji sərf edirdi və daha etibarlı işləyirdi. Vakuum lampa texnologiyasından tranzistorlara keçid ikinci nəsil maşınların xüsusiyyəti hesab olunur.

Maşınların yaddaşı da təkmilləşdi. Elektron-şüa boruları və ultrasəs yubatma xətləri tədricən ferrit nüvəcikləri və maqnit barabanları ilə əvəz edildi.

Prosessorda (hesablama maşınının ürəyində) da dəyişikliklər edildi. Komandaların da sayı artdı. Nəinki komandada göstərilən yaddaş ünvanlarını oxuyub yazmaq, həm də onları hesablamaq mümkün oldu, məsələn, xüsusi indeks registrinin qiymətini ədədlə toplamaq imkanı yarandı.



Tranzistor.

Giriş və çıxış əməliyyatlarını idarə etmək və informasiyanı ötürmək üçün xüsusi IO prosessorları (*ing.* "input-output" – "giriş-çıxış") işlənib hazırlandı. Onları mərkəzi prosessor heç nə ötürmədən idarə edirdi.

Məlum oldu ki, riyazi hesablamalar üçün dörd əsas hesab əməliyyatı (toplama, çıxma, vurma və bölmə) çatışmır, hansı ki, adətən yalnız tam ədədlərlə aparılırdı. Ona görə də həqiqi ədədlər üzərində effektiv hesablama əməliyyatlarının aparılması üçün əlavə olaraq xüsusi prosessorlar işlənib hazırlandı.

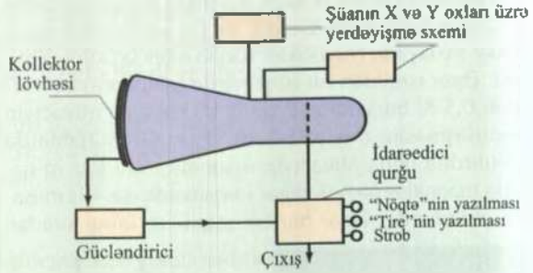
Prosesorun maşın dilində proqramlaşdırılması çox zəhmət tələb edən məsələdir. Bir EHM-də işləyən proqramları digər maşınlarda icra etmək üçün uyğunlaşdırmaq çətin idi, çünki EHM-lərin dilləri bir-birindən fərqlənirdi. Maşından asılı olmayan proqramlaşdırma dilləri yaratmaq lazım idi.

Yüksək səviyyəli proqramlaşdırma dilləri meydana gəldi: 1954-1957-ci illərdə IBM firmasında Con Bəkus tərəfindən yaradılan FORTRAN və Alqol dilləri (*ing.* Algorithmic Language). Qeyd edək ki, Algol-58 dilinin birinci versiyası FORTRAN dilinin sintaksisinin idcyalarına əsaslanırdı.

YADDA SAXLAYAN ELEKTRON-ŞÜA BORULARI VƏ ULTRASƏSLİ YUBATMA XƏTLƏRİ

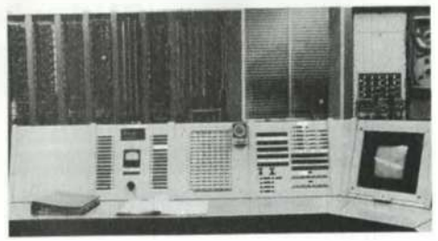
Birinci nəsil kompüterlərdə yadda saxlama mühiti kimi elektron-şüa borusunun ekranının lüminoforu çıxış edirdi. Yaddaşın ayrı-ayrı bitlərinə yazmaq və oxumaq elektron topun köməyilə yerinə yetirilirdi. Yazı zamanı 1 yazmaq üçün top ekranın böyük olmayan sahəsini tire şəklində, 0 yazmaq üçün isə nöqtə şəklində "yandırırdı". Saxlanacaq informasiyanı oxumaq üçün nazik ötürücü təbəqə kimi borunun ekranının üz hissəsində yerləşən köməkçi elektroddan istifadə olunurdu. Elektrodun şüası oxunacaq sahəyə (bitə) tuşlanırdı. Əgər həmin yerdə tire "işıqlanırdısa", onda köməkçi elektrodda müsbət impuls meydana gəlirdi, nöqtə işıqlanırdısa, onda elektrodun potensialı dəyişmirdi, yəni 1 və ya 0. Borunun yaddaşının həcmi çox az – 1024 və ya 2048 bit idi. Amma XX əsrin 50-60-cı illərində bu həcmdə yaddaş EHM üçün xarakterik olan yaddaş həcmi idi.





Elektron-şüa borularındakı yaddaşa alternativ – yubatmalı ultrasəs xətlərindəki yaddaş idi. Bu yaddaş növündə yadda saxlama mühiti elektrik ultrasəs impulsları idi. Onlar fasiləsiz olaraq civə ilə dolu olan borunun bir tərəfindən o biri tərəfinə doğru "səyahət" edirdilər. Bu proses bir sahildən digər sahilə qaçan dalgalara bənzəyirdi. Onu elektrik siqnalı hərəkətə gətirirdi . Dalğa borunun o biri tərəfinə çatanda buna oxşar bir siqnal əmələ gəlirdi. Beləliklə, siqnal ultrasəs yubatma xəttindən keçən zaman yadda saxlanırdı. Belə bir elementdə 400 mövqeli ikilik ədəd saxlana bilərdi.

İkinci nəslin ilk kompüterləri faktiki olaraq hibrid idilər, yəni həm yeni, həm də köhnə texnologiyalardan istifadə etməklə yaradılırdılar.



TX-O kompüteri.

Hesablama maşınlarının proqram təminatının işlənib hazırlanmasına başlandı. Bu, ilk növbədə *paket emalı* sistemlərinə (onda maşın məsələləri: məsələlərin daxil edilməsi, emalı, alınan nəticə paketə yığılırdı) aid edilirdi, onlar müasir əməliyyat sistemlərinin rüşeymləri idi. Tranzistoru element bazası kimi istifadə edən ilk kompüter eksperimental TX-O maşını (*ing.* Transistorized Experimantal Computer – "tranzistorlu eksperimental kompüter") idi. 1953-cü ildə TX-O maşınının istehsalı haqqında elan verdilər. Bu maşın Massaçusets Texnologiya İnstitutunda yaradıldı.

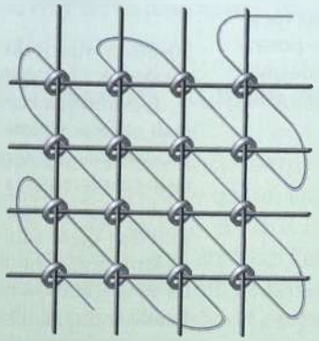
Elektron vakuum lampalı IBM 704 kompüteri indeks registrləri olan prosessora malik idi və sürüşkən vergüllü ədədlər üzərində əməliyyatları icra edirdi. Bu sıranın daha sonrakı modeli olan IBM 709-da giriş-çıxış prosessorları var idi ki, sonralar bunları *kanallar* və ya *kanal prosessorları* adlandırdılar. Onlar informasiya mübadiləsini həyata keçirirdilər, özlərini isə mərkəzi prosessor idarə edirdi və bu da giriş-çıxışın proqramlaşdırılması adlandırıldı.

FERRİT NÜVƏCİKLƏRİ VƏ MAQNİT BARABANLARI

Maqnitli (ferrit) nüvəcik kiçik diametrli halqadan ibarət idi. O asanlıqla maqnitləşirdi və bu vəziyyətini uzun müddət saxlayırdı. Nüvəciklər naqilə muncuq kimi düzülürdü. Əgər naqildən bir istqamətdə sabit cərəyan (təxminən 0,5 A) buraxılırdısa, onda bu cərəyan nüvəciyin maqnitləşməsini dəyişə bilirdi, əgər əks istiqamətdə buraxılırdısa onda nüvəciyin maqnitləşməsi (bir istiqamətdə maqnitləşmə – 0, digər istiqamətdə isə – 1) müşahidə olunmurdu. Əgər hər bir şaquli və üfüqi sıradan naqil keçirməklə nüvəciklər matrisini qursaq və ondan normanın yarısı qədər (0.25 A) cərəyan buraxsaq, onda şaquli və üfüqi xətlərin kəsişməsində yerləşən yalnız bir nüvəciyin maqnitləşməsini (0.25 + 0.25 = 0.5) dəyişmək olar. Digər nüvəciklər isə, cərəyan 0.5 A-dan kiçik olduğu üçün heç bir reaksiya verməyəcək. Nüvəciyin qiymətini müəyyən etmək üçün onu 0 (sıfır) vəziyyətinə (bir istiqamətli maqnitləşmə) keçirirlər. Əgər o, artıq bu vəziyyətdədirsə, onda heç nə baş verməyəcək. Digər vəziyyətə keçid, matrisin bütün nüvəciklərindən keçən əlavə naqildə (oxuma naqili) cərəyan əmələ gəlməsinə səbəb olacaq.



Maqnit nüvəciklərində yığıcı matris.



Ferrit nüvəcikləri matrisinin böyüdülmüş fraqmenti.

Maqnit barabanları maqnitsiz barabanların səthinə nazik təbəqəli (0,025 mm) maqnit materialının çəkilməsilə alınır. İkilik ədədlər daimi bucaq sürətilə fırlanan barabana kiçik sahələrin maqnitləşməsi yolu ilə yazılır və bu, həmin sahələr oxuma-yazma başlığının yanından keçərkən baş verir. Yazılmış ədəd barabanın bir dövrəsindən sonra oxuna bilər. Diametri 15 sm, uzunluğu 30 sm olan və dəqiqədə 6000 dövrə sürətlə fırlanan maqnit barabanı üzərində 1 sm²-də 60 rəqəm yerləşməklə 4 Kbayta qədər informasiya yazmaq olar.

IBM 7094

IBM-in ikinci nəslinə aid olan – IBM 7094 elmi kompüteri tamamilə tranzistorlar əsasında qurulmuşdu. XX əsrin 60-cı illərində o, bazarda olan ən sürətli kompüterlərdən idi – sürüşkən vergüllü ədədlər üzərində saniyədə 350 min əməliyyat yerinə yetirirdi. Maşının qiyməti təxmiən 3,5 mln. ABŞ dolları idi. Standart IBM 7094 maşını ferrit nüvəciklərində yerləşən 32×1024=32768 dənə 36-bitlik yaddaş sözünə malik idi və hər bir nüvəcik bir bit informasiya yadda saxlayırdı.



IBM 7094 kompüteri.

Xarici yaddaş kimi IBM 2302 modelinin bərk maqnit diskləri üzərindəki yığıcıdan istifadə edilirdi. 24 düym diametri olan diskin bərk maqnit lövhəcikləri vakuumda fırlanırdı. Maqnit başlıqları informasiyanı diskin maqnit səthinə yazır və oradan oxuyurdular (eyni qayda ilə səsi maqnit kasetinə yazıb və oxuyurlar). (Bu sıranın son modellərindən olan IBM 2302, 200 Mbayt-dan çox informasiya yadda saxlaya bilirdi). IBM 7094-ün prosessoru IAS maşınının prosessorundan indeks registrlərinə və həqiqi ədədlər üzərində (sürüşkən və qeyd olunmuş vergüllü) bütün əməliyyatları yerinə yetirən hesab qurğusuna malik olması ilə fərqlənirdi. Kanal prosessorları yaddaşa birbaşa müraciəti həyata



Maqnit disk qurğusu.

keçirirdilər. Mərkəzi prosessor və kanallar eyni zamanda yaddaşa müraciət edə bilmədikləri üçün prosessorlar arasında yaddaşa müraciəti paylayan xüsusi idarəedici qurğu nəzərdə tutulmuşdu.

IBM 7094-ün prosessorunun registrlərinin əksəriyyəti IAS maşınının prosessorunda olduğu kimi idi, məsələn:

- *IC* (*ing.* instruction counter) – göstərişlər sayğacı, *PC* komandasının analoqu idi;
- *AR* (*ing.* address register) – ünvan registridir və göstərişdə olan yaddaşın ünvanını saxlayır;
- *SR* (*ing.* storage register) – bufer registri, *DR* və *IBR* registrlərinin analoqudur;
- *RI* (*ing.* register instruction) – göstəriş registridir və cari əməliyyat kodunu saxlayır.

Bir takt ərzində prosessor yaddaşdan iki komanda-göstəriş yükləyirdi – ikinci göstəriş *SR* registrində yadda saxlanırdı, o proqramlardan gizli idi və xidməti təyinata malik idi.

İNDEKS REGİSTRLƏRİ

Adətən komandada yaddaş xanasının ünvanı göstərilir (verilənləri hara qoymaq və haradan götürmək lazımdır). Lakin, əgər 10 elementdən ibarət olan bir neçə cədvəli (massivi) sıfırlarla doldurmaq lazımdırsa, cədvəlin hər bir xanasının ünvanı aydın şəkildə göstərilməklə 10 eyni əməliyyat yerinə yetirilir:

0 ədədini 101 saylı xanaya qoymalı
0 ədədini 102 saylı xanaya qoymalı
...
0 ədədini 110 saylı xanaya qoymalı

Əgər prosessorda indeks registrləri yoxdursa, onda bunun başqa yolu qalmır. Lakin indeks registri, ədədi və indeks registrində yerləşən qiyməti toplamaqla xananın ünvanını hesablamağa imkan verir. Gəlin belə bir dövr quraq:

```
db 1 indeks registri üçün 1 dən 10 dək
     0-ı ünvan üzrə xanaya qoymalı
     indeks registri + 100
ds
```

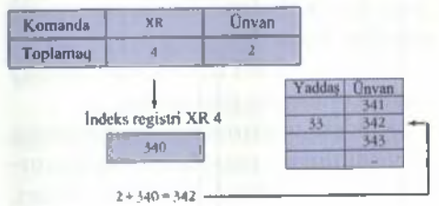
IBM 7094 maşınında yeddi indeks registri (XR) var idi ki, onların da hər biri 15 bitlik ünvanı yadda saxlaya bilirdi.

Komanda	Əlamət	Ünvan
18 bit	3 bit	15 bit

Ünvanı hesablayarkən hansı indeks registrindən istifadə olunduğunu prosessorun bilməsi üçün komandada 3 bit (2^3=8) yer ayrılmışdı. Bu da *ünvanlama əlamətlərini* – 0-dan 7-yə qədər olan ədədləri kodlaşdırmağa imkan verirdi. 1 ədədi birinci, 2 ədədi ikinci, ..., 7 ədədi isə 7-ci registrə uyğun gəlirdi. Əgər ünvanlama əlamətində 0 dururdusa, onda bu adi ünvanlamadır, yəni indeksli registrilə cəmləmə tələb olunmur.

Məsələn, komandada 4-cü indeks registri istifadə olunur (bu zaman 3 bitlik əlamət ikilik sistemdə yazılmış 100_2-ə və ya onluqda yazılmış 4_{10}-a bərabər olur). Bu zaman 4-cü indeks registrinin qiyməti göstərişdə yerləşən ünvan ilə (15 bit) toplanır. Bu da yaddaşdakı ünvandır.

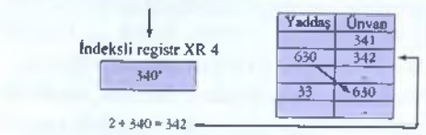
Aşağıda akkumulyatorda yerləşən 28 ədədilə 342-ci yaddaş sözündə yerləşən 33 ədədinin toplanması nümunəsi göstərilmişdir.



DOLAYI ÜNVANLAMA

Bundan əvvəlki misal komandalar üçün lazım olan verilənlərin göstərilən ünvanda yerləşdiyi *birbaşa ünvanlamaya* aid idi. Bizim *dolayı ünvanlama* misalında (bunu "*" simvolu göstərir) isə 342-ci sözdə *arqumentin ünvanı* yerləşir. Bu ona bənzəyir ki, siz dostunuz Kamranın evinə gedib onun özünü deyil, Nicatın yanında olmasını bildirən məktubu tapırsız. Əgər o, doğrudan da Nicatın evindədirsə, onda bu birinci səviyyəli dolayı ünvanlamadır. Əgər Nicatgildə də onların Kamranla birlikdə Kənangildə olmaları barədə belə bir məktub varsa, onda bu ikinci səviyyəli dolayı ünvanlamadır və s. Real olaraq prosessorlarda üç səviyyəli dolayı ünvanlama istifadə olunur.

Komanda	XR	Ünvan
Toplamaq	4	2



İndeksli və dolayı ünvanlama massiv və cədvəllərlə iş zamanı əvəzedilməzdir.

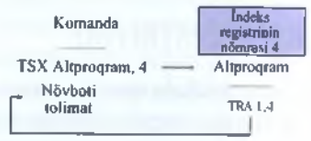
Bütövlükdə IBM 7094 maşınında 200-dən artıq komanda var ki, onları da bir neçə sinfə bölmək olar:

- Prosessorla yaddaş və ya prosessorun iki registri arasında informasiya mübadiləsini təmin edən verilənlərin göndərilməsi (ötürülməsi) göstərişləri;
- hesab göstərişləri: toplama, çıxma və s. (o cümlədən, sürüşkən vergüllü ədədlər üzərində);
- VƏ YA, DEYİL və s. tipli məntiqi (ədədi olmayan) göstərişlər;
- indeks registrlərinin qiymətlərinin dəyişdirilməsi, məsələn, 1 vahid artırmalı və ya azaltmalı və s. göstərişləri;
- şərti və şərtsiz keçid göstərişi (idarəetmənin ötürülməsi): proqramın və ya dövrlərin yerinə yetirilmə ardıcıllığının dəyişdirilməsi;
- giriş-çıxış qurğuları və kompüterin yaddaşı arasında verilənlərin ötürülməsi üçün giriş-çıxış əməliyyatları.

ALTPROQRAMLAR

İkinci nəsil maşınların mühüm xüsusiyyətlərindən biri, onlarda altproqramlara (köməkçi alqoritmlərə) keçid və onlardan qayıtma üçün xüsusi göstərişlərin olmasıdır. IBM 7094 maşınında altproqramın çağırışı üçün xüsusi TSX göstərişi və bunun əksi (altproqramdan qayıtma) üçün isə TRA göstərişindən istifadə olunurdu.

TSX komandasının yerinə yetirilməsi zamanı arqument qismində qəbul olunan indeks registrində cari göstərişin ünvanı (qayıtma ünvanı) yadda saxlanılırdı. Sonra altproqramın icrasına başlanırdı, yəni idarəetmənin ötürülməsi baş verirdi. Altproqramın icrası qurtardıqdan sonra orada, arqument qismində TSX komandasında olduğu kimi elə həmin indeks registrinin yerləşdiyi TRA göstərişi yerləşməli idi. Bu zaman proqrama qayıtma baş verirdi.



GİRİŞ-ÇIXIŞ

Kanal prosessorları (kanallar) xüsusi proqramları icra edirlər – bu giriş-çıxış proqramlarıdır. Mərkəzi prosessor kanalı işə salır və giriş-çıxış prosesinə "nəzarət" edir, yəni onun monitorinqini aparır. Mərkəzi prosessor giriş-çıxış prosesində özü iştirak etməsə də, o, kəsilmə üçün sorğu verə bilər, yəni cari işi dayandıra bilər. Adətən, bu giriş-çıxış əməliyyatları bitəndən sonra baş verir. İkinci nəsil kompüterlərin bəzilərində kəsilmə aparatının olması prosessorların məhsuldarlığını xeyli artırdı.

Hər bir kanal prosessoru eyni zamanda onlarla maqnit lentini, çap qurğusunu, perfokartdan giriş qurğusunu və hətta sərt maqnit diskli disk qurğusunu da idarə edə bilər. Mərkəzi prosessor kimi kanal da oxşar gizli registrlər dəstinə malikdir. Təbii ki, onun hesab əməlləri yerinə yetirmək üçün (akkumulyator və s.) adi registrləri və indeks registrləri yoxdur. Kanal prosessorları giriş-çıxışın unifikasiyasına xidmət edir.

IBM 7094 maşınında kanal prosessoru ilə əsas yaddaş arasında verilənlər 36-bitlik sözlərlə ötürülürdü, kanallar ilə giriş-çıxış qurğuları, məsələn, maqnit lenti arasında 6 bit ilə ötürülürdü. Buna görə də kanallar daxil olan verilənləri (6 bit olmaqla) 36-bitlik sözlərə yığmağı həyata keçirirdilər. Verilənləri yığmaq-paylamaq üçün bufer rolunu 36-bitlik *verilənlər registri* yerinə yetirirdi. Yaddaş ilə kanal arasında giriş-çıxış prosesi zamanı ötürülən sözlərin sayı sayğac-registrdə yadda saxlanırdı. Giriş-çıxış vəziyyəti status registrində saxlanılırdı. Əgər giriş-çıxış zamanı səhvlər baş verərdisə, onda mərkəzi prosessor bu haqda status registrində saxlanan koda görə məlumat ala bilərdi.



Giriş-çıxış prosesi aşağıdakı kimi baş verirdi:

1. Mərkəzi prosessor giriş-çıxış göstərişini qəbul edir. Ona əsasən giriş-çıxış proqramının harada olduğunu təyin edir.

KƏSİLMƏ APARATI

Sinifdə 20 şagird yoxlama yazı işini yerinə yetirir (şagirdlər – kanallardır, müəllim – mərkəzi prosessordur, yoxlama yazı işləri – kanalların proqramları və yaxud göstərişlərdir). Müəllim yoxlama yazı işinin qurtarmasını gözləyir və eyni zamanda jurnalı doldurur. Kəsilmə aparatının olmadığı halda müəllim bərabər zaman intervalından sonra jurnalın doldurulmasını dayandırır və hər bir şagirddən yoxlama işini bitirib-bitirməməsini soruşur. Suala müsbət cavab verən şagirdlər (öz proqramlarını icra edib qurtarıblar) dəftərlərini (giriş-çıxış verilənləri yaddaşda yerləşdirilib) müəllimə verirlər. Yerdə qalanlar yoxlama yazı işlərinin icrasını davam etdirirlər. Aydındır ki, bu halda müəllim şagirdlərin sorğusuna çox vaxt sərf edir.

Əksinə, kəsilmə mexanizminin mövcud olduğu şəraitdə müəllim şagirdlərə yoxlama yazı işləri verir və onlardan xahiş edir ki, işini bitirənlər əlini qaldırmaqla onu çağırsınlar. Bu halda o, jurnalı doldurur və heç kimdən də heç nə soruşmur. Hər hansı bir şagird işini bitirdikdə əlini qaldırmaqla siqnal verir (hansısa giriş-çıxış prosesi yekunlaşıb). Müəllim jurnalın doldurulmasını dayandırır və şagirdin yoxlama yazı işilə məşğul olur.



BÖYÜK SİSTEMLƏR

İlkin mərhələlərdə bütün proqramlar avtonom işləyirdi və yeni bir proqramı əl üsulu ilə kökləmək üçün hər dəfə kompüteri saxlamaq lazım gəlirdi. İkinci nəsil maşınlarda tapşırıqlar paketini – proqramlar dəstini hazırlayıb, bir yerdə saxlayıb, məsələn, maqnit lentinə yazıb, sonradan kompüterdə emal etmək mümkün idi. Bu zaman nəticə başqa maqnit lentinə yazılırdı. Bü cür emal üsulu *emalın paket metodu* adlandırıldı. Tapşırıqlar yazılmış lenti hazırlamaq üçün əlavə "böyük olmayan" köməkçi kompüterdən istifadə etməyə başladılar.

Emalın paket metodu daim kompüterin yaddaşında saxlanılan xüsusi monitor-proqramın olmasını tələb edirdi ki, o da tapşırıqlar axınını idarə edirdi. İdarəetmə multiproqramlaşma rejimində aparılırdı: prosessor müvəqqəti olaraq cari proqramın icrasını dayandırırdı, növbəti proqramın icrasına başlayırdı və sonra isə icrası kəsilmiş proqramın işini bərpa edirdi. Bu, kompüterin işinin effektivliyini artırırdı, xüsusilə də cari məsələ giriş-çıxışın bitməsini gözlədikdə və onun işi müvəqqəti olaraq kəsildikdə, mərkəzi prosessor boş dayandığı üçün, digər proqramın yerinə yetirilməsi mümkün olurdu. Multiproqramlaşdırma növbəti nəsil kompüterlərdə özünün sonrakı inkişafını tapdı.

2. Mərkəzi prosessor parametr kimi giriş-çıxış qurğusunun adını ötürür və proqramın icrasına başlayır.

3. Kanal prosessoru proqramı icra edir və fəaliyyət qurtardıqda (doğru və ya səhv ilə) status registrində başa çatma kodunu yadda saxlayır. Sonra giriş-çıxışın yerinə yetirilməsini bildirərək mərkəzi prosessora kəsilmə haqqında xəbər göndərir.

Giriş-çıxış komandaları 3 qrupa bölünür:

• giriş-çıxış qurğularını idarə edən göstərişlər.

Onlara aid olan göstərişlər bunlardır: maqnit lentinin geri fırlanması komandaları və ya oxuma/yazmadan öncə maqnit başlığının lazımi mövqeyə keçirilməsi və s.

• verilənlərin ötürülməsi göstərişləri.

Onlar "giriş-çıxış qurğusu ilə əsas yaddaş arasında *n* sözü ötürün" şəklindədir. Hər bir belə göstəriş sayğacda (sayğac-registrdə) saxlanılacaq *n* sayda sözdən və əsas yaddaşın ünvanından (verilənlər haradan və ya haraya daxil olur) ibarətdir.

Bundan sonra verilənlərin ötürülməsi baş verir:

```
db nə qədər ki sayğac > 0
|     sözü kanal ilə yaddaş arasında
|     ötürməli (ünvan registri)
|     sayğac := sayğac – 1
|     ünvan registri := ünvan registri + 1
ds
```

• idarəetmə göstərişləri və ya şərti və şərtsiz keçid göstərişləri.

Kanallar və mərkəzi prosessor öz aralarında kompüterin əsas yaddaşını paylaşırlar. İdarəetmə xüsusi qurğu və rabitə xətləri dəsti (naqillər) ilə icra olunur və bu naqillər eyni zamanda prosessora və kanallara birləşir. Əlaqə xətləri dəsti *sistem şini* adlanır. Giriş-çıxış əməliyyatlarının sürəti mərkəzi prosessorun yerinə yetirdiyi verilənlərin emalı sürətindən geri qalır. Buna görə də, yaddaş əsasən kanallara deyil, prosessora aid olur.

1965-ci il üçün, kifayət qədər baha olmasına baxmayaraq, 300-ə yaxın IBM 7090 və IBM 7094 kompüterləri müxtəlif kampaniyalarda qurulmuşdu.

ABŞ Silahlı qüvvələri ötən əsrin 80-ci illərinin ortalarında 30 il fasiləsiz işlədikdən sonra IBM 7094 maşınlarından birini "növbətçilikdən azad etmişdi".

KOMPÜTERLƏRİN ÜÇÜNCÜ NƏSLİ

1965-ci il üçüncü nəsil kompüterlərin meydana gəlməsi ili hesab olunur. İkinci və üçüncü nəsil kompüterləri arasındakı fərq birinci və ikinci nəsil arasındakı fərqlə müqayisədə o qədər də böyük görünmür. Hesablama maşınlarının istehsalı çox zəhmət tələb edən və bahalı iş idi. EHM-in əsas elementi olan çoxlu sayda tranzistorları bir yerə yığmaq lazım idi, saysız-hesabsız naqillər çoxluğu kompüterin blok və hissələrini əhatə edirdi. EHM-in mürəkkəbliyi günbəgün artırdı və bu da tranzistorlar sayının artmasına, montaj və layihələndirmə işlərində çətinliklərə gətirib çıxarırdı.

İnteqral sxemlərin meydana gəlməsi üçüncü nəsil EHM-lərin yaradılması texnologiyasında inqilab oldu. Bunun əsasında ilk dəfə 1958-1961-ci illərdə amerikalı mütəxəssislər tərəfindən irəli sürülmüş və yoxlanmış ideyalar dururdu.

Cek Kilbi elektron sxemin komponentlərini (tranzistorlar, kondensatorlar və rezistorlar) yarımkeçiricinin tam bir parçasına yığdı. Və beləliklə də, Texas Instruments şirkətinin istehsal etdiyi birinci inteqral sxem meydana gəldi. Kontaktlar sxemin konstruksiyasına mum vasitəsilə birləşdirilirdi. Bu, xüsusilə də istehsalda, o qədər də rahat və etibarlı deyildi və buna görə texnologiya öz sonrakı inkişafını tapmadı.

Cin Herni və Kurt Lehovets inteqral sxemlərin hazırlanması texnologiyasını təkmilləşdirdilər və elementlərin biribirindən izolə edilməsi sahəsində xüsusi olaraq böyük uğur qazandılar (izolyator kimi silisium oksidinin nazik təbəqəsindən istifadə olunurdu).

Robert Noys inteqral sxemin bütün elementlərinin çıxışlarını metalın tozlandırılmasının köməyilə bir üst səthdə yerləşdirməyə imkan verən üsul icad etdi. Bu isə gələcək sxemin laylarını – "şəkli" addımbaaddım çəkilməklə inteqral sxemlərin yüzlərlə kristallarından qurulan silisiumlu lövhəni tam şəkildə emal etməyə imkan verdi. Bundan sonra bu lövhəni kristallara "bölmək", plastik kütləyə bükmək və ona "ayaqcıqlar" lehimləmək lazım idi və nəticədə inteqral sxem hazır olurdu. Buna görə də bu texnologiyanı *planar*, yəni "müstəvi" texnologiyası adlandırdılar.

1954-cü ildə Qordon Til tranzistorları baha olmayan materialdan – adi silisiumdan hazırlamağı təklif etdi ki, bu da inteqral sxemlərin istehsalını ucuzlaşdırmağa imkan verdi.

Daha bir aparıcı ideya ondan ibarət oldu ki, inteqral sxemin bütün komponentləri – diodlar, rezistorlar və ya kondesatorlar ən mürəkkəb element olan tranzistorun hissələri kimi layihələndirilməyə başladı. Elə o vaxtdan inteqral sxemdə elementlərin sayından danışıldıqda, tranzistorların sayı nəzərdə tutulur.

İnteqral sxemin böyük olmayan həcminə yüzlərlə, minlərlə tranzistor yerləşdirmək mümkün oldu və bunun nəticəsində EHM-in ölçüləri kiçildi, montaj işləri sadələşdi. Buna görə də, inteqral sxemin yarımkeçirici yaddaşı ferrit nüvəcikli böyük ölçülü yaddaşı əvəz etdi.

Lakin bütün nailiyyətlər yalnız inteqral sxemlər ilə bağlı deyildi. Eyni zamanda daha bir yaddaş növü – DYQ (daimi yaddaş qurğusu – *ing.* Read Only Memory) meydana gəldi. DYQ yalnız verilənləri oxumaq imkanı verirdi. Bununla kompüter yaddaşını ucuzlaşdırmaqla yanaşı, həm də istehsal zamanı EHM-in bir çox vacib proqramlarını DYQ-yə yerləşdirməklə onların mühafizəsi problemini həll etmək mümkün oldu. Həm yazı üçün, həm də oxumaq üçün yararlı olan adi yaddaş OYQ (operativ yaddaş qurğusu – *ing.* RAM – Random Access Memory) adlandırıldı.

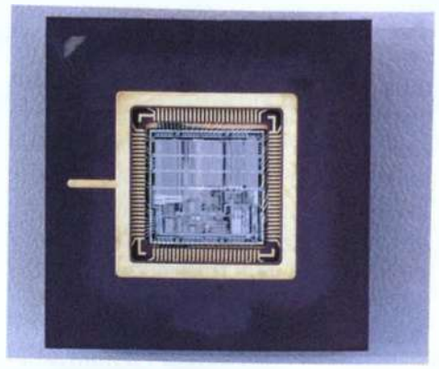


Cek Kilbi.



Robert Noys.

İnteqral sxemlər: mikroprosessor (solda), DYQ (sağda).





MİKROPROQRAMLAŞDIRMA

Mikroproqramlaşdırma texnikası (sadə komandalardan – mikrokomandalardan prosessorun mürəkkəb komandalarının qurulması) prosessorların layihələndirilməsində istifadə olunmağa başladı. Prosessorun öz komandaları mikroproqramlar (mikrokomandalardan ibarət olan proqramlar) kimi realizə olunurdu. Beləliklə, proqramlaşdırma prosessorun daxilinə "endi". İndi artıq alqoritmin bəzi göstəricilərinin dəyişilməsilə bütün proqramı yenidən yazmaq lazım gəlmirdi. Bu məqam, yəni proqramın "çevik" olması prosessorun da "çevik" olmasını şərtləndirirdi. Lakin mikrokomandalar bir qayda olaraq, kənar proqramlar üçün (prosessora nəzərən)



Moris Vilkes.

əlçatmaz idi, onlar yalnız layihələndirmə zamanı "görünürdülər".

1951-ci ildə Kembric universitetinin kompüter laboratoriyasının rəhbəri Moris Vilkes mikroproqramlaşdırmanın əsas konsepsiyalarını xülasə etdi:

• Prosessorun hər bir göstərişi (mikroproqram) mikrogöstərişlər ardıcıllığından ibarətdir və idarəedici yaddaş adlanan xüsusi yaddaşda saxlanılır.

• Mikrogöstəriş prosessorun göstərişlərinin yerinə yetirilməsi üçün zəruri olan bir sıra *mikroəməliyyatları* (məsələn, registrlər arası verilənlərin "ötürülməsi") icra edir.

• Prosessorun əsas yaddaşında yerləşən hər bir prosessor göstərişinin icrası idarəedici yaddaşdan mikrogöstərişlərin ardıcıl icrasını təmin edir.

Mikroproqramlaşdırma prosessorun layihələndirilməsinin daha sadə üsulunu təmin edir. "Mikroproqramlaşdırılmış" prosessorun göstərişlər dəstini yalnız idarəedici yaddaşda yerləşən *mikrokodun* əvəz edilməsilə dəyişmək olar. Bu məqam belə prosessoru digər maşının komandalar sisteminə kökləmiş olur və proqramı "yad" maşınla yerinə yetirməyə imkan verir.

Birinci və ikinci nəsil kompüterlərin layihələndirilməsində mikroproqramlaşdırmadan istifadə cəhdləri olsa da, o, yalnız XX əsrin 60-cı illərində IBM system/360 modelinin meydana çıxması ilə geniş tətbiq olunmağa başladı.

IBM SYSTEM/360

Üçüncü nəsil kompüterlər, modellərinin çoxluğu və istehsal firmalarının sayı ilə fərqlənirlər. Üçüncü nəsil kompüterlər arasında ən məşhurları IBM istehsalı olan kompüterlər sırasından 360 modelidir (*ing.* System/360). Bu ailə hesablama texnikasının tətbiqlərinin geniş spektrini əhatə etməyə cəhd edirdi. Müxtəlif modellər əhəmiyyətli dərəcədə uyuşan idi və buna görə də artıq proqram təminatının mobilliyindən danışmaq olardı. Yəni 360 sırasının bir modeli üçün yazılmış proqram (praktik olaraq dəyişikliksiz) bu sıranın digər modellərində də işləməli idi. Əlbəttə, yerinə yetirilmə müddəti fərqli idi və yaddaşda yerin çatışmaması ilə bağlı müəyyən çətinliklər yarana bilərdi.

Bütün IBM 360 sırası istehsal edilən kompüterlərin standartlarına kifayət qədər güclü təsir göstərdi.

IBM 360 prosessorunun ən sadə göstərişləri bir yarımsözdən, yəni 16 bitdən ibarət idi.

Komanda	Ünvan 1	Ünvan 2
8 bit	4 bit	4 bit

Burada ünvan 1 və ünvan 2 prosessorun registrlərinin adıdır. IBM 360 göstərişi IBM 7094 göstərişindən fərqli olaraq ikiünvanlıdır. Yaddaşda olan ünvanlar bir qədər mürəkkəb verilirlər. 16,7 mln baytdan istənilən birini ünvanlamaq mümkün olsa da, ünvan 24-bitlikdir (2^{24}=16 777 216), qiymətinin baha olmasına baxmayaraq, IBM 360-ın yaddaşı adətən, 32 kbaytı aşmırdı. Beləliklə, ünvanlama üçün yalnız 16 bit (2^{16}=65536) tələb olunurdu. Yaddaşa qənaət məqsədilə baza registri adlanan registrdən istifadə olunurdu.

Komanda	Ünvan 1	İndeks	Baza	Sürüşmə
8 bit	4 bit	4 bit	4 bit	12 bit



IBM 360.

Ən sadə halda icraedici ünvan – göstərişin icrasında iştirak edən ünvan baza registrilə komandada göstərilən 12-bitlik ünvanın (şürüşmənin) cəminə bərabərdir, yəni göstərişin uzunluğu 32 bitdir.

IBM 360 maşınında 16 ədəd 32-bitlik registr (ümumi təyinatlı) var idi ki, bunlar da indeks və baza registrlərinin cəmləyicisi kimi istifadə olunurdu. Bundan əlavə IBM 360-da hər biri ikiqat sözdən ibarət olan 4 ədəd sürüşkən vergüllü registr də var idi.

IBM 360 kompüteri üçüncü nəsil kompüterlərə aid olsa da, onlarda bir qayda olaraq köhnə ferrit nüvəcikli yaddaşdan istifadə olunurdu. Nüvəciklər kiçik ölçüdə idilər: daxili diametri – 0,48 mm, xarici diametri – 0,76 mm. IBM kompaniyası 8 192 və 284 912 ölçülü nüvəciklər matrisi istehsal edirdi.

1960-cı ildən IBM firması istifadəçilərə minlərlə tranzistorlu kompüterlər təklif etsə də, yalnız 360 sırasının meydana gəlməsilə firma öz böyük uğurunu bayram edə bildi. İndi artıq yaddaşın 32-bitlik sözünü istənilən baytdan başlayaraq ünvanlamaq olardı.

IBM yeni maşınlarında öz prosessorlarını quraşdırmaq üçün ilk dəfə mikroproqramlaşdırmadan istifadə etdi. Beləliklə, 360 sırasının hər biri eyni periferiyadan (xarici qurğulardan) və eyni proqram təminatından istifadə edə bilən 6 modelini yaratmaq mümkün oldu. 360 sırasının kompüterlərinə kənardan daxilolma həyata keçirildi (məsələn, telefon xətlərilə). Bundan sonra hesablama məsələlərini icra etmək və nəticə-

ləri almaq üçün kompüter zalına getmək məcburiyyəti qalmadı.

IBM 360 kompüterlər sırasını istehsal etmək üçün 60 min fəhlənin cəlb olunduğu 5 zavodun tikintisinə 5 milyard dollar vəsait qoymaq lazım gəldi. Məqsəd özünü doğrultdu. XX əsrin 70-ci illərində IBM 360 maşınlarına hər yerdə: xəstəxana və universitetlərdə, bank və korporasiyalarda, kitabxana və laboratoriyalarda rast gəlmək olardı.

ƏMƏLİYYAT SİSTEMLƏRİ

Kompüterdə eyni zamanda bir neçə prosesin paralel icra edilməsi kifayət qədər mürəkkəb proqramın olmasını tələb edir. Bu proqram onları idarə edir və kompüterin resurslarını əməliyyatlar arasında paylaşır, onları növbəyə qoyur, müxtəlif proqram-proseslər ilə qarşılıqlı əlaqədə olur və s. Burroughs (ABŞ) firmasının əməkdaşları onu idarəedici usta proqram adlandırdılar (adətən, onu *əməliyyat sistemi* adlandırırlar). Böyük kompüterin əməliyyat sistemi proqramlar kompleksidir, lakin bəzi funksiyalar



Atlas kompüteri.

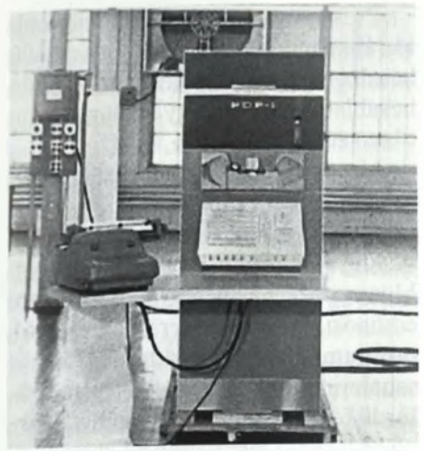
aparat vasitəsilə də icra oluna bilər. Bu cür sistemlərin geniş yayılması üçüncü nəsil EHM-lərin vacib xüsusiyyətidir. Əməliyyat sistemlərinin inkişafı 50-ci illərdə paket əməliyyat sistemlərinin (monitor proqramların) meydana gəlməsindən sonra başladı.

Atlas kompüteri (Mançester universiteti, 1961-ci il) bu tip sistemlərdən birinə malik idi. Bir neçə istifadəçinin eyni zamanda, praktiki olaraq bir-birini görmədən, bir kompüterdə işləməsini təmin edən vaxtın bölünməsi sistemlərinin layihələndirilməsinə 60-cı illərdə başladılar. *Multiproqramlaşdırma* və *multiemal* adətən, kompüterin əsas yaddaşını paylaşan proqramların paralel icrasından ibarətdir (dəyərinin baha olması səbəbindən əsas yaddaşın həcmi məhduddur).

Onun həcmi, adətən bütün yerinə yetirilən proqramların verilənlərilə birlikdə saxlanması üçün kifayət etmir. Buna görə də proqramlar üçün yaddaşı dinamik olaraq, "iş prosesində" ayırmaq lazımdır. Yaddaş ayrılmamış proqramlar müvəqqəti olaraq xarici yaddaşda yerləşməlidir (məsələn, diskdə). Avtomatik olaraq (əməliyyat sisteminin idarəsilə) proqramlar xarici yaddaşdan yüklənir və işini sona çatdırmış proqramlar operativ yaddaşdan diskə çıxarılır. Bu məqam üçüncü nəsil EHM-lərin əməliyyat sistemlərinin əsas funksiyalarından biri idi.

MİNİ KOMPÜTERLƏR

Böyük EHM-lərin fonunda mini kompüterlər adlanan kiçik ucuz kompüterlərin istehsalının başlanması bir təzad təşkil edirdi. Onlar 8 bitdən 32 bitədək "kiçik" maşın sözü, məhdud aparat və proqram vasitələrilə, həmçinin böyük olmayan ölçülərilə xarakterizə olunurdu. Mini kompüterlərin ucuz olması onların geniş yayılmasını şərtləndirdi. Məsələn, onlardan sənaye kontrollerləri kimi istifadə olunurdu, yəni bu maşınlarla elə dəzgahları və yaxud sadə istehsal prosesləri üçün ayrıca sexləri idarə edirdilər ki, bunlar üçün də çoxməqsədli "ağır" kompüterlərin istifadə edilməsi mənasız idi. Mini-kompüterlər sürətinə görə



PDP-1 kompüteri.

ikinci nəsil böyük kompüterlərilə müqayisə oluna bilərdi və birinci nəsil kompüterlərin də ən yaxşı göstəricilərini üstələyirdi. Mini kompüterlərin nümayəndələrindən biri də DEC firmasının PDP-1 maşını idi (*ing.* Programmed Data Prosessor). PDP-1 bir çox xarici qurğuları idarə edə bilirdi. Maşının qurulması nə elektrik təchizatının yenidən layihələndirilməsini (ABŞ-da bu, 110 volt idi), nə başqa kondisionerləşmə sistemini, nə də döşəmənin gücləndirilməsini tələb etmirdi: kompüter cəmi 2 m^2 yer tuturdu və bir neçə on kiloqramlıq çəkiyə malik 4 blokdan ibarət idi.

Mərkəzi prosessor hesab əməllərini yerinə yetirirdi, ünvanlaşmanı icra edirdi, idarəetməni və yaddaşa müraciəti həyata keçirirdi. Maşın sözünün uzunluğu 18 bit idi.

Toplama və çıxma komandaları 10 mikrosaniyəyə, vurma komandası – təxminən 20 mikrosaniyəyə yerinə yetirilirdi. İcra müddəti toplama üçün saniyədə təxminən 100 min əməliyyat təşkil edirdi. Prosessorun göstərişləri sadə – birünvanlı idilər.

Maşının əsas yaddaşı maqnit nüvəciklər əsasında olub standart olaraq 4096 ədəd 18 bitlik sözdən ibarət idi. Yaddaşı 4096 sözdən ibarət bloklarla artırmaq olardı və onun maksimal həcmi 65536 söz idi. Yaddaşdan oxuma və yaddaşa yazma 5 mikrosaniyəyə baş verirdi.

PDP-1 maşını məhz giriş-çıxış qurğularını idarə etmək üçün yaradılmışdılar. Bütün giriş-çıxış əməliyyatları yüksək sürətli kanallar vasitəsilə icra edilirdi. Mini-kompüter ikinci və üçüncü nəsil maşınlarında özünü yaxşı göstərmiş kəsilmə aparatından da istifadə edirdi.

PDP-11

70-ci illərdə buraxılmış EHM seriyasından olan PDP-11 çox böyük populyarlıq qazandı. Bu kompüterlərin maşın sözünün uzunluğu 16 bit idi.

Prosessor 16-bitlik 8 ədəd eyni, ümumi təyinatlı registrə malik idi. Son iki registrə əlavə funksiyalar da həvalə olunurdu: R7 registri komandalar sayğacı, R6 isə stekin göstəricisi idi. Stek – kompüterin yaddaş hissəsi olub, registrin məzmununu müvəqqəti olaraq yadda saxlamaq və altproqramdan qayıdan ünvanı saxlamaq üçün istifadə olunur (hara qayıtmağı da yadda saxlamaq lazımdır). Altproqrama müraciət adətən JSR R7 göstərişilə (*ing.* Jump to SubRoutine – "altproqrama keçid"), qayıdış isə – RTS R7 ilə (*ing.* ReTurn from Subroutine – "altproqramdan qayıdış") baş verir. PDP-11 standartlarına uyğun olaraq proqramda ünvanlar səkkizlikdir.

Burada ox işarəsi cari anda yerinə yetirilən göstərişi bildirir (altproqrama keçidə qədərki).



PDP-11-in 2,5 Mbayt tutumlu dəyişdirilə bilən maqnit diskinin disksürəni.

```
100000 Proqram          YADDAŞ  ünvan
. . . . . . . . . . . . . . . .
101000 JSR R7, #110000  000000  776
101004 . . . . . . . .  000000  1000
. . . . . . . . . . . . . . . .  170000  1002
110000 Altproqram       . . .
. . . . . . . . . . . . . . . .
110100 RTS R7

R6=001002    R7=101000
```

Növbəti addım – qayıtma ünvanının stekə yazılması (R7-də yerləşdirilir) və altproqram ünvanının R7-ə yazılmasıdır.

```
100000 Proqram          YADDAŞ  ünvan
. . . . . . . . . . . . . . . .
101000 JSR R7, #110000  000000  776
101004 . . . . . . . .  101004  1000
. . . . . . . . . . . . . . . .  170000  1002
110000 Altproqram       . . .
. . . . . . . . . . . . . . . .
110100 RTS R7

R6=001000    R7=110000
```

Altproqramın icrasından sonra RTS əmrinə görə stekdən qayıtma ünvanının "yada salınması"və onun R7 registrinə ötürülməsi baş verir.

▸ PDP-1140 və PDP-1170 kompüterləri.

```
100000 Proqram          YADDAŞ  ünvan
. . . . . . . . . . . . . . . .
101000 JSR R7, #110000  000000  776
101004 . . . . . . . .  101004  1000
. . . . . . . . . . . . . . . .  170000  1002
110000 Altproqram       . . .
. . . . . . . . . . . . . . . .
110100 RTS R7

R6=001002    R7=101004
```

PARALEL HESABLAMALAR

Hesablamaları paralel aparmaq bacarığı üçüncü nəsil kompüterlərin daha bir xarakterik xüsusiyyətidir.

IBM-in məşhurlaşmış 360-cı modelindən başqa bir çox firmalar da hesablama texnikasının istehsalına başladılar və maşınların məhsuldarlığını artırmağa

Prosessorun göstərişləri ikiünvanlı idi. Bəzi modellərdə sürüşkən vergüllü hesab əməllərini yerinə yetirmək üçün hesablama sürətini xeyli artıran qurğu əlavə edilirdi.

Maşının əsas yaddaşı 16-bitlik 32768 sözdən ibarət idi. Onu həm də genişləndirmək olardı. 64 kbaytdan artıq olan yaddaşa müraciət MM qurğusu (*ing.* Memory Management – "yaddaşın idarə edilməsi") vasitəsilə icra edilirdi. MM, proqramı operativ yaddaşın müxtəlif sahələrində yerləşdirməklə 8 yerə "bölürdü". Proqram bu "bölgünü" görmürdü, çünki tam olaraq 64 kbayt kimi icra olunurdu. Əlavə olaraq proqramın kodunu verilənlərin kodundan ayırmaq mümkün idi. Beləliklə, həm proqramın maksimal ölçüsü, həm də verilənlər sahəsinin maksimal ölçüsü 64 kbayt ola bilərdi. Əsas yaddaş (maqnit nüvəcikli yaddaş) kifayət qədər ləng olduğundan sürəti yarımkeçirici *keş*-yaddaş (həcmi 2048 söz) təmin edirdi.



nail oldular. Hesablama sürətinin artırılması üsullarından biri *çoxmaşınlı komplekslər* texnologiyası idi. Bu zaman ümumi hesablama məsələsinin həlli üçün bir neçə eyni tipli maşını birləşdirirdilər. Beləliklə, hesablamaları sürətləndirmək üçün iki IBM 7094 maşınını bir-birilə

birləşdirdilər, lakin bu xüsusi proqramlaşdırma üsullarının işlənilməsini tələb edirdi. Bundan başqa, xarici qurğuların təkrarlanması və məsələnin ümumi verilənlərinin iki müxtəlif kompüterdə saxlanması tələb olunan səmərəni vermədi.

Hesablama məsələlərinin həlli üçün yeni, verilənlərin paralel emalının *çoxprosessorlu* texnikası da geniş yayılmışdı. 1963-cü ildə Burroughs (ABŞ) bu prinsiplər əsasında qurulmuş yüksək məhsuldarlığa malik maşınlar istehsal etməyə başladı. Bu maşınların özəlliyi təkcə kompüterlərin işlənilməsinin qeyri-adi konsepsiyası deyildi ("kompüteri vahid tamlıq kimi yalnız bir adam layihələndirməlidir"), həm də sistemin məhsuldarlığını artırmaq üçün bir neçə eyni tipli prosessorların tətbiq edilməsi idi.

Birinci olaraq B5000 maşını, sonra isə daha bir neçə model meydana gəldi və 1973-cü ildə daha mükəmməl olan B 7700 modeli buraxıldı. Bəzi B 7700 maşınlarında yaddaş vasitəsilə verilənlərin mübadiləsini həyata keçirən yeddi tezişləyən prosessor quraşdırılmışdı. Operativ yaddaş səkkiz müstəqil modula bölünürdü. Modullu konstruksiya yaddaşa eyni zamanda səkkiz müstəqil girişdən müraciəti təmin edirdi. IBM maşınlarında olduğu kimi Burroughs maşınlarında da giriş-çıxış əməliyyatlarını təmin edən kanal prosessorları texnikasından istifadə olunurdu. Onlar verilənləri 32 xarici qurğudan da ala bilərdilər.

Çoxprosessorlu komplekslərin proqramlaşdırılmasında ixtisaslaşmış proqram təminatı tətbiq olunurdu ki, bu zaman hesablamalar *paralellənirdilər*



B 7700 kompüteri.

(yəni eyni zamanda müxtəlif prosessorlarda icra edilirdilər). Bunu dörd tam ədədin cəminin tapılması misalında aydınlaşdırmaq olar.

Adi cəm alqoritmi dörd ədədin cəmini hesablayır. *Sum* dəyişənini almaq üçün toplamanı üç dəfə icra etmək lazımdır. Köməkçi *s1* və *s2* (**Paralel cəm** alqoritmində) dəyişənlərindən istifadə etdikdə hesablamaları paralel aparmaq olar (bu misalda *s1* və *s2* dəyişənləri bir sətirdə ":" ayırıcısı ilə verilməklə göstərilir). Bunun üçün toplama əməlini cəmi iki dəfə icra etmək lazım gələcək, belə ki, ikinci alqoritm birinicidən tez icra olunacaq. Çoxprosessorlu komplekslərin səmərəli tətbiqi üçün xüsusi proqramlaşdırma dilləri yaradıldı. Hesablama kompleksi hesablamaları sanki özü "paralelləşdirirdi". Buna görə də hətta **Adi cəm** alqoritmi çoxprosessorlu komplekslərdə adi EHM-lərə nisbətən daha sürətlə yerinə yetirilirdi.

Çoxprosessorlu komplekslərin təkmilləşdirilməsi *konveyer sistemlərinin* meydana çıxmasına gətirib çıxardı. Bu komplekslərdə hesablamaların paralelliyi ideyası öz əksini tapmışdır. Xüsusi proqramlaşdırma üsullarının hesabına

```
alq Adi cəm (tam a, b, c, d)
verilib a, b, c, d
gərəkdir
baş tam sum
 | sum:=a+b
 | sum:=sum+c
 | sum:=sum+d
son
```

```
alq Paralel cəm (tam a, b, c, d)
verilib a, b, c, d
gərəkdir
baş tam sum, s1, s2
 | s1:=a+b : s2:=c+d
 | sum:=s1+s2
son
```

```
alq Adi hesablama (tam a, b, c, d, f)
verilib a, b, c, d, f
gərəkdir
baş tam sum1, sum2, s
 | s:=a+b
 | sum1:=s*c
 | s:=d+e
 | sum2:=s*f
 | qiym:=sum2
son
```

```
alq Konveyer hesablama (tam a, b, c, d, f)
verilib a, b, c, d, f
gərəkdir
baş tam sum1, sum2, s
 | s:=a+b
 | sum1:=s*c : s:=d+e
 | sum2:=s*f
 | qiym:=sum2
son
```

Bu sistemlər avtomobil quraşdırma konveyerinin analoqu kimi konveyer sistemləri adlandırılmışdı. Eyni tipli əməlləri yerinə yetirən xüsusi prosessorlar (toplama və çıxma üçün prosessor, vurma və bölmə üçün prosessor və s.) ixtisaslaşdırılmış yığma əməliyyatlarını yerinə yetirən fəhlələr ilə müqayisə olunurdu. Hesablama üçün zəruri olan verilənlər elə bil ki, konveyer boyunca irəliləyirdilər. Eyni zamanda müxtəlif verilənlər axını ilə müvafiq əməliyyatlar yerinə yetirilirdi.

xeyli sürət artımına nail olmaq mümkün oldu.

Adi hesablama alqoritmi iki toplama və iki vurma əməlini yerinə yetirir, ikinci alqoritm isə (konveyer kompüterdə) dörd əməliyyat yox (2+2=4), üç əməliyyat yerinə yetirir, belə ki, *sum1* hasili və *s* cəmi eyni zamanda müxtəlif ixtisaslaşmış prosessorlar kompleksində icra edilir (və bu məqam misalda *sum1* və *s* dəyişənlərinin bir sətirdə ":" ayıncısı ilə göstərilmişdir). Konveyer sistemində icra olunan real hesablamalarda hər iki alqoritm eyni cür işləyəcək, belə ki (çoxprosessorlu hesablama kompleksinə analoji olaraq) sistemin özü müəyyən edəcək ki, hesablamalarda necə qənaət etmək olar.

XX əsrin əvvəli. "Ford motor" kompaniyasının konveyerində iş.



Bundan əlavə paralelliyi prosessor səviyyəsində təmin etmək olar. Verilənlərin paralel emalının iki metodu prosessorların layihələndirilməsində öz yerini tutdu:

• Müəyyən əməllərin (məsələn, toplamanın) yerinə yetirilməsi üçün bir neçə qurğu (məsələn, hesab) nəzərdə tutula bilər; *n* sayda müstəqil cəmləyici eyni zamanda *n* sayda cəmi hesablaya bilər. Bu üsul massivləri tez emal etməyə imkan verir (yəni vektor əməliyyatlarını yerinə yetirir).

• Prosessoru müxtəlif *mikroəməliyyatları* eyni zamanda icra etməyə imkan verən konveyer üsulu ilə layihələndirmək olar.

SUPERKOMPÜTERLƏR

Konveyer sistemlərin istehsalçısı olan ABŞ-ın CDC korporasiyası (*ing.* Control Data Corporation) 1960-cı ildən kompüterləri layihələşdirməyə başladı. 1964-cü ildə o, CDC 6600 modelini, 5 il sonra isə daha güclü CDC 7600 modelini istehsal etdi. Sonralar bu maşınlar sırasını CYBER modeli davam etdirdi. CDC firmasının EHM-ləri üçün bir neçə giriş-çıxış prosessorunun olması xarakterik idi (onlar periferik prosessorlar adlanırdı) və bunların da hər biri mərkəzi prosessora nəzərən müstəqil idi. Əgər kanal adətən bir neçə sözdən ibarət olan yaddaşa və azsaylı spesifik göstərişlər toplusuna malik idisə, geniş həcmli yad-





CDC 7600 kompüteri (solda). CRAY1 kompüteri (sağda).

daşa və inkişaf etmiş əmrlər sisteminə malik olan periferik prosessor isə faktiki olaraq universal EHM idi. Kanal mərkəzi prosessor tərəfindən işə salınır və nəzarətdə saxlanılırdı, periferik prosessora isə təsir göstərə bilmirdi. On ədəd CDC 6600 periferik prosessoru 4096 ədəd 12 mərtəbəli sözdən ibarət olan yaddaşa malik idi.

CDC kompüterinin mərkəzi prosessoru öz növbəsində, bir-birindən asılı olmadan paralel işləyə bilən bir çox elementlərdən-prosessorlardan ibarət idi. CDC 6600 prosessoruna on funksional qurğu daxil idi: sabit vergüllü ədədlərin toplanması üçün qurğu, sürüşkən vergüllü ədədlər üzərində toplama əməli aparmaq üçün qurğu, vurma üçün iki qurğu, bölmə qurğusu, məntiq əməllərini icra etmək üçün qurğu və s. CDC 6600 kompüterinin orta məhsuldarlığı saniyədə 4 milyon, CDC 7600 kompüterinin orta məhsuldarlığı isə saniyədə 15 milyon əməliyyat idi (SuperEHM üçün məhsuldarlıq sürüşkən vergüllü ədədlər üzərində bir saniyədə aparıla bilən əməllərin sayı ilə ölçülür).

Superkompüterlərin istehsalında dünya liderlərindən biri Cray Research amerika firması idi. Onu 1972-ci ildə CDC 6600 və 7600 maşınlarının baş konstruktoru olmuş Seymur Krey təsis etmişdi. 1976-cı ildə firma Kaliforniya universitetinin nəzdində olan Los Alamos elmi laboratoriyasında özünün CRAY I maşınının birinci nümunəsini yaratdı. Onun orta məhsuldarlığı saniyədə 80 milyon əməliyyat təşkil edirdi. Bütün hesablama ehtiyatlarının cəlb edilməsi şəraitində – sürüşkən vergüllü ədədlər üzərində saniyədə 138 milyon əməliyyat, bəzən isə qısa zaman fasilələrində hətta 250 milyon əməliyyat yerinə yetirilirdi.

Belə yüksək məhsuldarlıq bir neçə konveyerin istifadə edilməsilə təmin olunurdu. Sistem vektorlar üzərində əməlləri yerinə yetirə bilirdi (prosessor vektor registrlərə malik idi) və prosessorun funksional qurğuları eyni zamanda vektor və adi (skalyar) əməlləri icra edə bilirdi. Prosessor hesab əməlləri yerinə yetirmək üçün 64-bitlik 8 ədəd registrə, aralıq nəticələri yadda saxlamaq üçün 64 ədəd 64-bitlik registrə və 8 ədəd 64 elementli vektor registrə (vektorun hər elementi – 64 mərtəbəli söz idi) malik idi.

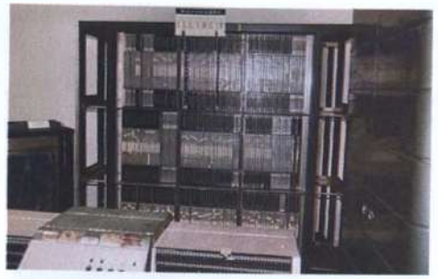
Yaddaş 1 milyona qədər 64-bitlik sözdən ibarət idi. Əslində yaddaşın sözü 72 bitdən ibarət idi və səkkiz "xidməti" bit yaddaşda baş verən sıradançıxmaları



Seymur Krey.

CRAY I maşınının sürətli olması onun ölçülərinin böyük olmaması hesabına başa gəlmişdi (hündürlüyü 2 m, diametri 2,5 m). Belə ölçülər, superEHM-in elementlərini "qısa" xətlərlə (naqillərlə) əlaqələndirməyə imkan verdi və bu da kompüter blokları arasında informasiya mübadiləsi zamanı yubanmaları xeyli ixtisar edirdi. Lakin EHM-in yığcamlığı avtomobilin mühərrikinin soyutma sisteminə oxşar maye soyutma sisteminin qurulmasını tələb etdi.

ILLIAC IV kompüteri.



tapıb islah etmək üçün istifadə edilirdi. 1967-1974-cü illərdə İllinoys universitetində (ABŞ) Burroughs firması tərəfindən cəmisi bir nüsxə olmaqla hazırlanmış digər superkompüter ILLIAC IV nəinki vektorlar üzərində, hətta matrislər üzərində də əməllər yerinə yetirirdi. Bu layihənin məqsədi saniyədə 1 mlrd. əməli yerinə yetirə bilən məhsuldarlığa malik olan sistemin qurulması idi. Bunun üçün maşın hər birində 64 prosessor elementi olan 4 matrisdə qruplaşdırılmış və ümumi idarəetmə altında olan 256 prosessor elementindən(faktiki olaraq universal prosessorlar) ibarət olmalı idi. Buna görə də ILLIAC IV maşınını *matrisli* EHM-lər sinfinə aid edirlər.

Matris sistemləri iki ölçülü cədvəllər – prosessor modullarının matrisləridir. Prosessor modulu yaddaşa malik idi və bir çox hesab əməlləri yerinə yetirə bilirdi: toplama və vurma, çıxma və bölmə, sürüşmə, müqayisə və s. Əgər konveyer sistemlərində bütün prosessorlar (CDC 6600 və 7600 olan funksional qurğular) müxtəlif əməlləri yerinə yetirirdisə, matris sistemi eyni zamanda bütün cədvəl – sözlər matrisilə eyni əməlləri icra edirdi. Böyük həcmli əməllər (məsələn, "matrisin bütün elementlərini hər hansı bir ədədə vurmaq", "matrisin elementlərini sıfıra çevirmək", bəzən isə "matrisin birinci və ikinci sətirlərini toplamaq" kimi daha mürəkkəb əməliyyatlar) bir iş kimi icra edilirdi və bu, ədədi ədədə vurmağı xatırladırdı.

Matris sistemləri universal olduqları üçün çox mürəkkəbdirlər. Buna görə də cəmi 64 prosessor elementindən ibarət olan yalnız bir matris quruldu və nəticədə məhsuldarlıq saniyədə 200 milyon əməliyyata qədər aşağı düşdü. Buna baxmayaraq 80-ci illərə qədər ILLIAC IV maşını dünyada ən yüksək məhsuldarlıqlı kompüter olaraq qalırdı. Bu layihənin dəyəri təxminən 100 milyon dollar idi.

ALEKSEY ANDREYEVİÇ LYAPUNOV

Riyaziyyatçı Aleksey Andreyeviç Lyapunov 1911-ci ilin 8 oktyabrında Moskvada anadan olmuşdur. Ailədə olan məlumata görə Lyapunovların nəsli öz köklərilə Aleksandr Nevskinin qardaşı, knyaz Konstantin Qalitskiyə gedib çıxır. Aleksey Andreyeviçin ulu babası olan Mixail Vasilyeviç N.İ.Lobaçevskinin tələbəsi, Kazan universitetində astronomiya professoru və observatoriyanın direktoru olmuşdur. Daha sonralar isə o, Yaroslavda ilk ali məktəb olan Demidov litseyinin direktoru olmuşdur. Onun çox yaxşı tanınmış üç oğlu var idi: riyaziyyatçı, mexanik və gəmiqayırma mütəxəssisi Aleksandr Mixayloviç, bəstəkar Sergey Mixayloviç, slavyan filologiyası mütəxəssisi, akademik Boris Mixayloviç.

Aleksey Lyapunovun elmi maraqları əsasən onun boya-başa çatdığı mühitlə təyin olunur. Onun ilk astronomiya, fizika, riyaziyyat və mineralogiya müəllimi Moskva universitetinin fizika-riyaziyyat fakültəsini bitirmiş, daha sonra isə Heydelberqdə oxumuş atası Andrey Nikolayeviç olmuşdu. Məhz o, oğluna riyaziyyatı sevdirmişdi.

A.A.Lyapunov "Kibernetikanın problemləri" elmi məcmuələrinin nəşrini təsis etdi, "Kibernetika monoqrafiyalarda" silsilə kitabları yaratdı və redaktə etdi, "Kibernetik məcmuə" seriyasında xarici müəlliflərin ən yaxşı əsərlərinin tərcüməsini təşkil etdi.

Onun ilk müstəqil tədqiqatları – məktəb illərində apardığı astronomiya üzrə müşahidələri dəfələrlə "Moskva astronomiyanı sevənlər cəmiyyətinin bülleteni"ndə çap edilmişdi. 1928-ci ildə məktəbi bitirmiş Lyapunov MDU-nun fizika-riyaziyyat fakültəsinə daxil olur, lakin il yarımdan sonra o "zadəgan mənşəli şəxs" kimi universitetdən gedir.

1930-cu ilin payızında o, Dövlət geofizika institutunda məşhur fizik alim, akademik P.P.Lazarevin (1878-1942) yanında laborant vəzifəsində işləməyə başlayır. O, ali təhsilini ekstern yolu ilə, MDU-da universitet kursları üzrə imtahan verməklə alır. Məşhur riyaziyyatçı akademik N.N.Luzinin dəstəyilə o, riyaziyyatı dərindən öyrənir və çoxluqlar nəzəriyyəsinin öyrənilməsinə dair tədqiqatlara qoşulur ki, onun 60 elmi işi məhz bu sahəyə həsr olunmuşdur. Aleksey Lyapunov çoxluqlar nəzəriyyəsinə dair əsərlərini çap etdirir və Böyük Vətən müharibəsindən bir qədər əvvəl namizədlik dissertasiyasını müdafiə edir.

Müharibə müəyyən müddətə onun elmi fəaliyyətini dayandırır. 1942-ci ildə Aleksey Andreyeviç könüllü olaraq cəbhəyə gedir və artilleriyada topoqrafiya alayının komandiri olur.

Müharibədən sonra Aleksey Andreyeviç F.E.Dzerjinski adına Artilleriya Akademiyasında riyaziyyat müəllimi işləyir.

Lyapunov elmi atmır, V.A.Steklov adına Riyaziyyat institutunda doktorluq dissertasiyasını müdafiə edir, SSRİ EA geofizika institutunda işləyir, professor elmi adı alır. Akademik M.V.Keldış onu EA-nın Riyaziyyat institutunun tətbiqi riyaziyyat bölməsinin kibernetika şöbəsinə dəvət edir.

Lyapunov ölkəsinin ilk alimlərindən biri idi ki, kibernetikanın əhəmiyyətini qiymətləndirmiş, onun təşəkkül tapması və inkişafına böyük töhfələr vermişdir. O zaman kibernetika sahəsində işləmək onun varlığı üçün mübarizə aparmaq demək idi.



A.A.Lyapunov.

MDU-nun riyazi məntiq və hesablama riyaziyyatı kafedrasının professoru olmaqla yanaşı o, 1953-cü ildə proqramlaşdırma üzrə seminar, 1954-cü ildən isə kibernetika üzrə seminarlar təşkil edir. Bu seminarlar alimlərin və mütəxəssislərin bir araya gəldikləri mərkəzə çevrilir, Lyapunov isə kibernetika sahəsində işləri koordinasiya edirdi. Onun təşəbbüsü ilə 1959-cu ildə SSRİ EA nəzdində kompleks problemlər üzrə Elmi Şura – "Kibernetika" yaradıldı ki, ona da akademik A.İ.Berq (1898-1979) rəhbərlik edirdi.

1961-ci ilə Aleksey Andreyeviç Lyapunov SSRİ EA Sibir şöbəsinin və Akademiya Şəhərciyinin yaradıcısı akademik M.A.Lavrentyevin dəvətilə Novosibirskə köçür. Orada o, Riyaziyyat institutunun kibernetika şöbəsini və Novosibirsk universitetində nəzəri kibernerika kafedrasını yaradır. Orada yaşadığı illərdə Aleksey Andreyeviç özünün bir sıra ideyalarını həyata keçirir. O, 1962-ci ildə ölkədə ilk dəfə Novosibirsk

"Computer pioneer" medalı. Üz tərəfində – Çarlz Bebbic, ilk (mexaniki) hesablama maşınının yaradıcısı. Arxa tərəfdə belə bir yazı vardı: "Kompüter cəmiyyəti Aleksey Andreyeviçi sovet kibernetikası və proqramlaşdırmasının banisi kimi tanıdı".



"Qısa müddət ərzində kibernetikaya münasibət aşağıdakı fazalardan keçmişdir: qəti inkar – varlığının etiraf edilməsi – faydalılığının etirafı və riyaziyyatçılar üçün məsələlərin olmaması – müəyyən riyazi problematikanın olması – kibernetikanın riyazi problematikasının tam etirafı".

A.A.Lyapunov.
Riyaziyyat üzrə IV Ümumittifaq qurultaydakı çıxışından.
1966-cı il



Lyapunov mühazirə oxuyur.

universiteti nəzdində fizika-riyaziyyat internat məktəbinin yaradılmasının təşəbbüskarı, onun elmi şurasının ilk sədri və fəal mühazirəçisi olmuşdur. O, Ümumsibir riyazi olimpiadalarının və Akademiya şəhərciyində yay fizika-riyaziyyat məktəblərinin təşkil edilməsini dəstəkləyirdi.

1964-cü ildə Lyapunov riyaziyyat bölməsi üzrə SSRİ EA müxbir üzvi olur.

Lyapunovun elmi maraq sahəsi çox genişdir. Onun işləri kibernetikanın ümumi məsələlərinə, proqramlaşdırmanın riyazi əsasları və alqoritmlər nəzəriyyəsinə, riyazi linqvistika və maşın tərcüməsinə, biologiyanın kibernetik aspektlərinə, habelə elmi fikrin inkişafının fəlsəfi və metodoloji problemlərinə həsr olunmuşdur. Onun tərəfindən maşın dilindən fərqli, Alqol tipli alqoritmik dillərin meydana çıxmasına qədər işlənilmiş ilk proqramlaşdırma dili – *operator metodu* yaradılmışdır.

Öz elmi, pedaqoji və təşkilati fəaliyyətinə görə o, dövlət mükafatları: Lenin ordeni, iki dəfə Qırmızı Əmək Bayrağı ordeni və "Şərəf nişanı" ordenilə təltif olunmuşdur. Amma alim ən çox Böyük Vətən müharibəsində Krımın azad edilməsi uğrunda gedən döyüşlərdə iştirakına görə aldığı Qırmızı Ulduz ordenilə fəxr edirdi.

Aleksey Andreyeviç nəinki görkəmli riyaziyyatçı idi, o, həm də ədəbiyyatı gözəl bilirdi, arxitektura və rəngkarlıqla maraqlanırdı, özünün minereoloji kolleksiyasını nümayiş etdirməyi sevirdi, fransız dilini mükəmməl bilirdi, Fransa tarixini, incəsənətini və mədəniyyətini dərindən bilirdi, baxmayaraq ki, heç vaxt öz vətənindən kənara çıxmamışdı.

A.A.Lyapunov 1973-cü ilin 23 iyununda vəfat edib və Vvedenskoye qəbristanlığında basdırılıb (onun qəbrinə gedən xiyaban müəllimi N.N.Luzinin uyuduğu yerin yanından keçir).

Aleksey Andreyeviçin informatika sahəsindəki xidmətlərinin beynəlxalq miqyasda tanınmasında ilk addım onun 1996-cı ildə "Computer Pioneer" medalı ilə təltif olunması oldu. Bu yüksək texnologiyalar sahəsində ən nüfuzlu beynəlxalq təşkilat olan – IEEE Computer Society tərəfindən edildi. Medalın üz tərəfində Çarlz Bebic təsvir olunmuşdur. Arxa tərəfini isə ingilis dilində belə bir yazı bəzəyirdi: "Kompüter cəmiyyəti A.A.Lyapunovu sovet kibernetikası və proqramlaşdırmasının banisi kimi tanıdı".

ÜÇLÜK HESABLAMA SİSTEMİ VƏ ÜÇLÜK EHM

1840-cı ildə mexaniki hesablayıcıların ixtiraçısı, fransız riyaziyyatçısı Leon Lalan tərəfindən tarazlaşdırılmış üçlük sistem təklif olunmuşdur. Bu sistemin əsasını 3 ədədi təşkil edir, amma 0, 1 və 2 ədədləri əvəzinə burada 0, 1 və -1 istifadə olunur (adətən "mənfi bir" əvəzinə $\bar{1}$ istifadə olunur). İkilik sistemin bitləri kimi bu cür işarələmələr *trit* adlanır.

Məlumdur ki, ədədin işarəsi onun üçlük təsvirindəki birinci sıfıra bərabər olmayan simvol ilə təyin olunur. Əgər bu 1 isə, onda ədəd müsbətdir, əgər $\bar{1}$ isə, onda mənfidir. Tarazlaşdırılmış üçlük sistemdə toplama və çıxma mərtəbələr üzrə çox sadə icra olunur (1+0=1, 1+$\bar{1}$=0, 1+1=1$\bar{1}$, 1+$\bar{1}$=0$\bar{1}$). Tarazlaşdırılmış üçlük sistemində 247 və 122 ədədlərini toplayaq:

$$\begin{array}{r} 100011 \\ + \\ \hline 1\bar{1}\bar{1}\bar{1}\bar{1}1 \\ \hline 1\bar{1}\bar{1}\bar{1}00 \end{array}$$

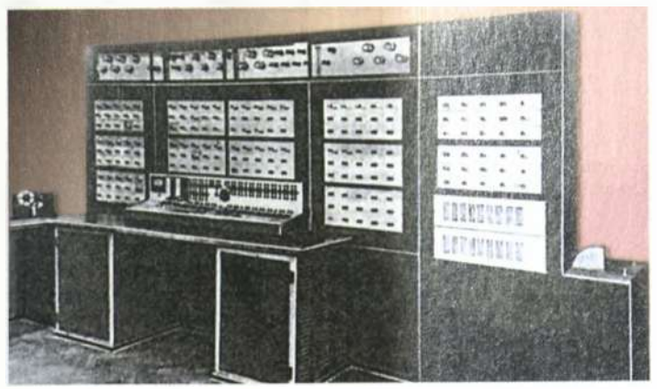
Asanlıqla yoxlamaq olar ki, $1\bar{1}\bar{1}\bar{1}00_3$=369. Çıxma da belə sadəliklə icra edilir: bunun üçün çıxılanın işarəsini əksinə dəyişmək və onu azalan ilə toplamaq lazımdır. Vurma da beləcə işarənin dəyişməsilə toplama əməlinə gətirilir.

Lalandan sonra 100 ildən artıq bir müddətdə tarazlaşdırılmış üçlük sistemi alimlərin böyük diqqətinə səbəb olmadı. Amma onluq sistemə əsaslanan mexaniki və elektromexaniki hesablama qurğuları elektron kompüterlər ilə əvəz olunduqda üçlük sistemə ikilik sistemin alternativi kimi baxıldı.

Bu sistemdə yazılmış ədədlər üzərində hesab əməlləri çox sadə icra olunur. Bu sistemin digər üstünlükləri də var: bu sistemdə ədəd ikilik sistemə nəzərən daha qısa yazılır; xüsusi işarə mərtəbəsi daxil etməyə ehtiyac yoxdur.

Hesablama üsulları sahəsində məşhur amerikalı mütəxəssis Donald Knut 1969-cu ildə yazırdı ki, əgər triggerə əvəz tapılarsa (iki vəziyyəti olan və 1 bit informasiyanı saxlamağa imkan verən məntiqi sxem) kompüterlərdə üçlük sistemin zamanı yetişəcəkdir. Lakin Knutun kitabının işıq üzü görməsindən xeyli əvvəl üçlük hesablama sistemi çox maraqlı və orijinal bir sovet layihəsində uğurla tətbiq olunmuşdu. Söhbət 1959-cu ildə akademik S.L.Sobolevin və N.P.Brusentsovun rəhbərliyilə MDU-nun Hesablama Mərkəzində yaradılmış "Setun" EHM-dən gedir. 1962-ci ildən başlayaraq maşın seriyalarla istehsal edilirdi və üç il ərzində onun 50 nüsxəsi istehsal olundu. Üçlük hesab əsasında qurulan qurğular nəinki nisbətən daha sadə və tez işləyirdilər,

"Setun" EHM-i.



ÇƏKİLƏR HAQDA BAŞE MƏSƏLƏSİ

Maraqlıdır ki, ədədlərin üçlük sistemdə təsvir edilməsi artıq məşhur olan "çəkilər haqda Başe məsələsi"ndə qeyri-aşkar şəkildə var idi, amma onu da qeyd edək ki, onu Başedən 400 il əvvəl XIII əsrdə Leonardo Pizanski (Fibonaççi) ifadə etmişdi. Məsələdə elə 4 çəki daşı tapmaq tələb olunurdu ki, onların köməyilə 1 kq-dan 40 kq-a kimi istənilən çəkini ölçmək mümkün olsun (və yaxud 3 daşdan ibarət olan və 1-dən 13 kq-a qədər çəkini ölçməyə imkan verən məsələ kimi də iki variantda məlum idi).

Məsələni həll etmək üçün yükün çəkisinə bərabər olan ədədi tarazlaşdırılmış üçlük sistemdə təsvir etmək lazımdır. Fərz edək ki, 33 kq ağırlığı olan yükü ölçmək lazımdır. 33 ədədini tarazlaşdırılmış üçlük sitemdə aşağıdakı kimi təsvir etmək olar: $33=27+9-3+0=1\bar{1}\bar{1}0_3$. Deməli, 27 kq və 9 kq olan çəki daşları tərəzinin bir gözünə, çəkilən yük və 3 kq daş isə o biri gözünə qoyulmalıdır. Əgər bu üç çəki daşına ağırlığı 1 kq olan dördüncü daşı da əlavə etsək, onda 40 kq çəkini aşmayan istənilən yükü tərəzilə çəkə bilərik.

$8_{10}=3^2+0-3^0=10\bar{1}_3$,

$1\bar{1}\bar{1}\bar{1}\bar{1}\bar{1}_3=3^5-3^4-3^3-3^2-3^1-3^0=243-81-27-9-3-1=122_{10}$,

$-17_{10}=3^3+3^2+0+3^0=\bar{1}101_3$,

$\bar{1}000\bar{1}\bar{1}_3=-3^5+0+0+0-3^1-3^0=-243-3-1=247_{10}$

S.L.Sobolev.



həm də çox etibarlı idilər. Onlar eyni elementlərdən düzəldilmiş ikilik qurğulardan daha az enerji sərf edirdi.

Bu maşın sabit vergüllü ədədlərlə əməlləri icra edirdi, sürüşkən vergüllü əməllər isə proqram vasitəsilə realizə olunurdu. Komandalar sistemində 3 şərti keçid komandaları və tritlərin vurulması komandası daxil olmaqla, cəmi 24 komanda var idi, lakin onlardan istifadə edərək müxtəlif məsələləri effektiv proqramlaşdırmaq olurdu. Yaddaş iki səviyyəli idi: həcmi 162 qısa sözdən (9 trit) ibarət olan ferrit nüvəcikli operativ yaddaş və həcmi 3888 qısa sözdən ibarət olan maqnit barabanlı əsas yaddaş. Toplama və çıxma əməllərinin icrası 180 mikrosaniyə təşkil edirdi, vurma isə 320 mikrosaniyə idi. Əlbəttə, indi bu ədədlər çox kiçik görünür, amma o zamanlar "Setun" məhsuldarlığına görə digər kiçik maşınları üstələyirdi. Bununla belə, o öz sinfinin maşınlarından daha ucuz idi.

Bir neçə il sonra 1970-ci ildə təkmilləşdirilmiş "Setun 70" maşınının eksperimental nümunəsi yaradıldı. Bu maşın bir sıra maraqlı arxitektur özəlliklərə malik idi və üçqiymətli məntiqin tətbiqi ideyaları onda öz əksini tapdı. "Setun" layihəsi indiyə qədər də üçlük sistemin kompüterlərdə istifadəsinin yeganə nümunəsi olaraq qalır və ola bilər ki, üçlük maşınların zamanı gəlsin.

SOVETLƏR ÖLKƏSİNDƏ İSTEHSAL OLUNAN EHM-lər

İlk baxışda elə görünə bilər ki, hesablama texnikası yalnız ABŞ-da istehsal olunurdu. Amma bu belə deyildi. Doğrudan da, yeni elm sahəsi böyük maliyyə qoyuluşu tələb edirdi ki, bu da İkinci dünya müharibəsinin əsas savaş meydanı olan müharibədən sonrakı Avropa üçün ağır yük idi. Hər şeyə rəğmən, kompüter quruculuğu yarışına girən azsaylı ölkələrdən biri də keçmiş SSRİ oldu.

1948-ci ildə kompüterlərin istehsalı sahəsində ölkənin pioneri olmuş akademik Sergey Alekseyeviç Lebedev (1902-1974) ilk sovet və Avropa kompüterinin – kiçik elektron hesablama maşınının (Малая Электронная Счетная Машина – МЭСМ) quraşdırılması prosesinə başladı. Onun yaradılması işləri tədqiqat, eksperiment xarakteri daşıyırdı. 1950-ci ildə Ukrayna EA elektromexanika institutunda bu МЭСМ-i istismara buraxdılar. 1952-1953-cü illərdə o, praktiki olaraq Avropada müntəzəm istismar olunan yeganə EHM idi.

S.A.Lebedev.



Maşının əsas parametrləri belə idi: sürəti – saniyədə 50 əməliyyat; yaddaşda 31 ədəd 16 mərtəbəli ədəd və uzunluğu 20 bit olan 63 komanda saxlamaq olardı; maşının tutduğu sahə – 60 m^2; istehlak etdiyi cərəyan – 25 kVt idi. Yalnız OYQ-da 2,5 min triod və 1,5 min diod istifadə olunurdu. Kiçik yaddaşın genişləndirilməsi üçün əlavə olaraq 5 min söz (hərəsi 16 bit) həcmə malik olan maqnit barabanından istifadə etmək olardı. Maşın dəyişdirilə bilən uzunmüddətli yaddaş qurğusuna da malik idi ki, ondan da ədədi sabitləri və tez-tez istifadə olunan əmrləri yadda saxlamaq üçün istifadə olunurdu (sonralar bu DYQ – daimi yaddaş qurğusu adlandırıldı).

Əlbəttə, maşın müasir ölçülərlə götürüldükdə ləng işləyirdi, amma onun əsas işləmə prinsipləri (Lebedev bu prinsipləri ABŞ-da aparılan işlərdən asılı olmayaraq təklif etmişdi) başqa kompüterlərin də qurulmasında istifadə edilmişdi. МЭСМ faktiki olaraq БЭСМ – böyük elektron hesablama maşınının (Большая Электронная Счетная Машина) modeli oldu. Hər iki maşın (МЭСМ və БЭСМ) bir nüsxədə hazırlanmışdı.

Praktiki olaraq МЭСМ yaratmış bütün kollektiv S.A.Lebedevin laboratoriyasının bazası əsasında yaranan Ukrayna EA Hesablama mərkəzinin nüvəsini təşkil etdi.

1952-ci ildə Hesablama mərkəzində БЭСМ üzərində iş yekunlaşdı və bir ildən sonra o, SSRİ EA-da artıq işə düşdü. БЭСМ haqlı olaraq XX əsrin 50-ci illər Avropasının ən yaxşı kompüteri kimi tanınmışdı. Maşın 39 mərtəbəli sözləri saniyədə 10 min əməliyyat olmaqla orta sürətlə emal edirdi. БЭСМ xarici yaddaş qurğusu olaraq hər birində 5120 söz olan iki maqnit barabanından istifadə edirdi. Barabandan oxunma sürəti saniyədə 800 söz idi. Maşına həmçinin ümumi həcmi 120 min söz olan maqnit lentlər də qoşulurdu.

БЭСМ rəqəmli hesablama maşınlarının tam bir seriyasının başlanğıcını qoydu. Operativ yaddaşın elementləri kimi istifadə olunan civəli ləngitmə xətləri 1954-cü ildə yadda saxlama qabiliyyətli elektron-şüa borusu ilə əvəz olundu. İki il sonra onları ferrit nüvəcikli 1024 ədəd 39 mərtəbəyə malik sözdən ibarət olan yaddaş əvəz etdi. Bu şəkildə maşın БЭСМ-1 kimi tanınmağa başlandı. Bu maşında müxtəlif məsələlər, o cümlədən Günəş sisteminin 700 kiçik planetinin hərəkət orbitləri hesablanırdı.

БЭСМ-4.

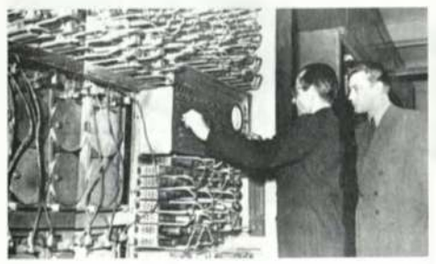


Sənaye üsulu ilə istehsalı üçün maşının konstruksiyasını dəyişdirdilər və 1958-ci ildə lampalı БЭСМ-2 maşınının seriyalı istehsalına başlandı. Onun elektrik məsrəfi 75 kVt idi.

1964 və 1966-cı illərdə bu tipli yeni БЭСМ-3М və БЭСМ-4 maşınları meydana gəldi. Bu maşınlar özlərinin sələflərindən fərqli olaraq yarımkeçirici elementlərdən yığılırdı. БЭСМ-4 maşınının yaddaşı 2×4096 45-mərtəbəli sözdən, hər birinin həcmi 16384 min söz olan dörd maqnit barabanından ibarət idi və o cəmi 8 kVt elektrik gücü istehlak edirdi.

1967-ci ildə çoxsaylı mürəkkəb hesablamalar aparılmasını tələb edən məsələlər üçün orta sürəti saniyədə 1 milyon əməliyyat olan yarımkeçirici əsaslı БЭСМ-6 maşını yaradıldı. БЭСМ-4 ilə müqayisədə yaddaş 8 dəfə (mərtəbələrin sayı 45 yox, 48 idi) artmışdı, maqnit barabanlarının sayı isə hər birinin həcmi 32 min söz olmaqla 16-ya çatdı.

БЭСМ-6 maşınında hesablama texnikasının inkişafının o zamankı bütün



S.A.Lebedev və V.A.Melnikov БЭСМ-6 maşınının qarşısında.

mütərəqqi meylləri əks olunmuşdu: multiproqram rejimi, kəsilmələrin aparat sistemi, "yaddaşın qorunması" sxemi və ünvanların avtomatik mənimsədilməsi (yəni faktiki olaraq yaddaş dispetçeri). Yaddaşın istənilən hissəsi stek kimi istifadə oluna bilərdi. Mərkəzi prosessor bir ünvanlı komandalar sistemindən və 16 ədəd sürətli registrlərdən istifadə edirdi.

Proqramlaşma üçün FORTRAN və ALGOL dillərindən istifadə olunurdu.

MÜHƏNDİS HESABLAMALARI ÜÇÜN MAŞIN

1965-ci ildə Ukrayna EA Kibernetika institutunda akademik V.M.Qluşkovun rəhbərliyi altında yaradılmış Mir EHM-i (mühəndis hesablamaları üçün maşın – машина для инженерных расчётов) faktiki olaraq dünyada birinci fərdi kompüter idi (kiçik otaqda yerləşə bilirdi). İlk maşınlar riyazi dilə yaxın olan MİR alqoritmik dilindən istifadə edirdilər. Hesablama apararkən 90-cı illərdə fərdi kompüterlərdə BASIC dilinin istifadəsi zamanı olduğu kimi burada da asanlıqla alqoritmi, düsturları, əmsalları dəyişmək olurdu. Proqramçı informasiyanın daxil və xaric edildiyi elektrik yazı makinası arxasında işləyirdi.

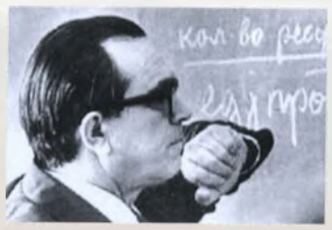
Maşında mikroproqramlaşdırma nəzərdə tutulmuşdu (maşının komandalarının məntiqini təsvir edən və ümumi həcmi 700 kbit olan xüsusi mikroproqram matrisi var idi). Həcmi 4096 bayt olan əsas yaddaş maqnit nüvəciklərdə yerləşirdi. Hesabi qurğu matrisli idi, bu isə emal olunan ədədlərin mərtəbələrini ixtiyari edə bilirdi və bu zaman məhsuldarlıq saniyədə 8 min əməliyyat idi.

1968-ci ildə maşının Mir-1 modifikasiyası çıxdı ki, bu da informasiyanı perfolentdən daxil və xaric edə bilirdi. Mir EHM-in bir nüsxəsini Londonda sərgidə IBM firması satın aldı.

1969-cu ildə Mir-2 maşını buraxıldı ki, o maşında ilk dəfə olaraq işıq qələmli displey tətbiq edildi və bu yenilik EHM ilə dialoqu sadələşdirdi və onu daha operativ etdi. Yeni maşında Analitik proqramlaşdırma dili istifadə edildi ki, bu da Mir alqoritmik dilinin sonrakı inkişafı idi. Bu dildə xətti cəbrin məsələlərini nəinki ədədi şəkildə, həm də analitik şəkildə həll etmək mümkün idi. Maşının sürəti saniyədə 12 min əməliyyata qədər artdı. Mir maşınları sırasının davamı da oldu.



MİR-1 maşını.



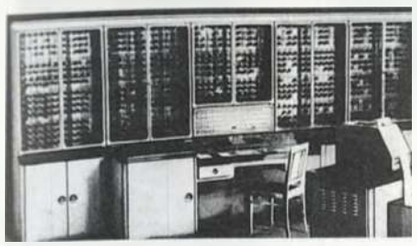
Akademik V.M.Qluşkov.

Maşın o qədər uğurlu və etibarlı oldu ki, o 90-cı illərə qədər istismarda qaldı. Tək-tük müasir kompüter tapılar ki, belə uzunömürlüyü ilə öyünə bilsin!

1958-ci ildə S.A.Lebedevin rəhbərliyilə SSRİ EA dəqiq mexanika və hesablama texnikası İnstitutunda M20 EHM-i yaradıldı. Bu maşın M220 və M222 maşınları ailəsinin əcdadı oldu. Onun orta məhsuldarlığı saniyədə 20 min əməliyyat idi. Həcmi 4096 ədəd 45 mərtəbəli sözdən ibarət olan yaddaş ferrit nüvəciklərdə idi. Üç maqnit barabanı 12 mindən cox söz yadda saxlaya bilirdi. Daxiletmə perfokartlar vasitəsilə, çıxış isə çap qurğusu vasitəsilə yerinə yetirilirdi. Maşın blok prinsipilə qurulmuşdu ki, bu da onun təmirini sadələşdirirdi. Onda 4,5 min elektron lampa və 3,5 min yarımkeçirici dioddan istifadə olunurdu.

1957-ci ildə Penza şəhərində birünvanlı lampalı "Ural-1" EHM-i yaradıldı. Maşının ölçüləri böyük olsa da, məhsuldarlığına görə onu kiçik maşınlara aid edirdilər. Hesab etmək olar ki, kiçik EHM-lərin tarixçəsi "Ural-1" maşınından başlayır. Sürətinin az olması üzündən (saniyədə 100 əməliyyat) maşının sürətli yaddaş qurğusuna ehtiyacı yox idi və buna görə də əsas yaddaş kimi həcmi 1024 dənə 36-bitlik sözdən ibarət olan maqnit barabanından istifadə olunurdu ki, sonralar da onlar ferrit yaddaş qurğuları ilə əvəz olundu. 1964-1971-ci illərdə aralarında proqram və aparat uyğunluğu olan "Ural-11", "Ural-14", "Ural-16" modelləri istehsal olundu.

"Minsk" seriyalı maşınlar XX əsrin 70 və 80-ci illərində əsasən mühəndis və elmi hesablamalar aparmaq üçün istifadə olunurdu. Onlardan biri, "Minsk-22" uzun müddət SSRİ-nin baş univermağı olan QUM-un hesablama mərkəzinin kompüteri idi ki, onun da xarakteristikaları belə idi: saniyədə 5 min əməliyyat, yaddaş – 37 mərtəbəli 8 min söz. Onda (maqnit lenti 1,6 mln.





Ural-1 maşını (solda). Minsk-22 maşını (sağda).

söz tuturdu) univermağın bütün anbarlarında olan məhsulların hərəkəti haqqında məlumat saxlanılırdı və əmək haqqı hesablanırdı. Lakin hesablama texnikasına müəyyən inamsızlıq üzündən univermağın rəhbərliyi paralel olaraq maşının hesablamalarını yoxlayan mühasiblərin böyük ştatını da saxlayırdı. EHM-in Assembleri kiril əsaslı mnemonikaya malik idi və SKD (simvolik kodlaşdırma dili) adlanırdı.

Bu sıranın digər kompüteri "Minsk-32" saniyədə 25 min əməliyyat aparırdı və onun yaddaşı 65 min 37 mərtəbəli sözdən ibarət idi. Maşın "Minsk-22" maşını ilə proqram uyğunluğuna malik idi. Ləng və sürətli kanallar ona maqnit barabanları qoşmağa imkan verirdi və bu da məhsuldarlığı xeyli artırırdı. "Minsk-32" kompüteri yüksək səviyyəli Algams (Algol dilinin modifikasiyası) və Cobol dillərindən kompilyatorlara malik idilər.

Sovet dövlətinin superEHM-lərinə (yüksək sürətli hesablamalar üçün nəzərdə tutulmuş maşınlar) 1970-1980-ci illərdə işlənib hazırlanmış çoxprosessorlu hesablama kompleksləri (ÇHK) olan "Elbrus" maşınları aid edilir. "Elbrus-1" saniyədə 10 milyon əməliyyat icra edirdi. Maşının konfiqurasiyasına 10-a qədər mərkəzi prosessor daxil idi ki, bunlar da ümumi yaddaşa müraciət edirdilər. Xarici qurğularla mübadiləni giriş-çıxış prosessorları yerinə yetirirdi və onlar da faktiki olaraq ixtisaslaşmış prosessorlar idi. Maşın maksimum dörd belə prosessoru idarə edə bilirdi. Digər xüsusi EHM-lər – verilənlərin ötürülməsi prosessorları – istifadəçilər ilə əlaqəni təmin edirdilər.

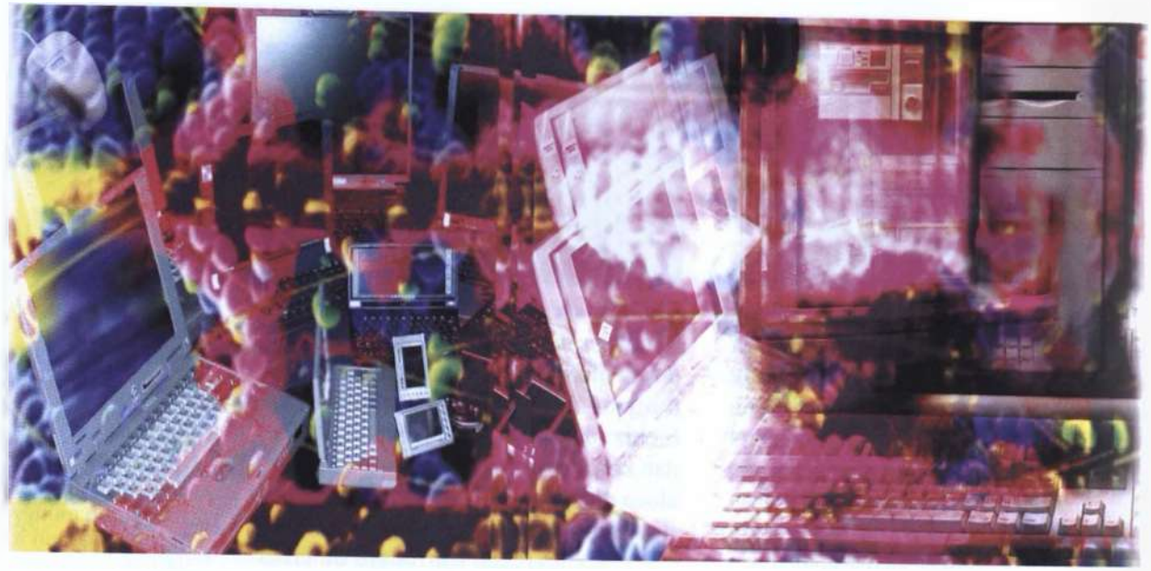
ÇHK-da çox sayda adi olmayan həllərdən istifadə edilmişdi, məsələn, yaddaşda olan hər bir kəmiyyət əlavə əlamətlə – *teq* (idarəedici mərtəbə) ilə təchiz olunmuşdu. Orada yadda saxlanan kəmiyyətin tipi haqda informasiya, eləcə də, oxuma və ya yazmadan mühafizə əlaməti saxlanılır. Mərkəzi prosessorun arxitekturası Burroughs amerika firmasının müvafiq komplekslərilə bir çox oxşar xüsusiyyətlərə malik idi.

70-ci illərin sonunda Sovet İttifaqında universal çoxprosessorlu komplekslərin dördüncü nəsil nümayəndəsi olan "Elbrus-2" maşınının istehsalına başlanıldı. Hər bir prosessorun məhsuldarlığı saniyədə 10 min sözdən çox idi. Yekun məhsuldarlıq saniyədə 100 milyon əməliyyat təşkil edə bilirdi.

Ölkənin kompüter quruculuğu keyfiyyətli elektron komponentlərin hazırlanması üzündən ciddi çətinliklərlə qarşı-qarşıya gəlirdi. Çox güman ki, məhz buna görə IBM firmasının System/360 seriyasının təcrübəsinə əsaslanan çox da uğurlu olmayan EC (Единая Серия – vahid seriya) EHM təcrübəsi təkrar edildi. Bir çox uğurlu (və o qədər də uğurlu olmayan) həllər qərb analoqlarından təkrarlanmışdı. Kiyev mini-maşını CM-4 və Zelenoqrad mini maşını "Elektronika-79"-un örnəyi isə amerikanın DEC firmasının PDP-11 maşını oldu. Lakin ölkə kompüterləri istifadəçinin əsas meyarı olan – etibarlığa görə qərb analoqlarına uduzurdu. IBM PC fərdi kompüterlərinin meydana gəlməsi və hər yerə nüfuz etməsilə nə qərb rəqibləri, nə də rusiyalı həmkarları bacara bilmədilər.

1961-ci ildə akademik V.M.Qluşkovun rəhbərliyi altında yarımkeçirici bazasında "Dnepr" EHM-i yaradıldı. Bu zaman dünya rekordu müəyyən olundu: maşının ideyasından istehsalına qədər cəmi üç ilə yaxın bir vaxt keçdi.

Akademik V.M.Qluşkovun nəzəri və tətbiqi kibernetika, rəqəmli avtomatlar nəzəriyyəsi ("Rəqəmli avtomatların sintezi" monoqrafiyaları, "Успехи математических наук" jurnalındakı "Avtomatların mücərrəd nəzəriyyəsi" məqaləsi) EHM-lərin nəzəriyyəsinin əsasını qoydu. Sovet kibernetiki Qluşkov dünyada amerikalı N.Vinerdən heç də az məşhur olmadı.



MÜASİR KOMPÜTER

KOMPÜTERLƏRİN DÖRDÜNCÜ NƏSLİ

BÖYÜK VƏ ÇOX BÖYÜK İNTEQRAL SXEMLƏR

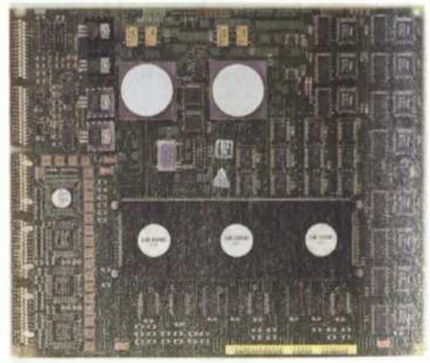
Hələ 1982-ci ildə Xu Min Tuq və Amar Qupta "Scientific American" jurnalında dərc olunmuş məqalələrində yazırdılar: "Əgər son 25 ildə hava nəqliyyatı sənayesi hesablama texnikası kimi sürətlə inkişaf etmiş olsaydı, onda "Boing-767" təyyarəsini bu gün cəmi 500 dollara əldə etmək və bu təyyarə ilə cəmi 19 litr yanacaq işlətməklə 20 dəqiqəyə yer kürəsinin başına fırlanmaq mümkün olardı. Bu analogiyaya görə (hətta tam dəqiq olmasa belə) enerji istehlakı dəyərinin düşmə tempi və hesablama maşınlarının sürətinin artımı haqqında mühakimə yürütmək olar. Kompüterlərin məntiqi elementlərinin qiymətləri hər il 25% azalır. Yaddaş qurğuları üçün isə bu göstərici daha yüksəkdir – 40%. Axırıncı 25 il ərzində hesablama sürəti 200 dəfə artıb; bununla yanaşı müasir EHM-lərin ölçüləri və elektrik enerjisinə tələbatı 25 il əvvəlki maşınlara nisbətən 10 min dəfə azalıb".

Hər şey isə XX əsrin 60-cı illərində başlamışdı, o dövrdə kompüter hissələrinin istehsalında əsas texnologiya, silisiumdan hazırlanan və izoləedici materiala "qablaşdırılmış" kiçicik düzbucaqlı yarımkeçiricilər – *çiplər* oldular. Onlar "inteqral sxemlər" adlandırıldılar. Sxemin əsas elementi tranzistor olduğundan, onun "tutumu" tranzistorlarla ölçülürdü. İnteqral sxemlərin istehsal texnologiyası, bir neçə tranzistordan ibarət olan *kiçik inteqral sxemlərdən* (KİS) başlayaraq, özündə minlərlə, yüz minlərlə və daha çox tranzistoru birləşdirən *böyük inteqral sxemlərə* (BİS), eləcə də *çox böyük inteqral sxemlərə* (ÇBİS) qədər bir inkişaf yolu keçmişdir. 1986-cı ildə yaranmış "çox böyük inteqral sxem" anlayışı yaddaş tutumu 1 Mbayt olan sxemlərə aiddir.

Hər bir belə çip milyondan çox tranzistordan ibarət idi. İnteqral sxemləri yalnız avtomatlaşdırılmış istehsal şəraitində hazırlamaq mümkündü, buna görə də çiplərin qiyməti böyük deyildi. Böyük inteqral sxemlərin kütləvi istehsalına 70-ci illərdə, EHM yaddaşlarının və cib kalkulyatorları üçün mikrosxemlərin istehsal edilməsilə başlanıldı.

Müasir kompüter, birləşmələr üçün metal kontaktları olan inteqral sxemlər çoxluğunun, qidalandırma mənbəyinin və digər kiçik elementlərin (rezistorlar, kondensatorlar və s.) toplusudur. İnteqral sxemlər bir-birindən formalarına görə fərqlənən korpuslarda yerləşdirilirlər. Ən "köhnə" və bu gün də geniş yayılmış inteqral sxemlər (mikrosxemlər) – DIP-lərdir (*ing.* dual in-line package). Onlarda çıxışlar ("ayaqlar") ən yaxın kontaktlar arasında 2,54 mm ara buraxılmaqla iki paralel sıra şəklində təşkil edilib. Bəzən çıxışlar mikrosxemin yalnız bir tərəfində yerləşdirilir, belə çip SIP (*ing.* single in-line package) adlanır. Daha mürəkkəb, yüz və daha artıq kontaktı olan çip başqa konfiqurasiyaya malikdir, burada çıxışlar mikrosxemin bütün kənarları boyunca yerləşdirilib. Bu çip PGA (*ing.* pin-grid array) adlanır.

İnteqral sxemləri naqillər dəstəsilə, yaxud da daha sadə qaydada əlaqələndirmək olar. Bunun üçün nazik izolyator təbəqəsində (məsələn, şüşə lifli təbəqədə) inteqral sxemlər və digər elementlər elə yerləşdirilir ki, onların kontaktları təbəqədə hazırlanmış (deşilmiş) deşiklərdən təbəqənin digər tərəfinə çıxsınlar. Bu təbəqə həm də mexaniki bərkidici rolunu oynayır. Sonra çıxmış "ayaqları", əvvəlcədən hazırlanmış elektrik sxem üzrə öz aralarında birləşdirmək tələb olunur. Birləşməni sadəcə olaraq hər hansı bir cərəyankeçirici

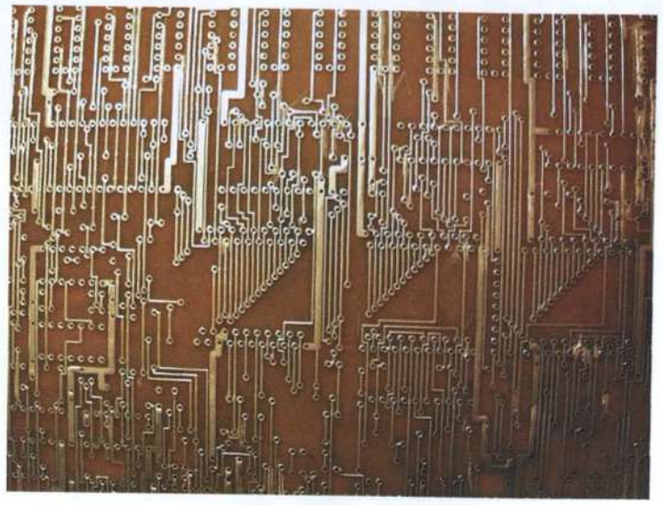


VAX EHM-in lövhəsi.

materialla, məsələn, mislə "çəkmək" olar. Belə bir "şəkil", birləşdirici naqilləri əvəz edəcəkdir. Sənaye istehsalı zamanı bu cür *çap lövhəsi* (*ing.* printed-circuit board – PCB) naqillərlə sadə birləşdirməyə nisbətən daha ucuz və texnolojidir. Belə ki, bütün lövhənin birləşdirmə şəklini bütov kəsəcək trafaret qoymaqla "çəkmə"ni bir mərhələdə yerinə yetirmək olar. Bəzən lövhədə kəsişmələr olmadan bütün lazımi birləşmələri yerinə yetirmək mümkün olmur. Bu halda çoxlaylı quraşdırmadan istifadə edilir. Mini-kompüterlərdə yeddi və hətta on laylı quraşdırmadan istifadə edilmişdi. Adətən, bir cüt lay lövhədəki mikrosxemlərə cərəyanın çatdırılması üçün ayrılır, qalanları isə birləşdirici rolunu oynayırlar. Bu cür lövhələrin hazırlanması çox əmək tələb edir. Lövhədə inteqral sxemlər və digər elementlər nə qədər çox olarsa, onun quruluşu da bir o qədər mürəkkəb olar.

Bundan əlavə başqa problem də mövcuddur. Çiplər iş zamanı tranzistorlar kimi istilik ayırırlar. Buna görə də bərk qızmanın, mikrosxemlərin və hətta bütün lövhənin sıradan çıxmasının qarşısını almaq üçün istehsalçılar mikrosxemləri bir-birindən təhlükəsiz məsafədə yerləşdirmək məcburiyyətində qalırlar. Bəzən temperatur rejimini gözləmək üçün inteqral sxemə metal radiator və ya hətta ventilyator yerləşdirirlər.

İnteqral sxemin "atası" sayılan Cek Kilbi, özünün bu dahiyanə icadına görə Nobel mükafatına yalnız 2000-ci ildə layiq görülmüşdür.



Çap lövhəsi.

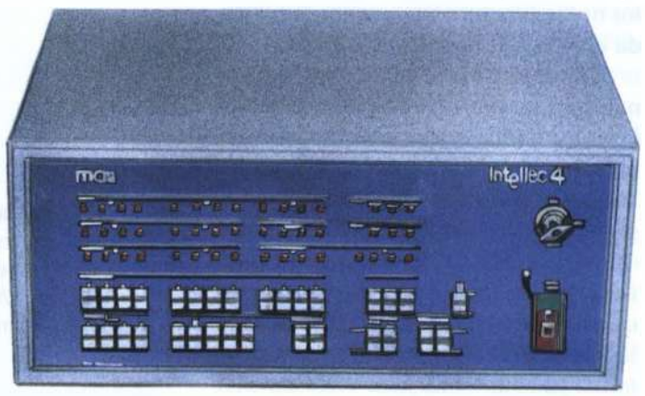
MİKROPROSESSORLAR

İnteqral sxemlərin yaranması ilə ilk dəfə olaraq bütov bir prosessoru (və ya hətta bütov bir kompüteri) yeganə bir çipdə hazırlamaq, yəni mikroprosessor yaratmaq imkanı yarandı.

Intel 4004 mikroprosessorunun məhdud imkanları, inteqral sxemlərin sıxlığı üzərinə qoyulan məhdudiyyətlərlə izah olunur (1971-ci ilin çipinin ölçüləri – 1 sm²).

Birinci əsl mikroprosessor 1971-ci ildə Intel (ABŞ) firmasının istehsal etdiyi Intel 4004 mikroprosessoru oldu. İlk olaraq ondan MSC-4 kalkulyatorlarında (böyük inteqral MOY-sxemləri) dörd çipin tərkibində istifadə edildi. Intel 4004-də maşın sözünün uzunluğu cəmi 4 bitdir. Bütün komandaları bu cür bir sözdə "yerləşdirmək" sadəcə olaraq mümkün deyil, çünki 4-bitlik sözün bitlərinin müxtəlif kombinasiyalarının sayı 2^4=16 olur, yəni bu "körpədə" bütün mümkün ünvanlaşdırma üsulları ilə birlikdə 16 komanda qəbul etmək imkanı vardı. Buna görə də göstərişlər 8 bit uzunluqlu idi: 2^8=256. Verilənlər və proqramlar müxtəlif yaddaşlarda saxlanılırdı. Verilənlər üçün cəmi 1 Kbayt yaddaş və proqramlar üçün 4 Kbayt yaddaş ayrılırdı. Intel 4004 mikroprosessorunda 12-bitlik komandalar sayğacı (PC) və alt proqramların (CALL) yerinə yetirilməsini və alt proqramlardan qayıdışı (RET) təmin edən, dərinliyi cəmi 4 çağırışa bərabər olan kiçicik stek nəzərdə tutulmuşdu. Bundan əlavə o, 8-bitlik 8 registrdə qruplaşa bilən 4-bitlik 16 registrə malik idi. Intel 4004-cəmi 46 göstərişə malik idi.

İntel 4004 mikroprosessoru əsasında Intellec 4 maşını.



MSC-4-dən sonra müxtəlif istehsalçıların çoxlu sayda mikroprosessorları meydana gəldi. İnteqral sxemlərin sıxlığının artması ilə maşın sözünün ölçüsü də 32 bitə qədər artdı. Bundan əlavə mikroprosessorların komandalar çoxluğu da genişləndi, baxmayaraq ki, "böyük" prosessorlarla müqayisədə bu çoxluq çox da böyük deyildi (məsələn, sürüşkən vergüllə hesablamalar aparmaq üçün komandalar yox idi), mikroprosessorların məhsuldarlığı, yəni bir saniyədə yerinə yetirilən komandaların sayı da artdı.

Mikroprosessorlar artıq, məsələn, yanacağın verilmə komponentlərini (avtomobil mühərriklərindəki "kompüterlər") və mürəkkəb məişət avadanlıqlarını idarə edə bilirdi. Onlar tədricən "böyük" EHM prosessorlarının cəhətlərini özündə əks etdirməyə başladılar. Mikroprosessor istehsalının artması ilə onların sənaye standartları yaranmağa başladı.

Z-80

1974-cü ilin aprel ayında Intel firması 8-bitlik Intel 8080 mikroprosessorunu istehsal etdi. Bu prosessor bütün göstəricilərinə görə əvvəlki bütün prosessorları üstələyirdi. Lakin Intel firmasına nəzərən daha kiçik olan Zilog firmasının hazırladığı və istehsal etdiyi Z-80 mikroprosessoru özünə əbədi şöhrət qazandı. O, Intel 8080 ilə tam uyğunlaşdırılmışdı, yəni Z-80 nəinki Intel 8080-in edə biləcəklərini bacarırdı, üstəlik daha artıq imkanlara malik idi. O vaxtdan etibarən Intel firması 16-bitlik, daha sonra isə 32 və 64-bitlik prosessorların istehsalına keçdi. Bu prosessorlar müasir fərdi kompüterlərin əsasını təşkil edir.

Z-80 indiyə qədər istifadə edilir, 80-ci illərdə isə o, əsl böyük müvəffəqiyyət qazanmışdı. 8-bitlik mikroprosessor 16-bitlik ünvanlaşmadan istifadə edirdi, yəni birbaşa olaraq 64 kbaytlıq yaddaşa müraciət edə bilərdi. Ona Intel 8080-ə nəzərən 80-dən artıq yeni göstərişlər əlavə olunmuşdu: 1, 4, 8 və 16-bitlik verilənlərlə əməliyyatlar, eləcə də bloklu komandalar (bir komanda 64 kbayta qədər verilənlərin göndərilməsi) və ya yaddaşda lazımi baytın axtarılması üçün xüsusi əmrlər. Prosessor iki registrlər toplusuna və onlar arasında qoşulma əməliyyatlarına malik idi ki, bu da qırılmanın təşkilini sadələşdirirdi (proqramda registrlərin vəziyyətini yadda saxlamaq tələb olunmurdu). Z-80-ə iki 8-bitlik indeks registri də (IX və IV) əlavə olunmuşdu.

Mikroprosessorda əməliyyatların yerinə yetirilmə sürəti inteqral sxemin xüsusiyyətlərilə təyin edilirdi və iş zamanı kvars generatorla – saatlarla müəyyən edilirdi. Bu prosessorun takt tezliyi adlanırdı.

Birinci Z-80 2,5 MHers, sonrakı Z-80 C isə artıq 8 MHers takt tezliyinə malik idi. Bir registrdan digərinə ən "qısa" göndəriş komandanı prosessorun 4 taktı hesabına yerinə yetirilirdi. Birinci Z-80 saniyədə 600 mindən çox komanda yerinə yetirə bilirdi, bu isə "körpə" haqqında kifayət qədər yaxşı təəssürat yaradırdı.

"CISC"-LƏR VƏ "RISC"-LƏR

Mikroprosessorların əksəriyyəti komandalar toplusu ilə verilən CISC (*ing.* Complex Instruction Set Computer) arxitekturalı standarta aiddir. Onun əcdadı IBM firması hesab edilir.

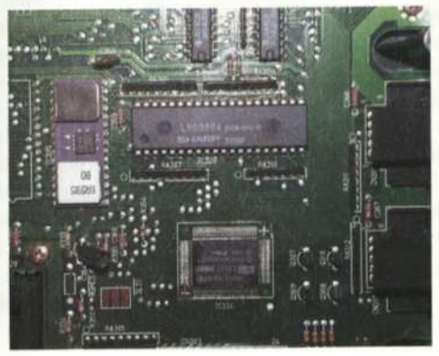
CISC mikroprosessorlarının işlənməsi sahəsində lider olan Intel firması özünün x86 və Pentium seriyaları ilə məşhurdur.

CISC prosessorları üçün registrlər sayının çox da böyük olmaması səciyyəvidir, lakin komandaların sayı, ünvanlaşdırma üsullarının dolayısılıq səviyyəsi, komandaların formatları yüksək olaraq qalır.

Bu cür arxitektura mikrokompüterlər üçün standart hesab edilir.

RISC (*ing.* Reduced İnstruction Set Computer) – azaldılmış komandalar toplusuna malik prosessorlardır. Onlar çox sadələşdirilmiş məntiq – sxemə, çox sadə göstərişlərə və çoxsəviyyəli dolayısı olmayan ünvanlaşdırma üsullarına malikdirlər. Buna görə prosessor kiçik və sadə olub sürətlə işləyir.

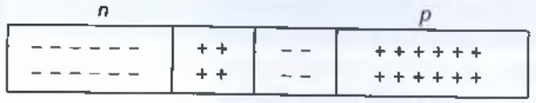
RISC-in tarixi CDC 6600 kompüterlərindən başlayır. Bu zaman Seymur Krey hesablamaların sürətləndirilməsi üçün komandalar yığınını sadələşdirilməsini təklif edir və bu arxitektura Cray seriyalı superkompüterlərin yaradılması zamanı müvəffəqiyyətlə tətbiq olundu. Lakin tamamilə RISC arxitekturası aşa-



Z-80 prosessoru əsasında MSX kompüterinin lövhəsinin fraqmenti.

İNTEQRAL SXEMLƏRİN NÖVLƏRİ

Tranzistor və diodların hazırlandığı yarımkeçiricilər, elektron – *n* (*ing.* "negative" – "mənfi") və deşik – *p* (*ing.* "positive" – "müsbət") keçiricili yarımkeçiricilərə bölünürlər. Yarımkeçirici diodların iş prinsipi, *p-n* keçidinin xassələrinə əsaslanır, bu zaman *p* və *n* tipli iki yarimkeçirici təmasda olurlar. Təmas yerində müsbət yüklərin (deşiklərin) *p* oblastından *n* oblastına, elektronların isə əksinə *n* oblastından *p* oblastına diffuziyası baş verir. Lakin xarici təsir (elektrik cərəyanı) olmadıqda prosess stabilləşir, çünki bağlayıcı adlanan bir lay əmələ gəlir.



P oblastına elektrik cərəyanı mənbəyinin "plyusunu" *n* oblastına isə uyğun olaraq "minusunu" birləşdirdikdə bağlayıcı lay dağılır və bu cür diod cərəyan keçirməyə başlayır. Əgər cərəyan mənbəyini əksinə qoşsaq, yəni *p*-yə "minus", *n*-ə isə "plyus" qoşsaq, onda cərəyan faktiki olaraq sıfıra bərabər olacaqdır və bağlayıcı lay genişlənəcəkdir. Yarımkeçirici diodların bu əsas xassəsi onlardan cərəyan düzləndiricisi kimi istifadə etməyə imkan verir. Yarımkeçiricilərin çoxu bəzi metalların oksidləri və məxsusi olaraq bu metalların özlərinin müxtəlif əlavələrilə silisiumdan və germaniumdan hazırlanırlar. Bu əlavələrdən asılı olaraq onlar *n* və ya *p* tipli olurlar.

Tranzistor belə yarımkeçirici materiallardan ibarət üç laylı struktura malikdir, lakin onun işinin əsasında bir deyil, iki *p-n* keçidi durur. Kənar laylar *emitter* və *kollektor*, orta (adətən, çox nazik bir neçə mikron qalınlıqlı) lay isə *baza* adlanır. Tranzistora cərəyan mənbəyi qoşduqda bazadan kollektora axan cərəyanın kiçik dəyişmələri, emitterdən kollektora doğru daha böyük cərəyan dəyişmələri yaradır. Tranzistorun bu xassəsi, ondan cərəyan gücləndiricisi kimi istifadə etməyə imkan verir.

Sadə üçlaylı tranzistor bipolyar tranzistor adlanır. Əgər baza *p*-tipli yarımkeçiricilərə, emitter və kollektor isə *n*-tipə aiddirsə, onda tranzistor *n-p-n* tipli sayılır, əgər əksinə, daxildə n-tipli, kənarlarda *p*-tiplidirsə, tranzistorun tipi *p-n-p* olur. Bipolyar tranzistorun əsas nöqsanı onun çoxlu enerji sərf etməsi və istilik ayırmasıdır.



Tranzistor kəsiyi.

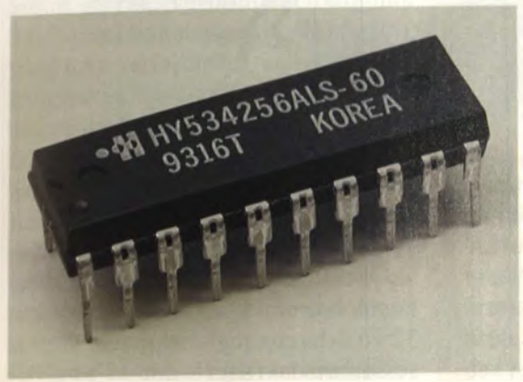


Səyyar tranzistor.

Alternativ olaraq səyyar tranzistor işlənib hazırlanmışdı. O, birpolyarlı yarımkeçirici qurğu olub, *başlanğıc*, *axar* və *bağlayıcı* adlanan çıxışlara malikdir. Bağlayıcı və axara (və ya uyğun olaraq başlanğıca) gərginlik verdikdə yük daşıyıcıları – elektronlar *n*-tipli keçiricilik olan sahələrdə (və ya *p*-tipli keçiricilik olan sahələrdə deşiklər) bağlayıcının altında yaranan nazik keçirici kanaldan keçirlər. Adətən kanalda cərəyan (bağlayıcı və axar arasındakı) keçidə əlavə olunan gərginliyin yaratdığı elektrik sahəsinin köməyilə dəyişdirilir. Əgər axar və başlanğıc *n*-tipli yarımkeçiricidən hazırlanmışsa, tranzistoru *n*-kanallı, *p*-tipli yarımkeçiricidən hazırlanmışsa *p*-kanallı adlandırırlar.

Bağlayıcısı kanaldan dielektrik layı ilə ayrılmış səyyar tranzistorlar da mövcuddur. Belə tranzistorlar böyük xüsusi müqaviməti olan yarımkeçirici təbəqədən ibarət olurlar. Bu təbəqə başlanğıc və axar elektrodların yerləşdirildiyi bir-birinə əks tipli elektrik keçiriciliyi olan iki sahədən ibarətdir. Onlar arasındakı yarımkeçiricinin səthi nazik dielekrik üzərinə metal elektrod-bağlayıcı yerləşdirilmiş qat ilə (adətən silisium oksidi SiO_2) örtülüb. Buna görə də izolə olunmuş bağlayıcılı – MDY (metal-dielektrik-yarımkeçirici) səyyar tranzistorları çox vaxt MOY- tranzistorlar (metal-oksid-yarımkeçirici) adlandırırlar. Bundan əlavə laylı dielektriklərdən də, məsələn SiO_2-Al_2O_3 istifadə olunur. Onlar uyğun olaraq MAOY (metal-alüminium-oksid-yarımkeçirici) tranzistorlar adlanırlar. Bipolyar tranzistorlardan fərqli olaraq MOY-tranzistorlar daha qənaətcildirlər.

1948-ci ildə icad olunmuş tranzistor bütün müasir mikrosxem və mikroprosessorların əsasını təşkil edir. Əbəs yerə deyil ki, onun müəllifləri Uilyam Şokli (1910-1989), Uolter Bratteyn (1902-1987) və Con Bardin (1908-1991) 1956-cı ildə fizika üzrə Nobel



Mikrosxem.

mükafatını almışdılar. Hesablama texnikasında kollektordan böyük cərəyan axdığı halda tranzistorun vəziyyətini şərti olaraq, məsələn, 1, kiçik cərəyan axan halda isə 0 qəbul etmək olar. Lakin tranzistordan istifadə şərtləri informasiyanın saxlanılması zamanı (0 və ya 1) onun enerjiyə olan tələbatı ilə məhdudlaşmır, burada onun bir vəziyyətdən digərinə: 0-dan 1-ə və ya 1-dən 0-a keçidi üçün lazım olan vaxt da əhəmiyyət kəsb edir.

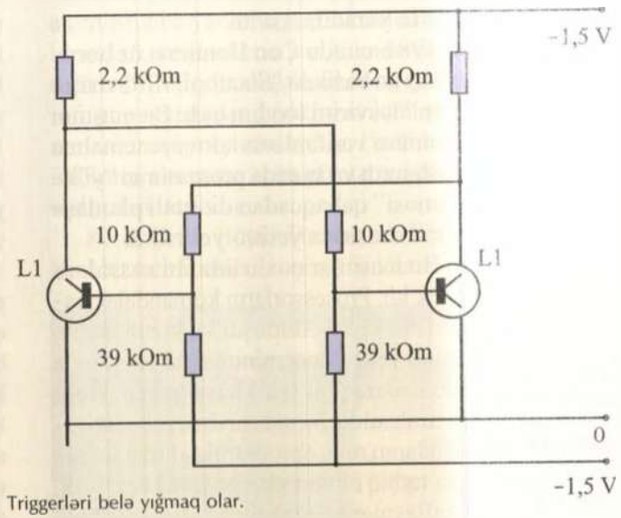
Əvvəllər tranzistorlar ayrı-ayrı elementlər şəklində hazırlanırdılar və bir neçə naqil çıxışları olan onlarla millimetr diametrli silindir şəklində idilər.

Lakin hesablama texniksında ayrı-ayrı tranzistorlardan deyil, onların kombinasiyalarından istifadə edilir. Adətən iki MOY-tranzistoru: *p*-kanallı və *n*-kanallı tranzistoru birləşdirirlər. Bu cür cütü KMOY-cüt (komplimentar MOY) adlandırırlar. KMOY-cütün yaxşı cəhəti odur ki, keçid iki mümkün dayanıqlı vəziyyətdən birində olduqda (bu vəziyyətlərə isə 0 və ya 1 uyğun gəlir) praktiki olaraq enerji işlətmir. KMOY-texnologiyalarının tətbiqi, səmərəliliyi əhəmiyyətli dərəcədə artırmağa, sxemlərin enerji sərf etməsini və uyğun olaraq istilik ayrılışını azaltmağa imkan verir. Bu texnologiya TTL-sxemlərin (tranzistor – tranzistor məntiqi) əsasını təşkil edir. MOY-qurğularının ölçüləri nə qədər kiçik olarsa, vahid sahədə onlardan bir o qədər çox yerləşdirmək olar və ya başqa sözlə desək, yerləşdirmə sıxlığını artırmaq olar. KMOY-cütləri çox az istilik ayırdıqlarından mikrosxemin bir kristalında milyonlarla element yerləşdirmək mümkündür. Bu zaman sxemlərin sıradan çıxmasına səbəb olacaq həddindən artıq qızmanın qarşısını almaq olur.

Tranzistorlardan iki və daha artıq çıxış siqnalını bir çıxış siqnalına (məsələn, məntiqi "və", "və ya" və s.) çevirən elektrik sxemi-ventil və ya bir bit informasiyanı yadda saxlama sxemi – trigger (yəni iki mümkün dayanıqlı vəziyyətdən: 0 və ya 1-dən birində olan) hazırlayırlar. Triggeri bir cüt tranzistordan da qurmaq olar, orta çətinlikli məntiqi ventilin hazırlanmasına isə adətən dörd tranzistor gedir.

İnteqral sxemlər (çiplər) onlara daxil edilmiş tranzistorların sayı və ya məntiqi ventillərin sayı ilə, yəni bu qurğuların sıxlığı ilə təsnif edilirlər. Kristalda 100 min tranzistor var deyilirsə, bu o deməkdir ki, sxem 50 min KMOY-cütünə və ya 12500 mətiqi ventilə malikdir.

İnteqral sxemlərin çoxunun ənənəvi olaraq silisiumdan hazırlanmasına baxmayaraq, digər materialdan - qallium arsenidindən hazırlanan yarımkeçiricilərdən də istifadə olunur. Belə inteqral sxemlərin istehsalı çox mürəkkəbdir, lakin daha yüksək iş sürətinə malikdirlər (silisium sxemlərə nəzərən təqribən beş dəfə çox).



Triggerləri belə yığmaq olar.



ğıdakı üç tədqiqat layihəsinin nəticəsi kimi formalaşdırılmışdır: IBM firmasının 801 prosessoru, Kaliforniya universitetinin (Berkli) RISC prosessoru və Stanford universitetinin MIPS prosessoru.

1970-ci ildə IBM tərəfindən ilk dəfə təklif olunmuş RISC texnologiyası heç vaxt dərc olunmamışdı – onun əsasında seriyalı kompüterlər hazırlanmamışdı.

1980-ci ildə Deyvid Patterson Kaliforniya universitetinin (Berkli) kollektivilə birlikdə RISC-I və RISC-II adlanan iki maşın yaratdı. Bu EHM-in əsas ideyası yavaş sürətli yaddaşın yüksək sürətli registrlərdən ayrılması, verilənlərin emalı üçün (hesablamalar üçün) ayrı-ayrı "sürətli" komandaların yaradılması və registrlərdən verilənlərin yaddaşa yüklənməsi və əksinə yaddaşdan registrlərə verilməsi üçün ayrı-ayrı "yavaş" komandaların yaradılması idi.

1981-ci ildə Con Hennessi öz həmkarları ilə birlikdə "Stanford MIPS maşınının" təsvirini təqdim etdi. Bu maşının işlənməsi verilənlərin konveyer emalına əsaslanırdı və burada prosessorun "yüklənməsi" qabaqcadan diqqətli planlaşdırma əsasında yerinə yetirilirdi.

Bu maşınlar çoxlu ümumi xassələrə malik idi. Prosessorların komandalar sistemi elə təşkil edilmişdi ki, komandanın yerinə yetirilməsi minimal takt hesabına (ideal olaraq bir takt) başa gəlsin. Həm də məhsuldarlığın artırılması üçün komandanın mikroproqramlaşdırma texnikası tətbiq olunmadan həyata keçirilirdi. Reallaşmanın maksimal sadələşdirilməsi və sürətinin artırılması üçün yalnız qeyd olunmuş uzunluqlu komandalardan istifadə olunurdu.

RISC-prosessorlarının digər xüsusiyyətləri arasında verilənlərin daha böyük həcmini saxlamağa imkan verən 32 və daha çox registrlərin (CISC-prosessorlarındakı sayı 8-dən 16-ya qədər olan registrlərlə müqayisədə) olmasıdır. İnformasiyanın emalı üçün adətən üç ünvanlı komandalardan istifadə edilir. Bu isə registrlərdə yenidən yüklənmə aparmadan çoxlu sayda dəyişənlərin saxlanmasına imkan verir.

RISC prosessorlarının universitet layihələrilə təsdiq olunmuş sadəliyi və böyük iş sürəti hesablama texnikası istehsalçılarının böyük marağına səbəb oldu və 1986-ci ildən başlayaraq RISC-prosessorlarının istehsalına başlandı.

Bu cür prosessorlardan biri Motorola və IBM firmasının birgə hazırladıqları Motorola Power PC 601 idi. Bu 80 MHers takt tezlikli 32-bitlik RISC-prosessordur. Növbəti prosessor Power PC 603 (KMOY-çip) 601-ə nisbətən iki-üç dəfə səmərəli olub cəmi 1,6 milyon tranzistordan ibarət idi və 80 MHers takt tezliyinə malik idi.

Motorola və IBM firmalarının sonrakı prosessorları bir-birinin ardınca istehsal olunurdu: PowerPC 604 (100 MHers) bir neçə aydan sonra meydana gəldi. PowerPC 615, Intel firmasının məşhur 8086 prosessoru ilə aparat uyğunluğuna malik olmaqla bu prosessora güclü rəqabət yaratmışdı. PowerPC 620 isə artıq 64-bitlik prosessor idi. RISC-prosessorlarının inkişafı xüsusi optimallaşdırıcı

kompilyatorların təkmilləşdirilməsi hesabına mümkün olmuşdu. "Ağıllı" kompilyator effektiv proqram kodunu təmin edərək, verilənlər çoxluğunu RISC-prosessorunun çoxlu sayda registrləri üzrə yerləşdirirdi və bununla da kompüterin "yavaş sürətli" yaddaşından verilənlərin yüklənməsi və boşaldılması komandalarından imkan daxilində az istifadə edilməsini təmin edirdi. Konveyer sistemləri kimi, kompilyator proqramı elə dəyişməli idi ki, prosessorun konveyerində müxtəlif hesablamaların eyni zamanda yerinə yetirilməsi kodunu verə bilsin.

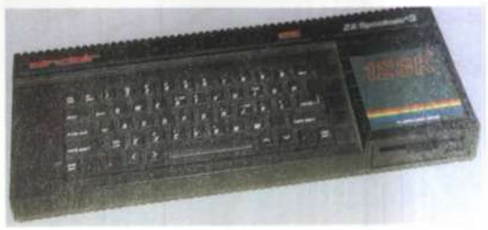
Geniş yayılmış CISC-prosessorları ilə uyğunlaşdırılması üçün RISC-prosessorları CISC-prosessorlarının xüsusi emulyasiyası rejiminə malik idilər və bu zaman kompilyasiya olunmuş proqram başqa prosessorda yerinə yetirilməsini hiss etmirdi.

Maraqlıdır ki, Intel firmasının (Pentium və Pentium Pro) prosessorlarında, eləcə də onun ardıcılları və rəqiblərində (AMD R5, Cyrix M 1, NexGen NX 586 və s.) RISC-mikroprosessorlarında realizə olunmuş, konveyer emal ideyası istifadə olunurdu. Buna görə də CISC və RISC prosessorları arasındakı bir çox fərqlər tədricən aradan qaldırılır.

MİKROKOMPÜTERLƏR

Tipik mikrokompüter, kifayət qədər yaddaşa malik, geniş periferiyası (giriş-çıxış qurğuları üçün çoxlu sayda portlar) olan, 50-yə yaxın inteqral sxemdən ibarət və yeganə çap lövhəsində realizə oluna biləcək bir qurğudur. Bu cür maşınları bir lövhəli mikroEHM-lər adlandırırlar.

Kiçik yaddaşlı və az sayda giriş-çıxış portları olan mikrokompüteri yeganə bir ÇBİS-çipdə istehsal etmək olar. Bir çipli mikroEHM-lər kompakt ölçülərinə, ucuz qiymətinə və sadə təmirinə (sıradan çıxanı atıb, yenisilə əvəz etmək)



Sinclair ZX Spectrum kompüteri.

görə geniş yayılmışdılar. Onların istehsalı birinci mikroprosessorların yarandığı dövrdən, 70-ci illərin ortalarından başlanmışdı. Bu mikroEHM-lər əsasən müxtəlif maşın və mexanizmlərin proqramlaşdırılan kontrollerləri kimi (məsələn, ədədi proqramla idarə olunan dəzgahlar, mikrodalğalı sobalar və s.) istifadə olunurdu.

Geniş tətbiqli EHM-in yeni sinfi – fərdi kompüterlər meydana gəldi. Bunlar ofisdə, alimin kabinetində, məktəblinin partasında quraşdırmaq üçün nəzərdə tutulan, kiçik və ucuz mikrokompüterlər idi. Onlardan bəziləri akkumulyatorlardan enerji alırlar və o qədər yığcamdırlar ki, onları özünüzlə səyahətə də götürmək olar.

Tipik fərdi kompüter klaviaturadan, videomonitordan və sistem blokundan (mikroprosessorlu lövhə burada yerləşdirilir) ibarət olur. Kompüter xarici aləmlə rabitə yaratmaq üçün telefon xətlərindən, verilənlərin saxlanması üçün müxtəlif maqnit disklərindən, qrafik informasiyanın daxil edilməsi üçün skaner və manipulyatorlardan, "möhkəm üz"lərin alınması üçün printer və plotterlərdən istifadə edir.

80-ci illərdə Böyük Britaniyada Klayv Sinkler tərəfindən işlənmiş Sinclair ZX



Ən birinci mikro EHM-lərdən olan intellec 8 (Intel 8008 prosessoru əsasında qurulmuşdu), təqribən 2,5 min dollar dəyərində idi.

Apple I kompüteri.



Spectrum modeli çox populyar idi. Maşın klaviatura ilə bir korpusda yerləşirdi, monitor kimi adi televizordan istifadə olunurdu. Bu maşın cəmi bir neçə yüz dollar dəyərində idi və buna görə də qiymət baxımından hər bir kəs üçün əlçatan idi. Bu ev kompüteri 48 kbayt yaddaşa malik idi və Z-80 mikroprosessoru əsasında qurulmuşdu.

1982-ci ildə Yaponiyada bu mikroprosessor əsasında digər məşhur maşın – MSX kompüteri meydana gəldi. Bu EHM Yaponiya, Koreya, Cənubi Amerika, Niderland və Fransada geniş yayıldı. SSRİ-yə gətirilmiş dörd min MSX, 1986-ci ildən məktəb informatikası üçün baza təşkil etmişdi. MSX kompüterlərində 3,58 MHers takt tezliyi olan Z-80 prosessorundan istifadə olunurdu. Texas Instruments (TI 9918 və ya TI 9928 çipləri) firmasının yaratdığı videoprosessor (ekranda təsviri formalaşdıran prosessor) IBM PC XT və AT (1983–1988-ci illər) videoprosessorunu təsvir etdiyi rənglərin sayına görə çox üstələyirdi. Bu maşın EHM-lərdə nadir rast gəlinən audiosistemlə – üç kanallı musiqi sintezatoru ilə təchiz olunmuşdu. MSX – 1,64 Kbayt yaddaşa, MSX-2 isə 128 Kbayt (MSX-2-nin maksimal yaddaş həcmi – 4 Mbayt idi) yaddaşa malik idi. Yəqin ki, MSX maşınları 8-bitlik prosessor əsasında qurulan ən yaxşı kompüterlərdir.

İlk dəfə 1981-ci ildə təqdim olunmuş IBM PC seriyalı maşın hal-hazırda fərdi kompüterlər üçün faktiki olaraq standart olmuşdur. Lakin IBM PC XT (Intel 8086 prosessoru əsasında olan) və IBM PC AT (Intel 80286 prosessoru əsasında olan) kompüterlərinin birinci modelləri videoprosessorlar baxımından digər maşınlardan geri qalırdılar. Düzdür, IBM PC tipli maşınlar AMD (*ing.* American Micro Device) firmasının mikroprosessorları əsasında hazırlanmağa başlanandan sonra öz xarakteristikalarını xeyli yaxşılaşdıra bilmişdilər.

Digər məşhur fərdi kompüter – Macintosh maşınıdır. ABŞ-da fərdi kompüterlərin yaradıcıları Stiven Cobs və Stefan (Stiv) Voznyak hesab edilirlər. 1976-cı il aprel ayının 1-də onlar Palo-Alto (Kaliforniya ştatı) şəhərində Apple firmasını yaratdılar. Onların məşhur qarajlarında yığdıqları ilk kompüter Apple I adlanırdı və cəmi 8 Kbayt yaddaşa malik idi. 1977-ci ildə onlar özlərinin birinci Apple II kompüterini işləyib hazırlamış və satmışdılar. Bu kompüter birbaşa televizora birləşdirilirdi və onda istifadə olunan 6502 markalı prosessor cəmi 25 dollara idi. Bu kompüterlərin seçilməsində qiymət əsas faktor idi. Prosessorun imkanları xeyli zəif idi, burada beş registr var idi ki, onlardan ikisi (PC və SP) xitmətçi registr, yerdə qalanları isə 8-bitlik idilər.

1983-cü ildə firma tərəfindən qrafik informasiyanın daxil edilməsi üçün nəzərdə tutulmuş çox qəribə manipulyator qurğusu olan Lisa kompüteri yaradıldı. Alim və mühəndislərin bir çoxu bu yeniliyə şübhə ilə yanaşdılar. Bu qurğunun adı – "siçan" idi. Lisanın qiyməti çox baha idi – təqribən 10 min dollar. 1984-ci ildə nisbətən ucuz olan Macintosh kompüteri meydana gəldi. Bu maşın



► Xircom elektron yazı kitabçası da Z-80 prosessorundan istifadə edir.

ağ-qara monitora, audiosistemə və qrafik interfeysli əməliyyat sisteminə malik idi. 1994-cü ildə Motorola Power PC RISC-prosessoru bazasında ilk Macintosh kompüteri təqdim edildi.

Fərdi kompüretlər cəmisi bir neçə on il ərzində bütün dünyaya yayıldı, indi onlar ofislərdə yazı makinaları kimi adi bir qurğuya çevrilmişlər. Son illərdə fərdi kompüterlərin imkanları mətn emalından başqa multimedia proqramları və dünya kommunikasiya vasitələri hesabına genişlənmişdir. Fərdi kompüterlər praktik olaraq böyük maşınlar kimi güclü maşınlara çevriliblər. Onların həll etdikləri məsələlərin dairəsi də genişlənib.

ÇBİS-texnologiyaları, mərkəzi prosessoru, yaddaşı və digər qurğuları ayrı-ayrı ÇBİS şəklində hazırlamağa imkan



Lisa kompüteri.

verdi. Bu isə praktik olaraq hər kəs üçün qiymət baxımından əlçatan olan kompüterlərin kütləvi istehsalını mümkün etdi. ÇBİS-texnogiyalarının inkişafı nəticəsində EHM-in yeni sinfi kimi ucuz fərdi kompüterlər, eləcə də minlərlə prosessoru olan çox yüksək sürətli paralel hesablama sistemləri yarandı. "Dördüncü nəsil" termini çox vaxt ÇBİS-texnologiyalara əsaslanan maşınların sinonimi hesab olunur.

MİKROPROSESSORLARIN İSTEHSALI

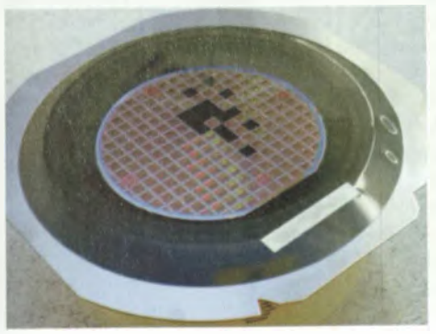
İnteqral sxemlərin hazırlanması çox mürəkkəb və baha başa gələn prosesdir. Mikroprosessorlar mərhələlərlə istehsal olunurlar. İstehsalın hər bir mərhələsində eyni zamanda yüzlərlə inteqral sxem – çoxlu sayda eyni cür detallardan – kristallardan ibarət olan bütov bir silisium təbəqəsi emal olunur. İxtiyari kristal milyonlarla elementdən – tranzistorlardan təşkil olunmuşdur, yəni bir texnoloji əməliyyat zamanı milyardlarla element emal olunur.

İstehsaldan qabaq inteqral sxem istehsalçısı tranzistorlardan, onların birləşmələrindən və digər elementlərdən ibarət yastı çoxlaylı konstruksiya şəklində elektron sxem təqdim edir. Konstruksiyanın hər bir layı – müəyyən bir materialdan (yarımkeçirici, metal, izolyator və s.) hazırlanmış mürəkkəb şəkildir.

İnteqral texnologiyası nəinki yeni kütləvi istehsala, mikroprosessor və standart çiplərdən kompüterlərin yığılmasına şərait yaratdı, eləcə də, kompüterlər üçün standart olmayanlar da (sifarişlə, xüsusi qaydada qurulanlar) daxil olmaqla çiplərin tez və baha olmayan layihələndirilməsi və istehsalına imkan yaratdı.

Hər bir yeni mikroprosessorun işlənməsi çoxlu sayda hesablamaların aparılmasını və ən müasir texnikanın tətbiqini tələb edir. Yeni nəsil mikrosxemlərin istehsalı yalnız əvvəlki nəsil mikrosxemlərin köməyilə mümkündür və öz növbəsində bu nəsil də sonrakı nəsil mikrosxemlərin yaradılması üçün yol açır.

İnteqral sxem xüsusi istehsalda mürəkkəb yolla laybalay işlənmiş sxemi təkrarlamaqla hazırlanır. Başlanğıc mər-



İnteqral mikrosxemlərin plastini.

Hal-hazırda poliqrafiya istehsalı texnologiyaları informasiya texnologiyaları aləminə gətirilir. Poçt markaları tək-tək deyil, bloklarla, hər sırada müəyyən sayda marka olmaqla, bir neçə sıra ilə çap olunur. Eləcə də, mikroprosessorlar (inteqral sxemlər) da adətən tək-tək deyil, silisiumdan hazırlanan yarımkeçirici təbəqə üzərində qruplarla yerləşdirilərək istehsal olunurlar.

Silisium təbəqələri emaldan əvvəl.



Fotoşablonlar mikroprosessorun sxemi çəkilmiş nazik şüşə təbəqələrdən ibarət olur. Onlar yüksək dəqiqlikli optik avadanlıqda hazırlanır, şəkil elektron şüa ilə çəkilir.

İNTEQRAL SXEMLƏRİN LAYİHƏLƏNDİRİLMƏSİ

Aydındır ki, bir neçə milyon elementdən ibarət mürəkkəb kristalı əl ilə hazırlamaq mümkün deyil. Bunun üçün güclü kompüterlər və inteqral sxemlərin avtomatik layihələndirilməsi üçün xüsusi proqram sistemləri – CAD (*ing.* Computer-Aided Design) tələb olunur.

Layihələndirmə prosesi üç mərhələdən ibarətdir.

İnteqral sxemin verilməsi. Sxem CAD-kompüterdə layihələndirilir. İstehsalçı onu ekranda çəkir. Həm sadə sxem, həm də mürəkkəb mikroprosessor elementlərinin hər biri ayrıca verilir. Sxem elementləri ya insan tərəfindən müstəqil layihələndirilir, ya da bunun üçün verilənlər bazasından müəyyən standart elementlər seçilir. Yaxud, məsələn, inteqral sxem sadəcə olaraq fayldan öz proobrazı ilə götürülür və korrektə olunur. Layihələndirmə zamanı mikroprosessor sxeminin təsviri üçün xüsusi dildən istifadə olunur.

İnteqral sxemin verifikasiyası. Bu inteqral sxemin yoxlanma prosesidir. Yaradılmış sxemin layihəyə uyğun olub-olmaması xüsusi proqram təminatının köməyilə testlərlə yoxlanılır. Həm də təkcə layihəyə uyğunluq yoxlanılmır, eləcə də gələcək mikroprosessorun komponentlərinin əməliyyatlarının öz aralarında uyğunlaşdırılması üçün zaman göstəriciləri də yoxlanılır. Bu mərhələdə insan layihəyə düzəlişlər edə bilər, sonra yenə inteqral sxemi yoxlayıb, onu yaxşılaşdıra bilər və ya onun hissələrini tamamilə dəyişdirə bilər. CAD-kompüterlərin tətbiqi olmadan əl ilə bu prosesi yerinə yetirmək praktik olaraq mümkün deyil. İkinci mərhələ inteqral sxemin hər bir elementinin ayrıca təsvir olunduğu faylın yaradılması ilə başa çatır.

İnteqral sxemin istehsalına hazırlıq. Bir qayda olaraq CAD-kompüterin verilənlər bazası istehsal olunmuş inteqral sxemlərin təsvirlərinə malikdir. Layihə olunan sxemin olduğu fayl, inteqral sxemin avtomatlaşdırılmış istehsal prosesinin xüsusi formatına çevrilir. Bu fayldan istifadə edərək, layihələndirilən inteqral sxemin fotoşablonlarının zavodda istehsal olunmasını sifariş etmək kifayətdir.

hələ hazırlıqdan başlanır. Silisium təbəqəyə yüksək tələblər qoyulur. Kristal şəbəkənin təyinatından asılı olaraq yarımkeçiricilərin xassələri güclü şəkildə fərqlənirlər.Diametri 200 və ya 300 mm olan süni surətdə hazırlanmış silisium monokristaldan almaz mişarla kəsməklə alınan nazik kəsik təbəqənin hazırlanması zamanı onu düzgün istiqamətləndirmək vacib şərtdir. Təbəqənin ölçüləri nə qədər böyük olarsa, onun üzərində bir o qədər də çox sayda kristal (mikrosxem) yerləşdirmək olar. Lakin bu yüksək texnoloji avadanlığın tətbiq edilməsini tələb edir, belə ki, təbəqənin emalı mərhələlərində onun bütün səthi boyu bərabər təsir etmək (məsələn, temperaturun sabitliyini təmin etmək) tələb olunur.

Birinci texnoloji əməliyyatdan qabaq hazırlıq işləri görülür: təbəqə yuyulur, qurudulur, cilalanır və s. Təbəqə ideal təmiz olmalıdır – bu bütün istehsal prosesində ən vacib şərtlərdən biridir. Kristala düşən kiçicik toz belə zədələnməyə səbəb ola bilər, çünki elementlər arasındakı məsafə çox kiçikdir.

Birinci mərhələdə təmiz silisiumdan ibarət təbəqəni, silisium dioksidindən (SiO_2) olan nazik dielektrik örtüyü ilə örtürlər. Bunun üçün təbəqəni kamerada 25 atmosfer təzyiqdə və 1000°C yaxın temperaturda oksidləşdirirlər və ya digər bir üsuldan istifadə edirlər – su buxarları ilə oksidləşmə aparırlar. Bu zaman oksigen, təzyiq altında təbəqə səthinə su buxarları ilə üfürülür, alınan nazik izolyator örtüyünün bərabər paylanması üçün (zədəsiz) kameranın bütün həcmi üzrə eyni bir temperatur ciddi gözlənilməlidir.

FOTOLİTOQRAFİYA

Bu planar texnologiyanın əsas əməliyyatlarından biri olub, texnoloji əməliyyatlarda təbəqəyə seçimli təsir göstərilməsinə təminat yaradır.

Təbəqədə müəyyən tip tranzistorların hazırlanması üçün silisiuma müəyyən qarışıqlar əlavə etmək lazımdır ki, *p* və *n*-tipli keçiriciliyi olan yarımkeçirici qurğular almaq mümkün olsun. Bu cür əlavələri təbəqəyə daxil edərkən qarışıqların vurulmayacağı sahələri, məsələn, izolyator ilə örtmək tələb olunur. Onda bütün təbəqənin səthini izolyatorla örtüb, burada kimyəvi üsulla *aşındırma* (təbəqənin səthindən lazım olmayan yarımkeçirici, dielektik, alüminium və ya mis keçirici sahələrinin çıxarılması prosesi) əməliyyatından əvvəl pəncərələr qoymaq olar. Bu *fotolitoqrafiya* – etalon fotoşablon üzərindəki şəklin təbəqə üzərinə köçürülməsi zamanı baş verir.

Emaldan sonra təbəqədə ultrabənövşəyi şüaların təsirinə həssas, xüsusi işığa həssas materialdan olan örtükdən ibarət şəkil qalacaqdır. Deməli, təbəqənin bu hissələrini ultrabənövşəyi lampa ilə şüalandırsaq (ekspozisiya etsək), bu hissələri örtən material aşındırmaya davamlı olacaqdır.

Fotolitoqrafiya bir neçə mərhələdən ibarət olan mürəkkəb prosesdir. Təbəqənin səthi sentrifuqanın köməyilə bütünlüklə xüsusi material ilə örtülür: təbəqə firlanan üfüqi diskə yerləşdirilir, yuxarıdan forsunkalardan üfürülən maye məhlul isə mərkəzdənqaçma qüvvəsinin təsirilə təbəqənin mərkəzindən kənarlarına doğru axıb paylanır.

Bu qatın qurudulması sobada ultraqırmızı şüalanmanın dəqiq hesablanmış dozasının təsirilə yerinə yetirilir. Sonra təbəqəni fotoşablonla müqayisə edirlər. Əvvəllər bu əməliyyatı mikroskopun köməyilə əllə yerinə yetirirdilər və bu zaman təbəqədəki və fotoşablondakı uyğun nişanlar üzrə müqayisə aparılırdı. Lakin indi bu işi artıq insana etibar etmirlər.

Şablon təbəqə ilə ciddi şəkildə uyğunlaşdırıldıqdan sonra, təbəqə səthi ultrabənövşəyi şüa ilə şüalandırılır və gələcək layın şəkli həkk olunmuş formada qalır. Ultrabənövşəyi işıq düşmüş hissələr aşılanmaya qarşı davamlı olur və növbəti mərhələdə silisium dioksid qatını qoruyacaqlar.

Təbəqənin aşılamaya hazırlamaq üçün fotolitoqrafiya üsulundan başqa üsullar da var, məsələn, *elektron-şüa litoqrafiyası* və *lazer litoqrafiyası*. Bu zaman optik litoqrafiyadan fərqli olaraq inteqral sxemin bütün şəklinin ekspozisiyası eyni zamanda aparılmır. Kompüterin idarəsi altında elektron (və ya lazer) şüa bütün şəkli ardıcıl olaraq keçir, yəni heç bir fotoşablondan istifadə edilməsi tələb olunmur. Emal sürəti kiçik olduğundan (bir qurğuda il ərzində 60-a yaxın təbəqə emal etmək olur, fotolitoqrafiyada isə bir qurğuda il ərzində on minlərlə təbəqə emal etmək olur) bu texnika əsasən fotoşablonların istehsalı üçün istifadə olunur.



İstehsal yerlərində mütləq təmizlik olmalıdır.

Silisium – silisium dioksidi (Si – SiO_2) cütünün iki əsas kimyəvi xassəsi, silisium texnologiyasının əsasını təşkil edir. Birincisi: bu materiallardan hər hansı birinə seçkili qaydada təsir göstərmək imkanıdır (belə ki, flüorit turşusu silisium dioksidini həll edir, silisiumun özünə təsir etmir); ikincisi silisium dioksidindən aşqar vurma (aşqarlar daxil edilməsi) zamanı istifadə etmək olar.

Müasir texnologiyalar fotolitoqrafiya zamanı elə bir qoruyucu materialdan istifadə etməyə imkan verir ki, bu material şüalanma nəticəsində aşılanmaya davamlı olur və ya özünün qoruyucu xassələrini itirir.



İnteraktiv sxemlərin istehsal mərhələlərindən biri.

AŞILAMA

Maye aşılamadan, məsələn, azot turşusu ilə aşılamadan istifadə etmək olar. Təbəqəni, radio həvaskarlarının öz çap lövhələrini aşıladıqları qaydada emal edirlər. Cərəyankeçirici layla (folqadan hazırlanmış) örtülmüş izolyatorda (tekstolitin) gələcək lövhənin birləşdirici yollarının şəklini çəkirlər, sonra lövhəni, məsələn, azot turşusuna salırlar. Folqanın lazım olmayan hissələrini aşılayırlar, nəticədə tekstolitin üzəri açılmış olur.

Bu zaman aşılamanın nəticələrinə turşunun qatılığı, temperatur kimi bir çox faktorlar təsir göstərir. Parametrlər düzgün seçilmədikdə turşu tədricən qoruyucu layın kənarlarından sızaraq lazımlı birləşmələri sıradan çıxara bilir.

İonlaşdırılmış qazın tətbiqilə quru (plazmalı) aşılama prosesi bir istiqamətdə qəti olaraq yuxarıdan aşağıya doğru aparmağa imkan verir. Qaz, emal olunan səthlə reaksiyaya girir, nəticədə alınan əlavə uçucu məhsullar aşılamanın aparıldığı kamera tərəfindən udulur.

İstehsal prosesində aşılama qeyd olunmuş ardıcıllıq gözlənilmədən dəfələrlə təkrarlanır. Aşqarların daxil edilməsi əməliyyatlarını kombinə etmə yolu ilə, yarımkeçirici laylarının tozlandırılması, lazım olmayan sahələrin aşılanması, silisiumlu səthdə mürəkkəb, çoxlaylı naxış əmələ gətirir. Bu naxış inteqral sxemin işlənmiş layihəsinin dəqiq təkrarı olub, yarımkeçiricilərin, izolyatorların və keçiricilərin bir-birinə birləşmələrindən ibarətdir.

Silisium dioksidinin aşılanması nəticəsində dielektrikdə silisiumun hissələrini üzə çıxaran deşiklər əmələ gəlir. Sonra fotoqoruyucu layın qalıqlarını çıxarırlar və silisium aşqarların daxil edilməsi üçün hazırdır.

DİFFUZİYA (AŞQARLARIN DAXİL EDİLMƏSİ)

Aşqarların daxil edilməsində məqsəd. silisiuiumun elektrik xassələrinin dəyişdirilməsi və təmiz yarımkeçiricinin *p*- və ya *n*-tipli yarımkeçiriciyə çevrilməsidir. *n*-tipli yarımkeçirici almaq üçün fosfordan, *p*-tipli üçün isə bor atomlarından istifadə etmək olar. Diffuziya iki addımdan ibarətdir: *çökdurmə* və bilavasitə diffuziya.

Birinci mərhələdə silisium səthinə az miqdarda *aşqar maddə* daxil edilir. Təbəqənin digər emal mərhələlərində olduğu kimi bu zaman da onun səthi ciddi şəkildə təmizlənməlidir ki, lazım olmayan əlavələr kristalın üzərinə düşməsin. 1000°C temperaturda aşqar maddə silisiumun kristalik şəbəkəsini dəlib keçir. İkinci mərhələdə diffuziya prosesi baş verir və aşqarlar silisiuma daha dərin daxil olub, bərabər paylanırlar.

BİRLƏŞMƏLƏRİN YARADILMASI

Planar texnologiya ilə alınan ayrı-ayrı qurğuları (tranzistorlar, KMOY-cütlər) öz aralarında və ola bilsin ki, xarici çıxışlarla birləşdirmək tələb olunur.

Bunun üçün silisiumla kontakda olmalı hissələrdən dioksid silinir, sonra isə təbəqənin səthi çökdürülmə yolu ilə metal layla örtülür. Metal onu buxarlandıra biləcək yüksək temperatura qədər qızdırılır. Sonra soyuyaraq qazşəkilli metal silisiumlu təbəqə üzərinə bircins örtük şəklində kondensasiya olunur. Bundan sonra metal (adətən alüminium) fotolitoqrafiyanın və aşılanmanın köməyilə onun olması məqbul olmayan hissələrdən çıxarılır. Bütün bu əməliyyat *metallaşdırma* adını daşıyır.

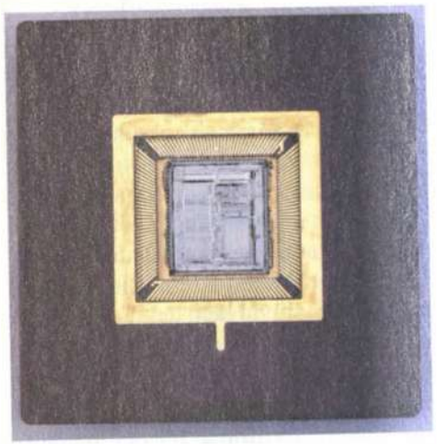
İnteqral sxemlərdə yerlədirilmiş tranzistorların sıxlığı hər dəfə artdığından, elementlər arasındakı birləşmələrin sayı da artır. Bu birləşmələri heç də həmişə bir metal layda yerləşdirmək mümkün olmur. İkinci, üçüncü və daha artıq sayda lay tələb oluna bilər. Göstərilən texnoloji əməliyyatlar çoxlu sayda təkrarlanır və bu zaman metallaşan laylar birbirindən dielektriklə izolə olunur.

Texnoloji əməliyyatlardan sonra yüzlərlə eyni cür kiçik düzbucaqlıdan (hər bir düzbucaqlı başqa təyinatlı gələcək mikroprosessor və ya inteqral sxemdir) ibarət olan təbəqə alınır.

Təbəqə hələ bir neçə əməliyyatdan keçməlidir: ayrı-ayrı çiplərə kəsilmə, inteqral sxemlərin yığılması, testləşdirmə. Axırıncı əməliyyat mikroprosessorlardan neçəsinin yararsız olacağını, neçəsinin isə kompüterlərin, avtomobillərin, mikrodalğalı sobaların və s. istehsalında istifadə ediləcəyini müəyyən edir.

İon aşqarlanması silisiuma aşqarların daxil edilməsinin müasir üsullarından biridir. Bu üsulda böyük temperatura ehtiyac olmur və aşqar maddələr yarımkeçiricinin səthinə çox nazik təbəqə ilə (1 mikron) paylanır.

Aşılama zamanı silisiumun səthi üzərində, təbəqə ilə olan kontakların keyfiyyətinə təsir göstərən pillələr əmələ gəlir. Yeni metod metalların naxışlanmasına bənzədiyinə görə Dəməşq texnologiyası adlanan üsulla metallaşmanı yerinə yetirməyə imkan verir. Bu ad, üsul müəyyən mənada metalda naxış açmaya bənzədiyindən verilmişdir. Dioksidin səthində qabaqcadan aşılanmış cığırlara metal layı hopdurulur. Metal bu cığırları doldurараq kontakt birləşmələri yaradır. Sonra səth cilalanır və artıq metal çıxarılır. Bu üsul yararlı mikrosxemlərin istehsalını artırmağa imkan verir. XX əsrin əvvəllərində bu texnologiyanın daha mütərəqqi formalarından istifadə edilməyə başlandı.

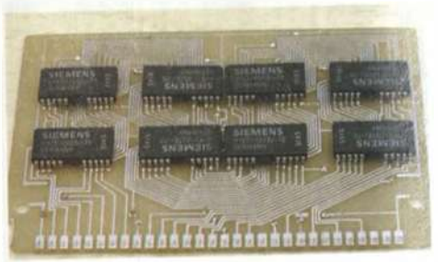


Mikroprosessor.

EHM YADDAŞININ TƏKAMÜLÜ

Məlum ifadəni çevirməklə demək olar ki, "kiçik və sürətli yaddaşı olan bahalı kompüter almaqdansa, böyük və az sürətli yaddaşı olan ucuz kompüter almaq daha yaxşıdır". Lakin böyük yaddaş kiçiyə nisbətən, sürətli isə, az sürətliyə nisbətən daha bahadır. Buna görə də istehsalçı və istehlakçı qiymət, yaddaş həcmi və sürət arasında kompromis axtarmağa məcbur olur və kompüterin optimal işləməsi üçün, müxtəlif tip yaddaşların kombinasiyasını qurmağa çalışırlar.

Kompüterin naqillərlə əlaqələndirilən komponentləri öz aralarında informasiya mübadiləsi edərkən elektrik siqnallarından istifadə edirlər. Əgər aksiom kimi qəbul etsək ki, prosessor və yaddaş arasında informasiyanın naqillər (*şin*) üzrə ötürülməsi prosesi şinin



Yaddaş.

Şin yalnız prosessor və yaddaşa deyil, həmçinin EHM-in digər komponentlərinə də xidmət edir. Şinlər verilənlərin ötürülməsilə yanaşı, ona qoşulan qurğuların iş sürətlərinin uyğunlaşdırılması üçün də istifadə olunur. Bunun üçün o, şinin informasiya qəbul edib və ya göndərməyə hazır olub-olmadığını təyin etməyə imkan verən məxsusi siqnallar hasil edir. Burada vacibdir ki, şinin və ona qoşulan bütün qurğuların məhsuldarlıqları öz aralarında uyğunlaşdırılmış olsun. Sürətli prosessor və zəif sürətli yaddaşı və ya sürətli prosessor və yaddaşı zəif sürətli şinlə bir araya gətirmək ağılsızlıq olardı.

Ferromaqnitiklər – güclü maqnit nüfüz etmə qabiliyyətli (məsələn, dəmir, nikel, kobalt) maddələrdir. Onlar xarici sahənin təsirilə maqnitləşmə qabiliyyətinə malikdirlər. Fransız alimi Andre Mari Amper belə bir fərziyyə irəli sürmüşdü ki, molekul və atomların daxilində elementar cərəyanlar dövr edir. Maddənin maqnitləşdirilmiş vəziyyətində elementar cərəyanlar elə istiqamətlənir ki, onların təsirləri toplanır.

məxsusi xarakteristikalarından (naqillərin sayı, siqnalın keçmə sürəti və s.) asılı deyil, əslində belə deyil, onda yaddaşı aşağıdakı üç əsas meyar üzrə qiymətləndirmək olar: qiymət, həcm və informasiyaya müraciət zamanı.

Yaddaşın prinsipial fərqli formalarının bütün çoxluğu içərisindən əsas iki qrupu: *operativ yaddaşı* və *xarici yaddaşı* ayırmaq olar. Kompüterin prosessoru operativ yaddaşa tez-tez müraciət edir, burada informasiyanın oxunması və yazılması kifayət qədər yüksək sürətlə baş verir. Xarici, uzunmüddətli yaddaşda



Qabarcıqlı yaddaşın domenləri belə görsənirlər.

isə kompüter böyük həcmli informasiyalar saxlayır və ona demək olar ki, az-az hallarda müraciət edir. Qənaətcillik baxımından operativ yaddaş xarici yaddaşa nəzərən həcmcə həmişə kiçik olur.

Hesablama texnikasının bütün tarixi boyunca həm operativ, həm də xarici yaddaş durmadan təkamül yolu ilə inkişaf edir: məhsuldarlıq və həcm artır, qiymət azalır. Tətbiq olunan texnologiyalar filmdəki kadrlar kimi bir-birini əvəz edir. Elektron-şüa borularındakı və ultrasəs xətlərindəki yaddaş ferrit nüvəciklərdəki yaddaşla əvəz olunur. Yarımkeçiricilər və KMOY-sxemlərdəki yaddaş ferrit yaddaşı əvəz edir. Bu proses uzun müddətli yaddaşla da baş verir.

60-70-ci illərdə yarımkeçiricilər əsasında olan yaddaş baha başa gəlirdi və cərəyan kəsilən kimi onun tərkibi də silinirdi (nə verilənlər, nə də proqramlar saxlanılmırdı). Buna görə də problemin yeni həllərini axtarırdılar. Yeni çox saylı araşdırmalar sürətli və etibarlı sayılan yaddaşın yaradılmasına səbəb olurdu, lakin tezliklə o da yeni texnologiyaların köməyilə sıxışdırılıb aradan çıxarılırdı, bəzən isə ümidləri doğrultmayan səhvlər də baş verirdi.

QABARCIQLI YADDAŞ

Bu cür uğursuzluqlardan biri 1966-cı ildə Bell Labsda (ABŞ) işlənib hazırlanmış qabarcıqlı yaddaş (*ing.* bubble memory) adlanan qurğudur. Bu qurğu verilənləri nazik maqnit material üzərində maqnit domenlər şəklində yadda saxlayırdı. Maqnit material kimi, məsələn, ferroqranatdan istifadə edilirdi.

Əgər təbəqəni əvvəlcədən maqnitləşdirib, sonra isə onun üzərinə maqnitləşdirmə istiqamətində müəyyən səviyyəli maqnit sahəsi qoysaq, onda domenlər əyri xətt formasından qabarcıq formasına çevrilir. Buna görə də yaddaş

qabarcıqlı yaddaş adını almışdır. Əslində domen silindr formasına malikdir, onun oxu isə təbəqəyə perpendikulyardır. Materialın müəyyən nöqtəsində domenin olması 1 kimi, olmaması isə 0 kimi qəbul olunurdu. 1 sm^2 layda 16 milyona qədər domen yerləşdirilə bilirdi, başqa sözlə desək qabarcıqlı yaddaşın həcmi 16 Mbayta qədər idi.

Bu cür yaddaş daha bir maraqlı xassəyə malik idi. Zəif xarici maqnit sahəsinin təsiri nəticəsində ferrormaqnit materiallı təbəqədə domenlər əvvəlcədən verilmiş istiqamətlərdə yüksək sürətlə hərəkət edirdilər. Bu isə mexaniki hərəkət edən hissələri olmayan yaddaş qurğusunu yaratmağa imkan verdi. Belə ki, yaddaşdan informasiya oxunması üçün domenləri oxuyan qurğuya doğru ardıcıl olaraq yönəltmək tələb olunurdu. Qabarcıqlı yaddaşın daha bir vacib xüsusiyəti, elektrik cərəyanı olmadığı halda özünün maqnitliyini saxlamaq qabiliyyəti idi. Yəni, bu yaddaşın əsasında asanlıqla, enerjidən asılı olmayan, proqramları dəyişdirilə bilən daimi yaddaş qurğusu DYQ və ya ROM (*ing.* read only memory) alınırdı.

Bu orijinal elmi ideya artıq 90-cı illərin əvvəlləri üçün daha ucuz və sürətli, proqramları elektron dəyişdirilə bilən daimi yaddaş – EEPROM (*ing.* electrically erasable program-mable read-only memory) tərəfindən tamamilə sıxışdırılıb çıxarılmışdı.

YÜK RABİTƏLİ QURĞU

Digər dahiyanə ixtira CCD (*ing.* charge coupled device) və ya YRQ-yük rabitəli qurğudur. İşığa həssas bu tip yarımkeçirici 1969-cu ilin sonunda Bell Labsda amerikalılar Uilyam Boyl və Corc Smit tərəfindən yaradılmışdı. CCD informasiyanın kompüterdə saxlanması üçün də nəzərdə tutulmuşdu, lakin qurğu yaddaş kimi istifadədə çox long idi, buna

Domenlər (*frans.* "domaine" – "sahə") – daxilindəki bütün metalları yalnız bir istiqamətdə maqnitləşdirilən kifayət qədər böyük atom qruplarıdır. Maqnitləşdirmə adətən əsas kristallik oxlardan birinin boyunca aparılır. Metalların maqnitləşdirilməsi zamanı, tətbiq olunmuş sahə istiqamətində maqnitləşdirilmiş domenlər qonşu atomların hesabına, "böyüyürlər". Əgər sahə zəifdirsə, onda bu dəyişiklikləri geri qaytarmaq olur və onlar çox da böyük olmur. Əksinə çox güclü maqnit sahəsində bəzi domenlər sahənin istiqamətini alırlar, baxmayaraq ki, bu istiqamət kristalın heç bir oxu ilə üst-üstə düşmür. Domenlər fiziki ölçülərinə görə çox kiçikdirlər, onları yalnız çox güclü mikroskopun köməyilə görmək olar. Lakin atomlara nəzərən bu elementar maqnitiklər çox böyük yığına bənzəyirlər.

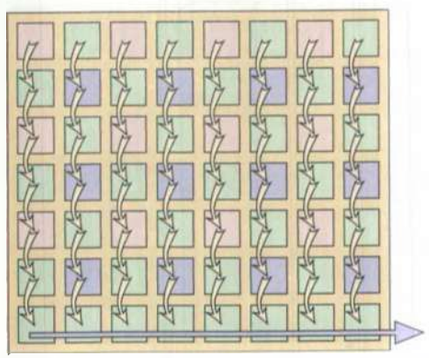
baxmayaraq sensor kimi bu qurğu ideal idi.

CCD ekspozisiya zamanı işıq axınını yüklər şəklində "yadda saxlayan" işığa həssas elementlər matrisindən ibarət idi. Sonra yük rabitəli qurğu şəkil haqqında informasiyanın ötürülməsini təmin edir. Bir addıma yüklər matrisin sıraları üzrə bir sətir aşağı hərəkət edirdilər. Axırıncı sətir irəliləmə registri adlanan hissəyə düşürdü. Buranı "tərk edən" signal gücləlnib rəqəm formasına çevrilirdi. Proses o vaxt sona çatdırılırdı ki, CCD matrisi bu qayda ilə tam şəkildə oxunmuş olsun.

CCD 1975-ci ildə videokamera və skanerlərdə, 1980-ci ildə isə ilk rəqəmli fotoaparatlarda öz tətbiqini tapmışdı.

Mikrosxemlərin əsas komponentləri kameralarda işığa həssas material kimi istifadə olunan KMOY-yarımkeçiriciləridir. 90-cı illərin əvvəllərində KMOY-sensorların həssaslığı xeyli

Yük rabitəli qurğu matrisin elementləri arasında yükün verilməsi qaydasına görə belə adlandırılıb.



YRQ (CCD)-nin iş sxemi.

▸ Lent çevirən mexanizm.

artmışdı və onların işlətdiyi gərginlik azalmışdı.

KMOY-sensorun aktiv elementlərindən yüklərin oxunması paralel sxem üzrə yerinə yetirilir. Bu isə ixtiyarı müraciəti təmin edir, matrisi bütövlükdə deyil, yalnız onun müəyyən sahələrini oxumaq olur. CCD-yə nəzərən matrisin oxunması sürəti daha yüksəkdir. Belə ki, CCD-ə bütün yüklər yeganə irəliləmə registrlərindən çıxır. KMOY-sensor həmçinin təsvirin ölçülərini kiçiltməyə də qadirdir. İqtisadi baxımdan CCD-sxemlərinin ayrıca istehsalını saxlamaq qeyri-rentabelli olur. Belə ki, analoji KMOY-sensorların istehsalını KMOY-mikroprosessorların istehsal olunduğu adi zavodda təşkil etmək olar.

Ehtimal etmək olar ki, XXI əsrin əvvəllərində CCD-matrisini KMOY-matrislər əvəz edəcək və CCD texnologiyası yaddan çıxacaqdır. Təsadüfi deyildir ki, CCD və qabarcıqlı yaddaş tranzistordan fərqli olaraq Bell Labsın "dünyanı dəyişmiş" on ən böyük ixtirası siyahısına düşmüşdür.

Çıxarıla bilməyən maqnit daşıyıcısının birinci qurğusu, hər biri 30 Mbayt tutumlu iki diskdən ibarət idi. Yaddaşın ümumi tutumu 30/30 kimi işarə edilirdi. Bu isə "Vinçester" firmasının məşhur tüfənginin adı ilə üst-üstə düşürdü. Hesab olunur ki, kompüterin bu vacib detalı öz adını buradan götürüb.

XARİCİ YADDAŞ QURĞULARI

Kompüterin operativ yaddaşı sürətlə işləməlidir, lakin hətta XXI əsrdə də bu əvvəlki kimi kifayət qədər baha başa gəlir. Nəzəri olaraq fərz etmək olar ki, operativ yaddaşın qiyməti o qədər ucuzlaşacaq ki, o, EHM-də yeganə yaddaş olacaqdır. Lakin hələ ki, bu yaddaşa çox böyük həcmli informasiya sığmır. Bundan əlavə hələ də operativ yaddaşın tərkibi EHM cərəyandan ayrılan kimi silinir. Bu isə sübut edir ki, kompüterin xarici yaddaşı ucuz və böyük həcmli olmalıdır.



Xarici yaddaşın bir qrupu *ixtiyarı müraciət* qurğuları, digəri isə *ardıcıl müraciət* qurğularıdır. Hal-hazırda çox nadir hallarda istifadə olunan maqnit lenti ardıcıl müraciət qurğularına tipik misal ola bilər. Burada verilənləri yalnız ardıcıl olaraq oxumaq və ya yazmaq olur. Ardıcıllığı pozduqda, lentin lazımi yerədək fırlanmasını uzun müddət gözləmək lazım gəlir.

Maqnit lentlər, böyük tutuma malik olsalar da, kifayət qədər ləng sürətli ardıcıl müraciət qurğularıdır. Maqnit lentlərlə iş üçün müasir qurğular – *strimerlər* artırılmış yazılış sürətinə (dəqidədə bir neçə meqabayt) malikdirlər. Maqnit lentlərin ucuzluğu, strimerlərdən informasiyanın uzunmüddətli dövr üçün saxlanılmasında istifadə edilməsinə şərait yaradır. Strimerin bir kassetinin həcmi yüzlərlə və minlərlə meqabaytla ölçülür.

İxtiyarı müraciət qurğuları, verilənlərin ixtiyarı hissəsini təxminən eyni bir *müraciət zamanında* almağa imkan verirlər. İxtiyarı müraciət qurğularından ən geniş yayılmışı *disketlərdir.* Disketlər



3,5 düym (1 düym = 2,54 sm) ölçülü plastik "zərfə" qablaşdırılmış kiçik maqnit diskləridir. Disketlər mexaniki zədələrdən qorunurlar və onları asanlıqla pencəyin cibinə yerləşdirib gəzdirmək olar. Verilənlər iki tərəfdən diskin səthində yerləşdirilmiş konsentrik maqnit cığırlarda saxlanılır. Disketlərdəki informasiyanı oxuyan və ona informasiya yazan qurğu *yumşaq maqnit disklər üçün disksürən* adlanır. Onun daxilində diskin sərhəddindən mərkəzə doğru addımlı mühərriklə radial hərəkət edən iki maqnit başlıq var. Disketin özü kiçik sürətlə fırlanır. Nəticədə, bir dövrün davam etməsindən çox olmayan zaman müddətində, disketin ixtiyari yerində yazılış və ya oxunma aparmaq olar.

Disketin maksimal tutumu 2,88 Mbaytdır. Disketlə iş zamanı verilənlərin ötürülmə sürəti çox da böyük deyildir: təqribən saniyədə bir neçə kilobayt. Orta müraciət zamanı isə 256 mikrosaniyədir.

3,5-düymlü disketlərə qədər 5,25- və hətta 8-düymlü disketlər mövcüd idi. Kiçik disketdən fərqli olaraq onlar yumşaq karton zərflərə yerləşdirilirdilər. 8-düymlü disket cəmi 0,3 Mbayt, 5-düymlü isə 1,2 Mbayta qədər informasiya tutumuna malik idi. 80-ci illərin sonunda 3,5-düymlü disketlərə çox bənzəyən tam 3-düym (plastik qablaşmada) ölçülu disketlər buraxılırdı. Lakin onlarda yerləşdirilə bilən informasiyanın həcmi az olduğundan, bu disketlər tezliklə dəbdən düşdülər.

Çıxarıla bilən maqnit disklər böyük EHM-də çoxdan istifadə edilirdilər. Çıxarıla bilməyən maqnit daşıyıcısına – vinçesterə (disk konstruktiv olaraq ötürücü ilə bir vahid blokda birləşdirilib) keçid diskin fırlanma sürətini (dəqiqədə bir neçə min dövrə qədər) və yazılışın sıxlığını kifayət qədər artırmağa imkan verdi. 2000-ci il kompüterlərində vinçesterlərin tutumu onlarla giqabaytlarla, verilənlərin ötürülməsi sürəti isə saniyədə yüzlərlə meqabaytla, müraciətin orta zamanı millisaniyələrlə ölçülür. XXI əsrin əvvəllərinin kompüterlərində bu xarici yaddaşın əsas formasıdır.

MAQNİT YAZILIŞ



Voldemar Poulsen.

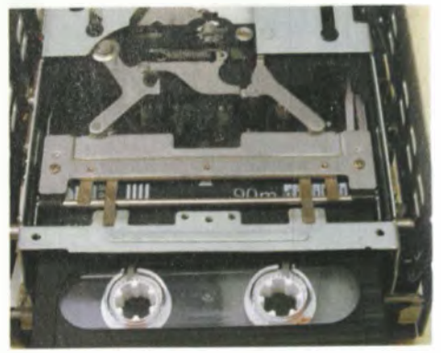
Maqnit yazılış artıq 100 ildən çoxdur ki, mövcuddur, bununla belə o, müasir cəmiyyətin ən qabaqcıl texnologiyalarından biri olaraq qalır. Yazılış üçün ilk qurğu 1898-ci ildə, o zamanlar Danimarka telefon kompaniyasında işləyən danimarkalı mühəndis Voldemar Poulsen (1869-1942) tərəfindən yaradılmışdı.

Poulsen laboratoriya boyunca çəkilmiş polad royal simindən istifadə etmişdi. Yazılış və oxunma sim boyunca mikrofon və xüsusi maqnitli yumşaq materialdan (praktik olaraq qalıq maqnitləşməsi olmayan) hazırlanmış nazik mil olan valı hərəkət etdirən vintli mexanizmin köməyilə aparılırdı. Sarğılar mili solenoidə (*yun.* "sollen" – "borucuq" və "eydos" – "forma") – maqnit sahəsinin həyəcanlandırıcısına çevirirdi. Solenoidə cərəyan verdikdə, milin sonunda maqnit sahəsi formalaşırdı və bu sahə sarğıdakı cərəyanın rəqslərinə tam uyğun olaraq dəyişdirilirdi. Bu maqnit sahəsi solenoidin maqnit sahəsinin gərginliyinə mütənasib qalıq maqnetikləri – polad sərt-maqnit naqildə (qalıq maqnitləşməni saxlayan) yazılış elementləri yaradır.

Yazılışın təsviri zamanı maqnit keçiricisi kimi işləyən həmin mil tətbiq olunurdu. O, sim boyunca hərəkət edərkən yazılmış maqnit sahəsinin dəyişmələrini qəbul edirdi. Sarğıda elektrik cərəyanı yaranırdı və sonra bu cərəyan səs rəqslərinə çevrilirdi.

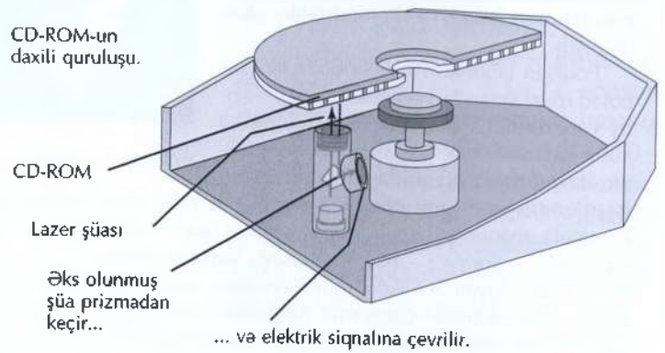
Poulsen birinci olaraq mili olan universal maqnit başlığı yaratdı. İndi o, kompüterlərin maqnit yazılışı qurğularında istifadə olunur.

Maqnit lent – asetat əsaslı toz halında olan ferromaqnit materiallı (dəmirin qamma-oksidi) lent XX əsrin 30-cu illərində BASF firması tərəfindən yaradılmışdı.



Strimer.

CD diski Philips firması tərəfindən musiqi yazılışı üçün işlənib hazırlanmışdı. Diskin tam tutumu təqribən 650 Mbayt, səsləndirmə müddəti isə 74 dəqiqədir. 44100 Hers tezlikli 16-bitlik kodlaşdırma üçün təqribən 170 kbayt/saniyə sürəti tələb olunurdu.



Məişət optik audikompakt-disklərinin (lazer disklər və ya CD) ardınca optik disklər geniş yayıldılar. CD-ni kompüterdə oxuya bilən qurğu CD ROM adlanır. O, kompakt disklərin səsləndiricilərilə çox oxşardır və mexaniki olaraq yumşaq disklər üçün nəzərdə tutulmuş disksürənlərin quruluşuna analojidir. Lakin onlarda informasiya olan yeganə spiral cığır (adi vinil plastinlərdə olduğu kimi) olur, oxunma zamanı dalğa uzunluğu 780 nm olan infraqırmızı lazerdən istifadə olunur. CD-ROM həmçinin kompüterdə audiokompakt disklərin səsləndirilməsinə imkan verir. Kompakt-disk diametri 120 mm, qalınlığı isə 1,2 mm olan şəffaf polimerdən (məsələn, polikarbonatdan) hazırlanır.

CD-ROM verilənləri əks etdirən səthdə 1,6 mkm addımlı spiral cığırlarda yerləşdirilən ikilik sıfır və birləri işarə edən mikroskopik oyuqlar formasında saxlayır. Bu oyuqların dərinliyi 0,13 mkm, eni 0,5-0,8 mkm, cığır boyunca minimal uzunluğu isə 0,834 mm-dir. Verilənlərin ötürülmə sürəti saniyədə bir neçə meqabaytdır, müraciətin orta zamanı isə onlarla millisaniyədir.

CD-ROM-un üstünlükləri onun münasib qiyməti (1 ABŞ dollarından da ucuz) və böyük informasiya tutumudur (600-700 Mbayt). Praktik olaraq bütün müasir proqram təminatı CD vasitəsilə yayılır. Onlara multimediali böyük tədris ensiklopediyaları, oyunlar, videofilmlər və s. yazılır.

CD-RW (*ing.* ReWritable) disklərində silinmə və yazılış proseslərini dəfələrlə təkrarlamaq mümkündür. Onları 700 Mbayt tutumu və informasiyanın qorunub saxlanmasının daha yüksək etibarlığı fərqləndirir.

XXI əsrdə buraxılan fərdi kompüterlərin çoxunda artıq disketlər üçün disksürənlər yoxdur. XXI əsrin əvvəlində kompüterdə istifadə olunan qurğuların tam olmayan siyahısına maqnit-optik diskləri, çıxarıla bilən maqnit kartları, DVD-ROM-u, kartlardakı flash-yaddaşı (yüzlərlə meqabayt tutumu olan) daxildir.





Yaddaş flash-kartları (solda); 8, 5 və 3-düymlü disketlər (sağda).

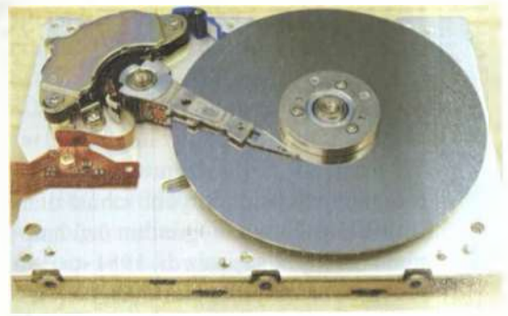
STANDARTLARIN MÜHARİBƏSİ

80-ci illərin ortalarında JVC firmasının mütəxəssislərindən biri qeyd etdi ki, əgər VHS-kassetlərinin yeni standartlarının dəyişmə tezliyi indiki sürətlə artsa, onda 2005-ci ildə standartlar hər 5 dəqiqədən bir dəyişəcək. Lakin VHS standartlı ən müasir videomaqnitafonda yazılmış kassetlərə yeni funksiyaların heç yarısını da təmin etməyən köhnə videomaqnitofonlarda baxmaq olar. İnsan yalnız VHS-kassetlərin hər hansı bir yeni avadanlıqla, məsələn, DVD-çoxfunksiyalı rəqəmli disk ilə əvəz olunmasından qorxmalıdır.

Buna bənzər bir şey XX əsrə qədər kompüterlərlə baş verirdi. İstifadəçi informasiyanı vaxtlı-vaxtında köhnə daşıyıcılardan yenilərinə, məs., 5-düymlü disketlərdən 3-düymlülərə, 3-düymlülərdən CD-ROM-a və s. keçirməli idi.

Lakin DVD informasiya daşıyıcılarının yaranması ilə hər şey dəyişdi: DVD CD-ROM-la eyni ölçülərə malik olsa da, formatdan asılı olaraq həqiqətən də nəhəng tutuma (4,7-dən 17 Gbayt qədər) malikdir. DVD-də yazılışın yüksək sıxlığına nail olmaq üçün dalğa uzunluğu 635-650 nm olan qırmızı lazer tətbiq olunur. Oyuğun minimal uzunluğu cəmi 0,4 mkm təşkil edir, cığır isə 0,74 mkm addımlı olur. DVD-lər ikiqat və ikitərəfli ola bilir.

DVD texnologiyasında bir zəif nöqtə var ki, bu da DVD diskləri və səsləndiricilərinin bir-birilə uyuşmayan müxtəlif standartlarda olmasıdır. 1999-cu ilin əvvəllərindən Hitachi, Toshiba və Panasonic firmaları adi kompüter DVD ROM-səsləndiricilərilə və DVD-disklərinin məişət səsləndiricilərilə pis uyuşan yazan DVD-ROM qurğuları buraxırdılar. Bir dəfə və bir neçə dəfə yazıla bilən DVD+RW disklərin formatını Hewlett-Packard, Ricoh, Philips və Yamaha firmaları dəstəkləyirdilər. Yazılmış disklər



Qapağı çıxarılmış vinçester.

o saat DVD-səsləndiricilərində oxudulurdu. Üçüncü, lakin axırıncı olmayan variant Pioneer firması tərəfindən hazırlanmış DVD-RW texnologiyası idi. Yazan qurğularda bir dəfə yazıla bilən disklərdən istifadə etmək olar. Onları Pioneer firmasının praktik olaraq ixtiyarı DVD-pleyerində və çox vaxt digər firmaların DVD-səsləndiricilərində oxutmaq olur.

İstifadəçi bu qədər çox saylı standartlardan necə baş çıxartsın, nə seçsin? Bəlkə bu firmalar arasında hansınınsa lider olmasını gözləmək lazımdır? Hələlik isə evə gəlib diski DVD-pleyerə və ya kompüterə daxil edərək, diskin oxunmamasına və ya kinonun göstərilməməsinə şahid olmaq olar.

KEŞ BİZƏ KÖMƏK EDƏCƏK

Prosessor hər şeydən tez verilənləri özünün çox sürətli yaddaşından (registrlərdən) oxuyur və ya yaza bilir. Daha sonra müraciət sürətinin azalma ardıcıllığı ilə əsas yaddaş və xarici yaddaş gəlir. Əgər bütün yaddaş çox sürətli olsa idi, kompüterin məhsuldarlığı kifayət qədər yüksələrdi. Lakin bunu yerinə yetirmək çox baha başa gəlir. Bu işdə, operativ yaddaşa nisbətən sürətli, böyük olmayan həcmli yaddaş – keş-yaddaş köməyə gəlir.

Prosessor əsas yaddaşa bayt yazarkən, bayt və onun əsas yaddaşdakı ünvanı eyni zamanda keş-yaddaşa da daxil edilir. Hər dəfə prosessor hər hansı bir baytı oxumaq istədikdə, əvvəlcə analiz aparılır: bu ünvanlı bayt sürətli keş-yaddaşda varmı? Əgər varsa, bayt keş-yadaşdan oxunur. Əgər yoxdursa, bayt əsas yaddaşdan keş-yaddaşa köçürülür və eyni zamanda prosessora ötürülür.

Nəticədə zamanın hər bir anında əsas yaddaşın ən çox istifadə olunan hissəsinin təkrarı keş-yaddaşa köçürülmüş olur və ona tez zamanda müraciət etmək mümkün olur. Keş-yaddaş prinsipindən vinçesterdə, CD-ROM-da və digər qurğularda istifadə olunur.

1976-cı ildə JVC (Victor Company of Japan) kompaniyası VHS (Video Home System) videoyazılış formatını və VHS kassetlərini işləyib hazırlamışdı. 1978-ci ildə birinci portativ VHS-videomaqnitofonu buraxılmışdı.

IBM FİRMASININ "ŞAN-ŞÖHRƏTİ VƏ ACİZLİYİ"

MAVİ NƏHƏNG BAZARA GƏLİR

XX əsrin 80-ci illərinin əvvələrində kompüter istehsalının yeni sahəsi olan fərdi EHM-lər günü-gündən özü haqqında daha çox səs salırdı. 1981-ci ildə 700 min fərdi EHM satılmışdı, növbəti il üçün satışın minimum iki dəfə artacağı proqnozlaşdırılırdı. Biznesin bu hissəsini böyük EHM-lərin istehsalında lider olan ABŞ-ın ən böyük elektron korporasiyalarından biri olan IBM firması da gözdən qaçırmamışdı.

1981-ci il avqustun 12-də IBM özünün IBM PC (Personal Computer) adlandırılmış ilk fərdi EHM-ni yaratdı. O, Intel 8088 prosessorundan (Intel 8086 prosessorunun 8-bitlik versiyasından), 64 Kbayt həcmli yaddaşdan, 160 Kbayt tutumlu yumşaq disklər üçün disksürəndən (və ya 2 diskovoddan) istifadə edirdi. Kompüterin DYQ-da (daimi yaddaş qurğusunda) interpretator – BASIC dili yerləşdirilmişdi. Eyni zamanda proqram və texniki spesifikasiyalar buraxılmışdı ki, onlar da digər firmaların mütəxəssislərinə kompüterin funksiyalarını genişləndirmək üçün lövhələr hazırlamağa imkan vermişdi.



IBM firmasının Mainframe kompüteri. XX əsrin 80-ci illəri.

Həmin il satılmış IBM PC maşınlarının sayı ümumi satış həcminin 4%-ini təşkil etmişdi. Fərdi EHM-lər bazarında liderlər aşağıdakı firmalar idi:

- Radio SHACK – 26%;
- Apple – 25%;
- Commodore – 10%;
- Nippon Electric – 5%.

1983-cü ildə IBM yeni – PC XT (eXtended Technology) modelini istehsal etdi. Maşın artıq 10 Mbayt tutumlu vinçestrerlə, əlavə lövhələr üçün üç qoşma konstruksiyalar ilə təhciz olunmuşdu. Operativ yaddaş 128 Kbayt idi, lakin 640 Kbayta qədər genişləndirilə bilərdi. Disklərə 360 Kbayt informasiya yazılırdı. Əməliyyat sistemi kimi Microsoft firmasının MS DOS dialoq sistemindən istifadə edilirdi.

IBM kompüterləri Apple firmasının maşınları ilə müqayisədə bütün göstəricilər üzrə uduzurdular. Lisa kompüteri IBM PC-nin yanında gələcəkdən gəlmiş qonağa bənzəyirdi. Takt tezliyi 8 Mhers olan 32-bitlik Motorola 68000 prosessorundan istifadə edən və pəncərə əməliyyat sistemilə təhciz olunmuş Macintosh firmasının maşını isə IBM-i kölgədə qoydu.

IBM firmasının fərdi kompüterlər bazarındakı mübarizəsinə kömək olan əsas səbəblərdən biri, korporasiyanın nəhəng maliyyə ehtiyatlarını göstərmək olar. Bu ehtiyatlar hesabına firma uzun müddət ərzində keyfiyyətdə uduzan kompüter buraxa bilmişdir. Digər əsas səbəb isə IBM PC maşınlarının arxitekturasının açıq tipli olmasıdır. Bu 1982-ci ildə IBM PC-nin birinci surətinin – IBM PC ilə uyğunlaşdırılmış Compaq I



Vinçester.

maşınının yaranmasına gətirib çıxardı. Sonralar bu cür maşınlar IBM PC compatible (uyğunlaşdırılmış) adlandırıldı. IBM isə heç də yaxşı olmayan IBM-compatible kompüterlərin istehsalını davam etdirirdi.

Çox vaxt IBM PC-nin həyata davamlılığının səbəbini Intel prosessorunda görürlər.

INTEL. "ÇATDIRARIQ – DÜZƏLDƏRİK"

Adətən yeni arxitekturanı mütəxəssislərin kiçik qrupu yaradarkən, ayrı-ayrı hissələr bir-birinə çox yaxşı uyğunlaşdırılmış olur. Lakin bunu 80×86 prosessorları haqqında demək olmaz. Bu, verilən arxitekturanı on illərlə inkişaf etdirmiş müstəqil mütəxəssislərin bir neçə qrupunun məhsuludur. Bu qruplar başlanğıc komandalar yığınına hər dəfə yeni-yeni imkanlar əlavə etmişdilər.

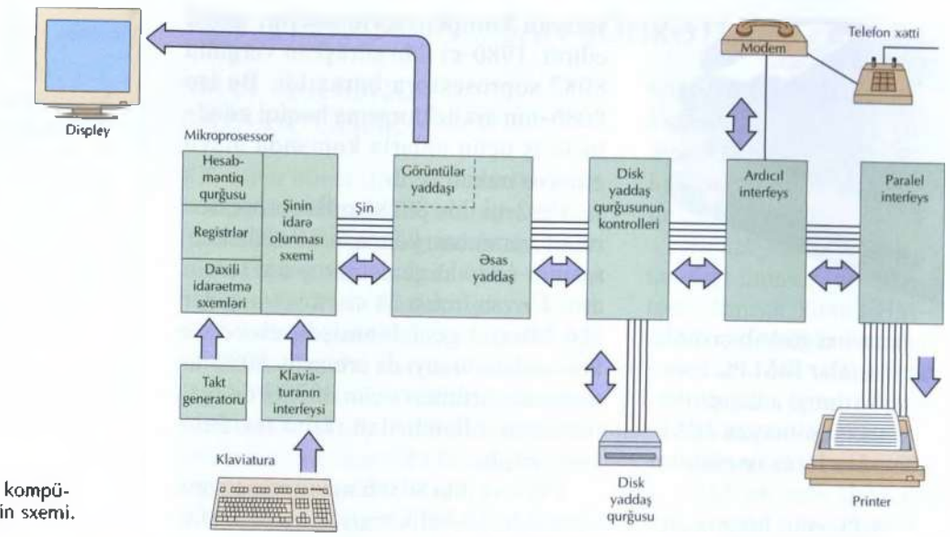
1978-ci ildə 8-bitlik 8080 mikroprosessorunun genişlənmiş variantı olan Intel 8086 prosessoru ortaya çıxdı. Prosessor 16-bitlik idi, bir cəmləyicinin bazasında qurulmuşdu, lakin yeni registrlərlə zənginləşdirilmişdi. Bu arxitekturanın praktik olaraq hər bir registri dəqiq müəyyən olunmuş təyinata malik idi. Proqramlaşdırmanın əlverişliliyi nöqteyi-nəzərindən Intel 8086 ümumi məqsədli registrləri olan daha mütərəqqi prosessorlar sinfinə uduzurdu. 8088 versiyası (verilənlərin 8-bitlik xarici şinilə) sonralar bütün dünyanı fəth etmiş IBM PC seriyalı kompüterlərin əsasını təşkil edirdi. 1980-ci ildə sürüşkən vergüllü 8087 soprosessoru buraxıldı. Bu isə 8086-nın arxitekturasına həqiqi ədədlərlə iş üçün onlarla komanda əlavə etməyə imkan verdi.

1982-ci ildə 80286 mikroprosessorunun yaranması köhnə 8086 arxitekturasının sonrakı genişlənməsini təmin etdi. Ünvan fəzası 24 mərtəbəyə qədər (16 Mbayt) genişlənmişdi, eləcə də komandaların sayı da artmışdı. 8086 ilə uyğunlaşdırılması üçün 80286-da real ünvanlar adlandırılan rejim nəzərdə tutulmuşdu.

1987-ci ildə 80386 mikroprosessoru yaradıldı. 32-bitlik registrlı və 32-bitlik ünvan fəzalı 32-bitlik arxitekturaya əlavə olaraq, 80386 mikroprosessorunda yaddaşın səhifəli təşkili imkanları və bəzi əlavə komandalar meydana gəldi. Nəhayət ki, sonsuz sayda əlavə işlər praktik olaraq 80386 prosessorunu ümumi təyinatlı registrləri olan prosessora çevirdi. Əvvəlki modellər kimi o, 8086 üçün yazılmış proqramların yerinə yetirilmə rejiminə malik idi. Yaddaşın səhifəli təşkili və 32-mərtəbəli arxitekturası bu platformada UNIX əməliyyat sisteminin quraşdırılmasına şərait yaratdı. Sonrakı 80486 (1989-cu il) və Pentium (1993-cü il) prosessorları məhsuldarlığın artırılmasına yönəldilmişdilər və çoxprosessorlu işin təminatı üçün yalnız bir neçə yeni komandanı əlavə etdilər.

80×86 prosessorlarının əvvəlki versiyaları ilə uyğunlaşdırılması kimi ağır bir miras, təəccüblü olsa da, Intel prosessorlarının geniş yayılmasına təsir etməmişdi. Fərdi kompüterlərin beş prosessorundan dördü məhz bu arxitektura əsasında qurulmuşdur.

i486-dan başlayaraq prosessorlarda RISC-prosessorlarının bəzi xüsusiyyətləri vardı. Məsələn, ən çox istifadə olunan komandalar bir takt ərzində yerinə yetirilirdi. Pentium prosessoru, RISC-



Fərdi kompüterlərin sxemi.

i486SX, i486DX, i486DX2 və i486DX4 prosessorları 8 Kbayt tutumlu daxili keş-yaddaşı və 32-bitlik verilənlər şini olan 32-bitlik prosessorlardır. i486SX prosessorunda sürüşkən nöqtəli soprosessor yoxdur. OverDrive və i486DX2 markalı prosessorda takt tezliyi iki dəfə, i486DX4 prosessorunda isə 3 dəfə artırılıb. Bu isə prosessorun məhsuldarlığını demək olar ki, iki dəfə artırır.

MMX-göstərişləri 8×8 ölçülu tam ədədli 8-bitlik matrislə sadə komandaları yerinə yetirməyə imkan verir. Bu MPEG-kinofilmlərin göstərilməsində, MP 3 formatlı musiqi dinlənilməsində və JPEG alqoritmlərilə sıxılmış şəkillərin nəzərdən keçirilməsində istifadə olunan diskret konus çevirməsinin yerinə yetirilməsini tezləşdirir.

arxitektura elementlərinin tətbiqilə 80×86 prosessorlarının inkişafında yeni bir mərhələ açdı. i486 prosessorlarından fərqli olaraq onlarda məhsuldarlığı artıran, 8 Kbayt tutumlu komanda və verilənlərin ayrı-ayrı keş-yaddaşlardan istifadə edilirdi. Prosessorun əsas komandaları iki müstəqil icra qurğuları – konveyerlər üzrə paylanırdı. Birinci konveyer ixtiyari komandaları yerinə yetirirdi, ikinci konveyer isə tam ədədli sadə komandaları və sürüşkən vergüllü bəzi komandaları yerinə yetirirdi. Komandalar konveyerlərlə eyni zamanda göndərilirdi, daha mürəkkəb komanda birinci konveyerə, daha sadəsi isə ikinci konveyerə verilirdi. Təəssüf ki, belə iş tam ədədli komandaların məhdud çoxluğunda mümkün idi.

Demək olmaz ki, Intel firması XX əsrin 80-ci illərində "yaradarıq – düzəldərik" texnologiyasından əl çəkmişdi. 90-cı illərdə Pentium prosessorlarının multimedialı genişlənməsi olan MMX (*ing.* Multi-Media eXtension) modifikasiyası yaradıldı. Multimedialı genişlənmədə 64-mərtəbəli 8 registr hesabına onlarla yeni komanda əlavə olundu. MMX-registrlərinə yiyələnmək üçün sürüşkən vergülli əmrləri yerinə yetirərkən istifadə olunan registrlərə yiyələnmə mexanizmi tətbiq olunurdu ki, bu da komandaların eyni zamanda yerinə yetirilməsinə imkan vermirdi.

Qeyd etmək lazımdır ki, növbəti Pentium II prosessorunda bu nöqsan düzəldildi. Buna baxmayaraq 90-cı illərin ortalarında bu, o qədər səs-küyə səbəb olmuşdu ki, firma yeni buraxılan prosessorların üzərində "Bug free" – "Səhv yoxdur" yazısını yerləşdirirdi.

KLONLARIN HÜCUMU. IBM HARADADIR?

IBM PC kompüterlərinin quruluşunun açıqlığı, satış həcminin artımı ilə yanaşı, həm də dünyanın başqa istehsalçıların çoxsaylı analoq maşınları ilə dolmasına səbəb oldu. 1984-cü ildə belə maşınları 50-dən çox kompaniya istehsal edirdi,

iki ildən sonra isə klonların satış həcmi, IBM firmasının istehsal həcmini üstələmişdi. Bundan sonra IBM firması bir daha liderliyi özünə qaytara bilmədi. O vaxtdan etibarən "IBM PC" adı, "kseroks" adı kimi rəmzi xarakter daşıyır.

Təbii ki, 1984-cü ildə IBM PCjr – ev fərdi kompüterinin istehsalını böyük səhv hesab etmək olar. Maşın yaxşılaşdırılmış (IBM PC-yə nəzərən) qrafikaya, səs kartına və simsiz klaviaturaya malik idi. Çox pis işləyən klaviatura ev maşınının hörmətdən düşməsinə səbəb oldu. PCjr-in yaxşılaşdırılması (yeni kartların əlavə edilməsi) cəhdləri nəticə vermədi və kompüter 1986-cı ildə istehsaldan çıxarıldı.

1984-cü ildə IBM kompaniyası fərdi kompüterlərin yeni – IBM PC/AT seriyasının buraxılmasını elan etdi. İlk maşınlar 6 Mhers tezliyində (lakin sonralar 12 və hətta 20 Mhers tezlikli prosessorlar yarandı) işləyirdi. Intel 80286-dan AT-klonlarda istifadə edilməyə başlandı. Belə bir əfsanə vardı ki, BIOS (ROM-da qurğuların əsas funksiyaları saxlanılan yer) o illərdə əsasən IBM firması tərəfindən buraxılırdı, onun kodları gizli saxlanılırdı və hər bir kompüterdə dəyişdirilmədən sadəcə yerləşdirilirdi. Klonlarda əsl IBM PC/AT-yə nəzərən daha yüksək takt tezlikli prosessorlardan istifadə edilməsin deyə, startda prosessorun sürəti yoxlanılırdı: əgər sürət 6 Mhersdən yüksək olardısa kompüteri saxlayırdılar. Deyilənə görə, bu məqsədlə klonlarda "Turbo" (tezləşdirici) düyməsi qoyulurdu. Startda 12 meqahersli prosessor 6 Mhersli kimi işə salınırdı, sonra istifadəçi düyməni basırdı və prosessor iki dəfə tez işləyirdi. Korpuslardan "Turbo" düyməsi Pentium prosessorlarının yaranması ilə yoxa çıxdı, baxmayaraq ki, artıq digər firmalar da heç bir yoxlama aparmadan BIOS buraxırdılar.

1987-ci ildə Compaq Computers kompaniyası tərəfindən 80386 prosesoru əsasında ilk kompüter istehsal olundu. Həmin ilin aprelində IBM firması liderliyi özünə qaytarmaq üçün fərdi kompüterlərin PS/2 (Personal System) ailəsini yaratdı. Bu kompüterlər şin və birləşmələrin digər standartlarından istifadə edirdilər, buna görə də onun istehsalçılarını uğursuzluq gözləyirdi. Compaq (AST, Epson, Hewlett-Packard, NEC, Olivetti, Tandy, Wyse və Zenith) firması başda olmaqla klon hazırlayan doqquz kompaniya IBM PC/AT ilə tam uyğunlaşdırılmış öz şin standartlarını buraxdılar. Mikroprosessor istehsalçıları da geri qalmırdılar. Advanced MicroDevices (AMD) və Cyrix firmaları i386 və i486 ilə uyğunlaşdırılmış prosessorlar işləyib hazırladılar ki, onların qiyməti daha ucuz, sürəti daha yüksək idi.

IBM firması əməliyyat sistemlərinin istehsalı zamanı da uğursuzluğa məruz qaldı. 1994-cü ildən, Microsoft firmasının Windows 95 sistemindən əvvəl tam 32-mərtəbəli çoxtapşırıqlı OS/2 Warp 3.0 əməliyyat sistemi yaradıldı. Onun təqdimatı çox təntənəli surətdə keçdi. Buna baxmayaraq bu boşluq artıq Windows 3.11 sistemi tərəfindən tutulmuşdu. İstifadəçilər artıq bu sistemə öyrəşmiş və səbirsizliklə Windows 95-in gəlişini gözləyirdilər. IBM çətin vəziyyətə düşmüşdü. Bir tərəfdən öz Power PC prosessorlarının əsasında gələcək



IBM PC XT-nin klonu.

Intel firmasının ən böyük uğursuzluqlarından biri 1982-ci ilin sonunda buraxılmış iAPX 432 idi. O, ümumi məhsuldarlığı saniyədə 2 milyon komanda olan dörd mikrosxem yığınından ibarət idi. Bu, əməliyyat sistemlərinin proseslərin planlaşdırılması və məlumatlar mübadiləsi kimi bir çox funksiyalarını təmin edən obyektyönlü prosessordur. iPX432, Motorola 68000 və Intel 80286-ya nəzərən 10 dəfə az məhsuldarlığa malik idi.

PS/2 kompüteri.



fərdi EHM-lər üçün OS/2 hazırlanırdı, digər tərəfdən IBM, öz IBM PC kompüterlərini əməliyyat sistemlərilə təmin etmək üçün Microsoft-la əməkdaşlıq edirdi. OS/2, firmanın daxili siyasətinin və biznes-strategiyasının uğursuzluğu üzündən görkəmli bir ideyanın məhvinə klassik misaldır.

IBM indi haradadır? Mavi nəhəng əvvəlki kimi superkorporasiyadır və XXI əsrə də adlayaraq özünü pis hiss etmir.

STİV COBS NƏ BARƏDƏ BİLMİRDİ

İlk ixtiraçılar çox vaxt özlərini başladığı işi tamamlamaq üçün ilahi hüquqa malik kimi aparırlar. Məqsədə çatmaq üçün onların nə özlərinə, nə həmkarlarına, nə əməkdaşlarına, nə də müdirlərinə yazığı gəlir. Onlar nəhəng iradəyə malikdirlər. Bütün bu deyilənləri Apple kompüter firmasının yaradıcılarından biri olan Stiven Cobsa aid etmək olar. Onun şüarı: "Bu mənim yolumdur və bu ən yaxşı yoldur".

Apple firması 1 aprel 1976-cı ildə yaradılmışdı. Firmanın prezidenti Mayk Skott əsas işi kompüterlərin istehsalı üçün maliyyə vəsaitlərinin axtarılmasında görürdü. O, biznes-planlar tərtib edir, marketinq araşdırmaları aparırdı. Lakin kompaniyanın maliyyə vəziyyəti uzun müddət ağır olaraq qalırdı. 1980-ci ildə artıq Apple-II kompüterlərinin 130 mindən çox nüsxəsi satılmışdı (1980-ci ildə Cobsun aksioner gəliri 256 milyon dollar təşkil etdi).

Stefan Voznyak kompüterlərin hazırlanması ilə məşğul idi, Cobs isə katalizator rolunu oynayaraq, nəinki yeni bazarın yaranmasına təkan vermiş, həm də bu riskli işi təşkil etmişdi. O, özü üçün qaydalar müəyyənləşdirir və özü də bu qaydaları, insanlara qarşı saymamazlıq göstərərək pozurdu. Seçilmiş məqsəd uğrunda Stiven hər şeyi qurban verməyə hazır idi və başqalarından da işə bu cür münasibət tələb edirdi.

Lakin Cobsun heç də bütün qərarları gəlir gətirmirdi. O, uyğunlaşdırılmış kompüterlərin yaradılmasına etiraz edirdi. Apple-II, Lisa, Macintosh və Apple-III hətta öz aralarında belə uyuşmurdular. Çox qəribədir ki, bu cür böyük səhv buraxmaqla kompaniya Apple firmasının maşınlarının istifadəçilərindən ibarət böyük bir fanatlar ordusu qazanmışdı. Bu rahat texnika əvvəldən tədris kompüterləri bazarında qabaqcıl yer tutmuşdu.

1981-ci ildə Stiven Cobsu kompaniyanın rəhbəri təyin etdilər. 1982-ci ilin yekununda ona fərdi kompüterlər üzrə "İlin adamı" mükafatı verildi (o vaxtlar Stivenin gəliri 500 milyon dollar təşkil edirdi).

1983-cü ildə Apple firması yeni maşın – Lisa maşını buraxdı. Bu GUI (graphic user interface) pəncərə sistemindən istifadə edən birinci kompüter idi. Digər yenilik, indi çox öyrəşdiyimiz direktoriya və altdirektoriya, açılan menyuları olan iyerarxik fayl sistemi idi. Lakin qiymətinin yüksək (10 min dollar) və məhsuldarlığının aşağı olması bu maşının Apple firmasının Macintosh kompüteri tərəfindən bazardan sıxışdırılmasına səbəb oldu. Lisa, Apple firmasının ən böyük istehsal səhvlərindən biri hesab olunur, belə ki, üç il ərzində bu kompüterin istehsalı kompaniyaya heç bir maliyyə xeyri gətirmədi.

Stiven Pol Cobs 1956-cı ilin fevralında Maunteyn – Vyu (Kaliforniya ştatı, ABŞ) şəhərində anadan olmuşdu. Bu rayon sonralar Silikon Vadisinin (mikrokompüter sənayesinin mərkəzi) ürəyi olmuşdu. Bir neçə ildən sonra burada Robert Noys özünün birinci inteqral mikrosxemini yaratmışdı.



Stiven Cobs və StefanVoznyak.

1983-cü ildə Apple firmasına Pepsi Cola kompaniyasının keçmiş icraedici direktoru Con Skalli başçılıq etməyə başladı. İki ildən sonra isə Cobsu idarəetmədən kənarlaşdırdılar.

1984-cü il uğurlu il oldu. Kompaniya bazar üçün yeni məhsul – Macintosh kompüterini hazırladı. IBM firmasına qalib gəlmək üçün düzgün marketinq hazırlığını aparmaq tələb olunurdu.

Reklam tərtibatçıları yeni addım atdılar. Belə ki, Amerika futbolu üzrə Superkubokun televiziya ilə oyununun yayımının tən ortasında ilk baxışdan bəsit süjetli bir reklam çarxı göstərdilər. Zal. İnsanlar. Nəzərlər ekrana dikilmişdir. Ekrandan IBM firmasını təcəssüm etdirən "Böyük Qardaş" yayımlanırdı. Hər şey məyus tondadır. Birdən bir qız qaçaraq daxil olur və ekrana çəkic atır. Ekran parçalanır və belə bir yazı çıxır: "Bu il Apple firması Macintosh kompüterini buraxır. Buna görə də 1984-cü il "1984" olmayacaq!" Bu klipdə Corc Oruelin məşhur antiutopik "1984" romanının mövzusundan istifadə edilmişdi. Bu cür təqdim olunmuş Macintosh böyük maliyyə müvəffəqiyyəti qazandı.

1988-ci ildə Apple kompaniyası kompüter istehsal edən digər firmalardan xeyli geri qalırdı.

Aşağıda bir il ərzində məhsul satışının nəticələri verilir (milyardlarla ABŞ dolları ilə):

- IBM – 59,681;
- Digital Equipment – 11,475;
- Hewlett-Packard – 9,831;
- Apple Computer – 4,071.

XX əsrin 90-cı illərində kompaniya fərdi kompüterlərin satış həcminə görə artıq IBM firmasını ötüb keçmişdi (həmin dövr üçün IBM praktik olaraq bu bazardan çıxmışdı). 1992-ci ildə Apple firması dünya bazarında fərdi kompüterlərin cəmi 10%-ni təşkil etsə də, onun illik gəliri 7 milyard dollara çatmışdı.

İlk modellərin dizaynı all-in-one stilində hazırlanırdı, yəni sistem bloku və 9-düymlü monitor ümumi çəkisi 9 kq olan bir korpusda buraxılırdı. Ekranın kiçik ölçülü olması təsvirin çox yüksək dəqiqliklə alınmasına imkan verirdi. Mərkəzi prosessor kimi Motorola 68000-dən, sonra isə daha güclü Motorola 68020 prosessorundan istifadə olunurdu.



Bu, həmin Macintosh-dur. İndi isə Classic adlanır.

90-cı illər Apple üçün itkilər və həyata keçməmiş ümidlər tarixi oldu. 1991-ci ildə kompaniya əməkdaşlarının 10%-ni işdən çıxardı. 1993-cü ildə isə Con Skalli Apple firmasını tərk etdi. Faktiki olaraq kompaniya üçün ən yaxşı dövr qurtardı. İki ildən sonra kompaniya ciddi itkilərə – təqribən 70 milyon dollar itkiyə məruz qaldı. Bir ildən sonra idarəetmədəki rotasiyalara baxmayaraq, bir kvartal ərzində itkilər 700 milyon dollar təşkil etmişdi.

1993-cü ildə Apple ən birinci PDA (*ing.* "personal digital assistant" – "fərdi rəqəmli köməkçi") – Apple Newton buraxdı. PDA-nın əsas reklam şüarı: "Newton sizin əlyazma mətnlərinizi oxuya bilir!" ilə onun bu imkanları dillərdə dastan oldu. Newton əlyazma mətnlərini çox pis oxuyurdu. Çoxlu

Cobs özünün prinsiplərindən birini KISS ("Keep It Simple, Stupid!" – "Bunu daha sadə et, sərsəm!") abbreviaturası ilə işarə etmişdi.

MACINTOSH ƏSRLƏRİN SƏRHƏDİNDƏ

Əksər istifadəçilər üçün naməlum olan kompüter, az saylı peşəkarlar üçün güclü köməkçidir. Macintosh özündə bir araya sığmayanları: bir tərəfdən etibarlıq və yüksək səmərəliliyi, digər tərəfdən ünsiyyətdəki sadəliyi özündə birləşdirir. Dişlənmiş alma şəkili loqotiplərin olduğu kompüterləri, Amerikada evlərdə və tədris müəssisələrində tez-tez görmək olur. Bu maşınlar arasında PowerBook – daşınan "çantalar", kub şəkilli orijinal Cube və daha çox öyrəşdiyimiz PowerMac G3 və PowerMac G4 "göydələnlər"i var. Çox böyük məhsuldarlığa malik iki G4 (bir baş yaxşıdır, ikisi daha yaxşıdır) prosessorlu modellər də var ki, bunlar kompüter qrafikası ilə məşğul olan rəssam və dizaynerlər arasında xüsusilə populyardırlar. iMac və eMac kompüterləri də var ki, onlarda monitor və sistem bloku bir korpusda yerləşdirlib. 2002-ci ildə qeyri-adi iMac-2 meydana gəldi. O, stolüstü lampalarda olduğu kimi kronşteyndə maye kristallar əsasında qurulmuş monitoru olan boz rəngli yarımsfera şəklindədir. iMac-ın məqsədli ucuzlaşdırılması bu "günəbaxanların" pərəstişkarlarının sayının artmasına səbəb oldu.

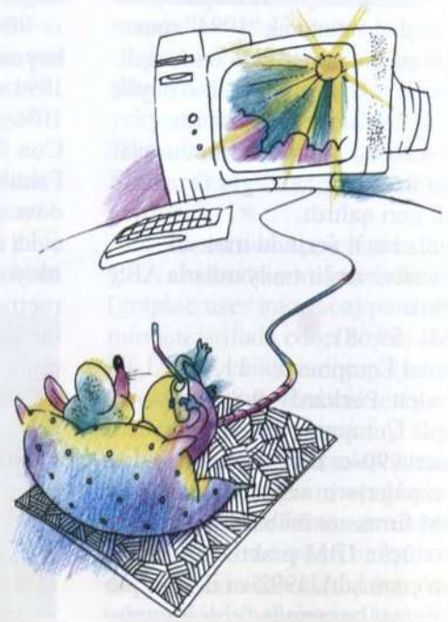


Macintosh (Mac-kompüterlər) əvvəlcədən "oturdun – işlə" prinsipilə işləyən yaradıcı insanlar üçün nəzərdə tutulmuşdu. Apple firması nəinki Mac kompüterlərini, həm də onlar üçün Mac OS (və ya System) əməliyyat sistemini istehsal edir. Bu sistem özünü elə aparır ki, bəzən onun olmasını belə hiss etmirsən. Proqramlar istifadəçinin səviyyəsindən asılı olmayaraq çox asanlıqla sazlanır. Proqram və drayverlərin yeni versiyaları tez-tez dəyişdirilmir və buna görə də bazara hər bir xırdalığa kimi düşünülmüş məhsullar çıxarılır. Köməkçi proqramlar çox az-az hallarda kompüterin imkanları uğrunda "ölüm-dirim" mübarizəsinə çıxırlar, buna görə də təsadüfi hallarda kompüterin işçi vəziyyətdən çıxması baş verir. Əlbəttə ki, səriştəli istifadəçilər üçün MacOS sistemində daha dərinə gedib, məsələn, tapşırıqları idarə edən AppleScript dilinin köməyilə yorucu işləri avtomatlaşdırmaq olar. Bu dil istifadəçinin verdiyi kompüter əməliyyatları ilə idarə edilən *skriptləri* (interpretasiya proqramlarını) təsvir edir.

MacOS X əməliyyat sistemi, özünü yaxşı göstərmiş Free BSD (azad yayılan UNIX) əməliyyat sisteminin əsasında qurulmuşdur və Macintosh-un çoxprosessorluğundan məharətlə istifadə edir. O, özünün kiçik "bacılarına" nisbətən daha dayanıqlıdır: yaddaşı qoruya bilir, hər hansı bir proqram "dayandıqda" bütün kompüteri yenidən yükləməyə ehtiyac yoxdur, həmin proqramı yaddaşdan çıxarıb işi davam etdirmək olar. Yeni Aqua (latıncadan "su") qrafik interfeysi həm görünüş baxımından, həm də rahat iş üçün nəzərdə tutulmuş bir çox özəlliklərə malikdir.

Bəziləri hesab edirlər ki, Macintosh kompüterləri PC-kompüterlərinin etdiyi geniş sinif məsələlərini həll edə bilmirlər. Bu səhv fikirdir. Hətta kompüter üçün oyun proqramları istehsalçıları belə Macintosh-u işsiz qoymurlar: praktik olaraq bu maşınlar üçün bütün "oyuncaqları" tapmaq olar. Şəbəkədə də "rəqiblər" özlərini birlikdə əla hiss edirlər. Əlbəttə ki, Macintosh, PC ilə uyuşmur, lakin xüsusi PC-Virtual PC emulyator-proqramını daxil etməklə, PC üçün yazılmış bütün tətbiqi proqramlar və oyunlar Macintosh-da da işləyəcəkdir. Axır zamanlar (Power MAC G4 kompüter modelindən başlayaraq) Macintosh-a əvvəlcədən PC üçün yaradılmış bir çox qurğuları: operativ yaddaşı və s. qoşmaq olur.

Nə vaxtsa Stiven Cobs Apple üçün belə bir şüar irəli sürmüşdü: "Fərqli düşün!". Kompaniyanın kompüter istifadəçilərini həmişə özünün qeyri-standartlığı ilə sevindirən "övladı" Macintosh bu şüarın ən yaxşı gerçəkləşdirməsidir.



sayda məzəli, gülməli səhvlər alınırdı. Buna baxmayaraq, Newton mətnlərin tanınmasında daha mükəmməl proqramların yaranmasına yol açdı.

1985-ci ildə Stiven Cobs Apple-dən çıxaraq NeXT Computer kompaniyasını yaratdı. Cobs tamamilə yeni kompüter yaratmaq istəyirdi. O, yeni ideyalarla zəngin idi. Qarşıya qoyulmuş məqsədə çatmaq üçün NeXT digər kompaniyaların ən yaxşı nəticələrini götürmək fikrinə gəldi. Bu UNIX əməliyyat sistemi, obyekt-yönlü Objective-C dili və Adobe PostScript-ə əsaslanan yeni vizuallaşdırma sistemi idi.

Beş il ərzində kompaniya öz hazırladığı kompüterləri buraxırdı, lakin 1993-cü ildə onların istehsalı dayandırıldı. Əməliyyat sistemləri isə əvvəlki kimi həyata keçirilirdi: əvvəlcə Intel prosessorları üçün NextStep yarandı, sonra o, Hewlett-Packard PA-Risc və Sun SPARC işçi stansiyası üçün uyğunlaşdırıldı.

NeXT Computer-də işlədiyi dövrdə Stiv Cobs özünü professional maliyyəçi kimi göstərdi. O, öz kompaniyasına 15 milyon dollar qoymaqla investorları cəlb etmişdi. O, milyarder Ross Pero ilə 20 milyonluq saziş bağlamış, Canon yapon konsernindən 100 milyonluq sərmayə ala bilmişdi. IBM şirkətilə sövdələşmə kompaniyaya on milyonlarla dollar gətirmişdi. Cobs, Businessland Computer mağazaları ilə saziş bağlayaraq, NeXT-in gələcək 100 milyon dollarlıq məhsulunu sata bilmişdi.

1997-ci ilin sentyabrında 12 illik fasilədən sonra Stiven Cobs yenidən Apple şirkətinin başına keçdi. Lakin bu artıq əvvəlki Cobs deyildi. Onun bir çox qərarları Apple kompüterlərinin pərəstişkarlarının etirazına səbəb oldu. Cobsun müvəffəqiyyətə aparan sadəlik və dürüstlük prinsipi kompaniyanın yeni strateqiyasında özünü göstərdi. Buraxılan texnikanın çeşidi azaldıldı, proqram təminatı sahəsindəki əsas səylər müasir əməliyyat sistemlərinə yönəldildi.



Stiven Cobs.

Geri qayıtmış prezidentin fəaliyyətinin nəticəsində satış həcmi artdı. Apple uzun sürən böhrandan çıxmağa başladı. Düzdür, burada Microsoft firmasının maliyyə dəstəyi olmadan keçinmək mümkün olmamışdır. Çox təəssüf ki, dünya elə qurulub ki, elmi və maliyyə müvəffəqiyyəti nadir hallarda yanaşı gedə bilir. Stiven Cobs hələ də başa düşmürdü ki, onun bütün dahiyanə ideyaları və işləri, sanki taleyin hökmü ilə əvvəlcədən uğursuzluğa düçar olmuşdu. Əgər belədirsə, onda Apple yenə də öz kompüterləri və əməliyyat sistemləriylə səs salacaqdır və onun ardınca yenə də maliyyə iflasına uğrayacaqdır.



Əgər sizin sevimli idman köynəyinizdə "Windows must die!" yazılıbsa, deməli, siz, çox güman ki, Microsoft firmasının pərəstişkarı deyilsiniz.

III UİLYAM (BİLL) HENRİ GEYTS

Fərdi kompüterlərin proqram təminatının aparıcı dünya istehsalçısı Microsoft korporasiyasının direktorlar şurasının sədri və yaradıcısı Bill Geyts 1955-ci il oktyabrın 28-də Sietldə anadan olmuşdur. O, ailədə üçüncü uşaq idi. Onun tam adı – III Uilyam (Bill) Henri Geyts idi (I Bill Geyts – babası mebel mağazasının sahibi idi, II Uilyam Geyts – atası vəkil idi). Onun anası Meri Geyts (qızlıq soyadı Maksvell) məktəb müəlliməsi, Vaşinqton ştatının universitetinin idarəetmə şurasının üzvü və United Way International xeyriyyəçilik təşkilatının sədri idi. Onun ana tərəfdən babası National City bankının yaradıcısı idi.

Bill Geyts öz təhsilini bələdiyyə məktəbində başlayır və Lakeside School elitar özəl məktəbində davam etdirir. Orada o, 13 yaşından proqramlaşdırma ilə maraqlanır.

1973-cü ildə Bill Geyts Harvard universitetinin birinci kursuna daxil olur. Burada o, ilk Altair 8800 kompüteri üçün BASIC proqramlaşdırma dilini işləyib hazırlayır. Lakin Bill Harvard universitetini bitirmədi. O, bu tədris müəssisəsini üçüncü tədris ilində yarımçıq tərk edərək, özünü fərdi kompüterlərin proqram təminatının yaradılması üzrə ixtisaslaşan Microsoft kompaniyasına həsr etməyə qərar verir. Bill bu firmanı 1975-ci ildə Pol Allenlə birgə yaratmışdı. O, başa düşürdü ki, fərdi kompüter hər bir iş yerində, hər bir evdə əvəzolunmaz qurğu olacaqdır.

Yanlış olaraq DOS əməliyyat sisteminin müəllifliyi Bill Geytsin adına yazılır. Əslində isə bu, sadəcə, uğurlu pul yatırımı idi. DOS sistemini hələ o vaxtlar Microsoft-da işləməyən Tim Paterson yaratmışdı. 1981-ci ildə Geyts və Allen IBM-in sifarişini yerinə yetirirdilər və Seattle Compute Productsda işləyən Patersondan onun Intel-in 16-mərtəbəli prosessorları üçün yaratdığı QDOS (Quick Disk Operating System – "sürətli disk əməliyyat sistemi") sistemini çox da böyük olmayan qiymətə almışdılar. Əlbəttə ki, bu əməliyyat sistemi üzərində işləmək tələb olunurdu, lakin bu sıfırdan başlayaraq yeni əməliyyat sisteminin yaradılması demək deyildi. IBM tərəfindən qoyulmuş bütün tələbləri nəzərə alaraq, kompaniyonlar, sistemi IBM PC-in təchiz olunduğu 8-mərtəbəli prosessorlar üçün yenidən işlədilər. O zamanlar praktiki olaraq heç kəsin tanımadığı Microsoft firması IBM ilə yeni əməliyyat sisteminin hazırlanıb təqdim olunması haqqında müqavilə bağlaya bildi. IBM bu sistemin istehsalı üçün tələb olunan maliyyə vəsaitini ödədi və kompüterləri yalnız PC DOS (MS DOS) versiyalı sistemlə satmağa boyun oldu. Microsoft satılan hər bir maşından faiz alırdı. İndi bu satış cavan menecerlər tərəfindən klassik bir sövdələşmə kimi öyrənilir. Belə ki, bu əməliyyat sistemi hər hansı bir böyük məbləğə satılsaydı, Bill Geyts hal-hazırda multimilyarder olmazdı.

1994-cü il yanvarın 1-də Bill Geyts Microsoft şirkətində işləyən Melinda Frençlə evlənir. Toy Havayda, Lanai adasında keçirilmişdi. Jurnalistlərin gəlməsinin qarşısını almaq üçün qabaqcadan bütün mehmanxana nömrələri və bütün avtomobillər, təyyarələr kirayələnmişdi. 2000-ci ilin əvvəlində Microsoft-un baş direktoru vəzifəsini tutmuş Stiv Ballmer toyda sağdış idi. 1996-cı il aprelin 26-da Geytslər ailəsində qız – Cennifer Ketrin Geyts dünyaya gəldi. 1997-ci il sentyabrında Bill Geyts Vaşinqton gölünün sahilində öz ailəsi üçün tam şəkildə kompüterləşdirilmiş ev tikdirdi. 1999-cu il mayın 23-də Geytsin oğlu – Rouri Con Geyts dünyaya gəldi.

1995-ci ildə Geyts, o zamanlar Microsoft-un vitse-prezindenti olan Natan Mirvold və jurnalist Piter Raynarsonla birlikdə "Gələcəyə gedən yol" adlı kitabını yazdı. Kitab ABŞ-da Viking nəşriyyatında çap olunmuş və 18 həftə ərzində bestsellerlər siyahısında olmuşdur. Onun 1999-cu ildə dərc olunmuş ikinci kitabı – "Xəyal sürətli biznes" Kollinz Heminqueylə birgə yazılmışdı və sonralar 25 dildə dərc olunmuşdu. Hər iki nəşrdən gələn gəlirləri Bill Geyts xeyriyyəçilik fonduna verdi. Bu fond tədris prosesində kompüter texnologiyalarından istifadə edən dünya müəllimlərinə yardımla məşğul olurdu.

Microsoft daima informasiya məhsul və texnologiyalarını yaradır və təkmilləşdirir. Onun uğuru – kompüterlə işi daha sadə və əlverişli etmək niyyətindədir. Kompaniya elmi-tədqiqat işlərinə külli maliyyə vəsaitləri xərcləyir.

Lakin pəncərə sistemlərindən istifadə etmək fikri nə Bill Geytsə, nə də Microsoft firmasının hansısa bir əməkdaşına aid deyildir. Bu sistemi Stiv Voznyakla birlikdə ilk fərdi kompüteri yaratmış Stiv Cobs artıq tətbiq etmişdi. Apple şirkətinin kompüterləri Lisa və Mac bazarda IBM-i pəncərə sistemlərilə üstələyirdilər. 80-ci illərin sonunda Atari oyun kompüteri pəncərə əməliyyat sisteminin və 32-bitlik Motorola prosessorunun imkanlarına görə IBM AT-dən öndə idi. Onda 1985-ci ildə Geyts Windows sistemini yaratdı. Birinci versiyalar (3.0 versiyasına qədər) bir qədər işə yaramırdılar. Təkcə pəncərə interfeysini deyil, həm də bir kompüterdə eyni zamanda bir neçə proqram yerinə yetirmək imkanını daxil etmək tələb olunurdu. MS DOS üzərində qurulmuş Windows 3.0 versiyasında bunu etmək mümkün oldu. Eyni zamanda bir neçə pəncərə açılırdı və bir məsələdən digərinə keçmək qalırdı. Belə bir mexanizm DOS-da, ancaq MS DOS-da deyil, 8-bitlik EHM-lər (CP/M) üçün əməliyyat sistemlərinin müvəffəqiyyətli yaradıcısı Digital Reserch şirkətinin DR DOS versiyasında alınmışdı. DR DOS çox yaxşı əməliyyat sistemi idi. Əgər Bill Geytsin rəqiblərilə mübarizə aparmaq bacarığı olmasaydı, bu, MS DOS-un sonu demək idi. Məsələ onda idi ki, Windows yalnız MS DOS-la işləyə bilirdi və digər əməliyyat sistemlərilə uyuşmurdu.

1997-ci ildə Bill Geyts öz ideyasını – baş rəqibi olan Apple şirkətinə 150 milyon dollar kredit verilməsini həyata keçirdi. Bundan əvvəl o, şirkəti bankrotluq və ləğv olunma təhlükəsi qarşısında qoymuşdu. Bu sövdələşmə Bill Geyts və onun şirkətinə antimonopol araşdırmaları və şirkətin məcburi bölünməsindən can qurtarmaq kimi uğurlu imkanlardan başqa nə gətirdi?



Bill Geyts.


Pol Allen.


Tim Paterson.

Əgər siz üç yaşlı uşağın "Səma nə üçün mavidir?" sualına Windows haqqında nə isə danışmağa başlayırsınızsa deməli, siz "Microsoft"-un məhsulu ilə işləyirsiniz.

Microsoft, İnternetə irişmə proqramının Apple əməliyyat sisteminə daxil ediləcəyini elan etməklə, öz rəqib şirkətləri olan Netscape, Sun, Oracle firmalarına zərbə endirdi. Eyni zamanda Microsoft, NeXT (öz adı ilə IBM PC XT maşını olmadığını göstərən maşın) əsasında Apple əməliyyat sisteminin bir rəqib kimi ləğv ediləcəyinə zəmanət verdi. Özündən asılı olmayaraq Apple şirkəti Java (Sun şirkətinin buraxdığı, platformalardan asılı olmayan tətbiqi proqramların yaradılması üçün əsas olan) dilinin faktiki olaraq ləğv edilməsinə şərait yaratdı.

Microsoft provayderlərlə saziş bağlayaraq, Netscape loqotipinin çıxarılması əvəzinə onların Microsoft məhsullarının informasiya kanalları siyahısına daxil edilməsinə icazə vermişdi! Bu Explorer-in Netscape üzərindəki qələbəsi deyilmi? Bu, yeniləşməyə heç bir əmək və vəsait qoymadan, yalnız aşkar üstünlük hesabına qalib gəlməkdir (1995-ci ildə Netscape bu cür NSCA Mosaic, Cello və digər web-naviqatorlara qalib gəlmişdi), sadəcə olaraq iki məhsulun müqayisə olunmasına imkan verməməkdir.

Windows-un hər bir yeni versiyası bədbəxt istifadəçilərdən onların kompüterlərinin ciddi şəkildə təkmilləşdirilməsini tələb edir. Nəticədə istehsalçıların ciblərinə pullar su kimi axırdı. Bununla əlaqədar yada aşağıdakı məşhur zarafat düşür: "Hər bir yüksək məhsuldarlıqlı kompüter üçün onda zəif sürətlə işləyən Word tapılar". Lakin bununla Microsoft öz alıcılarını aldatmır: firma heç bir şeyə söz vermir və faktiki olaraq heç bir dəstək göstərmir. İşləməyən əməliyyat sistemi (əlavə ödəmə aparmaqla) yenə də korlana bilən yeni əməliyyat sisteminə dəyişdirilir və bu sonsuzluğa qədər davam etdirilir... Ancaq heç kəs bir ay işləyib sonra xarab olacaq avtomobili almaq istəməzdi.

Microsoft-un proqram məhsullarını yayan və istifadə edən piratlarla mübarizəsi göründüyü qədər də uğurlu deyildir. Belə ki, qanunsuz, ucuz təkrarlar narkotik kimi milyonlarla vətəndaşı Microsoft firmasının məhsullarından istifadə etməyə sövq edirdi. İnsanların qalan hissəsi isə MS Word-də yazılmış və dünyanın müxtəlif yerlərindən gələn məktub və sənədləri oxuya bilmək üçün həmin məhsullardan istifadə etmək məcburiyyətində qalırdılar. Elə ki, başqa firmalar öz məhsulları ilə Word-ü bütün parametrlər üzrə vurmağa başlayırlar və istifadəçilərinə MS Word-də yazılmış sənədlərlə rahat işləyə bilmək imkanı yaradırlar, Microsoft özünün yeni versiyasını buraxır. Bu versiya əvvəlkilərdən yalnız bir cəhətlə – sənədlərin başqa formatı ilə fərqlənir. Nəticədə bu qovhaqov davam edir.

Microsoft proqramlarının yüksək səmərəliliyinin səbəbinin firmada işləyən superproqramçıların əməyinin nəticəsi olması fikri bir nağıldır. Əslində onlar Windows sisteminin sənədləşdirilməmiş funksiyalarından və xüsusi kitabxanalardan istifadə edərək proqramlar yazırlar. Bununla da Windows ilə öz məhsulları arasındakı işdə "vasitəçi proqramlardan" istifadə etmədən keçinə bilirlər. Yerdə qalan proqramlar üçün bu gizli olaraq qalır. Buna görə də Outlook poçt proqramının xarakteristikaları rəqiblərdəkindən heç də üstün deyildir, sadəcə olaraq bu proqram Windows-un daxili funksiyalarından daha səmərəli istifadə edə bilir.

Bill Geyts Microsoft şirkətindən başqa bütün dünyadakı şəxsi və dövlət kolleksiyalarında olan incəsənət əsərlərinin və fotoşəkillərin rəqəmsal arxivinin hazırlanması ilə məşğul olan Corbis Corporation şirkətini də yaratmışdır. Bundan əlavə o, Icos Corporation əczaçılıq kompaniyasının idarə heyətinə daxildir. O, Chiroscience Britaniya əczaçılıq korporasiyasının bir hissəsi olan və gen mühəndisliyilə məşğul olan Darwin Molecular kompaniyasının bir çox aksiyalarının sahibidir. O, Yer kürəsinin ətrafında aşağı orbitə verilənlərin ikitərəfli yüksək sürətli ötürülməsini təmin edəcək bir neçə yüz peykin buraxılması layihəsi üzərində işləyən Teledesic kompaniyasına vəsait qoymuşdur.

Bill Geyts təkcə kompüter texnologiyaları ilə məşğul olmur, o həm də biotexnologiyalarla da maraqlanır, çoxlu kitab oxuyur, həmçinin qolf və bric oynamağı da sevir.

III Geyts – marketinq üzrə böyük mütəxəssisdir. Varidatını (60 milyard dollardan çox vəsait) o, özü yığmış və tarixə ən cavan milyarder kimi daxil olmuşdur. Hələ 1993-cü ilin fevralında Amerika jurnalı "US News and World Report" Geytsi bizim günlərin Rokfellerı adlandırmışdı.



XXI ƏSRİN KOMPÜTERİ

KOMPÜTER VƏ SAĞLAMLIQ

Kompüterlər evdə, ofisdə qəti olaraq öz yerlərini tutmuşlar. Kompüterlər o qədər adiləşiblər ki, onlardan istifadə ilə əlaqədər təhlükələr istər-istəməz yaddan çıxır. Tərəqqinin nailiyyətləri olan avtomobil, təyyarə, televizor kimi kompüter də insana mənfi təsirlərin mənbəyidir, düzgün istifadə edilmədikdə insan səhhətinə ziyan vura bilər, professional xəstəliklərə səbəb ola bilər.

Həkim, kimsə xəstədir: ya mən, ya da kompüter



Lakin insan kompüterlə ünsiyyətinin bütün tərəflərini nəzərə alsa, texniki təhlükəsizlik qaydalarını gözləsə, əməyin elmi təşkili haqqında fikirləşsə, mütəxəssislər tərəfindən tərtib olunmuş sanitar normalarını gözləsə, onda kompüter insanın yaxın, ağıllı, yorulmaz köməkçisinə çevrilər.

Bu problemlər insanla maşın arasındakı münasibətləri öyrənən *erqonomika* (*yun.* "erqon" – "iş" və "nomos" – "qanun") elmi tərəfindən araşdırılır. Erqonomika insanın əmək fəaliyyətinin kompleks öyrənilməsilə məşğuldur və buna görə də bir çox elmi istiqamətləri: fiziologiya, əməyin gigiyenası, psixologiya və s. birləşdirir. Alimlər insanın kompüterdə işilə əlaqədar, onun orqanizminə düşən yükün azaldılması yollarını tapmağa çalışırlar, mükəmməl və təhlükəsiz texnikanın yaradılmasına cəhd edirlər. Onların qarşısında duran məsələlər – iş yerinin təşkili, kompüterin insana göstərdiyi zərərli təsirin profilaktikasıdır.

KOMPÜTERLƏ İŞ ZAMANI ALINAN ZƏRƏRLİ ŞÜALANMA

Kompüter bir neçə cür şüalanma və sahələrin mənbəyidir. Monitorun elektron-şüa borusu ionlaşdırıcı (rentgen) şüalanma yaradır. Müasir monitorlarda bu şüalanma cüzidir, çünki etibarlı surətdə ekranlaşır və təbii radiasiya fonu ilə müqayisə olunandır. Buna adi məişət radiometri vasitəilə asanlıqla əmin olmaq olar.

İxtiyari elektrik qurğusu kimi kompüter də elektromaqnit şüalanma yaradır. Kompüter işləyərkən ona daxil olan bütün qurğular (monitor, sistem bloku, klaviatura, printer, skaner və s.) və köməkçi elektrik avadanlığı (şəbəkə filtrləri, kəsilməz qidalandırma blokları) mürəkkəb elektromaqnit sahəsi formalaşdırır.

Elektromaqnit şüalanmasının təsiri üzrə aparılan araşdırmaların çoxu onun insan səhhətinə ziyanlı olduğunu göstərir.

Bundan əlavə, kompüter elektrostatik sahə də yaradır. Bu sahə monitorun ekranının yüklənmiş hissəciklər axını ilə şüalanması nəticəsində yaranır. Elektrostatik sahəsi toz və aerozol hissəciklərinin insanın üzünə, boynuna, əllərinə hopmasına səbəb olur. Nəticədə bu cür təsirlərə həssas olan insanlarda neqativ dəri reaksiyaları – quruluq, allergiya əmələ gəlir.

Elektrostatik sahə havanın ion tərkibinə də təsir göstərir. Monitorun kineskopu üzərində müsbət yük yaranır və o, havadakı mənfi yüklü xeyirli ionları neytrallaşdırır, bu isə kompüterin yerləşdiyi otaqdakı mühiti pisləşdirir.

KOMPÜTER VƏ GÖRMƏ QABİLİYYƏTİ

Artıq kompüterləşmənin birinci illərində displey istifadəçilərində görmə qabiliyyətinin spesifik zəifləməsi müşahidə olunurdu. Bunun ümumi adı "kompüter görmə qabiliyyəti sindromudur" (*ing.* Computer Vision Syndrome və ya CVS). Bunun yaranmasının bir neçə səbəbi var. Əsas səbəb ondadır ki, milyon illərlə davam edən təkamül nəticəsində insanın görmə sistemi obyektləri (təbiəti, şəkilləri, çap olunmuş mətnləri və s.) əks olunmuş işıqda qəbul etməyə öyrəşib, displey qarşısında oturub işləmək üçün nəzərdə tutulmayıb. Displeydəki təsvirlər göz tərəfindən müşahidə edilməsi adi hal olan təsvirlərdən prinsipial olaraq fərqlənirlər. Displeydəki təsvirlər işıqlanırlar, ayrı-ayrı nöqtələrdən – piksellərdən təşkil olunub, bu nöqtələr sayrışırlar (nöqtələr müəyyən tezliklə sönüb-yanırlar) və təbii rənglərə uyğun gəlmirlər.

Kompüter qarşısında fasiləsiz uzunmüddətli iş zamanı gözlərin dincəlmək üçün imkanı olmur, onlar gərginləşirlər, iş qabiliyyəti aşağı düşür. Göz orqanı ən

Mərkəzi və ürək-damar sistemləri kompüterin təsirinə ən çox həssas olan insan üzvləridir. Şərti-reflektor fəaliyyətdə pozuntular, beyinin bio-elektrik aktivliyində zəifləmə və neyronlararası əlaqələrdə dəyişikliklər müşahidə olunur. Endokrin sistemin işində də çatışmamazlıqlar mümkündür. Halsızlığın birinci simptomu artıq əsəbilik ola bilər, sonra fiziki və əsəbi-psixoloji zəiflik əmələ gəlir. Elektromaqnit sahəsinin xroniki təsirinin ümumi kliniki durumu üçün aşağıdakılar xarakterikdir: baş ağrısı, yorğunluq, halsızlıq, hipotoniya, bradikardiya, ürək əzələlərinin keçiriciliyinin dəyişməsi. Qeyd etdiyimiz hallar zəif, orta və aşkar görünə bilər. Bəzən qanda cüzi və adətən keçici olan dəyişikliklər müşahidə olunur.



Qoruyucu ekran.

böyük yükü informasiyanın EHM-ə daxil edilməsi zamanı alır. Belə ki, bu zaman istifadəçi gözünü tez-tez müxtəlif məsafələrdə yerləşən ekrandan mətnə, mətndən klaviaturaya və s. gəzdirməyə məcbur olur.

CVS-in əlamətləri görmə qabiliyyətinin zəifləməsi, yaxın məsafədəki əşyalardan gözün çətinliklə uzaqdakı əşyalara gəzdirilməsi, əşyaların ikiləşməsi, oxu zamanı gözlərin tez yorulması, gözlərin qaşınması, göz qapaqları altında "qum" olması təsəvvürü, gözlərin hərəkəti zamanı göz bəbəyində və alında ağrıların olması ilə müəyyən olunur.

İstifadəçilərin bir hissəsində CVS əlamətləri ekran qarşısında 2 saat, çoxlarında 4 saat və praktik olaraq hamıda 6 saat fasiləsiz işlədikdə əmələ gəlir. Bu zaman informasiyanın ekrandan oxunması, onun daxil edilməsinə nisbətən daha yüngül iş hesab olunur.

ƏZƏLƏ VƏ OYNAQ XƏSTƏLİKLƏRİ

Həkimlər EHM istifadəçilərində rast gəlinən bir neçə sindromu qeyd edirlər.

Onlardan biri – uzunmüddətli statik yük sindormudur ki, onun da təzahürləri əllərdə, boyunda, beldə olan ağrılardır. Rahat olmayan oturuş vəziyyətində ayaq, çiyin, boyun və əl əzələləri uzun müddət yığılmış halda qalırlar. Əzələ toxumaları uzun müddət gərgin vəziyyətdə qaldığından onlarda qan dövranı, maddələr mübadiləsi pozulur. Nəticədə əzələlər daimi yorğunluq vəziyyətinə düşürlər və zaman keçdikcə zəifləyirlər. Bu onurğa sütununun əyilməsinə, skeletin digər dəyişmələrinə, əzələ toxumalarının tamamilə dəyişib başqalaşmasına gətirib çıxara bilər.

Digər sindrom – bilək tunel sindromu və ya bilək kanalı sindromudur. Onun əmələ gəlməsinin səbəbi klaviaturada istifadəçi üçün rahat olmayan və uzun



müddətli iş nəticəsində barmaqları hərəkət etdirən əzələlərin vətərlərinin əlin orta sinirini sıxması ilə əlaqədardır. Narahat vəziyyətdə əl daraqlarının tez-tez təkrarlanan hərəkətləri nəticəsində əl vətərləri bilək sümüklərinə və oynaqlarına sürtülürlər. Nəticədə vətər və sinirlərin sıxılması üzündən ciddi xəstəliklər inkişaf edir. Xəstəliyin başlanğıc təzahürləri əldəki titrəmə, qaşınma və ağrılardır. Bu təzahürlər kompüterdəki iş qurtardıqdan bir neçə saat sonra özünü göstərir. Adətən insanların çox hissəsi bunu öz işlərilə bağlamırlar, bu isə nəticədə sindromun ağır hallarına gətirir. Tədricən əvvəlki təzahürlərə əldəki ağrılar, ağırlıq və keyləşmə də əlavə olunur. Ən ağır formada bilək tunel sindromu insanı iş qabiliyyətindən məhrum edir və cərrahi müdaxilə tələb edir.

SƏHHƏTİ NECƏ QORUMALI

Kompüterdə işlə bağlı peşə xəstəliklərinin profilaktikası avadanlığın qoyulmuş standartlara uyğun gəlməsinə və düzgün əmək rejiminə riayət olunmasına əsaslanır.

Adətən, istifadəçinin səhhətinə ən çox giriş-çıxış qurğuları: monitor, klaviatura və siçan ziyan gətirir. Fərdi kompüterin avadanlığının ekoloji təhlükəsizliyi üçün çoxlu sayda müxtəlif standartlar meydana gəlir. Müasir monitor təhlükəsizliyin qəbul olunmuş ümumi normalarına və erqonomikaya uyğun gəlməlidir.

Qapalı fəzada ətraf mühitin radio xətaların təsirindən qorunması üçün kommunikasiyalar üzrə Kanada federal komissiyası FCC Class B standartı işləyib hazırlayıb. FCC Class B standartının tələblərinə uyğun gələn avadanlıq tele və radio qurğularının işinə mane olmamalıdır.

Standartlar üzrə İsveç milli departamentinin buraxdığı MPR-II isə ofisdə isitifadə olunan kompüter monitorlarının və sənaye texnikasının verdiyi şüalanmaya məhdudiyyətlər qoyur.

İsveç həmkarlar təşkilatları konfederasiyası və İsveçin sənaye və texniki inkişafının milli şurasının tovsiyə etdikləri TCO-99 ətraf mühitlə qarışılıqlı münasibətləri reqlamentə gətirir. O, istifadəçinin qorunması məqsədilə elektrik və maqnit sahələrinin mümkün qədər azaldılmasını tələb edir. TCO-99 sertifikatını almaq üçün monitor kiçik şüalanma standartlarına uyğun gəlməlidir. Yəni, o, kiçik səviyyəli elektromaqnit sahəsinə malik olmalı, yanğın və elektrik təhlükəsizliyinin Avropa standartlarına uyğun gəlməlidir.

EPA Energy Star VESA DPMS (ABŞ) standartına uyğun olaraq monitor aşağıdakı üç enerjiyə qənaət rejimlərinə malik olmalıdır: gözləmə (standby), dayandırma (suspend) və "yuxu" (off). Belə monitor, kompüterlər uzun müddət işsiz vəziyyətdə qaldıqda enerjini qənaətlə işlədən rejimlərdən uyğun gələninə keçir.

Monitor təsvir parametrlərini (işıqlılıq, kontrast və s.) idarə etmək imkanlarına malik olmalıdır. Kompüterlə iş zamanı monitorun şaquli açılma tezliyinin 85 Hersdən aşağı olmaması məsləhətdir. Bu zaman istifadəçi gözün tez yorulmasına səbəb olan təsvirin titrəməsini hiss etmir.



Erqonomik klaviatura.

Kompüterin giriş qurğuları (klaviatura və siçan) üçün bu günə kimi ümumi qəbul olunmuş və geniş yayılmış standartlar mövcud deyildir. Bununla yanaşı bu avadanlıq istehsalçıları, öz məhsullarını reklam edərək, onlardan istifadənin erqonomikliyini artıran müxtəlif konstruktiv həlləri təsvir edirlər: düymələrin yerləşməsini tənzimləməyə imkan verən klaviatura, uzun müddətli iş zamanı əlin yorulmasını azaltmağa imkan verən formalı siçan .

Rusiyada da sanitar qayda və normalar kimi "Videodispleyli terminallara, fərdi elektron hesablama maşınlarına və işlərin təşkilinə qoyulan gigiyena tələbləri" – SanPiN 2.2.2.542-96 norması işləyir.



TCO-lu monitor.

TCO-03 loqotipi.

ƏL-QOLUN, BƏDƏNİN AÇILIŞMASI ÜÇÜN EDİLƏN HƏRƏKƏTLƏR

Əlinizi ovuc aşağı olmaqla stolun kənarına qoyun. Barmaqları o biri əlinizlə tutub, əlinizi geriyə çəkin və bu vəziyyətdə 5 saniyə saxlayın. Bu tapşırığı o biri əliniz üçün də təkrar edin. Əlinizi yüngülcə stola söykəyin və 5 saniyə ərzində barmaqları və biləyi gərin. Həmin tapşırığı o biri əliniz üçün də təkrar edin. Barmaqlarınızı yumruq şəklində bərk sıxın, sonra açın. Stulda düz oturub, ayaqlarınızı döşəməyə möhkəm sıxın (əgər stulun diyircəkləri varsa, onları bərkidin ki, tərpənməsin). Başınızı dizlərinizə çatdırmaq üçün irəliyə əyin. Bu vəziyyətdə 10 saniyə qalın, sonra isə ayaq əzələlərini gərməklə düz oturun. Bu çalışmaları 3 dəfə təkrar edin. Çoxları stolunun üzərində rezin oyuncaq və ya espander saxlayırlar, vaxtaşırı onun köməyilə əllərinin gərginliyini boşaldırlar.

Elekromaqnit sahələri üçün onlar beynəlxalq standartlara uyğun gəlir. Bundan əlavə bu sənəd kompüterlə iş zamanı əmək və istirahət rejimlərini və iş yerinin təşkilinə qoyulan tələbləri təyin edir. Xüsusi halda monitordan oxunan informasiyanın həcmi bir növbədə 60000 işarədən çox olmamalıdır, daxil edilən informasiyanın həcmi isə 40000 işarəni aşmamalıdır, monitor qarşısında keçirilən ümumi vaxt 6 saatdan çox olmamalıdır. İş günü boyunca hər 45-60 dəqiqədən bir 10-20 dəqiqə fasilə məsləhət görülür. Bu fasilələr zamanı fiziki məşqlər kompleksini yerinə yetirmək tövsiyyə olunur.

İş yerinin düzgün təşkili çox vacib şərtdir. Kompüterlə iş üçün nəzərdə tutulan otaqda həm təbii, həm də süni işıqlandırma olmalıdır. Buna görə də zirzəmilərdə iş yerlərinin təşkilinə icazə verilmir. Otaqda pəncərələrin şimal və ya şimal-şərqə açılması ən yaxşı haldır. Ofisləri nəinki isidici qurğular ilə, həm də kondisioner sistemi və ya effektiv ventilyasiya ilə təmin etmək lazımdır. Divar və tavanları tutqun rənglərlə rəngləmək lazımdır (parlayan, güzgü səthlər gözləri yorur və fikri işdən yayındırır). Otaqlarda hər gün yaş parçalarla təmizlik işləri aparılmalıdır.

İş yerinin sahəsinin ən azı 6 m^2, həcminin isə 20 m^3 olması arzu olunandır. Stolu pəncərənin yanında elə qoymaq lazımdır ki, işıq soldan düşsün. Kompüterlə iş üçün ən yaxşı işıqlandırma ekranda işıq ləkəsi verməyən, birbaşa olmayan paylanmış işıqdır.

İstifadəçinin gözü qarşısında tez-tez dəyişə bilən işıq olmamalıdır, buna görə də pəncərələri pərdə və ya jalüzlərlə örtmək məqsədə uyğundur. Süni işıqlandırma ümumi və müntəzəm olmalıdır, lakin yalnız stolüstü işıqlanma ilə kifayətlənmək olmaz.

Düzgün təşkil olunmuş iş yeri – mümkün xəstəliklərin profilaktikasında birinci addımdır. Kompüterlə iş insan səhhətinə zərər verməsin deyə, iş prosesində bədənin vəziyyətinə daim fikir vermək lazımdır. Düzgün qamət əzələləri maksimal olaraq boşaldır və daha az yorularaq daha çox iş görməyə imkan verir. Hesab olunur ki, düzgün qamətdə qulaqlar dəqiq olaraq çiyinlərin müstəvisində, çiyinlər isə budların müstəvisində yerləşməlidir. Başı çiyinlərə nəzərən düz tutmaq tələb olunur. Aşağı baxdıqda, baş irəliyə əyilməməlidir. Bel büküldükdə yəni çiyinlərin xətti dəqiq olaraq budların xətti və qulaqların xətti üzərində durmadıqda çiyin vətərləri və əzələləri üzərinə həddindən artıq yük düşür. Bu vəziyyətdə uzun müddətli iş xəstəliyə gətirib çıxara bilər.

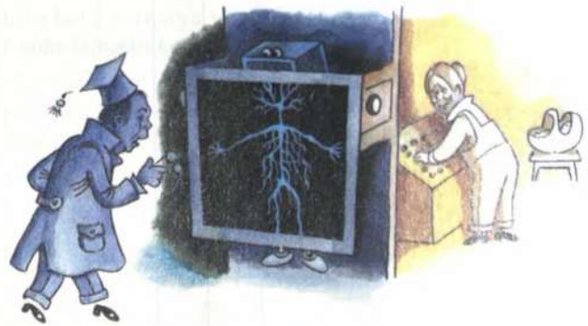


NEYROŞƏBƏKƏLƏR

İnsanın təfəkkür prosesi çox mürəkkəbdir və müasir alimlərə yalnız müəyyən mənada aydındır. Lakin elm insanın əsəb sisteminin quruluşunun öyrənilməsində müəyyən müvəffəqiyyətlər qazanmışdır. İnsanın əsəb sisteminin struktur vahidi əsəb hüceyrəsi – neyrondur. Bu cür hüceyrə gövdədən (nüvəli som) və ağacşəkilli çıxıntılar çoxluğundan – dendritlər və uzun aksondan ibarətdir. Dendritlər digər neyronlardan gələn əsəb impulsları üçün giriş kanalları rolunu oynayırlar. İmpulslar soma daxil olaraq onda spesifik həyacanlanma yaradırlar. Sonra buradan çıxış çıxıntısı olan akson üzrə yayılırlar. Neyronlar xüsusi kontaktların – sinapsların köməyilə birləşdirilirlər. Onlarda bir neyronun akson budaqlanması, digər neyronların som və ya dendritlərinə çox yaxın məsafədən (bir neçə on mikron məsafədən) keçir.

Reseptorlarda yerləşən neyronlar xarici qıcıqlanmanı qəbul edirlər. Onlar baş və onurğa beyninin boz maddəsində insanın hərəkətlərini (əzələ və vəzilərini) idarə edir, beyində hiss və hərəkət neyronlarını əlaqələndirirlər. Axırıncılar müxtəlif beyin mərkəzləri əmələ gətirirlər. Burada xarici qıcıqlandırıcılardan gələn informasiyanın hərəkət siqnallarına çevrilməsi baş verir.

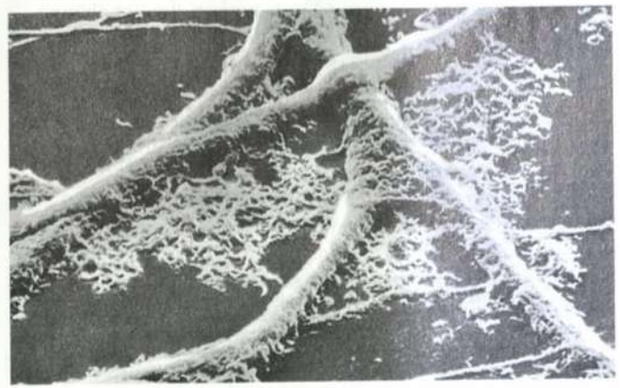
Bu sistem necə işləyir? Neyronlarda üç əsas proses baş verir: sinaptik həyəcanlanma, sinaptik tormozlanma və əsəb impulslarının yaranması. Sinaptik proseslər xüsusi kimyəvi maddələrin köməyilə baş verir. Bu maddələr bir neyronun çıxıntıları ilə ifraz olunur və digər neyronun səthilə əlaqəyə girir. Sinaptik həyəcanlanma neyronun cavab reaksiyasına səbəb olur və müəyyən həddə çatdıqdan sonra əsəb impulsuna keçir, çıxıntılar üzrə sürətlə yayılır. Tormozlanma isə əksinə, neyronların ümumi həyəcanlanma səviyyəsini azaldır.



Yüz milyardlarla neyron mürəkkəb sistem əmələ gətirir və super EHM-lər haqqındakı ən cəsarətli ümidləri hələ ki üstələyir.

Bu cür sistemlərin modelləşdirilməsi, yəni süni neyroşəbəkələrin yaradılması sahəsindəki təcrübələr hələ XX əsrin ortalarında aparılırdı. Burada məqsəd yalnız beynin funksiyalarını öyrənmək deyil, həm də süni neyroşəbəkələrdən istifadə edərək onları imkan daxilində təkrarlamaq idi. Neyronların şəbəkədə birləşmə sxemləri müxtəlifdir, lakin bütün şəbəkələr çoxqatlı fəza strukturlarından ibarətdir.

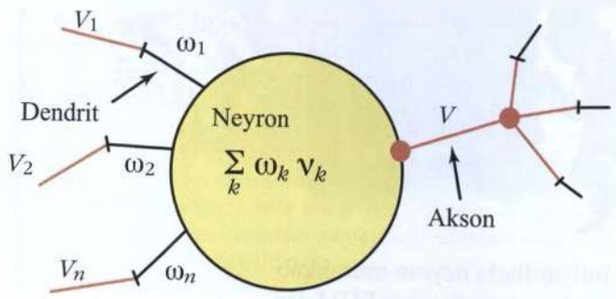
1949-cu ildə D.Hebb neyronlar şəbəkəsinin öyrənilməsini nümayiş etdirdi. Sonralar, 60-cı illərdə Marvin Minskinin də olduğu tədqiqatçılar qrupu süni neyronların bir qatından ibarət şəbəkə yaratdılar. Şəbəkələr müxtəlif məsələlərin



İnsan neyrosisteminin bir hissəsi

SÜNİ NEYRON ŞƏBƏKƏLƏRİ

Neyroşəbəkə düyünlərdən (neyronların analoqları) və birləşmələrdən (sinaptik əlaqələr) ibarətdir.



V_i – başqa neyronlar tərəfindən yaradılan və i dendritlərinə daxil olmuş impulslar, ω_i – dendritlərin çəkisi. V impulsu başqa neyronların dendritləri arasında budaqlanan aksonun köməyilə paylanır.

Neyroşəbəkədə sinaptik əlaqələrin çəkisini, şəbəkənin öyrədilməsi (və özünü öyrətmə) hədlərinin qiymətini vermək və dəyişmək olur. Belə sxem, həyəcanlanmanın şəbəkə boyunca keçid yollarını təyin edir və bununla "göndəriş – nəticə" rabitəsini formalaşdırır. Şəbəkədə xarici həyəcanlandırmanı qəbul edən giriş (reseptor) qat və məsələnin həll nəticələrini təyin edən çıxış qat mövcuddur. Şəbəkənin işi, keçən həyəcanlanmanın idarə edilməsi və imitasiyası üçün verilən impulslarla taktlaşır. Şəbəkə ya öyrədici rejimdə, ya da təyin etmə rejimində (iş rejimi) işləyir.

Süni neyroşəbəkə necə işləyir?

Tutaq ki, mətnin təyin edilməsi üçün şəbəkə qurulub. Giriş qatı təsviri, neyronlar çoxluğunun reseptorların müəyyən konfiqurasiyada həyəcanlandırılması ilə qəbul edir. Bir neçə sayda taktdan sonra çıxış qatının neyronlarından biri maksimal həyəcanlanır. Əgər girişdə işarənin etalonu, məsələn A hərfi verilirsə, onda çıxış qatının maksimal həyəcanlanmış neyronunu A hərfinin obrazı kimi qəbul etmək olar. Bu zaman sanki cavab alınır: "Bu *A* hərfidir, mən onu tanıdım!". Əgər sonra, nə vaxtsa yenidən *A* hərfi daxil edilərsə və bu daxil etmədə təhriflər olarsa, onda məntiqə görə həmin neyron həyəcanlanmalıdır, lakin bu başqa neyron da ola bilər və bu hal sistemin səhv davranışı hesab olunacaq. Sistemi "öyrətmək" üçün, çıxış qatındakı həmin neyronun maksimal həyəcanlanmasına nail olmaq lazımdır. Bunun üçün şəbəkədə həyəcanlanmanın keçid yolunda çəki və ya hədləri müəyyən alqoritmlərlə elə dəyişmək lazımdır ki, lazımi neyronu həyəcanlanmağa məcbur etmək mümkün olsun.

Şəbəkənin öyrədilməsi etalonlar çoxluğunun təqdim edilməsi və şəbəkənin parametrlərinin nizama salınması ilə həyata keçirilir. Öyrədilmə o vaxt dayandırılır ki, verilmiş şərtlərə uyğun olaraq səhv alınma ehtimalı kiçik olsun. "Öyrədilmiş" şəbəkəyə müəyyən mənada təhrif olunmuş müxtəlif hərflər daxil etməklə əmin olmaq olar ki, sistem onları təyin edəcək. Şəbəkə faktiki olaraq assosiativ təfəkkürü imitasiya edir.

həlli üçün, havanın proqnozlaşdırılması, elektrokardioqramların analizi və süni görmənin yaradılması üçün istifadə olunurdu. Sonralar Minski Paypertlə birlikdə "Perseptronlar" işində isbat etdilər ki, birqatlı şəbəkələr prinsip etibarı ilə bir çox sadə məsələləri, o cümlədən, VƏ YA-nın inkarını modelləşdirməyə imkan verən məsələni həll edə bilmirdi. Bu cür qiymətləndirmələr həddindən artıq pessimist idi. Kitabda sadalanmış məsələlərdən çoxu hal-hazırda çoxqatlı şəbəkələrdə standart prosedurlar yığınının köməyilə həll edilir.

80-cı illərin sonunda neyron şəbəkələrə maraq yenə də artdı. Artıq neyron şəbəkələrin öyrədilməsinə aid çoxsaylı nümunələr vardı: bir şəbəkəyə mətni fonetik ifadə şəklinə, sonra nitqə çevirməyi, başqa birinə isə əlyazma hərflərini tanımağı öyrətmişdilər. Süni neyroşəbəkələrdən istifadə nəticəsində təsvirin sıxılması sistemi qurulmuşdu.

Çoxqatlı şəbəkələrin əsas öyrətmə üsulu əks yayılma adlanır. Lakin bu üsul da şəbəkənin konkret zamanda öyrədiləcəyinə və təkrar öyrətmə nəticəsində daha yaxşı olacağına heç bir zəmanət vermir.

İnsanlar kompüterlərin səhv buraxmamasına öyrəşiblər, lakin kompüterdə modelləşdirilmiş süni neyroşəbəkələr, hətta düzgün işlədiyi halda belə, bəzən səhv buraxacaqlar. Neyron şəbəkələr məsələni necə həll etdiklərini "izah etmək" iqtidarında deyillər. Öyrətmə nəticəsində qurulan daxili ifadə o qədər mürəkkəb olur ki, onu analiz etmək mümkün olmur. İnsan da çox vaxt izdihamda uzun illər görmədiyi dostunu necə tanıdığını izah edə bilmir. Ekspert sistemi mühakimələr prosesini tərs qaydada elə izləyir ki, onların məntiqliyini yoxlaya bilsin.

Hesab olunur ki, süni neyron şəbəkələri süni intelekti əvəz edə bilir. Lakin ehtimal olunur ki, onlar, hər bir yanaşmanın daha yaxşı uyğun gəldiyi məsələnin həllində istifadə olunan sistemlərlə yanaşı mövcud olacaqlar. Obrazların tanınması çox vaxt sürətli reaksiya tələb edir. Əsəb sistemi yeni obyektin tanınmasında çətinlik çəkdikdə, bu sual beynin daha yüksək bölmələrinə ötürülür. Əlavə informasiyaya ehtiyac olarsa, onda həllin tapılması prosesi daha artıq zaman tələb edər. Lakin nəticənin keyfiyyəti bu halda daha yüksək olar. Neyron şəbəkə çox hallarda xarici mühitə belə reaksiya verərdi. Mürəkkəb hallarda, yəni səhv etmək ehtimalı böyük olan hallarda məsələnin həlli ekspert sisteminə tapşırılardı. Bu iki sistemin kombinasiyası daha effektiv olardı və bioloji təkamül nəticəsində alınmış modelə uyğun gələrdi.

Məşhur "Creatures" (*ing.* "yaradılma") oyununda norn adlanan gülməli məxluqların nəslini yaratmaq, tərbiyə etmək və öyrətmək tələb olunur. Buradakı personajların davranışını, xarici qıcıqlanmalara reaksiyasını və öyrədilməsini modelləşdirmək üçün neyron şəbəkələrindən istifadə edilmişdi.

NANOTEXNOLOGİYALAR

"NANO" NƏDİR?

Müasir elmi dildə "nano" önşəkilçisindən vergüldən başlayaraq vahidin milyardda bir hissəsinə bərabər fiziki kəmiyyətləri adlandırmaq üçün istifadə edilir. Məsələn, nanometr – metrin milyardda bir hissəsidir, yəni 1 nanometr (nm) = = 0,000 000 001 m = 10^{-9} m.

Alim və mühəndislər 1960-cı ildə anlayışın dəqiq mahiyyətini ifadə etmək üçün "nanos" (yunancadan tərcümədə "cırtdan" deməkdir) sözündən istifadə etmək haqqında razılığa gəldilər.

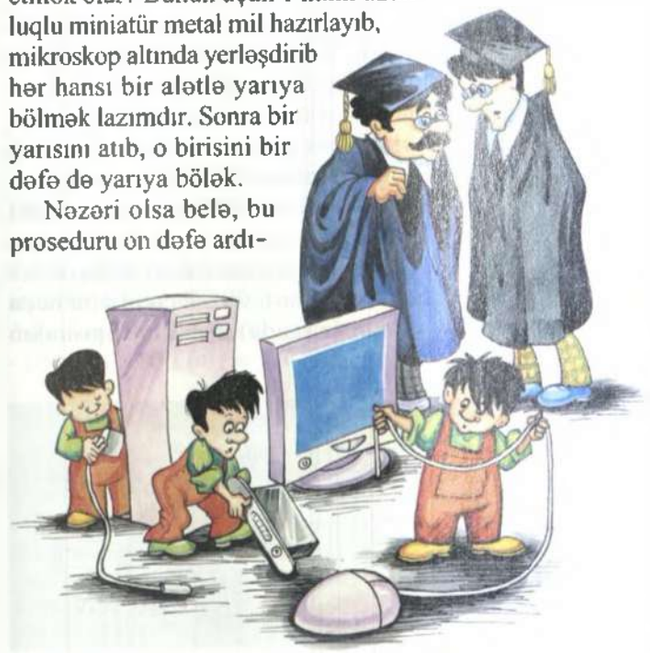
Beləliklə, nanotexnologiyalar dedikdə, nə isə cırtdan ölçülü, çox kiçik, nəyinsə milyardda bir hissəsini, məsələn metrin milyardda bir hissəsini təşkil edən şeylərlə məşğul olan texnologiyalar başa düşülür.

0,001 metr və ya millimetr ölçüsünə gündəlik həyatda tez-tez rast gəlinir. İynə başının ölçüsü 1 mm-dən bir qədər böyükdür. Lakin nanometrlə müqayisədə millimetr nəhəngdir, milyon nanometrə bərabərdir. Mühəndislər, artıq çoxdan millimetrin mində bir hissəsi olan mikrometrlə (mkm) və ya mikronla: 1 mkm=0,000 001 m=10^{-6} m=1000 nm ilə işləməyə öyrəşiblər.

İnsan tükünün diametri 1/20 mm, yəni 50 mkm-dir. Gül tozu insan gözü ilə görünməyən bir neçə mikron ölçülü tozdan ibarətdir. İnsan qanının analizi zamanı laborantlar mikroskop altında nümunələri nəzərdən keçirərkən birinci növbədə disk formalı bir neçə mikron diametrli qırmızı qan hüceyrələrinin sayını təyin edirlər.

Mikrondan kiçik ölçülərə tez-tezmi rast gəlinir? Hamı göydə və ya fəvvarənin su damlalarında göy qurşağını görüb. Göy qurşağında rənglər qırmızıdan bənövşəyiyədək dəyişir. Fizikadan bilirik ki, onlar işıq dalğasının uzunluğu ilə fərqlənirlər – 0,44 mkm (qırmızı) – 0,7 mkm (bənövşəyi).

Bəs, bir nanometri necə təsəvvür etmək olar? Bunun üçün 1 mkm uzunluqlu miniatür metal mil hazırlayıb, mikroskop altında yerləşdirib hər hansı bir alətlə yarıya bölmək lazımdır. Sonra bir yarısını atıb, o birisini bir dəfə də yarıya bölək.



Nəzəri olsa belə, bu proseduru on dəfə ardı-

Mikronun ölçü vahidi *μm* ilə işarə olunur. Lakin kompüterə mətnləri daxil edərkən yunan hərfini yazmaq çətinlik törətdiyindən qeyri-rəsmi olaraq daha çox *um* işarəsindən istifadə olunur. Əgər μ ("myu") yunan hərfində aşağıdakı quyruğu silsək, latın əlifbasının u hərfi alınacaq.

İşıq dalğasının uzunluğu o qədər də kiçik deyildir. İnsan tükünün diametrinə bərabər məsafədə işıq dalğasının cəmi yüz rəqsi yerləşdirilə bilir.



Skaner tunel mikroskopu.

cıl təkrar etmək mümkündürmü? Başlanğıc mil atomlardan təşkil olunub və bölünmədən sonra onun ölçüləri atomlar arasındakı məsafədən kiçik olmayacaqdır. Metalların çoxu üçün atomlar arasındakı məsafə 0,1–0,7 nm təşkil edir.

Beləliklə, 1mkm uzunluqlu mil üzərində cəmisi bir neçə min atom təbəqəsi yerləşər. Birinci bölmədən sonra təbəqələrin sayı təqribən iki dəfə, ikincidən sonra-təqribən dörd dəfə, onuncudan sonra-təqribən 1000 dəfə azalır, yəni on bölmədən sonra (əgər bunu yerinə yetirmək mümkün olarsa) milin uzunluğu 1nm olardı və mil cəmisi bir neçə (ondan az sayda) metal təbəqəsindən



ibarət olardı. Deməli, nanometr – kiçik sayda atomlardan təşkil olunmuş obyektlərin ölçülməsi üçün əlverişli ölçü vahididir.

İndi isə yenə "nanotexnologiyalar" sözünə qayıdaq. Əslində burada "nano" obyektlərin ölçülərini deyil, onların tipini göstərir. Nanotexnologiyalar ayrı-ayrı atomlarla fərdi işə istiqamətlənib.

Adi texnologiyalar (məsələn, xarrat dəzgahında detalın emalı, şüşənin bişirilməsi və ya beton konsrtuksiyaların hazırlanması) əşyaları hazırlayır, yəni pis idarə olunan atomların nəhəng qruplarıyla manipulyasiya edərək, atomları lazımi yığınlar şəklində qurur.

Nanotexnologiyalar informatika ilə sıx bağlıdır. Birincisi, qeyd edək ki, məhz R.Feynmanın informasiyanın kompakt şəkildə saxlanması məsələsinin qoyulmasına həsr olunmuş məruzəsindən sonra nanotexnologiyaların yaradılması zərurəti meydana çıxmışdır. İkincisi, nanotexnologiyalar, informasiya emalı alətləri (kompüterlər) ilə yanaşı onlardan istifadə üsullarını radikal olaraq dəyişəcəyinə ümid verir. Üçüncüsü, nanotexnologiyaların inkişafı informatika üsullarının tətbiqi olmadan mümkün deyil.

İLK LABORATOR UĞURLARI

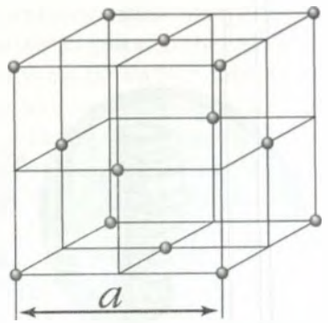
Hələ qədim Yunanıstanın filosofları maddənin atomlardan təşkil olunduğunu anlamışdılar. Sonralar bu nəzəri və eksperimental olaraq isbat olunmuşdu, lakin bir şahid and içə bilməzdi ki, "Mən atomu görmüşəm!". Ayrılıqda hər hansı bir atomu hər hansı bir optik qurğunun köməyilə görmək prinsip etibarı ilə mümkün deyil. Optik qurğular işıq rəqslərilə işləyir. Onların iş qabiliyyəti birbirindən ayırd edilə bilən iki obyekt arasındakı məsafə ilə ölçülür (bu məsafə işıq rəqslərinin dalğa uzunluğunu aşmır). Görünən işığın dalğa uzunluğu təqribən 500 nm-ə bərabərdir, bu isə atomlar

arasındakı məsafədən min dəfələrlə çoxdur. Atomları "güdmək" üçün daha çox tezlikli rəqslər lazımdır. Müasir fizikanın nöqteyi-nəzərindən elektron zərrəcik olmaqla yanaşı, həm də rəqs dalğasıdır. Bu dalğanın uzunluğu atomlar arasındakı məsafədən kiçikdir. Elektron dalğaların köməyilə "elektron işıqlı" mikroskopda ayrıca atomu görmək olar (onu başqa atomlardan ayırd etmək olar). 80-ci illərdə belə mikroskop yaradıldı və *skaner tunel mikroskopu* (STM) adlandırıldı.

Əsas ideya ondan ibarət idi ki, vakuumda bərk cismin səthində cərəyan buraxılmış iti iynənin ucu hərəkət etdirilir. Əgər nümunə ilə iynə ucu arasındakı məsafə kifayət qədər kiçikdirsə, onda elektronlar iynə ucundan nümunə üzərinə tunelləşirlər (atlanıb keçirlər) və beləliklə, tunelləşmə cərəyanı yaradırlar. İynəni nümunə uzərində gəzdirməklə və cərəyanı ölçməklə tədqiqatçılar nümunənin üzərindəki mikroskopik ölçülü (atom ölçülərində) "təpə" və "vadilərin" yerləşmə vəziyyətlərini "xəritəyə köçürmək" imkanı qazanırlar.

1986-cı ildə STM-i icad edənlər Nobel mükafatına layıq görüldülər. Skaner tunel mikroskopu insan ovcunda (vakuum kamerası olmadan) yerləşə bilir, şaquli istiqamətdə ölçüsü 0,1 Å olur, başqa sözlə desək, hidrogen atomunun diametrinin onda biri ölçüsünə malikdir. Eni cəmi bir neçə atom olan skaner iynəni üfüqi müstəvidə 2 Å ölçüyə qədər böyüdə bilir. Alimlər artıq eni cəmi bir atom olan iynə hazırlaya bilmişlər. İynənin ucu piramida şəklində hazırlanır. Burada axırıncıdan əvvəlki və axırıncı təbəqələr uyğun olaraq üç və bir atomdan ibarət olur.

STM atomları həm görə bilir, həm də onları hərəkət etdirə bilir. Məsələn, 1980-ci ildə IBM firmasının əməkdaşları nikel təbəqə üzərinə 35 ksenon atomu yerləşdirməklə, onlardan öz firmalarının adını düzəltmişdilər.



a – atomlar arasındakı məsafədir.

NANOTEXNOLOGİYALARDA "ƏMƏK BÖLGÜSÜNÜN" QAÇILMAZLIĞI

Ayrı-ayrı atom və molekullardan istifadə etməklə, prinsip etibarı ilə müxtəlif ölçülü yeni qurğular yaratmaq olur: insan gözü ilə görünməyən mikroskopik ölçülərdən tutmuş, Yer kürəsini də üstələyən planetlərin ölçülərinə kimi. Hətta ən xırda qurğular belə çox sayda atomlardan ibarət olur. Qurğunu nanotexnologiyaların köməyilə qurmaq üçün hər bir atomu onun üçün nəzərdə tutulmuş yerə keçirmək lazımdır. Xəyalımızda eksperiment keçirək. Bunun üçün bir tip atomlardan qurulmuş sadə bir fiqur – kub tələb olunur. Tili 5 mkm olan kubu yığmaq üçün nə qədər atomu hansı məsafədə yerləşdirmək lazım olacaq?

1 mkm = 1000 nanometr və ya 10000 anqstrem deməkdir. Əgər kubda qonşu atomlar arasındakı məsafəni 0,5 Å qəbul etsək, onda kubun tilindəki atomların sayı 100 001-ə, bütün kubdakı atomların ümumi sayı isə $100\,001 \cdot 100\,001 \cdot 100\,001 \approx 10^{15}$ olacaqdır.



"İxtiyari maddi əşya – atomların fəzadakı yığımından başqa bir şey deyildir. Bu atomların strukturda necə yığılacağı əşyanın nə olacağını təyin edir".

Stanislav Lem.

"ORADA – DƏRİNLİKDƏ..."



Riçard Feynman.

1959-cu ildə gələcək Nobel mükafatı laureatı amerikalı fizik Riçard Fillips Feynman Amerika fizika cəmiyyəti qarşısında "Orada – dərinlikdə istənilən qədər boş yer var (Fizikada yeni istiqamət açmağa dəvət)" adlı məruzəsilə çıxış etdi.

Nanotexnologiyalar bir neçə səbəbdən informatika ilə sıx bağlıdır. Tarixi olaraq nanotexnologiyaların müzakirəsi informasiyanın kompakt saxlanılması məsələsindən başlamışdı. Nanotexnologiyaların tətbiqi kompüterlər kimi onların tətbiq üsullarının da radikal dəyişdirilməsinə səbəb olur. Nanotexnologiyaların inkişafı informatika üsullarının tətbiqi olmadan mümkün deyil.

Məruzə gözlənilməz bir sualla başlayırdı: Britaniya ensiklopediyasının bütün 24 cildini iynə başlığında yazmaq olarmı?

Feynman bu suala müsbət cavab verdi. Onun düşüncələrini başa düşmək üçün məktəb proqramına daxil olan məlumatlardan əlavə adi metaldan hazırlanmış iynə başlığının atomları arasındakı məsafənin anqstremin (Å– uzunluq ölçü vahidi olub, 0,1 nm-ə bərabərdir) hissələri qədər olduğunu bilmək lazımdır.

"İynə başlığının ölçüsü təqribən 1,5 mm təşkil edir. Əgər iynə başlığının diametrini 25 min dəfə artırsaq, onun sahəsi təqribən ensiklopediyadakı bütün səhifələrin cəmlənmiş ümumi sahəsinə bərabər olacaqdır. Deməli, görülməsi lazım olan iş, 25 min dəfə xırda yazmağı öyrənməkdir. Bu mümkündürmü? İnsan gözünün görmə qabilliyyəti təqribən 0,2 mm-dir ki, bu da təqribən yarımton şəkillərin qurulduğu nöqtələrin diametrinə bərabərdir. Əgər bu cür nöqtəni 25 min dəfə kiçiltsək, diametri 80 Å olan sahə alınar, yəni ölçücə adi metalın atomları arasındakı məsafənin 32 misli-nə bərabər bir sahə alınar. Başqa sözlə, bizim nöqtə kiçildikdən sonra da minə yaxın atomdan ibarət olacaq. Deməli, bu prosesdə ona asanlıqla lazım olan formanı vermək olar və heç bir şübhə yoxdur ki, iynə başlığında onun üzərinə bütün Britaniya ensiklopediyasını yazmaq üçün kifayət qədər yer tapılar".

Sonra Feynman bildirdi ki, insanlar tərəfindən yığılmış bütün informasiyanı kiçik bir kitabçada yerləşdirmək olar. Həm də bunu informasiyanı kodlaşdırmadan, sadəcə olaraq başlanğıc materialı (bütün şəkil, reproduksiya və s. ilə birlikdə) ölçücə kiçiltməklə etmək olar. Sonra alim qeyd etdi ki, əgər informasiyanı maddə (iynə başlığı) səthində proporsional kiçildilmiş nüsxə şəklində deyil, maddə daxilində kodlaşdırılmış şəkildə saxlasaq, nəticələr daha heyranedici olacaqdır. Təsəvvür edək ki, informasiya nöqtə və tire işarələrilə kodlaşdırılıb, nöqtə hər hansı bir metal atomu, tire isə başqa bir metal atomu şəklində təsvir olunub. Əgər bu cür hər bir informasiya hissəsini 5x5x5 ölçülü atomlardan ibarət kubun daxilində yerləşdirsək, asanlıqla hesablamaq olar ki, "insanların bütün tarix boyu kitablarda yığdıqları informasiyalar gözlə güclə seçilən toz parçasına bənzər 0,1 mm-dən bir qədər kiçik kubun içinə yerləşdirilə bilər".

Məruzəsinin sonunda Feynman dedi: "Mənim bildiyimə görə fiziki prinsiplər, əşyalarla atom səviyyəsində manipulyasiyalar etmək üçün heç bir qadağalar qoymur. Bu cür manipulyasiyalar heç bir qanunları pozmur və prinsip etibarı ilə mümkündür. Lakin praktikada bu hələ edilməyib, çünki biz həddən artıq böyükük".

Kubu skaner tunel mikroskopundan istifadə edərək quraq. Fərz edək ki, hər bir atomun yerdəyişməsinə bir saniyə vaxt lazımdır, onda bütün yığıma təqribən 30 milyon il lazım olardı. Bu cür texnologiyanı, çətin ki, tətbiq oluna biləcək hesab etmək olar. İki STM bu işi iki dəfə tez yerinə yetirə bilərdi. On dənə STM-i isə, təəssüf ki, eyni zamanda tətbiq etmək mümkün deyildir – onlar yığılan kubun ətrafında yerləşdirilə bilməzdi!

Kubu ağlabatan zaman ərzində (məsələn, bir neçə saata, dəqiqəyə və ya saniyəyə) yığmaq üçün ya atomların yerdəyişmə sürətini artırmaq, ya da eyni zamanda işləyən "yığıcıların" (universal nanomaşınlar və ya özü-özünü təkmilləşdirən mikrorobotlar) sayını artırmaq lazımdır. Qeyd edək ki, onlar o qədər miniatür ölçülüdürlər ki, iş zamanı bir-birinə mane olmurlar.

Atomların yerdəyişməsini neçə dəfə tezləşdirmək lazımdır ki, bir "yığıcı" bütün işi 10 saniyəyə yerinə yetirsin? Bu halda o, bir saniyəyə atomların ümumi sayının onda bir hissəsinin yerdəyişməsini təmin etməlidir. Tutaq ki, hər bir atomu 10 mkm məsafəyə hərəkət etdirmək lazımdır, onda 1 saniyəyə "yığıcı" hər biri 10 mkm olan 10^{14} yerdəyişmə yerinə yetirəcək, yəni ümumi yerdəyişmə 10^{15} mkm $=10^{9}$ m $=10^{6}$ km >300 000 km təşkil edəcək.

Beləliklə, 1 saniyə ərzində "yığıcının" hər bir atom üçün yerinə yetirdiyi kiçik yerdəyişmələr ümumilikdə 1 milyon kilometr məsafə təşkil edər. Bunu yerinə yetirmək üçün o, müəyyən anlarda işıq sürətindən bir neçə dəfə artıq sürətlə hərəkət etməlidir, bu isə nisbilik prinsipinə ziddir.

Nanotexnologiyalar prinsipi üzrə ölçüləri millimetr və ya metr olan qurğuları yaradarkən bu çətinliklər bir qədər də dərinləşir. Beləliklə, mikronlardan millimetr və metrə keçid zamanı, atomların yerdəyişməsinə lazım olan məsafə uyğun olaraq min və milyon dəfə, atomların sayı isə 10^{9} və 10^{18} dəfə artacaqdır.

Deməli, nanotexnologiyalarda müəyyən məhsulun hazırlanması yalnız o zaman mümkün olar ki, "yığılma məntəqələri" yaradılan qurğunun bütün səthi (həcmi) üzrə paylansın və paralel işləsinlər.

Bu cür paralel işi ifadə etmək üçün qarışqaların dağılmış yuvasını və yuvanı bərpa etməyə çalışan qarışqaları təsəvvür edək. Qarışqalardan ibarət izdiham bir-birinə mane olmadan işləyir. İxtiyari qarışqanın ölçüsü onun yuvasının ölçülərinə nəzərən çox kiçikdir. Bəzi qarışqalar nisbətən azad işləsələr də, onların ümumilikdə işi vahid məqsədə tabe olur.

NANOTEXNOLOGİYALAR HƏYATA KEÇİRİLƏ BİLƏRMİ

Kubun yığılmasına aid misal kədərli fikirlərə gətirir. Yəni doğrudandamı nisbilik prinsipi istirahət günlərində ən yaxın ulduzlara uçuşun mümkün olmadığını, enerjinin saxlanılması qanunu isə daimi mühərrikin yaradılmasının mümkün olmadığını sübut etdiyi kimi, əsas fiziki prinsiplər və qanunlar da nanotexnologiyaların praktik olaraq həyata keçməyəcəyini sübut edir?

Nanotexnologiyaların nəzəriyyəçilərindən biri olan amerikalı alim Erik Dreksler belə bir misal göstərir. Əgər insanlar nanotexnologiyalara malik



Erik Dreksler.



olsaydılar, həkimlər bir neçə mikron ölçülü torbası, yapışa bilən pəncələri və xortumu olan qurğu yarada bilərdilər. Bu qurğunu insanın qanına daxil edərdilər və qurğu qandakı mikrobları tapıb, onlara yapışardı və xortumu vasitəsilə torbasında saxladığı antibiotikləri mikroba yeridərdi. Bir çox xəstəlikləri bu qayda ilə müalicə etmək asan olardı.

Lakin haradan məlumdur ki, nanotexnologiyalar daimi mühərrikdən fərqli olaraq mümkün ola bilər? "Cırtdan" qurğuların yaradılması realdırmı, bu hansısa fiziki qanunlara zidd getmirmi? Bu suala insan qanının bir damlasına mikroskop altında baxmaqla cavab verək. Qanda üzən və mikroblara qarşı mübarizə aparan, mahiyyət etibarı ilə bizim "qurğularla" eyni olan anticisimlər adlanan ünsürlər var. Onlar orqanizmdə genetik kod (dəqiq plan) üzrə addım-addım, molekul-molekul əmələ gəlirlər. Yer kürəsindəki bütün canlılar, bakteriya və sadə göbələkdən tutmuş insana və sekvoyyaya qədər amin turşuları və zülalların kiçik atom qrupları ilə aparılan manipulyasiya prosesslərinin nəticəsində yaranmışdır. Yəni müəyyən mənada hesab etmək olar ki, nanotexnologiyalar canlı təbiətdə artıq işləyirlər. Hal-hazırki zamanda nanotexnolgiyalar prinsipində yüzlərlə və hətta minlərlə atomlardan ibarət konstruksiyalar yaradılıb. Lakin onlar arasında mürəkkəbliyinə görə canlı hüceyrə ilə muqayisə oluna biləcək bir dənə də konstruksiya yoxdur. Lakin həyatın və bioloji proseslərin mövcudluğu nanotexnologiyaların praktik tətbiqinin mümkünlüyünü isbat edir.

İxtiyari özü-özünü yaradan obyektlər, istər insan bədənindəki bakteriyalar, nohurdakı yosunlar, şumlanmış torpaq sahəsindəki alaq otları, xəmirdəki maya və ya kompüter şəbəkəsindəki virus tez çoxalaraq ətraf mühiti insan üçün arzuolunmaz, hətta faciəli olan formaya çevirir. Bu cür faciələrdən biri "Sıyıq küpəsi" adlı məşhur alman nağılında təsvir olunub. Bu nağılda küpə o qədər sıyıq bişirir ki, axırda bütün şəhəri sıyıqla doldurur.

Lakin gerçəklik ixtiyari nağıldan daha qorxulu ola bilər. Nanotexnologiyalarla işləyən alimlər artıq indidən özü-özünü yaradan (və ya hətta özü-özünü mükəmməlləşdirən) mikrorobotların nəzarət altından çıxması təhlükəsi haqqında fikirləşirlər. Hipotetik faciə "boz selik" – torpağın, suyun, havanın mikroskopik robotlarla emalının nəzarətdən çıxması prosesi adını almışdı ki, bunun nəticəsində Yer kürəsi həddindən çox "quraşdırıcılar" kütləsilə örtülmüş olur.

Bu proses bir neçə gün ərzində həyata keçirilə bilər. Dalğa ilə dənizin qumlu sahilinə atılmış 10 mkm ölçülü özü-özünü yaradan robot təsəvvür edək. Tutaq ki, o, silisium, oksigen, hidrogen, azot və karbondan və duzları dəniz suyunda həll olunmuş bəzi metallardan təşkil olunub. Günəş şüaları altına düşdükdə mikrorobot özünə oxşar robotlar yaratmağa başlayır (yaxınlıqda lazımi atomların ehtiyatları olmaq şərti daxilində). Əgər "klonlaşma" prosesi yarım saat davam edərsə, onda birinci günün sonuna çimərlikdə ümumi sayı təqribən milyon ədəd və ümumi həcmi 1 mm^3 olan 20 nəsil robot əmələ gələcək. İkinci günün sonuna onların həcmi – 1 dm^3, üçüncü günün sonuna – 1000 m^3 və dördüncü günün sonuna, əgər qum kifayət

edərsə – 1 km^3 təşkil edəcək. Bütün çimərlik "boz seliyə" çevriləcək.

Nanotexnologiyaların nəzəriyyəçiləri artıq indidən "boz selik" faciəsinin qarşısını almağa imkan verən mikrorobot – "quraşdırıcının" quruluş prinsiplərini hazırlayırlar. Bu prinsiplər yazıçı-fantast A.Azimovun öz əsərində uydurduğu məşhur robototexnika qanunlarına bənzəyir. Lakin bu prinsiplər yazıçılar tərəfindən uydurulmayıb, alimlər tərəfindən praktik məqsədlər üçün verilib.

Prinsip 1. "Quraşdırıcı" yalnız xaricdən verilən komanda əsasında özünə oxşarların istehsalına başlaya bilər.

Prinsip 2. Enerjinin ayrılması ilə müşayət olunan qurma prosesslərini aparmaq qadağan olunur.

Prinsip 3. Təkrar istehsalda təbiətdə rast gəlinməyən maddələrdən istifadə edilməlidir.

CƏSARƏTLİ PROQNOZLAR

Nanotexnologiyaların tətbiqi üçün artıq çoxlu maraqlı üsullar tapılıb və proqnozların sayı günbəgün artır. Sabahkı günün texniki qurğularının yalnız üç təsvirini verək.

Yüzlərlə qiqahers tezliklərlə işləyə bilən, Bebbicin kompüterinə bənzər mexaniki nanokompüter. Bu kompüter hər biri bir neçə min atomdan yaradılmış kiçik detallardan ibatər olacaq.

Lazımlı yüklərin kosmosa qaldırılmasını təmin edən lifti olan qüllələri tikməyə imkan verən çox möhkəm və özü-özünü təmir edən materiallar. Belə qüllənin divarlarında birləşdirilmiş karbon borularla "təmirçilər" sürünərək hərəkət etməli, zədələri təyin edərək, aradan qaldırmalıdırlar.



Ayzek Azimov.

Oksigeni ağ ciyərlərdən orqanlara və karbon qazını əksinə nəql edə bilən, insan qanında üzən mikroskopik "sualtı qayıq". Əgər bu cür süni qan cisimlərindən yarım stəkanı insanın qanına vurulsa, onda insan havasız bir neçə saat qala bilər.

Nanotexnologiyaları kompüterlərsiz, kommunikasiyalarsız, proqramlaşdırmasız və informatikanın digər elementləri olmadan təsəvvür etmək mümkün deyildir. Robot-"quraşdırıcılar" informasiyanı xaricdən alıb emal edərək öz aralarında informasiya mübadiləsi edəcəklər. Mikrorobotların ötürücülərinin gücü və qəbuledicilərinin həssaslığı yalnız qısa məsafələrdə rabitəyə yarayacaq, buna görə də robotlar yəqin ki, kommunikasiya şəbəkələri yaradaraq informasiyanı retranslyasiya edəcəklər. Mikrorobotları və onlar üçün proqramları ən müasir texnikada yaratmaq tələb olunacaq. Əvvəlcə bu adi elektron kompüterlər, sonralar isə nanokompüterlər olacaq.

İNDİKİ NƏSİL NANOTEXNOLOGİYALARIN TƏTBİQİNİN NƏTİCƏLƏRİNİ GÖRƏCƏKMİ

Nanotexnologiyaların praktik tətbiqi uzaqda deyil. Artıq bu gün biznesmen və dövlət xadimlərinə nanotexnologiyalara qoyula biləcək



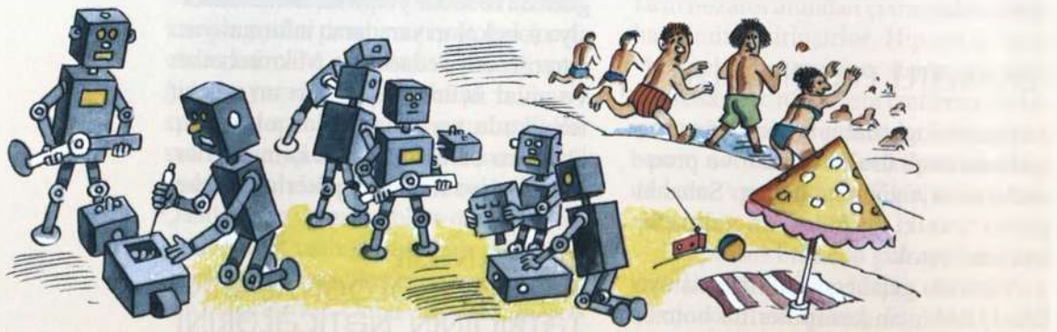
NANOROBOTLAR

Nanotexnologiya ideyalarının sistematik təhlili XX əsrin 80-ci illərində E.Drekslerin "Yaradan maşınlar" adlı kitabında dərc olunmuşdu. Əsas ideya verilmiş proqram üzrə yeni konstruksiyalar, həmçinin özlərini yığa bilən xüsusi robot – "quraşdırıcılarının" işlənib, kütləvi istehsala buraxılmasında idi. İnsan mürəkkəb alətlər yaratmaq üçün əvvəlcə daha sadələrindən istifadə edir. Sonra isə onları təkmilləşdirərək avtomobillər, təyyarələr, kompüterlər, televizorlar və digər lazımlı şeylər istehsal etməyə imkan verən texnologiyalar yaradır. Nanotexnologiyalarda da hələ ki, bir neçə nəsil universal nanoalətlər yaratmaq lazım gələcək ki, sonradan onların köməyilə nəsə mükəmməl bir şey istehsal etmək mümkün olsun.

Dreksler hesab edir ki, nanoalətlərin birinci nəslini canlı təbiətin konstruksiyalarından təkrarlama aparmaqla alacaqlar: "...çevik, proqramlaşdırıla bilən zülal maşını böyük molekulu (iş obyektini) tutacaq, bu zaman kiçik molekul həqiqi yerinin qarşısında yerləşdiriləcək. O, molekulları maya kimi bir yerə bağlayır. Molekulu molekul dalınca yığdığı hissəyə bağlamaqla, maşın böyük və daha böyük struktur yaradır. Bununla yanaşı atomların hansı qaydada düzülməsi ciddi nəzarətdə saxlanılır. Bu kimyaçıların malik olmadığı, ən vacib bacarıqdır. Ribosomlar kimi bu cür nanomaşınlar molekulyar lentlərin idarəsi altında işləyə bilərlər".

Sadə zülal maşınlarının köməyilə mürəkkəblərini, sonra isə daha mürəkkəblərini qurduqdan sonra birinci universal "quraşdırıcılar" yaratmaq mümkün olacaqdır.

"...Bu təkcə zülaldan deyil, daha böyük bir şeydən hazırlanan nanomaşınların ikinci nəslidir. Bəzi yeni nanomaşınlar molekulyar strukturların qurulması üçün mükəmməlləşdirilmiş qurğular kimi istifadə edilə bilər. Turşuya və ya vakuuma, dondurulmaya və ya qızdırılmaya davamlı, istifadə məqsədindən asılı olaraq ikinci nəsil fermentəoxşar maşınlar alət kimi, demək olar ki, hər bir kimyəvi molekuldan istifadə edəcəklər. Onlar praktiki olaraq ixtiyari dayanıqlı struktur almaq üçün atomları birləşdirə biləcəklər. "Quraşdırıcılar" bizə atomları, demək olar ki, ixtiyari ağlabatan formada yerləşdirməyə imkan verdiyindən, biz təbiət qanunlarına zidd olmayan, demək olar ki, hər şey qura biləcəyik. Xüsusi halda yeni "quraşdırıcılar" da bu siyahiya aiddir".



investisiyalar barəsində tövsiyyələr olan və 4 min dollara satılan 500 səhifəlik hesabat mövcuddur. Bu hesabata görə nanotexnologiyaların birinci sənaye tətbiqi artıq XXI əsrin 10-cu illərində baş verəcək. Daha on ildən sonra isə nanotexnologiyalar dünya iqtisadiyyatında böyük bir sektor yaradacaqdır.

HESABLAMA MAŞINLARININ GƏLƏCƏYİ

EHM "abbreviaturası" elektron hesablama maşını kimi açılır, kompüter isə ingilis sözü olub "hesablamaq" anlamını verən "to compute" sözündən yaranıb. Hesablama məsələlərinin həlli üçün ixtisaslaşmiş EHM-lər hazırlanırdı. Bunlar arasında uzun illər ən uğurlusu Cray Research amerika kompaniyasının buraxdığı Cray superkompüterləri idilər.

Birinci Cray-1 superkompüteri 1976-cı ildə ABŞ-ın Energetika Nazirliyinin Los-Alamos laboratoriyasında (atom silahına cavabdeh olan) quraşdırılmışdı. Bu superkompüter vergi ödəyicilərinə 8,8 milyon dollara başa gəlmişdi. O, 64-mərtəbəli sürüşkən vergüllü ədədlərlə saniyədə 160 milyon əməliyyat yerinə yetirə bilirdi, həm də yaddaşında milyon dənə belə ədəd saxlaya bilirdi. Kompüterdə inteqral sxemlər xüsusi texnologiya üzrə (bahalı qallium arseniddən istifadə edilməklə) hazırlanırdı. Cray özünün məhsuldarlıq zirvəsinə ayrı-ayrı ədədlərlə deyil, ədədlər yığını – vektorlarla əməliyyatlar aparmaqla çatmışdı. Bu kompüterdə bütün proqram təminatı, hətta proqramlaşdırma dili də hesablama məsələlərində rekord göstəricilər əldə etmək üçün modifikasiyaya uğramışdı.

1976-cı ildə 160 Mflops (sürüşkən vergüllü milyon əməliyyat) məhsuldarlıq fenomenal texniki göstərici hesab olunurdu. Lakin qiymətin məhsuldarlığa nisbəti baxımından Cray-1 o qədər də sərfəli deyildi. Saniyədə 1 milyon əməliyyat məhsuldarlığı Cray-1 alıcılarına 55 min dollara başa gəlirdi. Cray maşınları dövründə 1 milyon əməliyyat yerinə yetirən minikompüterlər də təxminən həmin qiymətə idi. Məhz bu iqtisadi şərait kompüter təkamülünün dinozavrları olan – ixtsaslaşdırılmış superkompüterlərin gələcək məhvini müəyyənləşdirdi.

Bir neçə ildən sonra fərdi kompüterlər yarandı. Əvvəlcə hesablama məsələlərilə məşğul olan riyaziyyatçı və mühəndislərdən heç biri böyüklər üçün bu yeni oyuncağı ciddi saymırdı. 1981-ci ildə hesablama gücünə görə Cray-1-i IBM kompaniyasının birinci fərdi kompüter modellərində istifadə edilmiş Intel mikroprosessorları ilə müqayisə etmək heç kimin ağlına belə gəlməzdi. Lakin superkompüterlərin tirajı onlarla və yüzlərlə ölçülürdüsə, fərdi EHM-in (və onlar üçün mikroprosessorların) tirajı milyonlarla və on milyonlarla ölçülürdü.

Qallium arsenid əsasında xüsusi inteqral sxemləri, silisium oksidlər əsasında buraxılan ənənəvi mikrosxemlərə nəzərən daha az buraxılırdılar. Buna görə də Cray superkompüterlərinin mikroprosessorlarının məhsuldarlığının artımı Mur qanunu ilə müəyyən olunmuş artımdan çox aşağı olub, ildə təqribən 10% təşkil edirdi. Birinci fərdi EHM-in yarandığı dövrdə, xüsusilə 1980-ci ildə onların məhsuldarlığı superEHM-in bir prosesorunun məhsuldarlığından aşağı olsa da, inkişaf tempi hiss olunacaq dərəcədə yüksək idi və 1995-ci ildə fərdi EHM-in mikroprosessorunun etalon məsələ üçün təyin olunmuş məhsuldarlığı superEHM-in mikroprosessorunun məhsuldarlığına bərabər olmuşdu.

Cray kompaniyası bu meyli hiss edərək 1993-cü ildən başlayaraq kütləvi masaüstü EHM-in adi silisiumlu mikroprosessorlarından istifadə edərək superEHM-lər istehsal etməyə başladı. 1998-ci il üçün dünyada istifadə olunan 500 superEHM-dən 65% kompüterlərdə kütləvi mikroprosessorlardan, daha 25%-də xüsusi silisiumlu mikroprosessorlardan və yalnız 10%-də veteranlardan – qallium arsenid əsasında qurulan mikroprosessorlardan istifadə edilirdi.

ASCI (Accelerated Strategic Computing Initiative – "sürətləndirilmiş strateji



Qordon Mur (solda) və Seymur Krey.

Kütləvi tələbat yaranan əşyaları, çox böyük sayda buraxmağa başlayırlar. Nəticədə onlar tez ucuzlaşır, onların keyfiyyəti isə yaxşılaşır. Müasir mədəniyyətin ən kütləvi texniki məhsullarından biri – fərdi kompüterlər üçün universal mikroprosessordur. Bu cür mikroprosessor bir neçə yüz dollaradır və ildə on milyonlarla tirajla buraxılır.

Cray Research firmasının yaradıcısı, onun baş ideoloqu və konstruktoru Seymur Krey dahi mühəndis və biznesmen idi (1996-cı ildə 71 yaşında avtomobil qəzasında həlak olmuşdur).

MUR QANUNU

1965-ci ilin aprelində müasir mikroelektronikanın əsasını qoyanlardan biri Qordon Mur "Elektronika" jurnalında çox maraqlı bir proqnoz dərc etdirmişdi. O, silisium oksidlər əsasındakı inteqral mikrosxemlərin üç nəslinin məhsuldarlığını müqayisə edərək aşağıdakı nəticəyə gəldi: inteqral mikrosxemlərin mürəkkəbliyi və məhsuldarlığı hər 18 ayda iki dəfə artacaqdır. Mur xüsusilə, iddia edirdi ki, 1975-ci ildə 65 min elementdən ibarət mikrosxemlər meydana gələcək.



Qordon Mur.

On il sonra məqalələrindən birində o, qeyd edirdi ki, bu proqnoz onun özünün də gözləmədiyi dəqiqliklə təsdiqləndi. Elə bununla da alim fərziyyələrinin sonrakı illərdə də təsdiqlənəcəyini zənn etmiş və onların texniki əsaslandırmalarını vermişdir. Məqalə çıxdıqdan sonra Murun pərəstişkarları onun proqnozunu Mur qanunu adlandırmışdılar. Bu qanun silisium əsaslı bütün inteqral sxemlərə aid olsa da, mikroprosessorlara aid olması ilə daha çox məşhurlaşdı. Bu hal belə ifadə olunur: mikroprosessorun məhsuldarlığı hər 18 aydan bir iki dəfə artır. Doğrudan da, Mur qanunu uzun illər boyunca təsdiq olunurdu və mütəxəssislər hesab edirlər ki, bu qanun ən azı 2012-ci ilədək aktual olaraq qalacaqdır.

Qordon Mur 1975-ci ildə və sonralar da öz qanununun texniki əsaslandırılmasını vermişdir. Lakin bu izahatları tam əsaslı hesab etmək olmazdı. İqtisadçılar və sosioloqlar Mur qanununu təbii olaraq öz-özünə yerinə yetirilən qanunlara, yəni hamının bildiyi və hamının inandığı qanunlara aid etmişdilər.

İqtisadçılar Mur qanununun uzun illər boyu yerinə yetirilməsini belə izah edirlər. Fərdi kompüterin mikroprosessorunun məhsuldarlığı öz-özlüyündə istifadəçi üçün bir o qədər də vacib deyil, onu daha çox kompüterin ümumilikdə nəyə qadir olduğu maraqlandırır. Mikroprosessordan başqa fərdi kompüterdə bir çox digər komponentlər var. Əgər onların məhsuldarlığı mikroprosessorun məhsuldarlığından azdırsa, onda axırıncının artıq məhsuldarlığı "əbəs yerə itir", ondan tam şəkildə istifadə etmək olmur. Əgər digər komponentlərin məhsuldarlığı mikroprosessorun məhsuldarlığından böyük olarsa, komponentlərin "artıq" məhsuldarlığı da "itir", çünki mikroprosessor tərəfindən istifadə edilə bilmir.

Fərz edək ki, hər hansı bir komponentin hər hansı bir istehsalçısı onun yeni versiyasını işləmək istəyir. Fərz edək ki, o, bu işi il yarım ərzində başa çatdırmağı planlaşdırır. İşin əvvəlində həll olunmalıdır ki, buraxılış anı üçün yeni versiyanın məhsuldarlığı nə qədər olacaqdır. Bu çox məsuliyyətli həlldir: əgər məhsuldarlıq il yarımdan sonra bazara çıxarılacaq prosessor üçün çox kiçik olarsa, komponentləri heç kəs almayacaq və istehsalçı ziyana düşəcəkdir. Əgər həddindən artıq məhsuldarlıq planlaşdırılsa, onda komponentlər prosessorların gələcəyini daha düzgün təyin etmiş rəqiblərdəkinə nisbətən daha baha olacaqlar və məhsullar yenə də alınmayacaq. Lakin istehsalçı bilir ki, Mur qanunu artıq çox illərdi ki, işləyir və o, bu qanundan cəsarətlə istifadə edir. O, fərz edir ki, 18 ay ərzində istifadəçilərin ala biləcəyi mikroprosessorların məhsuldarlığı iki dəfə artacaqdır.

İndi isə vəziyyətə mikroprosessor istehsalçısı nöqteyi-nəzərindən baxaq: əgər Mur qanununda nəzərdə tutulduğundan daha artıq məhsuldarlıq planlaşdırılsa, onda artıq məhsuldarlıq "itəcək", çünki komponent istehsalçıları bu artıma hazır olmayacaqlar.

Əgər əksinə hərəkət etsə, onda Mur qanununa riayət edən rəqiblərinə uduzmuş olacaqdır. Buna görə də həm mikroprosessor istehsalçıları, həm də kompüter komponentləri istehsalçıları yeganə düzgün davranış xəttinə – Mur qanununa uyğun inkişaf xəttinə riayət etməlidirlər. Nə çox tez, nə də çox gec.

İndi isə kiçik, lakin vacib bir qeyd edilməlidir. Mur qanununa texniki səbəblərdən riayət etmək imkanında olmayan mikroprosessor istehsalçısı hökmən iflasa uğrayacaqdır. Bu acınacaqlı qismət, həmin qanuna maliyyə imkanları üzündən riayət edə bilməyənləri də gözləyir.

Məhsuldarlığın nəzəri olaraq maksimal artımı, mikroprosessorun yeni modellərinin yaradılmasına istehsalçı tərəfindən qoyula biləcək vəsaitlərdən asılıdır. Mikroprosessorun tirajı və satış gəlirləri böyük olduqca, yeni araşdırmalar üçün bir o qədər artıq vəsaitlər ayırmaq mümkün olacaqdır. Və əksinə, mikroprosessorun kiçik tirajlarında və uyğun olaraq kiçik gəlirlər alındıqda, istehsalçı ciddi yatırımlar etmək iqtidarında olmur. Nəticədə, bu tip mikroprosessorun məhsuldarlığının artım tempi çox da böyük olmur.

İNTEQRAL TEXNOLOGİYALARIN UĞURLARI

Kristalın keyfiyyətini aşağıdakı üç parametrlə xarakterizə etmək olar:

- Layihə norması – texnoloji ölçü mikrometrlə (metrin milyonda biri) ölçülür; bu göstərici kristaldakı elementlərin yerləşmə sıxlığını təyin edir;
- Kristaldakı tranzistorların (və ya ventillərin) sayı kristalın məntiqi mürəkkəbliyini təyin edir;
- Takt tezliyi meqahers və gigaherslərlə ölçülür və kristalın işləmə sürətini, yəni faktiki olaraq saniyə ərzində ventilin maksimal mümkün keçirmələrinin sayını təyin edir;

1999-cu il üçün minimal layihə normaları 0,18 mk-ya, inteqral sxem kristalının maksimal ölçüləri – 25x35 mm, çipdəki tranzistorların maksimal sayı – 21 milyon ədəd, takt tezliyi 1,2 Ghers, metallaşma səviyyələrinin maksimal sayı – 7, qidalanmanın minimal gərginliyi – 1,5 V, təbəqənin maksimal diametri – 300 mm-ə bərabər idi.

İl	1989	1992	1995	1998	2001	2004	2007
Texnoloji ölçülər (mkm)	0,7	0,5	0,35	0,25	0,13	0,10	0,08
Kristaldakı tranzistorların sayı (milyon)	2,5	5	10	21	46	110	500
İşçi tezlik (Mhers)	100	175	300	600	1500	3500	7000

Yarımkeçiricilər istehsalının inkişafının axırıncı 30 ili belə xarakterizə olunur:

- Kristaldakı tranzistorların sayının hər 3 ildə 4 dəfə artımı;
- Təbəqələrin diametri hər 15 ildə 2 dəfə artır;
- Kristalın sahəsi hər 6 ildə 2,3 dəfə artır;
- İstehsalın dəyəri hər 3 ildə 2 dəfə artır və minimal layihə normaları hər 5 ildə 2 dəfə azalır.

Lakin bütün oriyentirlər, adətən düzəlişlər tələb edir. Belə ki, cədvəldə göstərilmiş 90-cı illərin proqnozu indiki real vəziyyətə (məsələn, mikroprosessorların tezliyi üzrə) uyğun gəlmir.

İnteqral texnologiyanın mükəmməllik dərəcəsi onun köməyilə hansı kristalları hazırlamaq mümkün olması ilə təyin edilir. Hal-hazırda XXI əsrin əvvəlində texnoloji ölçüləri 0,15 mkm, həcmi 1 Gbit olan yadda saxlama qurğuları buraxılır. Bizi KMOY ÇBİS ailəsinə aid olan yeni sənaye məhsulu – gigaçip gözləyir. Onun texnoloji ölçüləri 0,1 mkm-dir, 16 Gbit həcmli yaddaşa malikdir və onlar milyard tranzistordan ibarətdir.



Keçən əsrin sonunda yarımkeçiricilər sənayesi üçün təşkilin şaquli modeli tipik idi. Bütün texnoloji əməliyyatları özündə saxlayan tam istehsal dövrü bir müəssisədə gedirdi. Bu cür istehsalların yüksək qiyməti daha çevik üfüqi ixtisaslaşmaya keçməyi tələb edirdi. Beləliklə, inteqral sxemlərin avtomatlaşdırılmış layihələndirilməsi, fotoşablonların hazırlanması, təbəqədə inteqral sxemlərin hazırlanması, testləşdirmə və s. müstəqil istehsal kimi ayrıldılar.

Submikron layihə normaları sahəsindəki irəliləyiş, yeni məhsulların uzun ömürlülüyünə mənfi təsir göstərir: onların yaşama müddəti dayanmadan azalır. Tezliklə indiki ikiillik "yaşama dövrü" 9 ay və daha az olacaq. Bununla əlaqədar olaraq işlənmədən son məhsulun alınmasına qədər bütün mərhələlərdə sürətin artırılması ön plana çıxır. Əgər indiki meyllər yaxın gələcəkdə də saxlanılarsa, onda yeni işləmələr iqtisadi nöqteyi-nəzərdən sərfəli olmayacaq.

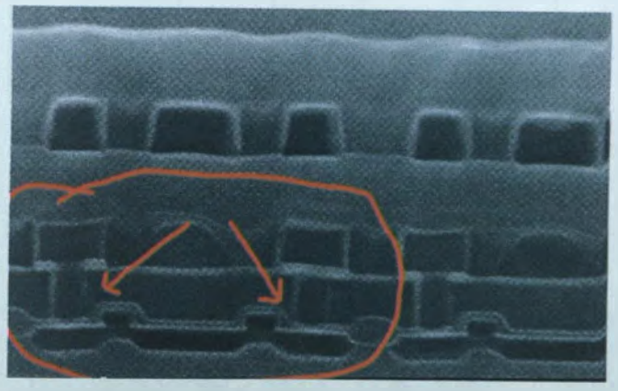
LAYİHƏ NORMALARI

Mikroprosessorlar və onları istehsal edən müəssisələr üçün əsas göstəricilərdən biri – *layihə normalarıdır*. Texnoloji dövrün hər bir əməliyyatı istehsala təsir göstərir, lakin istehsal olunan inteqral sxemlərə əsas məhdudiyyətləri fotolitoqrafiya qoyur. Burada layihə normaları, inteqral sxemin fotoşablonunun şəklinin elementləri arasındakı icazə verilən minimal ölçülərdir. Yerdə qalan əməliyyatlar bu "şəkli" korlamamalıdırlar. Standarta əsasən fotoşablonla təbəqə tutuşdurulduqda fərq layihə normasının yarısından artıq olmamalıdır.

İnteqral sxemin əsas ölçüsü – tranzistorun başlangıcı ilə axarı arasındakı məsafədir. Fotoşəklə baxsaq bu məsafənin təqribən bağlayıcının eninə bərabər olduğunu görərik. Bağlayıcı başlanğıc ilə axar arasındakı kəsiyi örtür və inteqral sxemin tezliyini və enerji sərfini təyin edir.

Vacibliyinə görə ikinci ölçü – başlanğıc və ya axarın kontakt meydançasının ölçüsüdür. O, proyeksiyada, bir növ quyu dibinə bənzər kvadrat əmələ gətirir, sonradan ora volframla doldurulur və xarici birləşmələri həyata keçirmək üçün metalın birinci səviyyəsinə çıxır.

Hər iki ölçü təqribən bərabərdir və onları layihə normaları adlandırırlar.



İki lay metallaşması olan kristalın kəsiyi. Aşağıdakı səviyyədəki qara lay, iki tranzistorun kombinə olunmuş strukturunun başlangıc və axarları ilə yaradılıb (soldan sağa doğru): başlanğıc 1 – axar 1, onların üstündə bağlayıcı, daha sonra başlanğıc 2 (axar 1 ilə üst-üstə düşür) – axar 2 və onların üzərində daha bir bağlayıcı. Başlanğıc 1 və axar 2-də metalın birinci layına çıxan quyular yerləşir. Daha yuxarıda metallaşmanın ikinci layı yerlişir.

kompüter təşəbbüsü") adlı ABŞ-ın Energetika Nazirliyi üçün superEHM-in inkişafı üzrə amerika proqramının yaradıcıları superEHM-də masaüstü EHM-lər üçün mikroprosessorlardan istifadə etdilər. Proqram müvəffəqiyyətlə yerinə yetirilirdi və 2000-ci il üçün dünyada ən məhsuldar EHM olan ASCI White (IBM kompaniyasının məhsulu) işə salındı. Onun məhsuldarlığı 7 Teraflops (sürüşkən vergüllü trilyon əməliyyat) idi. Bu superEHM hər biri təqribən 1 Gflops (sürüşkən vergüllü milyard əməliyyat) məhsuldarlıqlı 10 min mikroprosessordan ibarət idi.

Yaponiya hesablama məsələləri üçün mikroprosessor üzərində iş aparırdı.

2000-ci ildə belə mikroprosessor etalon məsələdə təqribən 10 Gflops məhsuldarlıq göstərdi. 2002-ci ildə isə fövqəlada xəbər gəldi: NEC kompaniyasının Earth Simulator yapon superEHM-i testdə təqribən 35 Gflops məhsuldarlıq göstərərək superEHM-lər siyahısında birinci yeri tutmuşdu. Bununla o, əvvəlki rekordçunu (ASCI White) beş dəfədən də artıq üstünlüklə "udmuşdu".

Earth Simulator superEHM-i peyklərdən daxil olan verilənlər əsasında iqlim dəyişmələrinin modelləşdirilməsi üçün istifadə olunurdu. Kompüterin yüksək məhsuldarlığı, xüsusi yaradılmış təqribən 8 Gflops məhsuldarlıqlı NEC Vector mikroprosessorunun tətbiq edilməsi hesabına alınmışdı. Bununla da vektor mikroprosessorları yenə də ümumi təyinatlı mikroprosessorlar üzərində qələbə qazanmışdı. Earth Simulator-da 5 mindən artıq prosessordan istifadə edilib, onlar üç tennis kortunun sahəsinə bərabər (50 m×65 m) sahə tuturlar və burada ümumi uzunluğu 2900 km olan kabellərdən istifadə edilib.

2004-cü ildə IBM kompaniyası (ABŞ hökümətinin sifarişilə) daha belə bir "div" hazırlamağı planlaşdırdı. Məhsuldarlığı 100 Gflops olan ixtisaslaşdırılmış Blue Gene superEHM-i biologigenetik araşdırmalar üçün nəzərdə tutulurdu. Bu elektron-hesablama maşını 2002-ci il buraxılışı olan kütləvi fərdi kompüterə nisbətən milyon dəfə məhsuldar oldu.

KVANT KOMPÜTERLƏRİ

Məlumdur ki, hesablama maşınları tək rəqəmsal deyil, həm də analoq tipli olurlar. Analoq maşınlarda hesablamalar hər hansı bir fiziki proseslə birbaşa modelləşdirilir. Məsələn, kondensator, müqavimət və induktivlik çarxlarından ibarət zəncir qurub hansısa müqavimətdə gərginliyin zamandan asılılığını ölçmək olar. Beləliklə, differensial tənliyin həllini tapmaq üçün gərginliyin bu tənliyə tabe olduğu zənciri tapmaq kifayətdir. Rəqəmsal maşınlarda həmin differensial tənlik tamam başqa cür həll edilir: ədədi qiymətlər bitlər yığını şəklində saxlanılır və onlarla hesab əməliyyatları yerinə yetirilir.

Tədricən analoq maşınlar rəqəmsal maşınlar tərəfindən sıxışdırılıb çıxarıldı. Bu ona görə baş verdi ki, analoq maşınlar müəyyən sinif məsələləri həll edə bilirlər, rəqəmsal maşınlar isə daha universaldırlar. Eyni bir rəqəmsal maşında proqramları dəyişdirməklə müxtəlif hesablamalar aparmaq olur. Bu zaman sürətdəki gerilik hesablamaların dəqiqliyilə kompensasiya edilirdi. Bəzən hesab edirlər ki, məhz bu səbəbdən kompakt-disklər qramofon vallarının yerini tutdular.

Kvant kompüteri ideyası insanları yenidən analoq maşınlara qaytarır. Bu zaman iki məqamı nəzərə almaq vacibdir.

Müasir hesablamaların mürəkkəbliyi nəzəriyyəsi göstərir ki, bəzi məsələlər, onlara gözlənilmədən çoxlu sayda başqa məsələlərin aid edilməsi hesabına "universal" olurlar. Məsələn, NP-tam adlanan məsələlər var. Onlardan biri – Hamilton yolu məsələsidir. Bu məsələdə verilmiş qrafda bütün təpə nöqtələrindən yalnız bir dəfə keçən yolun olub-olmaması yoxlanılır. Hal-hazırda məsələnin heç bir sürətli həll alqoritmi məlum deyildir. Lakin isbat olunub ki, əgər belə bir alqoritm olsa idi, bir çox digər məsələlər də tez həll oluna bilərdilər. Onda qrafda yolun tapılması bacarığı ədədlərin vuruqlara ayrılmasına da kömək ola bilərdi.

Adi rəqəmsal maşında klassik mexanikanın qanunlarını kifayət qədər tez modelləşdirmək olur, kvant proseslərinin modelləşdirilməsi isə ya mümkün deyildir, ya da çox uzun hesablamaların aparılmasını tələb edir. Beləliklə, əgər kvant kompüteri hansısa universal məsələni tez həll edə bilsəydi, onda müasir rəqəmsal hesablama maşınlarının gücü çatmadığı bir çox məsələlərin həlli üçün onlardan istifadə etmək olardı.



Piter Şor.

Bu ideyanı amerikalı alim Piter Şor həyata keçirdi. O, böyük ədədlərin sadə vuruqlara tez ayrılması üçün istifadə edilə biləcək xəyali maşını təsvir etdi. Vuruqlara ayrılma məsələsinin universal olub-olmadığını demək çətindir, lakin bu məsələ öz-özlüyündə kifayət qədər vacib məsələdir. Məsələ ondadır ki, praktikada geniş istifadə edilən müxtəlif şifrləmə sistemlərində böyük ədədləri vuruqlara ayırmaq çətindir. Əgər ayrılma üsulu tapılarsa, onda mövcüd şifrləmə sistemləri yararsız olacaqlar. Buna görə də təəccüblü deyil ki, kvant



kompüterinin yaradılması ideyası bir çoxlarını narahat edirdi.

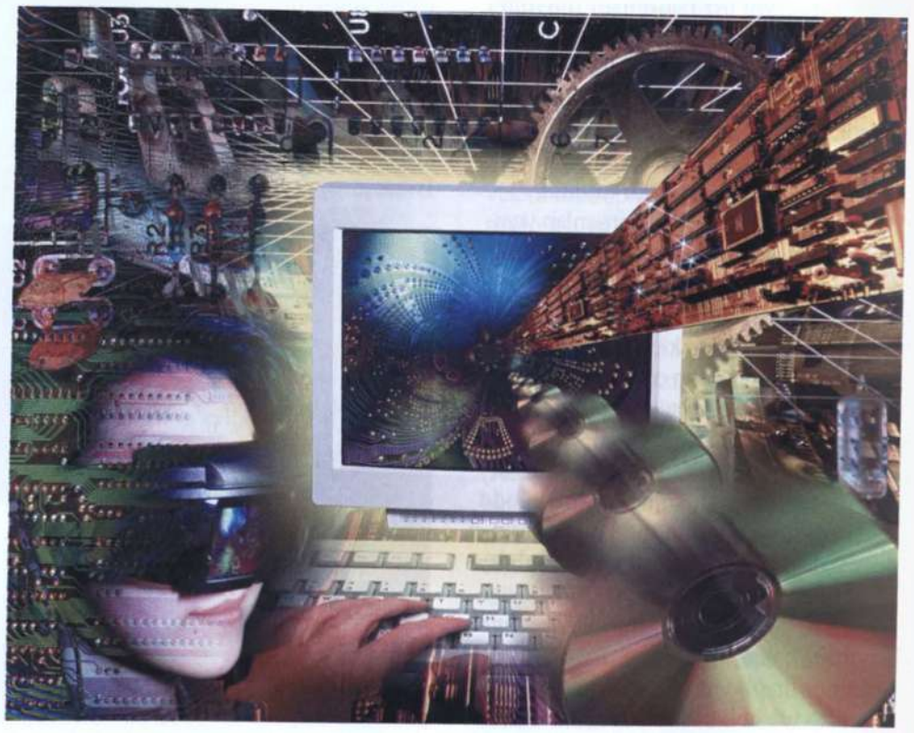
Şorun alqoritmi üçün minimum bir neçə min elementləri – kubitləri (kvant bitləri) olan maşın lazım idi. Halbuki müasir texniki eksperiment ancaq bir-iki belə elementi hazırlaya bilir. Bundan əlavə alqoritmə görə bütün kvant hesablamaları mütləq dəqiqliklə aparılmalıdır (praktikada bu mümkün deyildir). Deməli, bu cür alqoritmlərin praktik gerçəkləşməsi üçün yeni ideyalar tələb olunur: hesablamaları səhvlərə qarşı necə dayanıqlı etməli.

Kvant hesablamaları ideyası artıq həm hesablamaların mürəkkəbliyi nəzəriyyəsində, həm də fizikada (nəzəri və eksperimental) bir çox maraqlı araşdırmalara gətirib çıxarıb. Hesablamaların mürəkkəbliyi nəzəriyyəsində məsələlərin yeni sinifləri yaranmış, fizikada isə az sayda verilmiş fərdi kvant zərrəciklərindən ibarət sistemin vəziyyəti nəzərdən keçirilməyə və baxılmağa başlanılmışdır.

2010-CU İLİN KOMPÜTERİ

Əgər sizin fərdi kompüterinizin qiyməti yeni "Jiquli" maşınların qiymətindən bahadırsa, onda o, sizin həyatınızın mənasını təşkil edir.

2010-cu ildə hansı kompüterin kütləvi olacağı haqqında proqnoza iki mövqedən yanaşmaq olar. Onlardan biri texniki parametrlərə istinad edir. Prosessorun məhsuldarlığını (takt tezliyini), kompüteri buraxacaq firmanı, operativ və daimi yaddaşın həcmini, kompüterin qurulu-



şunu, monitorun tipi və xarakteristikasını, rabitə vasitələri və İnternet şəbəkəsindən istifadə edərkən mübadilə sürətini və bir çox şeyləri qabaqcadan müəyyən etmək tələb olunur.

Elektonikanın güclü inkişafı dövründə, hətta Mur qanunundan ("Mur

qanunu" əlavə oçerkinə bax) istifadə edilsə belə, gələcəkdən xəbər vermək heç də asan deyildir.

Digər yanaşma – kompüterin hələ olmayan, lakin yaxın onilliklərdə yarana biləcək məsələ və funksiyalar dairəsini təsvir etməkdir. İndi artıq EHM nəsilləri haqqında az danışırlar, belə ki, dördüncü nəsildən sonra beşincisi gəlməli idi (zarafat kimi Pentium prosessorunu bu nəslin nümayəndəsi hesab etmək olar).

Əvvəllər olduğu kimi süni intellekt sistemlərində həlli gözlənilən bir çox problemlər araşdırma səviyyəsində qalır.

Hələ də kompüterlərə insan kimi "düşünməyi" öyrətmək mümkün olmur, onlar ciddi məntiqi qayda və nəticələrə deyil, yalnız intuisiyaya əsaslanaraq tez qərar qəbul edirlər. Əgər gələcəyin EHM-ini "instinktiv məntiqlə" silahlandırmaq mümkün olarsa, onda maşınlar insan fəaliyyətinin bir çox sahələrində canlı ekspertləri əvəz etmək iqtidarında olacaqlar. Belə bir sistem tərəfindən yığılmış bilikləri digər kompüter asanlıqla əldə edib (məsələn, İnternet şəbəkəsilə) qərarların qəbul olunmasında istifadə edə bilər.

Son vaxtlarda fəal surətdə kompüterin, rabitə vasitələrinin və məişət cihazlarının vahid bir topluda birləşdirilməsi prosesi gedir. Portativlik, enerji sərfiyyatı, etibarlılıq, məhsuldarlığın həcmi elektron sənayesinin yeni məhsullarının qiymətləndirmə göstəriciləridir. Sonrakı tərəqqi bir inteqral sxemdə yerləşdirilmiş, həm mikroprosessoru və onun ətrafını, həm də proqram təminatını birləşdirən yeni sistemlərin yaradılması ilə bağlıdır.

Fərz etmək olar ki, universal kompüterlərin istehsalı getdikcə azalacaqdır və onların yerinə öz sahibinin konkret məsələlər spektrini həll edən inteqrallaşmış cihazlar: əl telefonu üstəgəl fotokamera – smartfon adlanan cihaz, web-



televizoru və s. gələcək. Müasir kompüterlər kifayət qədər mürəkkəbdirlər və istehsalçıların səylərinə baxmayaraq, bu cür sistemi işlək vəziyyətdə saxlamaq hər adamın işi deyil. Burada söhbət, ilk növbədə, hesablama texnikasının səriştəsiz istifadəçilərindən gedir. Belə insanlara universal kompüter almağın mənası yoxdur, onlara aldıqları cihazın konkret məsələlərin həlli üçün yararlı olması daha vacibdir.

Süni intellekt sistemlərinin əleyhdarları zarafat edirlər ki, süni intellekt, təbii intellekt çatışmayanda tələb olunur.



Növbəti upgrade zamanı vintlərin açılmasına vaxt itirməmək üçün sizin kompüterin qabı bu vintlərlə heç vaxt bağlanılmırsa, onda deməli, siz informasiya tərəqqisinin önündə dayanmısınız.

Şekspirin məşhur kəlməsini çevirərək belə demək olar: "Dəyişmək, ya dəyişməmək – sual budur, bu" (burada kompüter və digər texnika başa düşülür).

XX əsrin 80-ci və 90-cı illərində insanın səs əmrlərini qəbul edə bilən EHM-in yaradılması sahəsində fəal araşdırmalar aparılırdı və hətta müəyyən mənada uğurlu nəticələr də əldə olunmuşdu. Bu zaman maşın nəinki əmrlərin geniş spektrini yerinə yetirməli, həm də "eşitdiyi" mətni daxil edərək, katib kimi işləməyi bacarmalıdır. Onda ensiklopediya üçün məqalə yazmaq nə qədər də asan olardı. Daxil edilmiş nitq ilə artıq mövcüd texnologiya əsasında sintez olunmuş nitqin kombinasiyası, klavişin düyməsini sıxmaq əvəzinə kompüterlə maraqlı söhbət aparmağa imkan verərdi.

Nəhayət ki, kompüterlər səhiyyə sahəsində aparıcı yer tutacaqlar.Balaca EHM–lər insanların, məsələn qəza nəticəsində və ya sadəcə olaraq qocalıq üzündən zədələnmiş orqanlarını əvəz edə bilən mexaniki orqanları idarə edə biləcəklər. Kompüterlər, eynəklər kimi çox adi əşyalar olacaqlar, gözləri zəif görən insanlar özlərini eynəksiz necə pis hiss edirlərsə, gələcəkdə xəstələr də özlərini bu kiçik köməkçilər olmadan elə narahat hiss edəcəklər. Cib kompüteri, öz sahibini jurnallardakı məqalələr haqqında məlumatlandıracaq, dünyadakı hava durumu haqqında xəbərdar edəcək, avtoservisə zəng edərək gediş üçün biletlər sifariş edəcək, həmçinin aksiyalar alıb-sata və maliyyə hesabları üzrə əməliyyatlar apara biləcək (vergiləri vaxtlı-vaxtında ödəyə biləcək) və s.

Statistikanın göstərdiyi kimi fərdi kompüterin ömrü orta hesabla üç ili aşmır. Bu zaman kompüter iş qabiliyyətini itirmir, lakin mənəvi cəhətdən köhnəlir və avadanlıq, proqram təminatı istehsalçıları istifadəçilərə yeni-yeni EHM-lər təklif edirlər. Fərdi kompüterə qoyulmuş vəsaitlərin yarısı təqribən iki illik istifadədən sonra itir. Texnikanı dəyər-dəyməzinə satmağa adamın heyfi gəlir. Lakin, adətən, onu bütövlükdə deyil, hissə-hissə dəyişdirməklə, hər dəfə yaxşılaşdırılmış keyfiyyət almaq olar. Bu proses "upgrade" adlandırılır.

Maşında praktik olaraq bütün detalların (prosessorun, yaddaşın, disklərin) dəyişdirilməsi yeni kompüterin alınmasına nisbətən 150-250 dollara qənaət etməyə imkan verir. Şirkətdə 2 min maşın varsa, bu qənaət kifayət qədər böyük olur. Lakin burada nəzərə almaq lazımdır ki, upgrade sonsuz deyil, nəticədə bir çox təkmilləşdirmələrin sonunda alınmış kompüter, yeni yaradılmış kompüterlərdən geri qalacaqdır. Fərdi kompüterin hər bir komponenti özünün istifadə edilməsinin dəqiq sərhədlərinə malikdir. Hələlik yalnız 3,5 düymlü disketlərin disksürənləri üçün bu sərhədlər kifayət qədər genişdir. Faktiki olaraq onlar yalnız bir dəfə, 720 kbayt formatı 1,44 Mbayt ilə əvəz olunduqda dəyişdirilmişdi. Qalan hallarda disksürənlər ehtiyac olduqca bir kompüterdən digərinə keçirilə bilirdi. Ən az yaşayan hissə isə fərdi EHM-in ən bahalı hissəsi olan mikroprosessordur.





Proqram təminatı istehsalçıları EHM istifadəçilərini yeni keyfiyyət uğrunda sonsuz yarışmaya sövq edirlər. Çox vaxt növbəti yeniliklər, əməliyyat sistemləri və ya görüntülərin emalı proqramları, funksiyaların geniş yığınına sanballı heç bir yenilik gətirmirlər, həm də nəinki yeni versiyanı almağı, hətta kompüteri xeyli yeniləşdirməyi də tələb edirlər. Əməliyyat sisteminin ən müasir versiyasını aldıqda görmək olar ki, cəmisi bir il əvvəl alınmış avadanlığın bir hissəsi yeni əməliyyat sistemi tərəfindən dəstəklənmir, yəni bu avadanlıq bu əməliyyat sistemilə işləməyəcək. Burada proqram təminatı və avadanlıq istehsalçıları arasında qabaqcadan razılaşmanın olması da vacibdir.

Bundan əlavə müasir EHM-in xidmət müddətinin az olması, xüsusilə mexaniki hissələri olan kompüter komponentlərinin keyfiyyət səviyyəsinin aşağı düşməsinə səbəb olmuşdur. Bəzi məşhur firmaların müasir CD-ROM qurğularının daxilinə diqqətlə baxsaq onların nə üçün cəmi bir neçə il, bəzən isə heç zəmanət müddəti qədər də işləmədiyini asanlıqla başa düşmək olar.

İNFORMASİYANIN ÖTÜRÜLMƏSİ



İNFORMASİYANIN ÖTÜRÜLMƏSİ VƏ KOMPÜTER ŞƏBƏKƏLƏRİ

BÖYÜK MƏSAFƏYƏ İNFORMASİYANIN ÖTÜRÜLMƏSİ

Qədim zamanlardan informasiya mübadiləsinə ehtiyac olmuşdur. Düşmənin yaxınlaşmasını qulağı yerə söykəməklə bilmişlər, tonqallar yandırmaqla yenilikləri uzaq məsafədən bir-birinə ötürmüşlər. Yazı meydana çıxdıqdan sonra informasiya böyük məsafələrə poçt vasitəsilə ötürüldü. İlk zamanlar bu əlaqə fərdi xarakter daşıyırdısa, sonralar qədim Yunanıstanda, İranda, qədim Misirdə, Çində və Roma imperiyasında müntəzəm olaraq dövlət poçtu fəaliyyət göstərmişdi. Piyadalar və atlılar vasitəsilə məlumatlar estafet prinsipi əsasında ötürülürdü. Orta əsrlərdə Avropada qeyri dövlət poçtları – monastır və universitet poçtları inkişaf etməyə başladı. Onlar pullu xidmət əsasında məlumatları xüsusi adamlara da çatdırırdılar. Ticarətin inkişafı şəhərlər arasında poçt əlaqəsi olmadan mümkün deyildi.



X əsrdə Rusiyada *çapar* mövcud idi, əhali knyaz çaparlarını atlar və arabalarla təmin etməli idi. XIII əsrdə məktubları göndərmək üçün xüsusi növ xidmətdən – *çaparxana çatdırması* xidmətindən istifadə edilməyə başlandı və bu xidmət XIX əsrin ikinci yarısına kimi davam etdi. 1782-ci ildə Poçt Departamentinin yaradılması ilə çaparxana həyətləri poçt stansiyaları adlandırıldı.

Məktublaşma həm Rusiya daxilində, həm də xarici ölkələrlə aparılırdı. XVII əsrin ikinci yarısında Moskvadan Riqaya və Vilnoya olan poçt marşrutu fəaliyyət göstərməyə başladı. Bu zaman məktub 10 gündən çox yolda qalırdı.

1837-ci ildə dünya ölkələri arasında ilk dəfə Rusiyada poçtun daşınması üçün dəmir yolundan istifadə edildi. 1843-cü ildə Peterburqda, 1845-ci ildə Moskvada şəhər poçtu yaradıldı.

Poçtun tarixində romantik səhifəni XIX əsrin 60-cı illərində Rusiyanın qərbindən şərqinə dəmir yolu qurulmazdan əvvəl təşkil olunmuş "Poni-ekspress" adlı Amerika poçt xidməti yazmışdır ki, həmin vaxtlar poçtun daşınması həm təcili, həm də qorxulu idi.

Birinci olaraq müntəzəm poçt xidməti Sent-Cozef şəhərilə (Missuri ştatı) Sakramento şəhəri (Kaliforniya ştatı) arasında yaradıldı. Atlarla poçt xidməti "Poçt istənilən halda çatdırılmalıdır" şüarı altında işləyirdi. Kuryerlər poçtu 3 min km-dən çox olan məsafəyə 10 günə çatdırırdılar.



XX əsrin ikinci yarısında ABŞ-da 32 min, Böyük Britaniyada 25 min, Yaponiyada 20 min, Fransada 18,7 min poçt müəssisəsi fəaliyyət göstərirdi.

XIX əsrin başlanğıcında poçt məlumatlarını daha sürətlə daşıyan vasitələrin – paravoz və paraxodların, XX əsrin əvvəllərində təyyarələrin meydana gəlməsi poçt xidmətini xeyli sürətləndirdi. Teleqrafın (1832-ci il), telefonun (1876-cı il) və radionun (1895-ci il) kəşfindən asılı olmayaraq poçt praktik olaraq XX əsrin sonuna qədər ən kütləvi və ucuz əlaqə vasitəsi olmuşdur.

XXI əsrdə daha sürətli (kompüterlərlə əlaqədar) kommunikasiya xidmətləri yaransa da, informasiyanı praktik olaraq bir neçə saniyə ərzində dünyanın istənilən yerinə ötürmək mümkün olsa da, poçt xidmətlərinə tələbat olacaqdır.

1878-ci ildə dünyanın inkişaf etmiş 22 ölkəsi arasında poçt xidmətlərini nizamlayan ümumdünya poçt konvensiyası bağlandı.





Keçən əsrin əvvəlində poçt rabitəsi xidmətləri.

KOMPÜTERLƏR ARASINDA İNFORMASİYANIN ÖTÜRÜLMƏSİ

Arpanet (İnternetin ilk sadə variantı) şəbəkəsinin rəsmi olaraq 1972-ci ilin oktyabrında nümayiş etdirilməsi paketlərin kommutasiya üsuluna xüsusi maraq yaratdı və onun inkişaf etməsinə səbəb oldu.

Cəmiyyətin müasir inkişafı birinci növbədə bütün dünyanın qlobal informasiyalaşması ilə əlaqədardır. 1970-2002-ci illər ərzində yüzlərlə milli və beynəlxalq kompüter şəbəkələri qurulmuşdur. Bunun nəticəsində ölkələrin çoxunda informasiya texnologiyalarının tətbiq olunması və İnternetə çıxış təmin olunmuşdur.

Kompüterlər arası qarşılıqlı əlaqəyə (*kompüter şəbəkələrinə*) keçid informasiyanın ötürülməsində keyfiyyət sıçrayışını səciyyələndirir və kompüterlərin uzaqdan qarşılıqlı əlaqəsini, onların resurslarından şəbəkə istifadəçilərinin birgə istifadə etmələrini təmin edir.

Verilənləri ötürmək üçün kompüterlər müxtəlif fiziki *kanallar* istifadə edirlər. Bunlar ənənəvi elektrik kabelləri (bəzən izolə olunmuş bir neçə metal keçiricilər, bəzi hallarda ekranlaşdırılmış); retranslyator vasitəsilə olan radio əlaqələri, peyk əlaqələri; infraqırmızı şüalar (məsafədən idarətemə üçün olan televiziya pultları kimi); işıq siqnallarını mikrolazerlə ötürən müasir optik-lifli kabellər; adi telefon şəbəkəsi ola bilər. Bu kanallar vasitəsilə informasiyanın tam paketləri göndərilir.

Mexanika-riyaziyyat fakültəsinin şəbəkə avadanlığı.



Kompüterlərin öz aralarında necə birləşməsindən asılı olmayaraq (naqillər, radiokanallar, optik-lifli kabellər vasitəsilə) informasiyanın böyük məsafələrə ötürülməsi birbaşa deyil, kompüterlər qrupunun və ya digər köməkçi qurğuların vasitəsilə həyata keçirilir.

Verilənlərin ötürülməsi texnologiyasında bir-birindən prinsipial olaraq fərqlənən iki müxtəlif üsuldan – *kanal kommutasiyası* və *paket kommutasiyası* üsulundan istifadə edilir.

Birinci üsulda birləşmənin bütün yolu – göndərəndən qəbul edənə qədər – müəyyən edilir. Bundan sonra qaytarılan paket yolun tam hazır olduğu haqqında məlumat verir: verilənlərin ötürülməsinə başlamaq olar, bütün kanallar hazırdır. Verilənlərin ötürülməsi qurtardıqdan sonra ayrılan yol bağlanır.

Üsulun üstünlüyü alqoritmin sadəliyində və həm verilənlər, həm də rəqəm şəklində olan nitq vasitəsilə mübadilə imkanına malik olmasındadır.

İkinci üsulda məlumatlar əvvəlcə kiçik paketlərə bölünür, nömrələnir və ünvanlarla (kimdən kimə) təmin edilir. Bu paketlərin özləri marşrutdan asılı olmayaraq yollarını müəyyən edirlər. Bu halda şəbəkə də müntəzəm yüklənir. Bu üsul informasiyanın ötürülməsinin etibarlığını təmin edir (şəbəkədə kəsilmə halı olduqda yolda bütün paketlər itmir). Ancaq paket kommutasiyası üsulunun icrası zamanı *kommutasiyanın düyün* nöqtələrində və hər bir paketin emalı və saxlanmasında itkilər əmələ gələ bilər (məsələn, paket yeni marşrut seçərək yolu dəyişə bilər). Bu çatışmamazlıq müxtəlif sürətli kanallar vasitəsilə birləşdirilən kompüterlər arasında informasiyanın əlverişli ötürülməsilə kompensasiya olunur.

ÖTÜRÜLMƏ PROTOKOLU

XX əsrin 70-ci illərində müxtəlif modelli kompüterlərin geniş yayılması təkcə proqram təminatına görə onların uyğunluğu deyil, həm də onların qarşılıqlı əlaqəsi problemini yaratdı.

70-ci illərin ortasında bu problem öz həllini tapdı – "Açıq sistemlərin qarşılıqlı əlaqəsinin etalon modeli" (OSI) yaradıldı. Açıqlıq xassəsi *protokol* adlanan universal razılaşmaları, qaydaları əhatə edən proqram təminatının dəqiq yerinə yetirilməsi nəticəsində müxtəlif tipli kompüterlər arasında əlaqəni təmin edir.

OSI modelini Standartlaşma üzrə beynəlxalq təşkilat (International Standardization Orqanization – ISO) hazırlamışdır. Bu təşkilat informasiyanın şəbəkədə ötürülməsini və kompüter avadanlığını müəyyən edir. OSI modeli proqram və aparat təminatının yeddi səviyyəli şəbəkə modelidir.

I səviyyə – *fiziki* səviyyə adlanır. Bu səviyyədə verilənlər paketinin yuxarı kanal səviyyəsindən necə alınması, onların bir və sıfırlar axınına, elektrik və ya optik siqnallara necə çevrilməsi təsvir olunur. Burada fiziki mühit haqqında aşağıdakılar – kabel və birləşmə yerlərinin tipləri, kontaktların ayrılması, siqnalların kodlaşdırılması sxemi və IEEE-802.3 – Ethernet məlum standartı haqqında məlumatlar olur.

II səviyyə – *kanal* səviyyəsi adlanır. Bu səviyyə əlaqə kanalı üzrə birbaşa birləşmə zamanı verilənlər qrupunun yaradılmasını, ötürülməsini və qəbulunu təmin edir. Burada növbəti şəbəkə səviyyəsinin sorğuları emal olunur və paketlərin qəbulu və ötürülməsi üçün fiziki səviyyənin servisindən istifadə edilir.

III səviyyə – *şəbəkə* səviyyəsində paketlərin şəbəkədə yerdəyişməsi üçün marşrut seçilir.

Institute of Electrical and Electronics Engineers (IEEE) – şəbəkə və digər elektron kommunikasiya ilə əlaqədar olan standartları müəyyən edən professional təşkilatdır (ABŞ). IEEE 802.X qrupunda şəbəkənin xüsusiyyətləri təsvir edilir və şəbəkə və telekommunikasiya üçün standartlar, məsləhətlər, informasiya sənədləri yerləşir.

IV səviyyə – *nəqliyyat* səviyyəsində birləşmələrin hazırlanması, saxlanması və kəsilməsi prosedurları yerinə yetirilir.

V səviyyə – *seans* səviyyəsi adlanır. Bu səviyyə maşınlar arasında verilənlərin ötürülməsi seansının baş verməsinə, eləcə də əlaqə kanalının uzun sürən qırılmasından sonra ötürülmənin davam etdirilməsinə cavab verir.

VI səviyyə – *təqdimetmə* səviyyəsi adlanır. Bu səviyyədə verilənlərin çevrilməsi – kodlaşdırılması, sıxılması və s. təmin olunur.

VII səviyyə – *tətbiqi* səviyyə tətbiqi proqramların şəbəkəyə çıxışını təmin edir: faylların köçürülməsi (FTP – faylların köçürülməsi protokolu), poçt məlumatlarının mübadiləsi (X400 – elektron poçt, SMTP – poçt mübadiləsinin sadə protokolu), istifadəçinin uzaq məsafədə olan kompüterə çıxışı (Telnet – terminalla mübadilə protokolu) və s.

Protokol səviyyələri bir-birinin üzərində yerləşir. Daha yuxarı səviyyə daha aşağı səviyyələrin funksiyalarından istifadə edir və öz imkanlarını sxem üzrə növbəti səviyyəyə verir. Ona görə də verilənlərin ötürülməsinin protokollar səviyyələrinə protokollar *steki* də deyirlər.

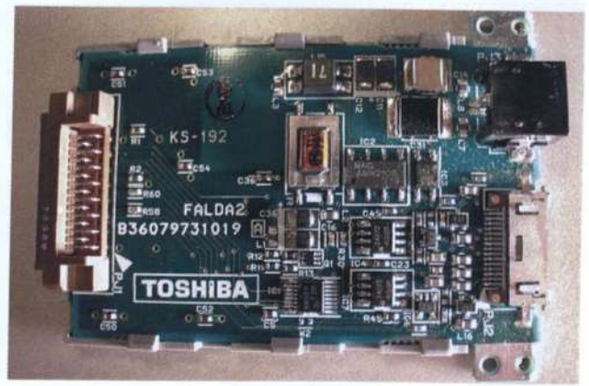


MODEM

Müasir telekommunikasiya vasitələrinin inkişaf tendensiyası analoq informasiyaların (səs, şəkil) rəqəm şəklində təsvir olunmasına imkan verir: rəqəmli CD, rəqəmli fotoşəkil və s.

Mövcud telefon xətləri rəqəmli informasiyaların ötürülməsini təmin etmir. Bu halda əvvəlcə rəqəmli informasiyanı analoq formaya çevirmək və sonra onu bu şəkildə telefon xətlərinə vermək, telefon xəttinin digər ucunda isə informasiyanın əks çevrilməsini yerinə yetirmək tələb olunur. Bu çevirmələri yerinə yetirən qurğu *modem* adlanır. Modem iki məntiqi qurğunu özündə birləşdirir: *modulyator* – rəqəmli informasiyanı analoq informasiyaya çevirir və *demodulyator* – əks çevirməni yerinə yetirir.

Müasir modemlər nəzəri olaraq informasiyaların ötürülməsində 50 kbit/san sürətini təmin edir. Doğrudur, real telefon şəbəkələrində aparılan eksperimentdə bu sürət çox nadir hallarda alınır, stabil əlaqə zamanı sürət adətən 2 dəfə azalır, 20 kbit/san olur. Daha yüksək sürət zamanı çox saylı səhvlər baş verir, qırılan paketlərin ötürülməsi sonsuz olaraq təkrarlanır və nəticədə real sürət xeyli azalır, ya da əlaqə kəsilir. Bunun səbəbi telefon şəbəkəsinin quruluşundadır.



Noutbukun daxili modemi.



ÜMUMDÜNYA ŞƏBƏKƏSİNƏ QOŞULMA: UUCP VƏ PPP – PROTOKOLLARI

Ümumdünya şəbəkəsi yaranana qədər istifadəçilər bir-birilə ünsiyyət yaratmaq üçün modemdən istifadə edirdilər. Modem ilk növbədə elektron poçt vasitəsilə işləməyə imkan verir. Bu halda UUCP protokolundan istifadə edilir.

İki kompüter arasında verilənlərin ötürülməsinin bu protokolu UNIX sistemində işlənmişdir. UUCP protokolu əsasən İnternetə qoşulan kompüterdən elektron poçtu istifadəçisinin fərdi kompüterinə və əksinə göndərmək üçün istifadə edilir. Fərdi kompüterlərdə UUPC proqramları (PC üçün protokol) oxşar qaydada işləyir. Belə ki, modemin köməyilə poçt serveri rolunu oynayan əsas kompüterə zəng etmək və UUCP birləşməsi yaratmaq, elektron poçtu fərdi kompüterə və əksinə göndərmək olar. Bundan sonra birləşmə qırılır, belə ki, UUCP protokolu telefon xəttini çox məşğul etmir. Poçtun oxunması və hazırlanması zamanı telefon xətti sərbəst qalır.

UUCP-poçtu çox əlverişlidir və çox resurs tələb etmir. Yüzlərlə istifadəçi bir-birinə mane olmadan bir telefon xəttindən istifadə etməklə poçtla işləyə bilərlər. Buna baxmayaraq, axır zamanlar UUCP protokolu çox az tətbiq olunur. Bu ümumdünya şəbəkəsinin getdikcə artan populyarlığı ilə əlaqədardır. Müasir provayderlər modem vasitəsilə İnternetə birbaşa qoşulmanı təmin edir və bu avtomatik olaraq elektron poçtun imkanlarından və İnternetin mövcud xidmətlərindən istifadə etməyə imkan verir.

Modem vasitəsilə İnternetə qoşulmaq üçün iki protokol vardır: SLIP və PPP. XXI əsrin əvvəllərində SLIP protokolu köhnəlib və demək olar ki, bütün provayderlər PPP protokolunu dəstəkləyirlər.

Modem vasitəsilə serverə qoşulma zamanı PPP protokolu fərdi kompüteri şəbəkənin tam hüquqlu qovşağına çevirir: belə kompüterə İnternet-ünvan verilir, İnternet-protokolun formatı əsasında hazırlanan paketləri qəbul edir və göndərir. Bir çatışmazlığı var: telefon xəttini tutur. Ödəniş haqqı göndərilən informasiyanın həcmilə deyil, istifadə olunan vaxtla müəyyən edilir.

Modem vasitəsilə informasiyanın ötürülməsinin sürəti və etibarlığı yüksək deyil və müasir kompüter şəbəkələri üçün qətiyyən məqbul sayılmır. Buna baxmayaraq, modemin böyük üstünlüyü ondan ibarətdir ki, o, kompüter şəbəkəsinin infrastrukturunun yaradılmasını gözləmədən mövcud telefon xəttindən istifadə etməyə imkan verir.

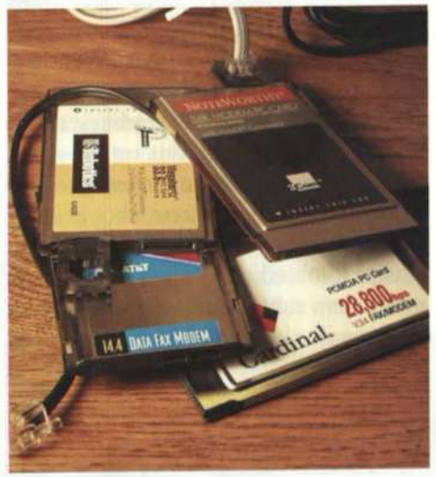
Yəqin ki, tezliklə müasir kompüter şəbəkələri də telefon şəbəkələri kimi adi bir şey olacaqlar. İndi artıq iri şəhərlərdə bir çox evlər müasir optik-lifli xətlər vasitəsilə İnternetə qoşulmuşlar. Belə şəbəkələr hazırlandıqca modem, 25 il bundan əvvəl EHM-in ayrılmaz hissəsi olan perfokart giriş qurğusu kimi köhnə bir texniki qurğuya çevriləcəkdir.

ISDN VƏ MOBİL TELEFON ŞƏBƏKƏLƏRİ

Mövcud telefon xətlərində olan çatışmazlıqları aradan qaldırmaq üçün telefon şəbəkələrinin yeni standartı – ISDN, yəni şəbəkənin inteqrallaşması xidməti (Integrated Service Digital Networks) hazırlandı. ISDN adi səsi və kompüter verilənlərini rəqəm şəklində ötürməyə imkan verir. Verilənlərin ötürülmə sürəti 64 kbit/san-dir. Danışıq zamanı səs rəqəmlər formasına çevrilir və bu formada xətlərə göndərilir,yəni modemin yerinə yetirdiyi çevirməyə əks olan çevirmə yerinə yetirilir. Təəssüf ki, ISDN telekommunikasiya sferasında soyuq qarşılandı. Birincisi, ona görə ki, onun inkişafı üçün çox güclü maliyyə vəsaiti tələb olunurdu. Ona görə də istifadəçilərin çoxu köhnə qaydada modemdən istifadə etməyə üstünlük verirlər.

İkincisi, ISDN-nin təmin etdiyi sürət müasir kompüterlər üçün münasib deyildir. Onun sürəti müasir modemlərin keyfiyyətli telefon şəbəkəsindəki sürətindən o qədər də fərqlənmir (lakin ISDN-nin etibarlığı daha yüksəkdir).

XXI əsrin əvvəllərində ölkəmizdə praktiki olaraq ISDN geniş yayılmamışdı. Lakin tezliklə mobil telefonlar şəbəkəsi yayılır. Mobil şəbəkələrdə tətbiq olunan protokollar kompüterlər arasında rəqəmli ötürmələr üçün çox əlverişlidir. Ona görə də İnternetə mobil telefonlar vasitəsilə qoşulmaq çox ucuz və əlverişlidir.



Kredit kartları ölçüsündə olan noutbuk modemləri.

MODEM NECƏ İŞLƏYİR

İki modemin birləşməsi zamanı onlar öz aralarında avtomatik olaraq hər ikisi üçün ötürmənin mümkün maksimal sürəti və *kommunikasiya protokolunun* seçilməsi barədə "razılığa" gəlirlər. Daha sürətli modem öz sürətini azaltmaqla az sürətli modemə qoşula bilər.

Əgər modemin dinamiki qoşulubsa, bu zaman insan fışıltı və müxtəlif tezlikli siqnallar eşidir. Əvvəlcə qoşulan

modem protokol təklif edir. Bu protokolda səhvlərin korreksiyası və sıxılması ilə maksimal sürət göstərilir. Əgər ikinci modem bu protokolu qəbul edə bilmirsə (məsələn, telefon birləşməsinin aşağı keyfiyyətli olmasına görə), onda birinci modem onu daha sadə, aşağı sürətə malik protokolla əvəz edir və bu proses o vaxta qədər aparılır ki, hər iki modem üçün qəbul oluna bilən protokol seçilir. Adətən, modemə verilənləri korrektə etmədən ən primitiv protokol səviyyəsinə "enmək" də qadağan edilir, əks halda adi telefon şəbəkəsində hətta ən qısa sözü belə səhvsiz ötürmək mümkün olmur.



Protokolda telefon xəttlərilə baytların ötürülməsi üsulları: analoq siqnalların modulyasiya tipləri, əsas tezlik, əlavə xidməti bitlər, baytların paketlərdə necə birləşməsi, paketlərin formatı, paket təhrif olunduğu halda ötürmənin təkrarlanması qaydası və s. təsvir olunur. Hər paketdə göndərilən baytlardan başqa əlavə informasiya, o cümlədən nəzarət cəmi də olur. Nəzarət cəminin köməyilə informasiyanı qəbul edən qurğu ötürmə zamanı paketin təhrif olub-olmadığını yoxlayır, ehtiyac olan halda informasiyanın ikinci dəfə ötürülməsini tələb edir.

Modem səhvləri düzəldən və həmçinin verilənləri sıxlaşdıran kodlardan istifadə edə bilər. Kodların ideyası ondan ibarətdir ki, hər bir bayta bitlər əlavə olunur ki, bu bitlərin birində səhv olarsa, onda birqiymətli olaraq ilkin bayt bərpa olunur. Bu zaman məlumatın uzunluğu artsa da, bir çox hallarda təhrif olunmuş paketlərin ötürülməsini təkrar etməyə ehtiyac qalmır. Bu, fasiləsiz əlaqə olduqda xüsusilə vacibdir, telefon xətləri isə olduqca etibarsızdır.



Verilənlərin sıxılması məlumatların məzmununu yığcamlaşdırmaq üçün istifadə olunur və nəticədə məlumatın həcmi xeyli azalır. Yığcam verilənlərin ötürülməsi (məsələn, şəkillərin JPEG formatında və ya ZIP-faylların göndərilməsi) zamanı zamanı əlavə sıxlaşdırma praktik olaraq məlumatın həcmini azaltmır. Verilənlərin sıxlaşdırılması səhvləri düzəldən kodlarla bir yerdə istifadə oluna bilər. Belə ki, əvvəlcə verilənlər yığcamlaşdırılır, sonra isə yığcamlaşdırılmış məlumatın hər bir baytı səhvləri düzəldən kodun köməyilə göndərilir.

Hazırda çoxlu sayda kommunikasiya protokolları mövcuddur, onların hər birinin öz işarələməsi var: MNP4, MNP5, V42b, V.90 və s. Əgər ilk modemlər yuxarı səviyyəli protokollardan istifadə etmədən ancaq ayrı-ayrı baytları göndərirdisə, indi verilənlərin ötürülmə sürəti bir neçə yüzdən 56 min bit/saniyəyə qədər artmışdır.

MODEMİN PROQRAMLAŞDIRILMASI

İnformasiya ardıcıl baytlar (8 bitdən ibarət olan qruplar) şəklində modemlə ötürülür. Hər bayta bir və ya bir neçə bitlər (start-biti, stop-biti) əlavə olunur. Baytların hər biri ayrıca, yaxud paketdə birləşdirilərək paket şəklində də göndərilə bilər. Hər bir paket üçün nəzarət cəmi haqqında məlumat verilir. Onlar üst-üstə düşmədikdə paketin ötürülməsi təkrarlanır.

Müasir modemlərin çoxu Hayes-uyarlıdır (Hayes firmanın adıdır). Bu modemin quruluşu sadələşdirilmiş şəkildə aşağıdakı kimi təsvir oluna bilər. Kompüterdə kommunikasiya portu vardır ki, bu portun köməyilə baytlar modemə verilir və modemdən qəbul edilir. Göndərilən və qəbul edilən baytlar iki asılı olmayan axın əmələ gətirirlər ki, bu da istənilən anda baytları portdan göndərməyə və portdan oxumağa imkan verir.

Modem aşağıdakı iki rejimdən birində ola bilər: *komandalar rejimində* və *verilənlər rejimində*. Komandalar rejimində modem port vasitəsilə ötürülən baytları komandalar kimi qəbul edir. Bu komandalar ya modemin daxili vəziyyətini (onun daxili registrlərinin məzmununu) dəyişə bilər, ya da modemin müəyyən əməliyyatı yerinə yetirməsini təmin edə bilər. İstənilən komanda iki AT simvolu ilə (*ing.* "attention" – "diqqət") başlayır və sətrin sonu simvolu ilə qurtarır. Məsələn:

- ATL3 – "daxili dinamikə maksimal səs qoy";
- AT S0 = 0 – "gələn zənglərə cavab vermək qadağandır";
- AT S0 = 2 – "iki zəngdən sonra dəstəyi qaldır";
- ATZ – "modemin inisiallaşdırılması".

Belə AT-komandaları onlarladır və onların köməyilə kompüter-modem ünsiyyəti baş verir. Ən geniş yayılmış komanda – telefon nömrəsinin yığılması komandasıdır:

AT DP 539-17-86,

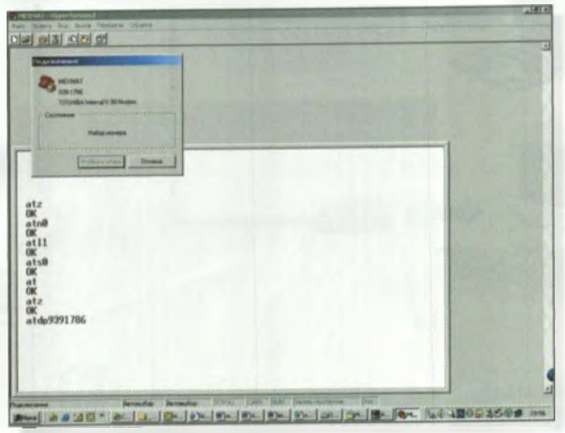
burada D (*ing.* Dial) – "nömrəni yığmaq", P (*ing.* Pulse) – "impuls" deməkdir. Azərbaycanda nömrənin impuls yığılması qəbul edilmişdir, tonal yığılmada isə DT prefiksindən istifadə edilir.

Komandalar rejimində hər bir komandanın yerinə yetirilməsinin nəticəsini modem kompüterə xəbər verir. Verilənlər uzaq məsafədə olan modemdən kompüterə necə ötürülürsə, həmin qaydada komandanın nəticəsinin kodu modemdən kompüterə ötürülür.

Verilənlər rejimində kommunikasiya portundan modemə verilən bütün baytlar uzaq məsafədəki modemə göndərilir.

İstənilən əməliyyat sistemi xüsusi "terminal" proqramına malikdir ki, bu proqram istifadəçiyə, məsələn, AT komandaları vasitəsilə modemə müraciət etməyə imkan verir. MS Windows sistemində belə proqram Hyperterminal adlanır. Uzaqda yerləşən modem ilə əlaqə yarandıqdan sonra klaviaturada yığılan işarələr uzaqdakı kompüterə göndərilir və əksinə, uzaqdakı kompüterin ekranında çıxan bütün işarələr modem vasitəsilə terminal proqramına göndərilir və onun pəncərəsində əks olunur. Beləliklə, terminal proqramı, fərdi kompüteri terminala (ekran+klaviatura) çevirməyə və evdən çıxmadan uzaqdakı kompüterlə işləməyə imkan verir. Onun

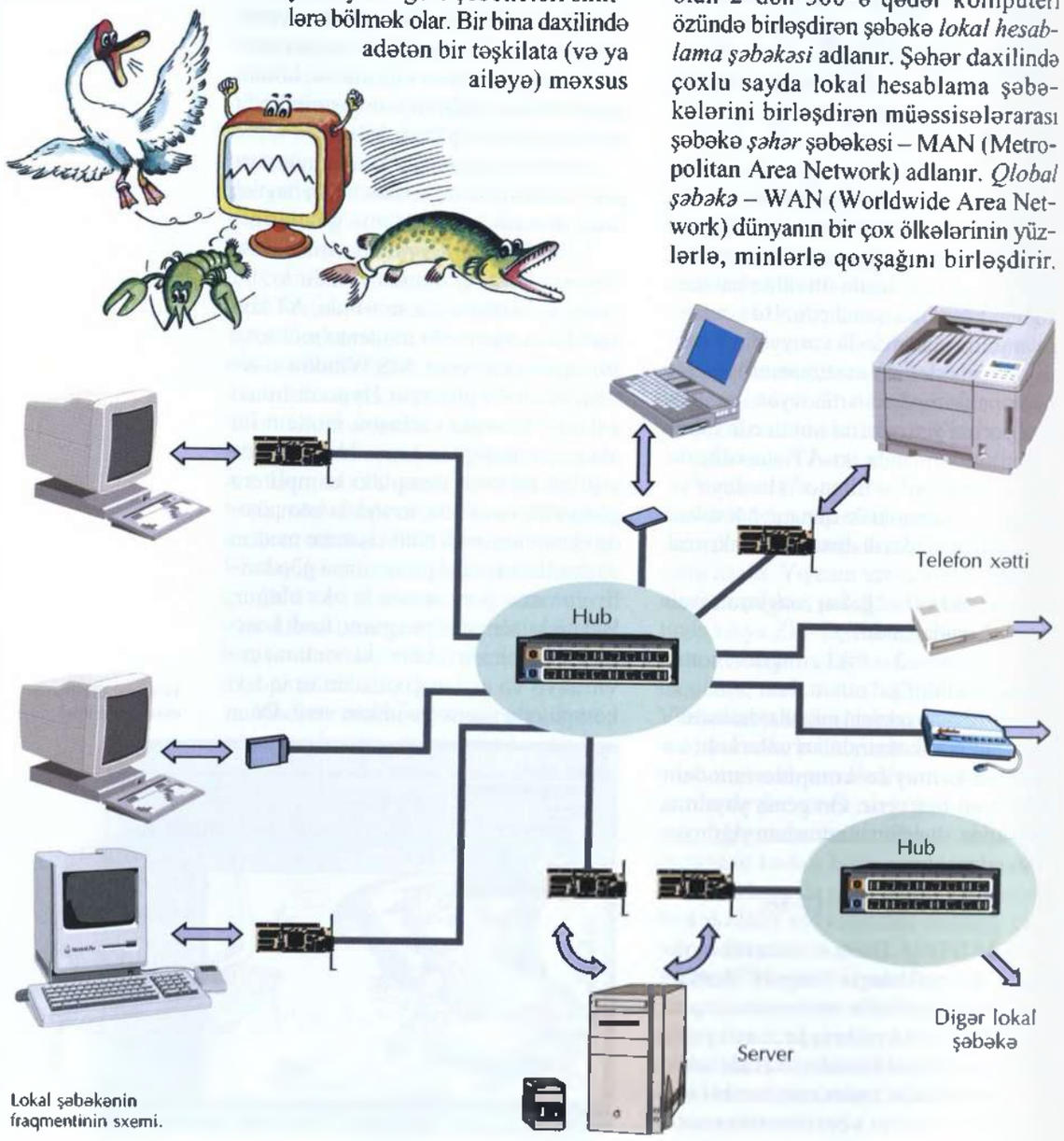
Terminal proqramın ekran görüntüsü.



köməyilə hətta Z-modem protokolundan istifadə etməklə fayl da göndərmək olar. Bu zaman Z-modem uzaqda yerləşən kompüterin qəbul rejiminə yüklənir, terminal proqramı isə komandaların yığılma ardıcıllığına müvafiq olaraq klaviş kodları kimi faylın mahiyyətini bitlər ardıcıllığı ilə ötürür.

HANSI KOMPÜTER ŞƏBƏKƏLƏRİ VARDIR

Böyüklüyünə görə şəbəkələri siniflərə bölmək olar. Bir bina daxilində adətən bir təşkilata (və ya ailəyə) məxsus olan 2-dən 300-ə qədər kompüteri özündə birləşdirən şəbəkə *lokal hesablama şəbəkəsi* adlanır. Şəhər daxilində çoxlu sayda lokal hesablama şəbəkələrini birləşdirən müəssisələrarası şəbəkə *şəhər* şəbəkəsi – MAN (Metropolitan Area Network) adlanır. *Qlobal şəbəkə* – WAN (Worldwide Area Network) dünyanın bir çox ölkələrinin yüzlərlə, minlərlə qovşağını birləşdirir.



Lokal şəbəkənin fraqmentinin sxemi.

Lokal şəbəkə kompüter avadanlığından rasional istifadə edilməsi və əməkdaşların səmərəli işləmələri üçün yaradılır. Hazırki zamanda hər hansı bir firmada, hətta mənzildə, ən azı iki kompüter varsa, onların şəbəkədə birləşməməsini təsəvvür etmək çətindir. Şəbəkə faylların bir maşından digərinə göndərilməsini, birgə arxivdə saxlanmasını (adətən kompüterlər eyni güclü deyillər və birinin disk sahəsi böyük ola bilər) və çap edilməsini təmin edir.

Adətən ofisdə bir server (verilənləri saxlamaq və çap etmək üçün), əməkdaşlar üçün iş stansiyaları, faks, elektron poçt, elektron ödənişləri qəbul etmək və göndərmək üçün İnternetə çıxışı təmin edən bir-iki modem (və ya birbaşa kabel birləşməsi), bir neçə şəbəkə printeri, onlarla telefon nömrəsi olan daxili ATS yerləşdirilir. Lokal şəbəkənin daxilində əməkdaşların yazışmasının təşkili sənədlərin dövriyyəsini sadələşdirir. Evdə isə böyük diski olan masaüstü serverlə, printerlə, skanerlə və bir-iki "zəif" maşınla (bu noutbuk da ola bilər, sahibi onu işə aparır və hərdən evdə işləyir) kifayətlənmək olar. Daha böyük həcmə malik informasiyaların ötürülməsi üçün yazan CD-ROM-un olması o qədər də lazım deyil, belə ki, müasir noutbuklar birləşmə üçün olan hörülmüş cütlərdən (kabeldən) istifadə etməklə bu məsələlərin həllini asanlaşdırır (lokal şəbəkələrdə ötürmə vasitəsi kimi bir-birinə hörülmüş adi naqillərə oxşayan *ekranlaşmamış hörülmüş cütlərdən* daha çox istifadə edilir).

Ethernet texnologiyası əsasında kompüterləri birləşdirmək üçün birtipli protokolların və eynitipli avadanlıqların tətbiq edilməsi yeni maşınların şəbəkəyə qoşulmasını asanlaşdırır. Baxmayaraq ki, Ethernet XX əsrin 80-ci illərinin əvvəlində yaranmışdı, bu texnologiyanın geniş şəkildə tətbiq olunması 80-ci illərin sonunda oldu. Bu IEEE 802.3 Komitəsinin (Standartlaşma üzrə beynəlxalq təşkilat) 10 BaseT standartını hazırlaması ilə bağlı idi. Bu standart lokal şəbəkənin təbiətini dəyişdirdi: ekranlaşmamış hörülmüş cütlərdən istifadə edilməyə başlandı.

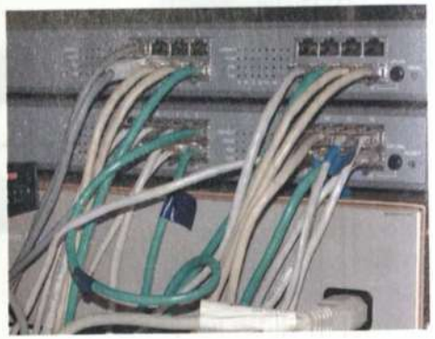
Əvvəllər lokal şəbəkədə birləşdirilən bütün maşınlar arasında vahid koaksial naqil (televizor antenasının kabelinə oxşar) yerləşdirilirdi. O, qırıldıqda bütün şəbəkə sıradan çıxırdı. Qəza nəticəsində naqil üzərinə yüksək gərginlik (hətta 220 V) düşdükdə şəbəkədə olan bütün kompüterlər sıradan çıxmış olurdu. Belə naqil şəhər informasiya şəbəkəsinə çıxış üçün istifadə edilə biləcək *qalın*



Hub və nazik Ethernet.



Qalın Ethernet.



Hub və hörülmüş cütlər.

Ethernet texnologiyasının geniş yayılmasına baxmayaraq, digər protokolları (və avadanlıqları) da qeyd etmək olar. Məsələn, Token ring.

Ethernet-dən fərqli olaraq *nazik Ethernet* adlanırdı.

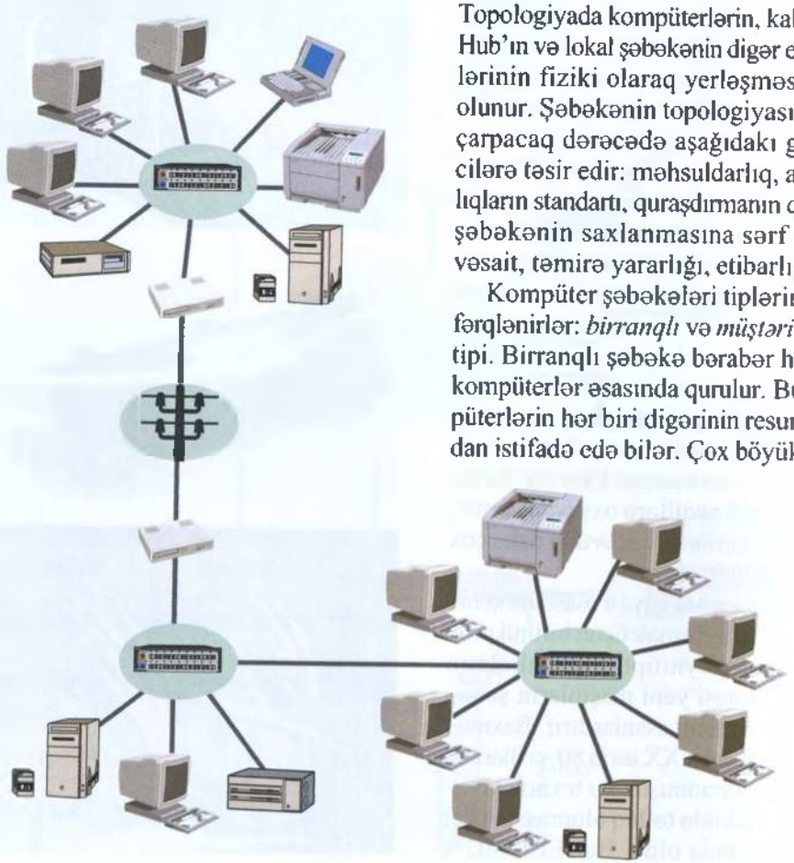
MAN və WAN ərazidə paylanmış şəbəkələrdir, uzaq məsafədə olan lokal şəbəkələr onlara sürətli optik-lifli əlaqə kanalları vasitəsilə birləşdirilir. Lifli-optik kabellər hazırda hər yerdə tətbiq olunur və onlar özlərinin misdən olan əvvəlki sələflərini müvəffəqiyyətlə sıxışdırıb aradan çıxarırlar.

10 BaseT standartı "ulduz" texnologiyasından istifadəni müəyyənləşdirir. Bu texnologiyada istənilən kompüter Hub adlanan qurğuya qoşulur. Huba eyni zamanda 8, 16 və ya 32 kompüter qoşmaq olar. Göründüyü kimi, sanki Hub – ulduzun mərkəzidir, kompüterlər isə onun şüalarıdır. Hub'ın fəaliyyəti telefonçunun işini xatırladır, o, paketləri maşınlar arasında paylamaq üçün kommutasiya edir. Daha mürəkkəb Hub eyni zamanda bir neçə cüt maşın arasında verilənlərin ötürülməsini təmin edir.

Əgər maşınlar binada paylanmışdırsa, onda bir Hubdan istifadə etmək səmərəli deyil (hətta şəbəkədə maşınların sayı az olarsa), çünki kompüterlərin ona birləşdirilməsi çox çətin olur. Bu problemi hər bir otaqda Hub yerləşdirərək orada olan kompüterləri ona qoşmaqla və bu Hubları mərkəzi Huba birləşdirməklə asan həll etmək olar. Əgər evdə iki kompüteri əlaqələndirmək tələb olunarsa, onda Hub lazım deyildir, xüsusi kabellərdən istifadə etmək kifayətdir.

Şəbəkəni təsvir etmək üçün "şəbəkə topologiyası" terminindən istifadə edilir. Topologiyada kompüterlərin, kabellərin, Hub'ın və lokal şəbəkənin digər elementlərinin fiziki olaraq yerləşməsi təsvir olunur. Şəbəkənin topologiyası nəzərə çarpacaq dərəcədə aşağıdakı göstəricilərə təsir edir: məhsuldarlıq, avadanlıqların standartı, quraşdırmanın qiyməti, şəbəkənin saxlanmasına sərf olunan vəsait, təmirə yararlığı, etibarlığı və s.

Kompüter şəbəkələri tiplərinə görə fərqlənirlər: *birranqlı* və *müştəri-server* tipi. Birranqlı şəbəkə bərabər hüquqlu kompüterlər əsasında qurulur. Bu kompüterlərin hər biri digərinin resurslarından istifadə edə bilər. Çox böyük sayda



Paylanmış şəbəkə fraqmentinin sxemi (WAN, MAN).

istifadəçiyə malik olan şəbəkədə istifadəçilərin şəbəkənin bütün kompüterlərinə müraciət etməsi arzu olunan deyil. Ona görə də birranqlı şəbəkələr eyni bir layihə üzərində işləyən kiçik qrup üçün əlverişlidir.

Şəbəkənin "müştəri-server" tipi daha geniş yayılmışdır. Proqramın *müştəri* və *server* hissəsinə ayrılması zamanı sadə terminal kimi istifadəsi səmərəli olmayan masaüstü kompüterlərin məhsuldarlığından daha yaxşı istifadə etmək olur. Belə yanaşma proqramları əsas kompüterdən – meynfreymdən (*ing.* "mainframe") lokal şəbəkələrə əsaslanan sistemlərə keçirməyə imkan verir. İstifadəçinin maşınında yerinə yetirilən proqram şəbəkəni verilənlərin ötürülməsilə az yükləyir. Belə ki, istifadəçi şəbəkədən yüklənən səhifəyə baxan zaman İnternetin naviqasiya proqramları, məsələn, Internet Explorer proqramı şəbəkəyə mane olmur.

ŞƏBƏKƏ ƏMƏLİYYAT SİSTEMİ (ƏS)

Vinçesterin qiyməti baha olan və onun hər bir kompüterə qoyulması sərfəli olmayan zamanlar şəbəkə əməliyyat sistemlərindən istifadə edilirdi. Bu sistemlər bir neçə fərdi kompüteri birləşdirərək çox da böyük olmayan proqramlardan və verilənlərdən birgə istifadə etməyə imkan verirdi. Əməliyyat sisteminin yüklənməsi mərkəzi maşından yerinə yetirilirdi. Terminalların bir kompüterə qoşulduğu hal üçün şəbəkə əməliyyat sistemi kollektiv istifadə mərkəzlərini xatırladırdı. Terminal olaraq fərdi kompüterlər götürülürdü və bu terminalların "intellekt"i olduqca yuxarı idi. Çünki proqramlar mərkəzi maşında deyil, istifadəçinin kompüterində yerinə yetirilirdi. Şəbəkə əməliyyat sistemi universitetlərdəki tədrisdə geniş istifadə olunurdu. Belə ki, bir tərəfdən tələbələrə böyük kompüter resursları tələb olunmurdu, digər tərəfdən şəbəkə əməliyyat sistemilə idarəetmə əlverişli idi. Bu gün şəbəkə əməliyyat sistemləri bütün işləri görürlər: köhnə fayl sistemlər ilə işləyir, eyni zamanda "müştəri-server" proqramlarını emal edir. Novell Netware ən geniş yayılmış şəbəkə əməliyyat sistemidir.

SİMSİZ RABİTƏ TEXNOLOGİYASI

İNFORMASİYANIN SİMSİZ ÖTÜRÜLMƏSİ. RADİO

1920-ci ilə qədər radio rabitəsində uzunluğu 100 metrdən 10 kilometrə qədər olan dalğalar tətbiq olunurdu. 1922-ci ildə radiohəvəskarlar atmosferin yuxarı qatlarından əks olunaraq istənilən məsafəyə yayılan qısa dalğaların xassəsini kəşf etdilər. Belə dalğalar uzaq radiorabitə üçün əsas vasitə oldu.

XX əsrin 30-cu illərində birbaşa görünən zonalara yayıla bilən metrli, 40-cı illərdə isə diametrli və santimetrli dalğalar müəyyən olundu. Bu dalğaların təsir radiusu 40-50 km-lə məhdudlaşırdı. Lakin qısa dalğaların tezlik diapazonunun eni (30 MHs-dən 30 GHs-ə qədər) 10 metrdən uzun olan dalğaların tezlik diapazonunun enindən min dəfə böyükdür, yəni bu tezlik vasitəsilə böyük informasiya axınını ötürmək və çoxkanallı rabitəni təşkil etmək mümkündür.



Yayılmanın kiçik məsafədə olması və siqnalların ensiz sahələr üzrə istiqamətləndirilməsi eyni uzunluqda olan dalğaları bir-birinə mane olmadan yaxın məsafədə yerləşən məntəqələrdə istifadə etməyə imkan verir.

Hers dipolu (Hers vibratoru) – 1888-ci ildə Henri Hersin öz təcrübəsində istifadə etdiyi sadə antenadır. O, metal şarları olan mis çubuqdan ibarət idi. Bura Rumkorf çarxı adlanan qurğu da əlavə olunurdu. Dipolda elektrik rəqsləri həmin vaxtlar elektrik rəqslərinin yeganə mənbəyi olan qığılcım boşalmasının köməyilə alınırdı. Ona görə də bu üsuldan istifadə edən ötürücülər *qığılcımlı* adlanırdı.

RADİOYA GEDƏN YOLDA

Hələ XIX əsrin 80-ci illərində Amerika ixtiraçısı Tomas Edison radiorabitəyə aid təcrübələr keçirirdi və hətta patent almışdı. 1886-1889-cu illərdə alman fiziki Henri Hers elektromaqnit dalğalarının varlığını eksperimental olaraq isbat etdi və elektromaqnit dalğalarının qığılcımlı şüalandırıcısını yaratdı. 1895-ci ildə rus alimi və elektrotexniki Aleksandr Stepanoviç Popov naqilsiz (radiorabitə) elektrik əlaqəsini kəşf etdi. 1895-ci ilin 7 mayında Popov dünyada birinci olaraq radioqəbuledici nümayiş etdirdi. 1897-ci ildə İtaliya radiotexniki Qulelmo Markoni analoji aparat layihələşdirdi. O, həmin il İngiltərədə, 1899-cu ildə isə ABŞ-da Kabelsiz teleqraflaşdırma və siqnallaşdırma şirkətini qeydiyyatdan keçirdi. Markoni 1901-ci ilin dekabrında Atlantik okeanının üzərindən radioteleqraf ötürülməsini təmin etdi.

Çox uzaq məsafələrə ötürülmə radiorele rabitə xətləri adlanan çox təkrarlanan retranslyasiya yolu ilə və ya yer səthindən 40 min km yüksəklikdə yerləşən peyklərin köməyilə həyata keçirilir.

1953-cü ildə "Strela-M" radiorele əlaqəsi qurğusu hazırlandı ki, bunun vasitəsilə eyni zamanda 24 telefon danışığını ötürmək mümkün oldu. Bir neçə il sonra onu 120 telefon kanalını təmin edən P-60/120 radiorele stansiyası əvəz etdi.

Bir radioverilişin digərlərinə mane olmaması üçün radio verilişlərinin tezliklərinin paylanması və hesabatının aparılması tələb olunurdu.
1927-ci ildə radio konfransının qərarı ilə tezliklərin enli zolağında qığılcımlı radio ötürücülər qadağan olundu (onlar ancaq qəbul etmənin ehtimalını artırmaq üçün SOS siqnallarının ötürülməsində saxlanıldı).

RƏQƏMSAL RADİORABİTƏ

Mobil telefon əlaqəsində səsin (və digər informasiyaların) ötürülməsi üsulları radiorabitənin prinsiplərinə əsaslanır.

İnformasiyanın ötürülməsində ümumi geniş radio kanalı necə bölmək olar? Çox sadə həll vardır: istifadə olunan diapazon bərabər hissələrə elə bölünür ki, ayrılan zolaqda ötürülən informasiyanın bütün spektri yerləşsin. Əgər bu telefon abonentinin səsidirsə, onda hər bir abonentə insan nitqinin spektrindən çox olan 10 kHs vermək kifayətdir. Bir abonentin danışığının digərlərinə mane olmaması üçün kifayət qədər böyük ehtiyat ayırmaq lazımdır.



CDMA və GSM mobil telefonları.

Birinci radiorabitə şəbəkələrində 50 kHs addımı istifadə olunurdu, sonra bunu 25 kHs-ə qədər azaltmaq mümkün oldu. Mobil əlaqənin analoji sistemlərində tezlikləri bölməklə çoxsaylı müraciət və ya kanalların tezliklərə görə bölünməsi – FDMA (*ing.* Frequency Division Multiple Access) üsulları tətbiq olunur. Səsin rəqəmli üsulla sıxılmasından istifadə etməklə texniki olaraq addımı 6-8 kHs-ə qədər azaltmaq mümkündür. Bu kəmiyyəti sonsuz olaraq azaltmaq real deyildir, çünki tezliklərin bir-birinə yaxın olması səbəbindən güclü təhriflər baş verə bilər. FDMA üçün artıq limit vəziyyət alınmışdır.

Digər üsullar daha dayanıqlı, kodlaşdırılması və şifrlənməsi asan olan rəqəmsal radiorabitəyə aiddir. Abonentin səsi diskretləşdirilərək rəqəm təsvirinə çevrilir və efirə bu şəkildə ötürülür.

Rəqəmsal radiorabitənin ən geniş yayılmış növü – *kanalların zamana görə bölgüsüdür*, TDMA (*ing.* Time Division Multiple Access). Radioşəbəkə üçün ayrılan tezliklərin ümumi geniş zolağına bu üsulu təmiz şəkildə tətbiq etmək olmaz. Məsələn, Avropada ən geniş istifadə edilən GSM mobil şəbəkəsində kanalların tezlikli və zamanlı bölgüsü üsulların kombinasiyasından (FDMA+ + TDMA) istifadə edilir. Əvvəlcə FDMA metodu ilə 25 MHs-lik ümumi zolaq 200 kHs-lik kanallara bölünür, sonra isə TDMA metodu ilə kanallar 8 istifadəçi kanalından ibarət olaraq qruplaşdırılır. Hər danışıq üçün (25 kHs-lik enə ekvivalent olaraq) kiçik zaman kvantı ayrılır ki, həmin zaman anında kanalla informasiya ötürülür, sonrakı zaman anında isə digər danışıq informasiyası ötürülür və proses dairəvi olaraq davam edir. Eyni bir danışığın bir kanalla ötürülməsi vacib deyildir. Stansiyalarda alınan hissələr abonentə görə "aydınlaşdırılır" və "birləşdirilərək" kəsilməz danışığa çevrilir.

Gözləmək olar ki, yaxın vaxtlarda bir abonentin zolaq itkisi 4 kHs-ə qədər azalacaqdır. Oxşar metodlarla kodlaşdıran IS-54 Amerika standartında zolaq itkisi GSM-ə nəzərən üç dəfə azdır.

Bunlardan əlavə, kanalları kodlarla ayıran rəqəmsal radiorabitə – CDMA (*ing.* Code Division Multiple Access) mövcuddur. Kanal tezliklər üçün ayrılmış bütün zolağı tutur və ona görə də digər kanallara mane olur. Elə görünə bilər ki, bu problemi daha da dərinləşdirir. Lakin təsəvvür edək ki, restoranda stol arxasında müxtəlif dillərdə danışan adamlar oturmuşlar. Yad nitq ayrı-ayrı danışıqlara az mane olur. Belə ki, söhbət edənlər ümumi səsdən öz ana dilində başa düşdükləri sözləri seçirlər. CDMA şəbəkəsi də bu prinsipə əsaslanmışdır.

1935-ci ildə Leninqrad Elektrotexnika İnstitutunun aspirantı Dmitriy Vasilyeviç Aqeyev "Xətti seleksiyanın nəzəri əsasları" adlı məqalə yazdı. Bu məqalədə o göstərdi ki, iki radiosiqnal bir radiotezlik zolağını tutarsa və onlar vaxta görə də üst-üstə düşərlərsə, onda radioqəbuledicidə ötürülən siqnalların forması haqqında informasiya varsa, bu siqnallar ayrılır və radioqəbuledici tərəfindən qəbul edilir.

Bu ideya hərbiçilərin böyük marağına səbəb oldu, belə ki, informasiyanın ötürülməsinin bu texnologiyasından istifadə etməklə başqalarının qulaq asmasından yaxşı qorunmaq olar. Məsələn, amerikalılar hərbi aviasiyada radioəlaqə üçün siqnalların koda görə ayrılmasından istifadə edirdilər. Belə bir imkan ilk dəfə 1989-cu ildə Qualcomm Amerika şirkəti tərəfindən nümayiş etdirildi və mobil əlaqə üçün CDMA standartı hazırlandı.

Belə bir təəssürat yarana bilər ki, CDMA radio şəbəkəsinin ümumi potensial buraxılış qabiliyyəti azalır, lakin CDMA sisteminin real effektliliyi hətta TDMA-dan yüksəkdir. CDMA sistemində diapazona bölünmə zamanı yaxın yerləşən kanallar arasında müəyyən bir ehtiyat saxlamaq tələb olunmasına baxmayaraq itki yoxdur. Maneəyə *dayanıqlı kodlaşdırmanın* xüsusi üsullarının köməyilə rabitənin keyfiyyətini yüksəltməklə yanaşı danışıqlar arasındakı zolağın ümumi ehtiyatlarını daha aktiv danışanın xeyrinə dəyişmək olar. Düzdür, CDMA qəbuledicilərı TDMA-nın oxşar qəbuledicilərindən mürəkkəbdir.

MOBİL İNTERNET

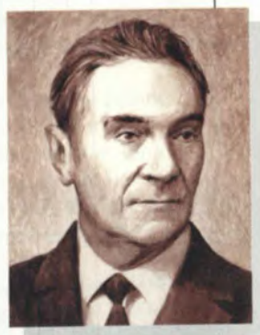
Mobil əlaqənin köməyilə nəinki danışmaq, həm də İnternetdən istifadə etmək olar. Mobil telefona modem birləşdirmək olar. Ancaq nəzərə almaq lazımdır ki, modem rəqəmli siqnalların analoq metodları vasitəsilə ötürülməsindən istifadə edir. Mobil əlaqə isə, bir qayda olaraq, ancaq rəqəmli əlaqədir. Bundan əlavə, mobil əlaqə səsi və ya çox sürətli olmayan informasiyaları: faksı, SMS-i (qısa mətn məlumatlarını) ötürmək üçün istifadə edilir. Modem vasitəsilə verilənlərin ötürülməsinin real sürəti 10 kbit/san-ni aşmır. Əlbəttə, bu İnternetdən tez-tez istifadə edən insanların tələblərini ödəmir. Bu sürətlə ancaq kiçik həcmli elektron poçtu almaq olar.





Mobil telefonda İnternet var.

D.V.Aqeyev Qorki Politexnik İnstitutunda dünyada ən güclü radioelektronika məktəblərindən biri olan radio-texnika fakültəsinin əsasını qoymuşdur.



D.V.Aqeyev.

Naqilsiz rabitə qurğusu Bluetooth.

CDMA-2000 verilənlərin yüksək sürətlə (təxminən 150 kbit/s) ötürülməsini (o cümlədən İnternetə qoşulmanı) təmin edir.

Mavi dişli – Danimarka kralı Haraldın (890-940-cı illər) ləqəbidir. Bu ləqəb ona ön dişi tünd olduğuna görə verilmişdir. Harald böyük istilaçı və xristianlığı Danimarkaya gətirən bir insan kimi tarixə düşmüşdür.

GSM şəbəkəsində sürəti artırmağın çıxış yolu tapıldı – mobil əlaqə şəbəkəsində verilənlərin paketlə ötürülməsi – GPRS (*ing.* General Packet Radio Service) metodu hazırlandı. Bu metodun köməyilə informasiya paketlərə yığılır və səslərin ötürülməsi arasındakı "boşluqları" doldurараq efirə verilir. Əgər modemin köməyilə verlənlərin adi ötürülməsində əvvəlcə birləşmə qurularaq sonra verilənlərin ötürülməsinə başlanırdısa, paket ötürülməsi zamanı telefon açıq olduğundan, mobil əlaqəyə müraciət mümkün olduğundan birləşmə həmişə vardır.

Verilənlərin paket üsulu ilə ötürülməsi zamanı radiokanal, onunla paket ötürüldüyü müddətdə, yəni ancaq verilənlərin ötürülməsi prosesində məşğul olur. Ona görə də ödəniş qoşulma vaxtına görə deyil, *trafikə* – verilən və qəbul edilən informasiyanın həcminə görə müəyyən edilir.

GPRS verilənlərin 107 Kbit/s və ondan yüksək sürətlə ötürülməsini təmin edə bilər. Bu tam qəbul edilən bir sürətdir və hətta modemdən istifadə edən oxşar adi telefon xətlərinin imkanlarından çoxdur. GPRS-in birinci nəsli 13,4 kbit/san sürətilə informasiyanın verilməsini, 40,2 kbit/san sürətilə informasiyanın qəbul edilməsini təmin edir.





SİMSİZ RABİTƏ

1998-ci ilin əvvəllərində 5 kompaniya – Ericsson, Nokia, IBM, Intel və Toshiba kabelsiz rabitənin yeni standartını hazırlamaq üçün SIG (Spesial Interest Group) adlı işçi qrup yaratdılar. Simsiz rabitəyə Bluetooth adı verildi ki, bu da ingiliscədən tərcümədə "mavi dişli" mənasını verir.

Bluetooth standartı ISM (*ing.* Industry, Science and Medicine – "sənaye, elmi və tibbi") adlanan diapazonda 2,4-2,8 GHs tezliyi istifadə edir ki, bunun üçün mobil rabitədən fərqli olaraq lisenziya almaq tələb olunmur. Verilənlərin ötürülmə sürəti 720 kbit/san-yə qədər çata bilər. Bluetooth standartı 10 m-dən çox olmayan məsafədə əlaqəni saxlamağa imkan verir, lakin XXI əsrin birinci illərində mikrosxemlər hazırlandı ki, bu sxemlərin köməyilə məsafəni 100 m-ə qədər artırmaq mümkün oldu.

Belə tezlikli şüalar maneələri keçmək qabiliyyətinə malikdir, ona görə də qurğunun birbaşa görünüş zonasında yerləşməsi vacib deyildir. Bluetooth-qurğu əlaqə mümkün olan sahədə yerləşən kimi birləşmə gedir və standart geniş yayımı təmin edir.

ISM-in lisenziyalaşmaması onun həm üstünlüyü, həm də çatışmayan cəhətidir. Müxtəlif tibbi cihazlar, məişət texnikası və s. də beləcə işləyir ki, bu da onlar arasında zidiyyətin yaranmasına səbəb ola bilər. Bundan qaçmaq üçün Bluetooth tezliklərin sıçrayışa bənzər (saniyədə 1600 sıçrayış) qurulması prinsipilə işləyir. Yeni tezlik təsadüfi ədədlər generatoru vasitəsilə seçilir və bu digər qurğulara lazım olan tezlikləri azad edir.

Bluetooth texnologiyası sayəsində mobil rabitənin abonenti otaqda olan belə modullu telefonla simsiz qulaqlıqdan istifadə etməklə mikrofonla danışa bilər. Güman olunur ki, mobil telefonlara, kompüter və digər qurğulara simsiz rabitənin tətbiqinə Bluetooth qoymaq mümkün olacaq.

HAMIYA MƏLUM NAMƏLUM...

Naqilsiz texnologiya da "Axilles dabanına" malikdir. Problem sanki havada qalıb: informasiya naqilsiz ötürülürsə, deməli, o, başqalarının ona qulaq asa biləcəyindən sığortalanmayıb. Düzdür, quraşdırıcılar bu problemi də ötürülən informasiyanın şifrlənməsi üsulu ilə həll etməyə çalışırlar. Belə görünə bilər ki, məlumatların şifrlənməsinin mövcud texnikası onu "kənar qulaqlardan" xilas edəcək ("Müasir kriptoqrafiya" məqaləsinə bax). Lakin naqilsiz əlaqə vasitələrinin işlənilməsi zamanı buraxılan və hakerlər tərəfindən aşkar edilən səhvlər təkcə konkret bir avadanlığı, konkret bir istifadəçini (söhbətinə qulaq asılan) deyil, həm də bütövlükdə müəssisəni zərbə altında qoyur.

XXI əsrin ən populyar naqilsiz texnologiyası olan Bluetooth ilə də belə oldu.

NƏYİ NƏ İLƏ BİRLƏŞDİRMƏLİ

Bu kompüter alıcısının ənənəvi sualıdır. Əgər kompüter birində monitor, birində isə sistem bloku yerləşən iki böyük qutudan ibarətdirsə, onları birləşdirmək üçün heç olmasa kabel olmalıdır. Əgər siçan və klaviatura verilənləri ötürmək üçün radio və ya infraqırmızı əlaqələrdən istifadə etmirsə, onda əlavə bir cüt kabel lazımdır. Bundan əlavə, musiqi səsləndirmək üçün səsucaldanlara, DVD disklərə monitordan deyil, ekrandan baxmaq üçün televizorlara kabellər lazımdır. Əgər kompüter oyunları ilə maraqlanırsınızsa, onda 3D eynəkləri (və ya dəbilqə) və pedalı olan sükanlar üçün də kabellər lazımdır. Əlavə printer, skaner, modem (əgər o, xarici modemdirsə), lokal şəbəkə üçün hörülmüş cütlər... Əgər kompüter video və ya səsi emal etmək üçün tətbiq olunarsa, onda kabelləri peşəkar mütəxəssis də qarışıq sala bilər.

Əlbəttə, hesablama texnikasını hazırlayanlar istifadəçinin kabelləri səhv qoşmaqla kompüteri sıradan çıxarmaması üçün foolproof ("axmaqdan müdafiə") haqqında düşünmüşlər. Ona görə də müxtəlif tipli birləşmələr üçün kompüter müxtəlif formalı birləşmə yerlərinə malikdir. Kabelləri nəinki düzgün birləşdirmək, həm də səliqəli birləşdirmək lazımdır ki, təsadüfən maşından ayrılmasın və s.

Eyni "xəstəlik" musiqi mərkəzləri və videomaqnitofonlara da aiddir. DVD-pleyeri, bir cüt video maqnitofonu və musiqi mərkəzini bir televizora birləşdirmək lazım gələrsə, onda necə olar? Sadəcə olaraq bütün çıxışların bir girişə sadə birləşməsi kömək etməz. Çünki müxtəlif qurğuların siqnalları bir-birinə mane olacaqdır. Bu halda xüsusi kommutasiya qurğusundan istifadə etmək lazımdır.

Paltaryuyan maşınlar və soyuducular daha mükəmməl hazırlanmışlar. Belə ki, onlar ancaq bir kabellə şəbəkəyə birləşdirilirlər. Paltaryuyan maşın ev informasiya şəbəkəsinə, soyuducu isə İnternetə birləşdirilərsə nə baş verəcəyi məlum deyil.

Əgər şəbəkə artıq ev boyu çəkilmişsə, həm enerjini ötürmək üçün (Azərbaycan dilində "şəbəkə" dedikdə həm cərəyanla təminat, həm də informasiya rabitəsi başa düşülür, ingilis dilində isə power – "güc şəbəkəsi", network isə – "informasiya şəbəkəsi" deməkdir) həm də informasiya kanalı olaraq yeganə şəbəkə kabelindən istifadə etmək olmazmı?

Elektrik xətti – informasiyanı ötürmək üçün ən mürəkkəb mühitlərdən biridir. Əvvəldən elektrik şəbəkəsi aşağı tezlikli (50 Hs) elektrik enerjisini ötürmək üçün istifadə olunur. Bu zaman elektrik enerjisinin sənayedə istifadəsi nəticəsində güclü maneə yaranır. Əgər elektrik şəbəkəsi şəhər informasiya şəbəkəsinə informasiyanın ötürülməsi üçün istifadə edilərsə, bu halda yüksək səviyyəli maneələrə və siqnalların kəsilməsinə görə problemlər yarana bilər. Kodlaşdırılan informasiya güc şəbəkəsinin tezliyi üzərinə qoyulduqda elektrik ötürülməsində enerji təchizatı azalmamalıdır. Belə sistemlər hazırlanmışdır və artıq mövcuddur, lakin onların geniş tətbiqi simsiz rabitə ilə müqayisədə iqtisadi cəhətdən səmərəli deyildir.

2004-cü ilin aprelində Böyük Britaniyanın qanunpərəst vətəndaşı olan Olli Uaythauz Bluetooth-qurğusunu sındıra bilən metodikanı nümayiş etdirdi. İki belə qurğunun köməkçi avadanlıqla birləşməsi zamanı haker partnyorlar arasında mübadilə olunan identifikasiya olunmuş informasiyanı ələ keçirmək imkanı qazanır. Qurğularda birinci birləşmə zamanı eyni dörd simvollu PIN-kod yığılmalıdır. Xüsusi Bluetooth alqoritmlərindən istifadə etməklə, təqribən 10 min variant yığmaqla haker həmin PIN-kodu tapır. Daha sonra o, həmin PIN-kodu xüsusi 128-bitlik gizli açarın generasiyası üçün istifadə edərək, faktiki olaraq həmin "gizli" açarı ələ keçirir. Beləliklə, bu iki qurğu arasındakı şifrlənmiş "söhbət"in bütün qalan hissəsi haker üçün açıq kitaba dönür.

Ancaq bu açıqlama Bluetooth rabitə texnologiyasına elə də əsaslı zərər gətirmədi. Belə ki, əgər hakerdə birinci "əlaqə" alınmamışdırsa, onda qurğular arasında verilənlərin sonrakı mübadiləsində risk yaranmırdı.

Lakin Bluetooth-un quraşdırıcıları çox tez sakitləşmişdilər. 2005-ci ilin yazında Təl-Əviv universitetindən olan iki İsrail vətəndaşı Arişay Vul və Yakiv Şaked populyar naqilsiz protokolu sındırmaq üçün yeni, təkmilləşdirilmiş və "gözəl" üsul təklif etdilər. "Bizim təklif etdiyimiz hücum üsulu iki Bluetooth-qurğusu arasında verilənlərin mübadiləsinin ixtiyari seansını, hətta bu onlar arasında birinci əlaqə olmadıqda belə, sındırmağa imkan verir" deyərək Yaniv Şaked izah verirdi.

Onlar Bluetooth-qurğularının birinci birləşməsi prosedurunu süni olaraq nəzərə almağa nail oldular. Qulaq asan qurğular özlərini başqası kimi "təqdim edərək", "şikara" guya ki, açarın itməsi barədə yanlış məlumat göndərərək kontakta "müdaxilə" edirdilər. "Ağıllı" qurğu protokola müvafiq köhnə açarı ləğv edir və yenidən birinci birləşmə proseduruna başlayırdı. Sonra haker, Olli Uyathauzun yuxarıda təsvir edilmiş sxemi üzrə hərəkət edirdi.

Açarın "itməsi" barədə yalan məlumatın göndərilməsi üçün hakerə ancaq bütün Bluetooth-qurğularının fəaliyyət zonaları çərçivəsində avtomatik olaraq necə gəldi payladıqları yad identifikator lazım idi.

Pentium IV sinfindən olan kompüterdə kontakt qurğulardan sonra 128-bitlik gizli açarın hesablanması proseduru cəmi 0,06 saniyə təşkil edir.

"Bu – nəinki nəzəriyyədə, həm də praktikada fəlakətdir", – deyə, Kolifornıyanın Mountin-Vyu şəhərindən olan sistemlərin təhlükəsizliyi üzrə ekspert Bryus Şnayer öz təəssüratlarında qeyd edirdi. Bu aşkarlamadan sonra bütün Bluetooth-qurğuları faktiki olaraq tamamilə yoxa çıxdılar.

Bluetooth-qurğusunun təsirinin hətta kiçik radiuslu (10 m-lik) olması belə onu hakerlərdən xilas edə bilmədi. Kiçik məsafələr üçün nəzərdə tutulmuş antenaların köməyilə hakerlər "uzaqdan sındırmanı" öyrəndilər, belə sındırma üzrə dünya rekordu 2005-ci ildə 1 km-i aşırdı.

BU NƏ VAXT QURTARACAQ?

Köhnə rabitə texnologiyalarının birləşmə səbəblərindən biri – simsiz rabitə və kommunikasiya (mobil telefon, peyk rabitəsi) bazarının sürətlə artmasıdır.

Yaxın vaxtlarda simsiz cihazların sayı o qədər artacaq ki, verilmiş diapazonda sonlu sayda kanalların necə yerləşdirilməsi sualı yenidən ortaya çıxacaqdır. Belə ki, əgər istifadə olunan tezlik resursları tükənərsə, onda tezlik pillələrilə daha yüksəklərə qalxmaq lazım gələcəkdir...

Radio rabitəsinin elementar problemləri (elektromaqnit enerjisinin yağış





Divar telefonu. 1922-ci il.

Telefon-güzgü.

zamanı və ya rütubətlik artan zaman tez sönməsi) ilə yanaşı olaraq digər problemlər də yarana bilər: daha yüksək tezlik insan orqanizmi üçün zərərli ola bilər.

Onda insanlar köhnə əziz simləri yada salacaqlar.

SUSMAQ – QIZILDIR (VERİLƏNLƏRİN ÖTÜRÜLMƏSİNİN YENİ ÜFÜQLƏRİ)

Dünən, hətta arzuedilməz olan – bu gün gerçəkliyə çevrilib. CDMA standartlı mobil telefon xidməti təklif edən "Sonet" kimi möhtəşəm şirkət XXI əsrin ilk illərindən öz fəaliyyətini faktiki olaraq dayandırdı. Bu perspektivli texnologiya bazardan GSM-operatorları tərəfindən sıxışdırılırdı. Onun yerinə isə sehirli Feniks quşu kimi CDMA-nın əvvəlki standartını inkişaf etdirərək eyni işləmə prinsiplərinə əsaslanan CDMA-2000 kimi daha mükəmməl texnologiya yarandı. Yeni texnologiya nəinki verilənlərin 150 Kbit/san ilə ötürülməsinə imkan verirdi, həm də bu sürətin bir neçə dəfə artacağı barədə sevindirici gələcəkdən xəbər verirdi. CDMA sistemlərində siqnalların ötürülməsi üçün 1,25 MHs tezlikli radiokanallardan istifadə olunur, sürət göstəricilərinin artırılması isə bir neçə (üç və daha artıq) radiokanaldan paralel istifadə olunması hesabına əldə oluna bilər.

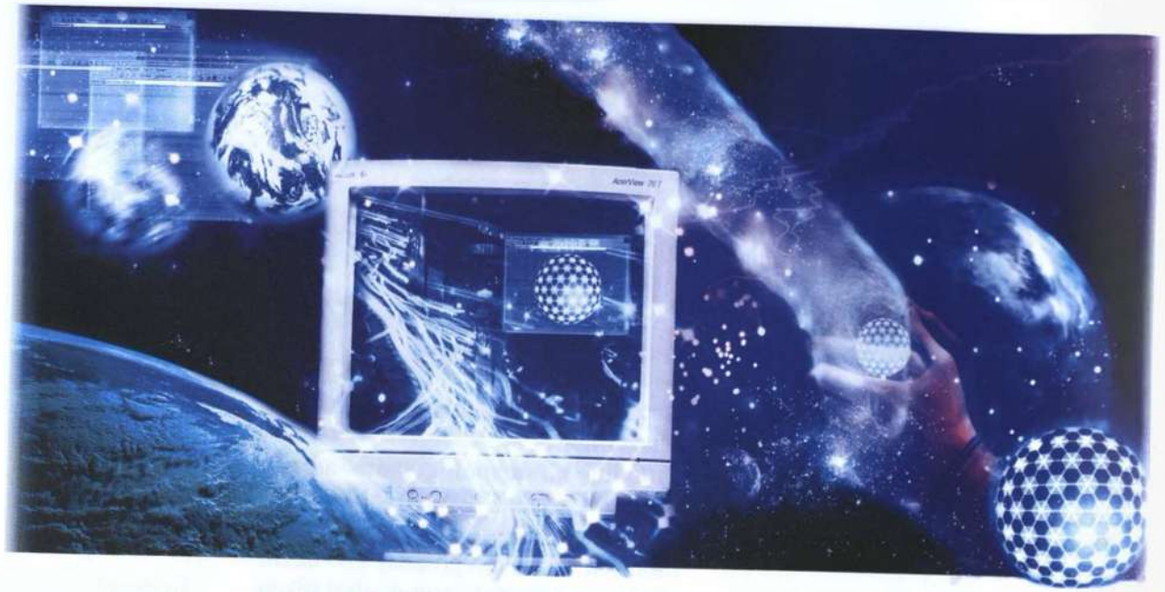
Odur ki, texnologiyanın adında "Multi-Carrier" – "çoxtezlikli" söz birləşməsindən istifadə olunur. Lakin həmişəki kimi, elə görünür ki, "daxili ehtiyatları" istifadə etmək olar: təkcə siqnalların kodlaşdırılması və modulyasiyası sisteminin təkmilləşdirilməsi hesabına verilənlərin ötürülməsinin arzu olunan sürəti 153,6 Kbit/san-yə çatdırılmışdır. Bu fakt öz əksini 1x – bir radiokanal adında əks etdirir.

Üç radiokanaldan istifadə olunduqda isə verilənlərin ötürülməsinin 2,4 Mbit/san (IMT-MC-1x-EV-DO) və IMT-MC-1x-EV-DV standartında 5,2 Mbit/san-yə kimi maksimal sürətinə nail olmaq olar.

Ancaq ilk terminalların (İnternetə yüksək sürətlə daxil olan telefonları indi hörmət əlamoti olaraq belə adlandırırlar) real olaraq testdən keçirilməsi zamanı verilənlərin sürətinin 25 Kbayt/san ətrafında olduğu müəyyənləşdirilmişdir ki, bu da elan olunmuş sürətdən 15 dəfədən də xeyli aşağıdır və təqribən GPRS-in inkişafının analoji təkamülü olan EDGE-yə uyğundur. Güman ki, uğurlu nəticənin alınmasına mobil rabitə radiokanallarını amansızlıqla istismar edən abonentlərin danışıqları mane olur. Bu anda məşhur atalar sözü yerinə düşür: "Susmaq – qızıldır!".

Gələcəyin texnologiyası EV-DO (*ing.* Evolution – Data Only; yalnız verilənlərin təkamülü) adlanır, formal olaraq isə IMT-MC-1x-EV-DO yazılır.

IMT-MC abbreviaturası Beynəlxalq Elektrorabitə institutu (ITU) tərəfindən hazırlanan mobil rabitənin standartlar qrupu mənasını verir (*ing.* International Mobile Telecommunications Multi-Carrier).



İNTERNET

İNTERNETİN YARANMASI, QURULUŞU VƏ İNKİŞAFI

20 oktyabr 1969-cu ildə Los-Ancelesdə Kaliforniya Universitetinin professoru Leonard Kleynrok öz kompüterindən Stanford tədqiqat mərkəzindəki kompüterlə əlaqə yaratmağa və tarixdə birinci elektron məktubu göndərməyə cəhd göstərdi. Əslində bu heç məktub da deyildi. Kleynrok Stanforddakı kompüterə LOG komandasını ötürmək istəyirdi. Yalnız iki hərfi göndərə bildi, G hərfi ötürülən zaman sistemdə nasazlıq yarandı. İki kompüter arasında əlaqə yaratmaq cəhdi zəncirvari proseslərin başlanğıcı oldu və gələcəkdə İnternet adlanan qlobal kompüter şəbəkəsinin yaranmasına gətirib çıxartdı. Belə ki, indi 20 oktyabr İnternetin yaranma tarixi hesab olunur.



Leonard Kleynrok.

İNTERNETİN YARANMA TARİXİ

Bəzən tarixdə elə hadisələr baş verir ki, kəşflər və yeniliklər sonralar əvvəlcədən təyin olunmuş məqsədlər üçün istifadə olunmur. Konyakın yaranmasını buna misal gətirmək olar. Holland tacirləri gömrük vergisini azaltmaq məqsədilə Fransadan şərabı göndərməzdən əvvəl distillə edib, həcmini kiçildirdilər, sonra öz ölkələrində yenidən şərabı su ilə qarışdırırdılar. Yalnız bir neçə ildən sonra fransızlar palıd ağacından hazırlanmış çənlərdə saxlanılan şərab spirtinin yüksək keyfiyyətli olmasını müəyyən etdilər. İnternetlə də buna oxşar hadisə baş verdi.

İkinci dünya müharibəsindən sonra ABŞ ilə keçmiş müttəfiqi Sovet İttifaqı arasında qarşıdurma yarandı. 50-ci illərin ortalarında hər iki tərəf nüvə və hətta istilik-nüvə silahına sahib olmuşdular və artıq bu silah üçün raket daşıyıcılarını işləyib hazırlayırdılar. Bu qarşıdurmanın gedişində ABŞ prezidenti Eyzenhauzerin sərəncamı ilə Müdafiə Nazirliyinin nəzdində Qabaqcıl Tədqiqatlar Agentliyi (Advanced Research Projects Agency – ARPA) yaradıldı.

Agentlik öz tədqiqatlarını kompüter texnologiyalarının hərbi sahədə tətbiqinə yönəltdi. Tədqiqatların əsas məqsədi elə kompüter şəbəkəsi yaratmaq idi ki, həmin şəbəkənin ayrı-ayrı sahələri zədələrə, məsələn düşmənin raketnüvə hücumuna qarşı dayanıqlı olsun. Bu cür şəbəkə qurmaq üçün 1967-ci ildə ARPANET (Advanced Research Projects Agency NETwork) adlanan layihə hazırlandı. 1969-cu ildə isə Yuta ştatında olan universitetin, Los-Anceles və Santa-Barbarada Kaliforniya universitetlərinin, Stanford tədqiqat mərkəzinin kompüterləri ilk dəfə olaraq şəbəkəyə qoşuldu. 1971-ci ildə şəbəkəyə artıq 23 kompüter qoşulmuşdu. 1973-cü ildə ilk beynəlxalq qoşulmalar həyata keçirildi.

ARPANET şəbəkəsi inkişaf etdikcə kompüterlərarası informasiya ötürülməsinin müxtəlif metodlarından istifadə edən ayrı-ayrı şəbəkələrin birləşdirilməsi problemi meydana çıxdı. Bunun üçün fərqli kompüter platformalarında işləyən müxtəlif proqramlar arasında informasiya mübadiləsini təmin edən qaydalar, yəni xüsusi protokol işlənib hazırlanmalı idi. 1974-cü ildə belə protokol yaradıldı. Bu protokol şəbəkədə qarşılıqlı əlaqənin təmin edilməsi, informasiyanın emalı və ötürülməsi üçün nəzərdə tutulmuş qaydalar və göstərişlər toplusu idi. Protokol TCP/IP (Transmission Control Protocol / Internet Protocol) adlandırıldı. TCP – informasiyanın ötürülməsinə nəzarət edən protokoldur. IP-informasiyanın rabitə kanalı ilə ötürülməsi zamanı paketləri marşrutlayır.

80-ci illərin ortalarında şəbəkəyə artıq minə qədər kompüter qoşulmuşdu və TCP/IP standart kimi qəbul edildi. Eyni zamanda UNIX əməliyyat sistemli masaüstü işçi stansiyalar yaradıldı və onlar İnternet adlanan şəbəkəyə qoşulmaq üçün quraşdırılmış vasitələrlə təchiz edildi.

1986-cı ildə ABŞ Milli Elm Fondu (National Science Foundation – NSF) öz altı superkompüter mərkəzlərini birləşdirmək üçün şəbəkə yaratdı. Şəbəkə TCP/IP protokolu əsasında qurulmuşdu. İnformasiyanın kanalla ötürülmə sürəti 56 kbit/s idi. Sonralar şəbəkəyə universitetlər də qoşulmağa başladı.

1989-cu ildə İnternetə qoşulan kompüterlərin-hostların sayı 100 mindən çox idi. İnternetdən yalnız dövlət və elmi tədqiqat məqsədləri üçün deyil, eyni zamanda kommersiya məqsədləri üçün də istifadə edilməyə başlandı. İnternetə MCI Mail kommersiya şəbəkəsi qoşuldu. Bu müddətdə şəbəkə artıq Birləşmiş Ştatların sərhədlərini aşaraq, Avstraliya, Böyük Britaniya, İsrail, Almaniya, İtaliya, Niderland, Yeni Zelandiya və Yaponiya ölkələrinə də gəlib çıxdı. 90-cı illərin əvvəllərində Rusiya və Şərqi Avropa ölkələri də İnternetə qoşuldu. 1991-ci ildə Azərbaycanda ilk dəfə olaraq beynəlxalq şəbəkəyə bağlantı həyata keçirildi. Artıq İnterneti Ümumdünya şəbəkəsi adlandırmaq olardı.

1991-ci ildə Cenevrədə Avropa Nüvə Tədqiqatları mərkəzinin elementar zərrəciklər fizikası laboratoriyasının işçisi Tim Berners-Li World Wide Web (WWW) sistemini işləyib hazırladı. Beləliklə də, İnternet tarixində yeni səhifə başlandı.



Tim Berners-Li.

İNTERNETİ KİM İDARƏ EDİR

İnternet bir-birilə əlaqədə olan magistral şəbəkələrdən, başqa sözlə, dayaq şəbəkələrdən (back bone) ibarətdir. Orta səviyyəli regional şəbəkələr yüksək sürətli dayaq şəbəkəyə birləşdirilir. Hər bir şəbəkə, öz daxilində dövr edən trafiki (informasiya həcmini) idarə edir və düzgün hesab etdiyi kimi marşrutlayır (istiqamətləndirir). Trafik rabitə kanalları ilə ötürülən informasiyanın həcmidir, yəni rabitə xəttinin iş yüküdür. Hər bir

şəbəkə, yüksək səviyyəli şəbəkələrə qoşularkən məsuliyyət daşıyır, maliyyələşdirməyə görə cavabdehdir və öz inzibati prosedurlarını təyin edir. Beləliklə, vahid şəbəkə xidməti təchizatçısı mövcud deyil. Lakin bütün şəbəkələr İnternetin bərabər hüquqlu hissələri hesab olunur.

İnternetə bütövlükdə hər hansı dövlət müəssisələri, maliyyə-sənaye qrupları və ya şirkətlər nəzarət etmir. O, vahid sistem kimi fəaliyyət göstərir və beynəlxalq ictimaiyyətin birgə səylərilə inkişaf edir. Lakin vahid standartların saxlanması, İnternetin bir hissəsini təşkil edən regional və milli şəbəkələrin fəaliyyətlərinin uyğunlaşdırılması, yeni texnologiyaların işlənib hazırlamasında səylərin cəmləşdirilməsi və şəbəkə cəmiyyəti üçün mühüm olan digər məsələlərin həll edilməsindən ötrü bir sıra təşkilatlar yaradılıb.

Dünya şəbəkəsinin inkişafını İnternet cəmiyyəti ISOC (Internet Society) idarə edir. ISOC beynəlxalq ictimai təşkilatdır. Əsas məqsədi İnternet vasitəsilə qlobal informasiya mübadiləsinə yardım etməkdir. ISOC texniki idarəetməyə cavabdeh olan İnternet arxitekturası üzrə komissiyanı (Internet Architecture Board – IAB) təyin edir. IAB İnternet protokollarının tədqiqatına və inkişafına nəzarət edir, şəbəkə ünvanlarının verilməsi qaydalarını təyin edir. IAB çərçivəsində mütəxəssislərdən ibarət bir neçə işçi qrup mövcuddur. Onlardan biri – mühəndislər qrupu (Internet Engineering Task Forse – IETF) İnternet arxitekturası və protokolları üçün standartların işlənib hazırlanmasına cavabdehdir. Digər tədqiqatçılar qrupu (Internet Research Task Forse – IRTF) işini texnologiyaların gələcək inkişafı üzərində cəmləşdirir. Fövqəladə hallar üzrə şöbə (Computer Emergency Response Team – CERT) şəbəkədə təhlükəsizlik məsələləri üzrə ixtisaslaşır. O, informasiyanın qorunması istiqamətində görülən işlərə nəzarət edir. İnternetin ünvan fəzasınının bölünməsi sisteminə ICANN (The Internet Corporation for Assigned Names and Numbers) beynəlxalq təşkilatı cavabdehdir.



İNTERNETİN QURULUŞU

İnternetin necə işləməsini təsəvvür etmək istədikdə, əvvəlcə onun analoqu olan telefon şəbəkəsi yada düşür. Hər ikisi elektron rabitə vasitələridir, hər iki halda birləşmə baş verir və informasiya ötürülür. Çoxları İnternetə məhz telefonla qoşulur. Əsaslı fərq şəbəkədən istifadə prinsiplərindədir.

Telefon şəbəkəsi – kanalların kommutasiyası şəbəkəsidir. İki abunəçi arasında əlaqə yarandığı zaman kanal ayrılır və hətta heç bir informasiya ötürülmədiyi halda, məsələn, hər iki həmsöhbət susduqda belə, başqa heç kim ondan istifadə edə bilmir. İnternet paketlər kommutasiyası şəbəkəsidir və əlaqə kanallarından daha effektiv istifadə edir. İnformasiya hissələrə – paketlərə bölünərək, məktubun poçt ilə göndərildiyi kimi ötürülür. Bu zaman kanal eyni bir vaxtda bir neçə birləşmə üçün istifadə olunur. İnternetin müxtəlif hissələri "marşrutlayıcılar" – kompüterlər sistemi vasitəsilə birləşir. Marşrutlayıcılar paketlərin başlıqlarını araşdırır, onların təyinat məntəqəsinə çatdırılması üçün hansı qovşağa ötürüləcəyini müəyyənləşdirir, həmçinin şəbəkədə birləşmə pozulduğu halda, alternativ marşrut seçir.

Bu təxminən adi məktubların ötürülməsinə bənzəyir. Məktub zərfə qoyulur, üzərində məktub alanın və göndərənin ünvanı qeyd edilir və marka yapışdırılır. Poçt xidməti məktubun çatdırılması zamanı müəyyən olunmuş qaydalara riayət edir. Eyni qayda ilə, İnternetin işini nizamlayan qaydalar (protokollar) da mövcuddur.

Ünvanlaşdırmaya və paketlərin çatdırılmasına IP (Internet protocol) cavabdehdir. IP-yə görə verilənlər paketinin ölçüsü 1 baytdan 1500 bayta qədər ola bilər. Paketə təyinat məntəqəsi haqqında zəruri məlumatlar əlavə edilir. İnternetə qoşulan hər bir kompüterə unikal ünvan – IP-ünvan mənimsədilir. O, bir-birilə nöqtə ilə ayrılan dörd ədəddən ibarətdir. Ədədlərin qiyməti 0 ilə 255 aralığında dəyişir. Məsələn 62.3.250.189 və ya 193.20.1.147.

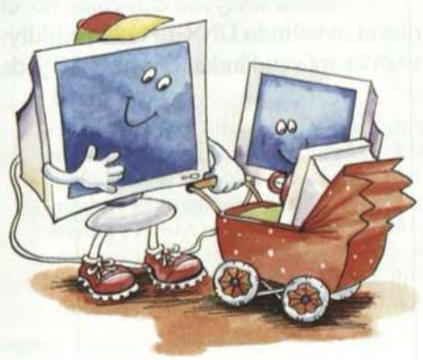
Bununla belə IP protokolu paketin itkisiz çatdırılmasına zəmanət vermir. O, həmçinin paketlərin ardıcıl çatdırılmasını yerinə yetirməyə təminat vermir. Əksər hallarda şəbəkə ilə göndərilən informasiyanın həcmi 1500 baytdan çox olur. Bu problemi verilənlərin ötürülməsinə nəzarət protokolu – TCP (Transmission Control Protokol) həll edir. TCP protokolunun proqram təminatı göndərilən tərəfdə informasiyanı paket adlanan hissələrə bölür və hər birinə sıra nömrəsi mənimsədir. TCP təyinat məntəqəsində paketləri yığır və onları nömrələnməyə uyğun olaraq düzür. Hər hansı hissənin çatışmadığı halda protokol təkrar ötürülmə tələb edir. Bütün informasiya düzgün qaydada yerləşdirildikdən sonra bu məlumatlar tətbiqi proqrama ötürülür. Bundan başqa TCP xüsusi düsturla paketin nəzarət cəmini, yəni bütün baytlarının cəmini yoxlayır. Əgər nəzarət cəmi uyğun gəlmirsə, deməli, paket təhrif olunmuş formada göndərilmişdir və təkrar göndərilmə tələb olunacaq.



TCP/IP protokolunun istənilən platformada işinin etibarlığı, əlverişliliyi və sadəliyi onun müvəffəqiyyətini müəyyən edir. Əgər əvvəllər TCP/IP yalnız UNIX əməliyyat sistemi və İnternet üçün nəzərdə tutulmuş protokol hesab olunurdusa, sonralar o daha geniş tətbiqini tapdı. TCP/IP lokal şəbəkələrdə də əsas protokol oldu.

DOMENLƏR – İNTERNETDƏ ADLARDIR

Yalnız rəqəmli ünvanlardan istifadə etməklə, İnternet bu cür rahat informasiya mühitinə çevrilməzdi. Rəqəmli ünvanlar bilavasitə kompüterlərin qarşılıqlı əlaqəsi üçün yaxşıdır, lakin insanlar adlara üstünlük verirlər. Bu problem hələ İnternetin formalaşdığı illərdə yaranmışdı. O zaman şəbəkəyə daxil olan bütün kompüterlərə xüsusi adlar mənimsədilirdi. Hər biri unikal olmaq şərtilə ad kimi sadə sözlər seçilirdi. Şəbəkəyə qoşulan kompüter Şəbəkə İnformasiya Mərkəzində (Network Information Centre – NIC) qeydiyyatdan keçirdi. Bütün kompüterlər





haqqında məlumatlar xüsusi fayla yazılırdı və bu fayl şəbəkə ilə müntəzəm olaraq göndərilirdi. Hər hansı kompüterə adı ilə müraciət etdikdə, sistem bu siyahıya baxırdı və IP-ünvanı hesablayıb tapırdı. Tezliklə İnternet sürətlə genişləndi və ad siyahısı olan faylın həcmi o qədər böyüdü ki, onu operativ göndərmək mümkün olmurdu: unikal adların qeydiyyatı çətinləşmişdi, axtarış sürəti azalmışdı. Bu problemin həlli üçün indiyə kimi istifadə olunan domen ad sistemi (Domain Name System – DNS) yaradıldı.

DNS-in mahiyyəti ondan ibarətdir ki, ad mənimsədilməsi məsuliyyəti İnternetin bu və ya digər sahəsinə – domenə (domain – "sahə") nəzarət edən müxtəlif istifadəçilər qrupunun üzərinə qoyulur.

Domen adı, rəqəmsal addan fərqli olaraq soldan sağa araşdırılır. Adətən ən birinci IP-ünvanı olan kompüterin adı, sonra kompüterin daxil olduğu domenin adı, ondan sonra daha böyük domen adı və s. dayanır. Əgər domen sahibi şəbəkəyə yeni kompüter qoşmaq istəsə, heç kimdən icazə almalı deyil. Ondan yalnız müvafiq qeydi öz domeninin ad serverində yerləşən verilənlər bazasına düzgün daxil etmək tələb olunur.

Ən yüksək səviyyəli domenlər, 80-ci illərin əvvəlində DNS-in tətbiq edildiyi vaxtdan mövcuddurlar. Əvvəlcə ABŞ-da *gov* (dövlət idarələri), *mil* (hərbi idarələr), *com* (kommersiya təşkilatları), *edu* (təhsil müəssisələri), *net* (şəbəkə təşkilatları), *org* (digər təşkilatlar) domenləri yaradıldı. Bu kodlaşdırma XXI əsrə də keçdi. Başqa ölkələrin şəbəkələri İnternetə qoşulduqda, onların şəxsi domenləri öz ixtiyarlarına verildi. Beynəlxalq standartlara uyğun olaraq dövlətlərin iki hərfli kodlaşdırılması qəbul edildi. Sonra *int* (Beynəlxalq təşkilatlar) domeni meydana gəldi.

2000-ci ildə ICANN yeddi yeni yüksək səviyyəli domenin yaradılmasını qərara aldı. Hava xətlərli, hava limanları və biletlərin sifariş sistemi üçün *aero* domeni nəzərdə tutuldu, *name* şəxsi istifadəçilər üçün, *biz* kommersiya təşkilatlarının saytları üçün, *coop* işguzar ittifaqlar üçün, *info* informasiya layihələri üçün, *museum* muzeylər üçün təyin olundu. NATO Şimali Atlantika bloku üçün isə *nato* domeni müəyyən edildi. Avropa Birliyi öz şəbəkə layihələri üçün *eu* domenindən istifadə etməyi planlaşdırır.

İnternetə qoşulan kompüterlərdə işləyən bütün proqramlar domen ad sistemindən istifadə edir. Hər dəfə bu və ya digər kompüterə adı ilə müraciət olunduqda, müraciətin IP-ünvanına çevrilməsi baş verir. Əgər həmin ünvan verilənlər bazasında varsa və ya bu ünvana əvvəllər müraciət olunmuşsa və o keş yaddaşda saxlanılıbsa, kompüter ünvanı tanıyacaq. Əgər ünvan kompüterdə yoxdursa, öz domeninin adlar serverindən müvafiq domenin adlar serverinin ünvanını soruşur, o da məlum olmadıqda daha yüksək səviyyəli domenin adlar serverindən ünvanı soruşur və i.a.

Domen ad sistemilə işləyərkən yadda saxlamaq lazımdır ki, domen adın hissələri yalnız onun kimə məxsus olduğunu göstərir. Bu halda kompüterin fiziki olaraq harada yerləşməsini həmişə asan təyin etmək mümkün olmur. Həm də bir domenin kompüterləri bəzən müxtəlif

Adlarda domenləri bir-birindən nöqtə işarəsi ayırır, məsələn, azerbaijan.az, burleson.rtp.epa.gov, www.informir.ru, abcd.abdn.ac.uk. Adda ixtiyari sayda domen ola bilər, lakin, bir qayda olaraq, beşdən artıq domenə rast gəlinmir.

www.altay.az ünvanında "altay" eyni adlı kompaniyanın şəbəkəsinə (qrupun domen adı – altay) daxil olan kompüterin adıdır (çox güman ki, bu kompüterdə www-server yerləşdirilib), az isə Azərbaycan Respublikasının milli domenininin adıdır. alpha.math.msu.su ünvanında birinci yerdə olan Moskva Dövlət Universitetinin (msu – Moscow State University), mexanika-riyaziyyat fakültəsinin (math) şəbəkəsindəki kompüterin adıdır. Bu ünvan Sovet İttifaqından miras qalan su domeninə aiddir.

şəbəkələrdə yerləşir. Buna baxmayaraq, IP-ünvanı əvəzinə domen adlarından istifadə etmək daha məqsədəuyğundur. Birincisi, istifadəçilər domen adını daha asan yadda saxlaya bilirlər. İkincisi, kompüterlərin IP-ünvanı bir sıra səbəblərdən dəyişə bilər (məsələn lokal şəbəkənin yenidən qurulması nəticəsində), adı isə demək olar ki, heç vaxt dəyişmir. Nəhayət, qəbul edilmiş addan başqa kompüterin bir neçə sinonim adı ola bilər. Adətən, belə adlar kompüterin təqdim etdiyi xidmətə müvafiq olaraq əmələ gəlir.

İNTERNETƏ İRİŞMƏ

İnternet o dərəcədə məişətə daxil olmuş və hər bir insan üçün asan əldə edilən olmuşdur ki, təcrübəsiz istifadəçi onu kompüterin bir hissəsi kimi qəbul edir. Satıcılar tez-tez alıcılar tərəfindən verilən "Bu kompüterdə İnternet varmı?", "Onun İnterneti haradadır?" kimi suallarla rastlaşırlar.

Əfsus ki, İnternet kompüterin daxilində olmur. Kompüter yalnız İnternetə çıxış üçün vasitədir.

Praktiki olaraq, bütün müasir əməliyyat sistemləri İnternetlə işləmək imkanı verir, başqa sözlə desək, onlar TCP/IP protokolunu dəstəkləyən sistem ilə təchiz olunur. Belə sistemli kompüter İnternetə qoşulmağa və onun bir hissəsi olmağa hazırdır. Bununla yanaşı kompüter modem və şəbəkə kartı ilə təchiz olunmalıdır. Bu qurğular adətən kompüterin daxilində yerləşdirilir, lakin bəzi hallarda onları ayrıca əldə etmək lazım olur.

İNTERNETƏ QOŞULMA. İRİŞMƏ NÖVLƏRİ

İnternetə çıxış provayderlər (Internet service provider – "İnternet xidməti təchizatçısı") adlanan xüsusi təşkilatlar tərəfindən təmin olunur. Provayderin kompüter şəbəkəsi İnternetə daimi qoşulu olur. Bunun üçün ayrılmış sürətli rabitə kanalı icarə olunur və ya özəl olaraq çəkilir. Provayderlərin şəbəkələri dayaq magistral şəbəkədə birləşdirilir və nəticədə İnterneti əmələ gətirirlər.

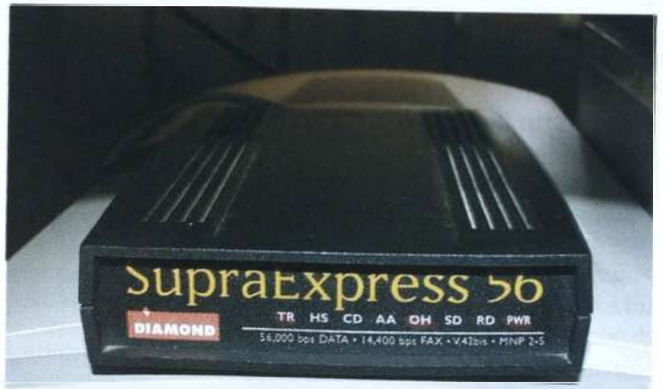
İnternetə irişmə üsulları qoşulma sxemləri, əlaqə xətləri və şəbəkənin təqdim etdiyi müxtəlif irişmə xidmətləri təyin edən protokollarla fərqlənirlər. İnternetə irişmə üsulları nə qədər çox imkanlar təqdim edərsə və nə qədər sürətli olarsa, bir o qədər baha olar.

Əgər modem V34 protokolunu dəstəkləyirsə, verilənlərin ötürülmə sürəti 28,8 Kbit/san-i aşmır. Əgər modem V34+ protokolunu dəstəkləyirsə, ötürülmə sürəti 33,6 Kbit/san-dən çox olmur. Nəhayət, əgər modem V90 protokolunu dəstəkləyirsə, onda ötürülmə sürəti 57,6 Kbit/san-yə çata bilər.

DIAL UP

XXI əsrin əvvəllərində İnternetə telefon xəttilə kommutativ çıxış (dialup) geniş yayılmışdır. Kompüter İnternetə qoşulan andan onun bir hissəsinə çevrilir və onun bütün resurslarına giriş imkanı







Xarici (*yuxarıda*) və daxili modem.

həmişə eyni olmur. Məsələ ondadır ki, kompüter şəbəkəsinin provayderlə bağlantısı müxtəlif səbəblərdən hərdən pozulur və ya tamamilə kəsilə bilər. Bir qayda olaraq, gecələr və istirahət günləri İnternetdən istifadə daha ucuz olur. Bəzən provayderlər abunə haqqı təyin edir və İnternetdən limitsiz istifadə müddətini təmin edirlər. Yüksək səviyyəli provayderlər müştərilərinə rabitənin keyfiyyətilə tanış olmaq üçün pulsuz test girişi təklif edirlər.

Modemli irişmənin bir sıra nöqsanları var. Telefon xətti öncədən rəqəmli verilənlərin ötürülməsi üçün nəzərdə tutulmadığından, onun bu məqsəd ilə istifadəsi bir sıra məhdudiyyətlərə və narahatçılığa səbəb olur. Telefon xəttilə verilənlərin ötürülmə sürəti maksimal 56 kbit/san-dir. Əslində ATS-in (avtomatik telefon stansiyası) imkanlarından və telefon məftillərinin keyfiyyətindən asılı olduğu üçün, bu sürət qat-qat aşağıdır. Bundan əlavə, İnternetə telefon xəttilə qoşulmaq üçün əvvəlcə provayderə zəng etmək lazımdır (provayderin müştərilərinin sayı modemin girişlərinin sayından çox olarsa, bu saatlarla davam edir). Keyfiyyətsiz telefon xətləri modemlərlə uzun müddət etibarlı bağlantını saxlamağa imkan vermir. Kəsilən bağlantı nəticəsində məlumat itir və onu yenidən yükləmək lazım gəlir.



qazanır. Kommutativ giriş xüsusi PPP (Point to-Point Protocol – "kompüter-kompüter əlaqə protokolu") protokolundan istifadə edir. Bu irişmə növü üçün modem (ədədi verilənləri telefon xəttinin analoq siqnallarına və əksinə çevirən xüsusi qurğu) lazımdır.

Adətən, dial up modemli irişməni təmin edən provayderlər, öz xidmətlərinin "tarif planı" adlanan müxtəlif ödəmə sxemlərini təklif edirlər. Onlar İnternetlə bağlantı vaxtına görə qiymətləri təyin edirlər. Ödəniş faktiki olaraq provayderlə telefon bağlantısının vaxtına görə hesablanır. Lakin bunlar heç də



AYRILMIŞ XƏTT VASİTƏSİLƏ BAĞLANTI

Artıq modem vasitəsilə məlumətların ötürülmə sürəti İnternetin bir sıra fəal istifadəçilərini, xüsusilə də bunun üçün əlavə pul ödəyənləri qane etmir (istifadəçi təkcə provayderin xidmətlərinə görə deyil, həm də telefon şəbəkəsinə görə, əksər halda istifadə müddətinə görə pul ödəyir). Bundan başqa telefon adi danışıqlar üçün lazım ola bilər. Beləliklə, dial up bağlantısı digər, daha mükəmməl bağlantı üsulu ilə əvəzlənir. Məsələn, böyük ötürmə sürətli ayrılmış rabitə kanalı icarə edilir (variantlardan biri kimi optik-lifli xətti göstərmək olar). Ayrılmış xəttin ötürmə qabiliyyəti, əsasən 64 kbit/s-dən aşağı olmur. Bu halda, İnternetə eyni anda bir neçə kompüteri – lokal şəbəkəni qoşmaq olar. Belə şəbəkənin hər bir kompüteri İnternetin tam hüquqlu iştirakçısına çevrilir.

Telefon xəttilə modemli qoşulmaya nisbətən ayrılmış kanal vasitəsilə İnternetə çıxışın bir sıra üstünlükləri var. Verilənlərin ötürülməsi üçün ayrılmış kanal gecə-gündüz yüksək sürətlə (məsələn, 100 Mbit/s) fasiləsiz iş üçün nəzərdə tutulub (provayderə zəng etməyə ehtiyac olmur). Bu zaman telefon xətti sərbəst qalır. Verilənlərin ötürülmə

Ayrılmış xəttdən istifadəyə görə abunə haqqı kanalın ötürmə imkanlarına görə təyin olunur. Bundan əlavə İnternet vasitəsilə yüklənən informasiyanın həcminə görə – trafikə görə də haqq alınır. Müəyyən trafikə görə məbləğ abunə haqqına daxil edilə bilər, bundan artıq istifadə edilən trafikə görə əlavə haqq verilir. Birləşmə vaxtına görə adətən haqq ödənilmir.

sürətinin yüksək olması hesabına, qəbul edilmiş informasiyanın qiyməti modem sisteminə nisbətən orta hesabla daha ucuz olur.

Provayderlər modem və ayrılmış optik-lifli xətlər vasitəsilə bağlantıdan başqa, həm də simsiz əlaqə kanalları – radio körpü vasitəsilə, rabitə peyklərindən istifadə etməklə bağlantı üsulu təklif edirlər. "Peyk" bağlantısı zamanı kompüterə parabolik peyk antenası – boşqabı və verilənlər çeviricisi (peyk televiziyasında istifadə olunanlardan) qoşulur. İnternet-provayderlə bağlı olan Yerin süni peyki verilənləri olduqca yüksək sürətdə (45 Mbit/s-yə kimi) ötürə bilər. Bu bağlantı zamanı istifadəçi sorğu və komandaları adi modem vasitəsilə göndərə bilir.

İnternetə bağlantının daha perspektivli və surətlə inkişaf edən üsullarından biri – ADSL-dir (Asymmetric Digital Subscriber Line). Bu abbreviaturanın açılışı "asimmetrik rəqəmsal abunə



RABİTƏ KANALLARI

İnternetə qoşulan kompüterlər bir-birilə rabitə kanalları vasitəsilə birləşdirilirlər. Kanallar *ötürmə qabiliyyətilə*, yəni vahid zamanda ötürülən informasiyanın miqdarı ilə xarakterizə olunurlar. Ötürmə qabiliyyətinin ölçü vahidi bit/san-dir.

Müxtəlif rabitə kanallarının ötürmə qabiliyyəti bir neçə kbit/san-dən yüzlərlə mbit/san qədər ola bilər. Kanalın ötürmə qabiliyyəti *orta* və *zəmanətli* olur. Orta ötürmə qabiliyyəti müəyyən vaxt ərzində orta hesabla ölçülür. Zəmanətli ötürmə qabiliyyəti kanalın daha əlverişli şəraitdə təmin edə biləcəyi minimal ötürmə qabiliyyətinə deyilir.

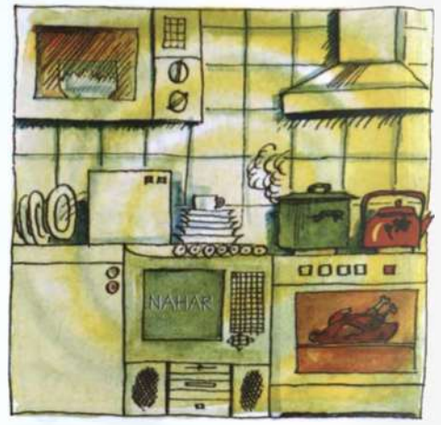
Ötürmə qabiliyyəti istənilən kanalın əsas xarakteristikası hesab olunur və kanalın istifadə üçün münasib olub-olmamasının göstəricisidir.

Ötürülən siqnalların ötürmə tipindən asılı olaraq *rəqəmli* və *analoq* kanalları fərqləndirirlər. Rəqəmli kanal giriş və çıxışda rəqəmsal (impulslu) siqnallara malik olur. Analoq kanal isə əksinə girişə fasiləsiz siqnallar daxil olur və çıxışda fasiləsiz siqnal qeydə alınır. Siqnalların parametrləri də fasiləsiz ola bilərlər və ya diskret qiymətlər ala bilərlər. Siqnallar informasiyanı ya hər bir zaman anında (zamana görə kəsilməz analoq siqnallar) və ya *müəyyən*, diskret anlarda (rəqəmli, diskret, impuls siqnallar) saxlaya bilərlər. ISDN sistemli kanallar, T1/E1 tipli kanallar və s. rəqəmsal kanallar adlanırlar. XXI əsrin əvvəllərində bir sıra üstünlüklərə malik rəqəmsal kanallar analoq kanalları sıxışdırmağa başladılar.

Analoq kanallar (məsələn, telefon xətləri) asanlıqla həyata keçirilirlər, lakin onlar maneələrə qarşı davamsızdırlar və kiçik ötürmə qabiliyyətinə malikdirlər (saniyədə onlarla kilobayt). Məlumatların ötürülməsi zamanı analoq kanalların giriş və çıxışında rəqəmli verilənləri analoq siqnallarına və tərsinə çevirən qurğu olmalıdır.

Kanallar həmçinin *kommutasiya* və *ayrılmış kanallara* bölünürlər. Kommutasiya kanalları müştərinin tələbinə əsasən ona bağlantı vaxtı verir. Adi telefon aparatları kommutasiya kanallarından istifadə edir. Bundan başqa, inteqrasiya olunmuş xidmətlərin rəqəmsal şəbəkəsi (ISDN – Integrated Services Digital Network) kommutasiya kanallarına əsaslanır. Ayrılmış kanal başqa cür işləyir: bu halda bağlantı daimidir və istənilən vaxtda məlumatları bir kompüterdən digərinə ötürmə imkanı var. Ayrılmış kanal kommutasiya kanalından yüksək ötürmə sürətinə görə (saniyədə onlarla meqabitə qədər) fərqlənir. Onun keyfiyyəti kommutasiya kanalının keyfiyyətindən yüksəkdir, çünki telefon stansiyasındakı kommutasiya aparatlarının vəziyyətindən asılı deyil.

Kanallar fiziki quruluşuna görə *elektrik simli, optik* və *simsiz* olurlar. Simli kanallar çox vaxt mürəkkəb quruluşlu elektrik kabellərindən təşkil olunurlar. Belə kanallarda məlumatların ötürülməsi elektrik impulslarının köməyilə baş verir. Optik rabitə kanallarında işıq ötürücülərindən istifadə olunur, siqnal isə lazerlərin köməyilə ötürülür. Simsiz kanallar (peyk kanallar, radio kanallar) televiziya və radioda tədbiq olunan iş prinsipilə işləyirlər.



xətti" deməkdir. Ad əvvəlcədən bu texnologiyanın əsasında istifadəçiyə və geri ötürülən verilənlərin mübadilə sürətindəki fərqin durduğunu nəzərə çarpdırır. Ötürmə sürətinin asimmetrikliyi qəsdən daxil olunur: adətən istifadəçi İnternetdən böyük həcmli informasiyanı öz kompüterinə yükləyir, əks istiqamətdə isə yalnız sorğular ya da çox kiçik həcmli informasiya axını gedir. XXI əsrin əvvəllərində ADSL 8 Mbit/san-yə qədər giriş, 1 Mbit/san çıxış sürətini təmin edir. Qoşulma, başlıqlarında xüsusi qurğu quraşdırılmış telefon xəttilə həyata keçirilir. Bu xətdə eyni vaxtda həm İnternetlə iş, həm də telefon əlaqəsi həyata keçirilə bilər.

Bu mümkündür, çünki verilənlərin ötürülməsi üçün adi insan səsindən (4 kHs) yüksək tezlik diapazonu istifadə edilir. İqtisadi cəhətdən bu sərfəlidir, çünki ayrıca kabel qoymaq lazım gəlmir, mövcud telefon xəttindən istifadə olunur. Əksər hallarda yeni tikilən binalarda İnternetə qoşulmaq üçün ayrılmış kanallar elektrik xətləri, su təchizatı və kanalizasiyada olduğu qayda ilə çəkilir. Bu, nəinki fərdi kompüterlərin, hətta digər məişət qurğularının da, məsələn, kondisioner və ya qısa dalğalı sobanın İnternetə qoşulma imkanlarının artması ilə əlaqədar olaraq daha da aktuallaşır.

HANSI KANALI SEÇMƏLİ?

Vinçesterlərin baha qiymətə və onların hər bir kompüterə qoşulmasının faydasız olduğu bir vaxtda şəbəkə əməliyyat sistemlərindən istifadə olunurdu. Bu cür sistemlər az sayda verilənlərdən (məlumatlar) və əsas tətbiqi proqramlardan birgə istifadəni təmin edərək, bir neçə fərdi kompüteri birləşdirməyə imkan verirdi. Əməliyyat sisteminin yüklənməsi prosesi mərkəzi maşınla həyata keçirilirdi. Terminallar bir maşına qoşularkən şəbəkə əməliyyat sistemi kollektiv istifadə mərkəzlərini xatırladırdı. Terminallar qismində fərdi kompüterlər çıxış edirdi. Terminalların "intellekt"i də əhəmiyyətli dərəcədə yüksək idi, çünki proqramlar mərkəzi maşında deyil, istifadəçinin kompüterində yerinə yetirilirdi.

Şəbəkə əməliyyat sistemi daha uğurla universitetlərdə təlim-tədris zamanı tətbiq edilirdi. Belə ki, bir tərəfdən tələbəyə elə də böyük kompüter resursu lazım deyil, digər tərəfdən isə şəbəkə əməliyyat sistemini idarə etmək asandır. Bu gün şəbəkə əməliyyat sistemi bütün əməliyyatları yerinə yetirməyi – köhnə fayl sistemlərilə işləməyi və eyni vaxtda müştəri-server tətbiqi proqramını emal etməyi bacarmalıdır.

Ən geniş yayılmış şəbəkə əməliyyat sistemi Novell Netware-dir.

KLASSİK İNTERNET XİDMƏTLƏRİ

Kütləvi şüurda İnternet barədə təsəvvürlər "World Wide Web" ümumdünya hörümçək torunun səhifələrilə sıx bağlıdır. Lakin bir zamanlar vəziyyət tamamilə fərqli idi.

İnternet ilkin olaraq ABŞ Silahlı Qüvvələri və Kəşfiyyat Xidmətinin məlumat-komanda sisteminin bir komponenti kimi yaradılmışdı. Sonradan isə o, mülki alimlərin və kompüter şəbəkələri üzrə mütəxəssislərin əlində bir alətə çevrildi. Qeyd etmək lazımdır ki, o zaman bu alət indiki kimi rahat deyildi.

Yeri gəlmişkən, hazırda klassik adlandırılan xidmətlər paketi də (uzaqdan irişim, faylların ötürülməsi və elektron poçt) hələ XX əsrin 70-ci illərində işlənib hazırlanmışdır.

UZAQDAN İRİŞİM

Bir vaxtlar kompüterlər çox böyük ölçülərə malik idilər və onlar xüsusi maşın zallarında yerləşdirilirdi. Kompüterdə işləməyə imkan verən terminallar – displey və klaviatura isə adətən başqa bir yerdə yerləşdirilirdi. Displeylər hərf-rəqəm rejimində işlədiyindən kompüterlə dialoq simvollardan ibarət komandaları daxil etməklə həyata keçirilirdi ki, buna cavab olaraq maşın müvafiq

məlumatları ekrana çıxarırdı. Uzaqdan irişim sistemi yaradılarkən kompüterlə bu cür dialoq üsulu saxlanılmışdır.

Uzaqdan irişim proqramı TELNET (teletype network) adlanır. O, sınaqdan



keçirilərək geniş yayılmışdır. TELNET istifadəçiyə məsafədən asılı olmayaraq şəbəkəyə qoşulmuş istənilən kompüterə daxil olmaq və məsafədən terminalla işləyərək kataloqları dəyişmək, faylları nəzərdən keçirmək, proqramları işə salmaq imkanını verir.

TELNET bir-birilə qarşılıqlı əlaqədə olan iki komponentdən – müştəri və serverdən ibarətdir.

Müştəri-proqramı bağlantı üçün sorğunu göndərən kompüterdə yerinə yetirilir. O, tələb olunan kompüterlə şəbəkə bağlantısını yaratmalı, istifadəçidən giriş verilənlərini qəbul etməli, bu verilənləri standart formata gətirərək serverə göndərməli, sonra serverdən çıxış verilənlərini standart formatda qəbul etməli, alınmış çıxış verilənlərini istifadəçinin displeyində əks etdirmək üçün yenidən formatlaşdırmalıdır.

Server-proqramı fon rejimində tələb olunan xidməti həyata keçirən kompüterdə yerinə yetirilir. O, öz xidmətləri üçün sorğu gözləyir və sorğular daxil olduqca onları emal edərək nəticələri standart formada müştəri-proqramına

TELNET PROQRAMININ İŞİ

İstifadəçi uzaq məsafədə yerləşən kompüterə girmək üçün aşağıdakı komandanı daxil edir.

telnet *kompüterin_adı*

Bağlantı yaradıldıqda uzaqdakı sistem giriş üçün adı (Login:) və parolu (Password:) soruşur. İş seansının bitməsi və uzaqdakı sistem əlaqənin kəsilməsi üçün istifadəçi logout komandasını daxil etməlidir. Bununla da bağlantı kəsiləcək.

TELNET proqramı məsafədə yerləşən sistemlə işləmə rejimindən komanda rejiminə keçmək imkanı verir. Bu, bağlantı yaradıldıqda escape-ardıcıllıq (Esc klavişi deyil!) daxil edilərkən baş verir. Adətən bu ardıcıllıq Ctrl və Esc klavişlərinin birgə kombinasiyasıdır. Bundan başqa komanda rejiminə kompüterin adını göstərmədən telnet komandasını daxil etməklə keçmək olar.

TELNET-in əsas komandaları (məhz KOMANDA rejiminin!)

? – komandalarının siyahısını göstər;
close – hazırkı bağlantını kəs;
quit – TELNET proqramından çıx;
open *ad* – göstərilən kompüterlə bağlantı yarat;
set – bağlantının müxtəlif parametrlərini təyin et. Məsələn, set echo komandası lokal əks-sədanı, yəni komandalara cavab olaraq məlumatların verilməsini aktivləşdirir və ya dayandırır, set escape *simvol* göstərilən simvolu escape-simvolu kimi elan edir. Escape-simvolunun əvəz edilməsi bir neçə uzaq kompüterlə bağlantı qurmaq üçün səmərəlidir. Məsələn, məsafədə yerləşən bir sistemdən digərinə daxil olduqda və susqunluqla istifadə olunan Ctrl-| klavişini basdıqda tələb olunmadığı halda belə hər iki sistemlə bağlantı kəsiləcək;
display – seansın iş parametrlərini göstər;
status – bağlantının vəziyyəti barədə məlumatları göstər;
z – məsafədə yerləşən sistemlə bağlantını fon rejiminə keçir.

Enter klavişini basdıqda komanda rejimindən çıxış baş verir.

(Məsafədə yerləşən sistemlə iş rejimində siz bilavasitə uzaqda yerləşən kompüterdə işləyirsiniz, komanda rejimi isə sadəcə rabitə seansının müxtəlif parametrlərini dəyişir.)

ötürür. Əgər server-proqramı aktivləşdirilməyibsə, uzaqdan irişim xidməti qeyri-mümkün olacaq.

Server-proqramı və müştəri-proqramı informasiya mübadiləsini müəyyən qaydalar toplusu (protokol) əsasında həyata keçirir. Protokola dəqiq riayət olunması statusundan asılı olmayaraq, kompüterlər arasında bağlantı yaradaraq əlaqə seansı keçirməyə imkan verir. Bu həm fərdi kompüter, həm işçi stansiya və ya server ola bilər. Aralarında bağlantı yaradılan kompüterlərin əməliyyat sistemləri də müxtəlif ola bilər.

İnternetin baza xidmətlərindən biri kimi uzaqdan irişim bədniyyətli istifadəçilərin hədəfindən yayına bilməzdi. Məlum oldu ki, istifadəçinin məlumatları TELNET protokolu ilə "açıq mətn" şəklində ötürülür və onları "dinləyərək" gizlicə ələ keçirmək olar. XX əsrin sonunda bu problemin həlli məqsədilə uzaqdan irişim üçün kriptoqrafik mühafizəyə malik protokol və onun əsasında SSH (Secure Shell) proqramı işlənib hazırlandı.

FAYLLARIN ÖTÜRÜLMƏSİ

Uzaqdan irişim sistemi olan TELNET-də işləmək üçün proqram doğrudan da bir çox imkanlara malikdir, lakin o, faylları bir kompüterdən digərinə köçürmək imkanını vermir. Bunun üçün adını eyni adlı tətbiqi protokoldan (File Transfer Protocol – "faylların ötürülməsi protokolu") götürmüş FTP proqramı mövcuddur. FTP fayllardakı verilənləri həm ikilik informasiya, həm də mətn şəklində göndərmək, bir direktoriyadan digərinə keçmək, bu direktoriya və faylların içərisindəki verilənləri nəzərdən keçirmək, faylları həm təktək, həm də qrup şəklində, direktoriya-

FTP PROQRAMININ İŞİ

FTP protokolu ilə məsafədə yerləşən kompüterlə bağlantı qurmaq üçün aşağıdakı komandanı daxil etmək lazımdır:

ftp *kompüterin_adı*

Bu zaman göstərilən kompüterlə bağlantı qurulur. Bundan sonra məsafədə yerləşən kompüterdə istifadəçinin adını və parolu daxil etmək lazımdır.

FTP-nin əsas komandaları:

quit – FTP-dən çıx;
dir *fayl* – məsafədə yerləşən kompüterdə kataloqun məzmununu nəzərdən keçir. Burada universal simvollardan (* və ya ?) istifadə etməklə, parametr qismində faylın və ya fayl qrupunun adını göstərmək olar;
cd *kataloq* – məsafədə yerləşən sistemdə cari kataloqu dəyiş. Kataloqların adları bir-birindən əyik xətlə / ayrılır, əgər kataloqun adı əyik xətlə başlayırsa, o zaman mövqeləndirmə əsas kataloqa nəzərən aparılır. Bir səviyyə yuxarı keçmək üçün cd.. və ya cdup komandası daxil edilməlidir;
binary – faylların ikilik ötürülmə rejimini aktivləşdir. Bu zaman bütün baytlar heç bir çevirmə və ya ixtisar aparılmadan göndərilir. Bu cür üsulla arxivlər, icra olunan fayllar, görüntülər və s. göndərilir;
get *fayl1 fayl2* – məsafədə yerləşən sistemdən fayl1 faylını göndər və istifadəçinin kompüterində fayl2 adı ilə onun lokal nüsxəsini yarat. Əgər ikinci parametr göstərilməsə, bu zaman lokal sistemdə eyni adlı fayl yaradılacaq. get fayl komandası göstərilən fayl ekrana çıxarılacaq;
put *fayl1 fayl2* – get komandasının əksi olan əməliyyatı həyata keçirir, yəni faylları lokal sistemdən məsafədə yerləşən sistemə göndərir;
delete *fayl* – göstərilən faylı məsafədə yerləşən sistemdən sil;
hash – hər bir verilənlər blokunu ötürdükdən sonra # işarəsini göstər. Bu komandadan iri həcmli faylları ötürərkən istifadə etmək əlverişlidir, belə ki, verilənlərin həqiqətən ötürülməsinə əmin olmaq imkanı verir. Bu rejimi ləğv etmək üçün hash komandası təkrar verilməlidir;
help – FTP-nin bütün mümkün komandalarının tam siyahısını göstər.



ları bütün alt direktoriyları ilə birlikdə göndərmək imkanı verir. TELNET proqramı kimi FTP proqramı da məsafədə yerləşən bir kompüterdə fon rejimində yerinə yetirilən FTP-serverindən və müştəri-proqramından ibarətdir. FTP protokolu ilə işləyərkən bağlantı quran kompüterlər arasında iki kanal açılır ki, bunlardan biri komandaların, digəri isə verilənlərin ötürülməsi üçündür.

ANONİM FTP

FTP protokolunun yaratdığı ən əsas imkanlardan biri bu və ya digər fayllara bütün şəbəkə istifadəçilərinin əlyetərliyini təmin edən anonim FTP-dir. Anonim FTP bu xidmətin *quraşdırıldığı* kompüterdə istifadəçinin qeydiyyatdan keçməsini tələb etmir. Anonim giriş üçün xüsusi olaraq ayrılmış "Anonymous" *istifadəçi adından* istifadə edilir. Anonim FTP serverlərdə ən müxtəlif məlumatları tapmaq olar. Bunlar müxtəlif drayverlər, müxtəlif əməliyyat sistemləri üçün əlyetərli proqram təminatı, serverlərində proqram məhsullarının yenilənmiş və nümayiş versiyalarını, eləcə də sənədlərə əlavələr və aşkar olunmuş səhvlər haqqında məlumatları yerləşdirən kompüter şirkətlərinin arxivləridir. Orada müxtəlif sənədlər, kompüter təsvirlərindən ibarət kitabxanalar, xəritələr, sxemlər, reproduksiyalar (üzü çıxarılmış şəkillər), fotoşəkil kolleksiyaları, müxtəlif mətnləri – İncil, Quran, Tövrat, romanlar, elmi fantastika və qara magiya, tarixi sənədlər, hava proqnozları, ulduz falı, kulinariya reseptləri və nəhayət, musiqi və kinofilm fayllarını, video-klipləri tapmaq olar.

ELEKTRON POÇT

Elektron poçt (E-mail – Electronic mail, *ing.* "mail" – "poçt") – adi poçtla birbaşa bənzərliyi olduğundan asanlıqla mənimsənilən bir vasitədir. İstifadəçi məlumatı elektron poçt vasitəsilə göndərə və öz elektron poçt qutusunda ala, cavabı avtomatik olaraq məktubun gəldiyi ünvana göndərə, öz məktubunun surətini eyni zamanda bir neçə alana çatdıra, aldığı məlumatı başqa ünvana ötürə bilər.

Elektron poçtun hər bir istifadəçisinə unikal poçt ünvanı mənimsədilir. Adətən, bu ünvan istifadəçinin adını onun kompüterinin adına bağlamaqla alınır. Kompüterlə istifadəçinin adı bir-birindən "quyruqlu a" (*ing.* "at commercial") adlanan xüsusi işarə, ilkin olaraq ad və soyadlarda rastlaşmayan simvol kimi seçilmiş "@" ilə ayrılır. Məsələn, əgər mekteb.edu.az adlı kompüterə istifadəçi "bahar" adı ilə daxil olursa, onun elektron ünvanı bu şəkildə olacaqdır: bahar@mekteb.edu.az.

Elektron poçtdan istifadə etmək üçün kompüterdə poçt serveri və yalnız sahibinə əlyetərli olan virtual "poçt qutusu" quraşdırılır. Daxil olan bütün məktublar bu qutuya yığılır və istifadəçinin onları xüsusi müştəri-proqramının köməyilə oxuyacağı ana qədər gözləyir. İstifadəçi elə bu proqramda öz məktubunu hazırlayıb, göndərə bilər. Bu zaman müştəri-proqram bu məktubu şəbəkə vasitəsilə lazımi ünvana çatdıracaq server-proqrama yollayır.

Elektron məktubda da adi məktubdakı kimi alanın və göndərənin ünvanı, "rabitə şöbələrinin möhürləri" – son ünvana çatmazdan əvvəl məktubun keçdiyi kompüterlərin adları göstərilir. Bundan başqa (Subject) alıcıya məktubların təsnifatını aparmağa, onların nə dərcədə təcili olmasını, dərhal cavab yazmağın zəruriliyini təyin etməyə imkan verən bir mətn sətri – məktubun başlığı da olur. Müasir poçt müştəri proqramları dialoq və menyular şəklində rahat interfeysə, həmçinin daxil olan və göndərilən məktubların saxlanılması üçün lazım olan bütün imkanlara malikdir. Bundan başqa, onlar adi mətn şəklində olan məktuba şəkil və ya istənilən digər faylı əlavə etmək imkanı verir. Əslində bu fayl hərflər ardıcıllığı kimi kodlaşdırılaraq məktubun mətninə daxil olunur, alanının proqramı isə onun kodunu avtomatik olaraq açır.

Hər bir poçt müştərisində ünvan kitabı, yəni daha çox istifadə olunan ünvanların siyahısı olur.

"ELEKTRON" MƏKTUBUN ÜSLUBUNUN XÜSUSİYYƏTLƏRİ

İstifadəçilərin əksəriyyəti qeyri-formal ünsiyyət üslubuna meyllidir və elektron aləmdə qəbul olunan ənənələrə riayət edirlər. Onlara müxtəlif ixtisarlar aiddir. Bunlardan bəziləri aşağıdakılardır:

IMHO – in my humble opinion – məncə
FYI – for your information – bilin ki
ASAP – as soon as posible – mümkün qədər tez bir zamanda
NC – no comment – şərhsiz
BTW – by the way – yeri gəlmişkən
THX (və ya TNX) – thanks – sağ ol

İnsanın sifət mimikasının üslublaşdırılmış təsvirlərindən – smayllardan (smile – "təbəssüm") istifadə etmək olar. Məktubların mətninə emosional rəng vermək üçün adi simvollara 90° bucaq altında baxmaq lazımdır:

:-) – təbəssüm
;-) – göz vurmaqla hiyləgər təbəssüm
:-(– qaşqabaqlı sifət
:-> – istehzalı təbəssüm
:-o – heyrət.

TELEKONFRANSLAR

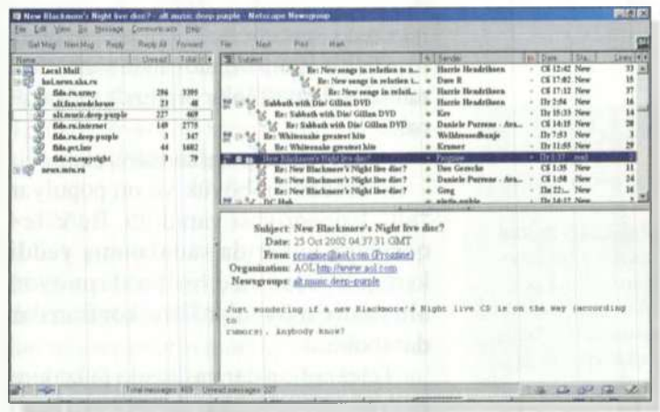
Telekonfranslar sistemi – paylanmış Usenet virtual şəbəkəsi açıq müzakirələrdə iştirak etmək, xəbər qrupuna məlumatlar göndərmək və onları oxumaq imkanı verir.

Adətən provayderlər elektron poçtla bərabər öz xəbərlər serverinə də müraciəti təmin edirlər. Serverə qoşulduqda onun dəstəklədiyi qrupların tam siyahısını yükləmək, sonra da sizi maraqlandıran qruplara abunə olmaq lazımdır. Abunəçiliyiniz bütün sonrakı əlaqə seanslarında saxlanılacaq. Konfransda yerləşdirilən məlumatları məqalələr (articles) adlandırmaq qəbul edilmişdir.

Bir neçə min xəbər qrupu mövcuddur, buna görə də, Usenet konfransları iyerarxik prinsip əsasında təşkil olunmuşdur. İlk əvvəl bu iyerarxiyanın əsasında yeddi kateqoriya yaradılmışdır:

• comp – hesablama texnikası, informatika, proqram təminatı və əlaqəli sahələr;
• sci – elmi-tədqiqat fəaliyyəti;
• soc – müxtəlif millətlərin sosial məsələləri və mədəniyyət problemləri;
• news – şəbəkə xəbərləri, xüsusilə xəbərlər sistemi haqqında;

Əgər ailə üzvləriniz sizi nahara elektron poçtla çağırırsa, o zaman ailənizlə daha çox vaxt keçirməyiniz məsləhətdir.



"Dip Pöpl" qrupuna həsr olunmuş telekonfrans.

ELEKTRON POÇTUN İCAD EDİLMƏSİ

Demək olar ki, son yüzilliklərdə edilən bütün ixtiraların müəllifləri məlumdur. Məsələn, 1807-ci ildə amerikalı Robert Fulton ilk paroxodu düzəltdi, 1903-cü ildə amerikalı Orvill və Uilber Rayt qardaşları dünyada ilk təyyarəni sınaqdan keçirdilər. Elektron poçtu icad edən şəxsin adı da məlumdur. Bu, yenə də Amerikadan olan professor Rey Tomlinsondur (Ray Tomlinson).

1971-ci ildə Tomlinson öz dahiyanə ixtirasını edərkən müasir İnternetin prototipi olan ARPANET şəbəkəsini (ABŞ Müdafiə Nazirliyinin Perspektiv Tədqiqatlar İdarəsinin Şəbəkəsini) yaradan BBN şirkətində işləyirdi. O, elektron poçtu rəhbərliyin tapşırığı olmadan, sadəcə şəxsi marağı üçün yaratmışdı. E-poçtun əsasında Deck firmasının kompüterlərilə işləyən tədqiqatçı və proqramçıların istifadə etdikləri və artıq mövcud olan mətn məlumatları ötürən bir proqram dururdu. Lakin bu proqramın köməyilə yalnız bir kompüterin istifadəçiləri bir-birilə məlumat mübadiləsi apara bilirdi. Tomlinsona sadəcə digər kompüterlərə də məlumatların ötürülməsi imkanını təmin etmək qalırdı və o, verilənlərin ötürülməsi protokolunu bir qədər dəyişdirdikdən sonra poçt proqramının ilk versiyası alındı.

Proqramı sınaqdan keçirmək üçün Tomlinson öz otağındakı şəbəkəyə qoşulmuş iki kompüterdən istifadə etdi. Elektron poçtla ilk məktubu o, elə özünə göndərdi. Tomlinsonun xatirələrində deyilir ki, həmin məktub klaviaturanın yuxarı hissəsində yerləşən böyük hərflərdən ibarət idi, təxminən QWERTYUIOP kimi bir sətir.

Tomlinson dünyada ilk poçt proqramını yaratdıqdan sonra onu iş yoldaşlarına göstərdi və onlardan bu haqda heç kimə danışmamağı xahiş etdi, görünür, müdiriyyətin onu xidməti vəzifəsinə etinasızlıq göstərməkdə günahlandıracağından ehtiyat edirdi. Lakin sonradan elektron poçtdan ilk istifadə edənlərdən biri də onun müdiri Larri Roberts oldu. Bir müddətdən sonra o, bütün işgüzar yazışmalarını elektron formaya çevirdi və nəticədə onun işçiləri də istər-istəməz müdirlərinin yolu ilə getməli oldular.


Rey Tomlinson.

Elektron poçtun yayılması XX əsrin 80-ci illərinin ortalarında fərdi kompüterlər meydana gəldikdən sonra kəskin şəkildə sürətləndi. Onun istifadəçilərinin sayı adi şəxslərin hesabına artdı. Əlverişli qrafik interfeysə malik Eudora proqramının yaradılması da elə həmin vaxta təsadüf edir.

Hazırda praktiki olaraq bütün İnternet istifadəçilərinin ən azı bir, çox vaxt isə bir neçə poçt ünvanı var. Analitiklərin hesablamalarına görə, 2004-cü ildə ABŞ əhalisinin yarısından çoxu elektron poçtdan istifadə edib, İnternet vasitəsilə hər gün 16 milyarda qədər elektron məktub göndərilib. Azərbaycan Respublikasında isə təqribən 1 milyon insan E-mail istifadəçisidir.



Rey Tomlinson bu möhtəşəm ixtirasına görə heç bir maddi mənfəət əldə etməyib. O, hazırda elə əvvəlki kimi BBN şirkətində işləyir, yeni elmi işlərlə məşğul olur.

Elektron poçtun üstünlükləri göz önündədir. O, E-mail-in fəal istifadəçiləri olan amerikalıların "ilbiz poçtu" (snail mail) adlandırdıqları adi poçtdan fərqli olaraq vaxta qənaət edir.

• rec – hər cür hobbi və əyləncələr;
• talk – istənilən mövzuda söhbətlər, mübahisəli məsələlər ətrafında müzakirələr;
• misc – yerdə qalan mövzular.

Sonradan ən böyük və ən populyar "alt" kateqoriyası yaradıldı. Bu kateqoriyaya yuxarıda sadalanmış yeddi kateqoriyadan heç birinə düşməyən alternativ mövzular üzrə konfranslar daxil oldu.

Telekonfranslar vasitəsilə müzakirə edilən mövzular son dərəcə geniş və müxtəlifdir və insan həyatının praktiki olaraq bütün fəaliyyət sahələrini əhatə edir. Xəbər qruplarının adları bunlardır:

• alt.music.deep-purple – "Dip Pöpl" Britaniya rok-qrupunun pərəstişkarları;
• alt.fan.wodehouse – Amerika yazıçısı Pelem Qrenvill Vudhausun pərəstişkarlarının ünsiyyət yeri;
• comp.lang.fortran – FORTRAN proqramlaşdırma dilinin məsələləri üzrə müzakirələr;
• rec.arts.tv.uk.comedy – ingilis telekomediyalarının həvəskarları üçün;

• sci.bio.paleontology – paleontoloqların konfransları;
• rec.pets.cats – pişiksevərlər üçün;

Word Wide Web-in populyarlığının artması və geniş yayılması ilə əlaqədar olaraq telekonfranslar sistemilə veb-səhifələrin köməyilə işləmək üçün vasitələr yaradılmışdır.

WORLD WIDE WEB

İnternet məsafə və sərhədlərə baxmayaraq, azad şəkildə informasiya mübadiləsini mümkün etdi. XX əsrin 90-cı illərinin əvvəlində qlobal şəbəkədə böyük həcmdə müxtəlif informasiya toplanmışdı. Lakin bu resurslar çox az mütəxəssis üçün əlçatan idi. Bunun üçün onlar yalnız faylların ötürülməsinə, uzaq terminal iş rejiminə və məlumatlar mübadiləsinə istiqamətlənmiş proqramlardan istifadə etməli olurdular. Vəziyyət yalnız World Wide Web (WWW) meydana gəldikdən və yayıldıqdan sonra dəyişdi.

World Wide Web – qlobal paylanmış hipermətnli multimedia informasiya sistemidir. O, bütün dünyada müxtəlif kompüterlərdə saxlanılan informasiyanı vahid bir sistemdə birləşdirməyə imkan verdi.

WWW-nin başlanğıcı 1989-cu ildə Cenevrədə yerləşən Avropa Nüvə Tədqiqatları Mərkəzində (CERN) işləyən britaniyalı proqramçı Tim Berners-Li (Tim Berners-Lee) tərəfindən qoyulmuşdu. Həmin vaxt o, CERN-nin bütün informasiya resurslarını (elmi hesabatlar və təcrübələrin nəticələrinin verilənlər bazasını, kompüter sənədlərini, poçt ünvanlarının siyahılarını, verilənlər yığınlarını və s.) birləşdirəcək bir informasiya sisteminin yaradılması üzərində işləməyə başlamışdı. Hipermətn texnologiyası bir sənəddən digərinə asanlıqla keçmək imkanı verməli idi.

Tim Berners-Li World Wide Web-i yaradarkən çətin təsəvvür edə bilərdi ki, tezliklə demək olar ki, hər bir şirkət, universitet, klub, mağaza və ya jurnalın öz saytı olacaq və bu sistem ətrafında bütöv bir sənaye sahəsi yaranacaq. WWW insanlara mehmanxanalarda yer tutmaq, qatara bilet sifariş etmək, elmi tədqiqatlarla tanış olmaq, müxtəlif ölkələrdə çap olunmuş qəzetləri oxumaq və planetin o biri tərəfində hava haqqında məlumatı öyrənmək imkanı verir.



Hipermətnin əsas ideyası ondan ibarətdir ki, onun sənədlərini (mətnin səhifələrini) ardıcıl deyil, kitab oxuyarkən adət etdiyimiz kimi, sərbəst şəkildə oxumaq olar. Hipermətndə xüsusi hissələr – istinadlar qeyd olunub. Onların köməyilə oxucu bir andaca lazım olan bölmələrə, şəkil və təsvirlərə daxil ola bilər. Nəticədə oxuma prosesi kökündən dəyişir: hiper mətni çoxlu sayda müxtəlif yollarla nəzərdən keçirmək olar və oxucu ona ən rahat olan yolu özü seçir. Hipermətnə misal kimi ensiklopediyanı göstərmək olar.

WWW texnologiyasının əsasında üç tərkib hissəsi dayanır: hipermətnin işarələnməsi dili HTML (HyperText Markup Language), şəbəkədə resursların universal ünvanlaşdırılma üsulu URL (Uniform Resource Locator) və HTTP (HyperText Transfer Protocol) hipermətnli informasiyanın mübadilə protokolu.

HTML

1986-cı ildə Beynəlxalq standartlaşdırma təşkilatı (ISO) tərəfindən istənilən növ mətnlərin məntiqi strukturunun işarələr sistemini qurmağa imkan verən, ümumiləşdirilmiş metadil standartı olan Standard Generalized Markup Language


Tim Berners-Li.

HTML dilinin standartının işlənməsi və təsdiqilə qeyri-kommersiya beynəlxalq təşkilatı olan World Wide Web Consortium (W3C) məşğul olur. XXI əsrin əvvəllərində HTML dilinin dördüncü versiyası geniş yayılmışdır.



www.crx.ru. saytının başlanğıc səhifəsinin ekran görüntüsü.

Hipermətn dilini sənədlərin vahid ünvanlaşdırma sistemi tamamlayır. WWW-də müxtəlif növ resurslar tapmaq mümkün olduğundan, onlara müraciət üçün ilk növbədə məxsusi proqram təminatı tətbiq etmək lazım gəlirdi. Tədricən bu məsələ asanlaşdı və hazırda yalnız ünvan və serverin növünü, yəni protokolu göstərmək tələb olunur.

HTTP

WWW sistemi ənənəvi "müştəri-server" sxemi üzrə qurulmuşdur. WWW serveri və müştəri proqramı arasında qarşılıqlı əlaqə HTTP hipermətninin ötürülməsi protokolu üzrə həyata keçirilir.

Qarşılıqlı əlaqə prosesində müştəri mətnə daxil edilmiş qrafik təsvir üçün sorğu göndərə, onun parametrlərini qəbul edərək ötürə, şəbəkədə hər hansı resursun yeni ünvanını əldə edə bilər.

WWW ilə iş lazımi sənədlərin hazır fayllarını almaqla məhdudlaşmır. Müştəriyə, həmçinin serverdə proqramları icra etmək imkanı verilir. Bu zaman həmin proqramların çıxışa verdiyi məlumatlar sənədlər şəklində müştəriyə göndərilir. Məsələn, WWW vasitəsilə məlumat axtarış sistemləri və istifadəçilə dialoqun tələb olunduğu istənilən interaktiv proqram və sistemlərlə iş bu cür təşkil olunmuşdur. Proqramların icrası üçün sorğunun verildiyi və müştəriyə nəticələrin ötürüldüyü standart Common Gateway Interface (CGI) adlanır.

BAŞLANĞIC SƏHİFƏ, SAYT

WWW sistemində informasiya səhifələr şəklində təşkil olunmuşdur. İnternetə çıxışı olan hər bir insan şəbəkədə istənilən məlumatı yerləşdirə bilər ki, bu məlumatlardan da bütün dünya istifadə edə bilər. Adətən, istifadəçilər özləri (SGML) qəbul edilmişdir. Bu işarələmə ilə mətnə daxil olunan idarəedici kodlar sadəcə onun məntiqi quruluşunu verirdi, yəni sənəddən sənədə istinad yaradırdı və onun xarici görünüşü haqda heç bir məlumat daşımırdı.

WWW-nu işləyib hazırlayanlar hipermətn sənədlərin xüsusi işarələmə dilinin əsası kimi SGML-i seçdilər. Bu dil Hypertext Markup Language (HTML) adlandırıldı.

Hipermətn, istinadlarının xüsusi formatlı ayrıca fayllarda saxlanıldığı SGML-dən fərqli olaraq, HTML-də hipermətn istinadlar, eləcə də sənədin xarici görünüşünü idarə edən kodlar sənədin mətninə *teq* adlandırılan xüsusi mətn parçaları şəklində daxil edilir.

URL daha çox WWW səhifələrinə istinad üçün istifadə olunur:
http://www.thehighwaystar.com,
http://www.infomir.ru/KuMir/updates.html.

haqda məlumatı öz provayderlərinin və ya şirkətlərinin serverlərində yerləşdirirlər. Məlumatların bu cür təqdimatı başlanğıc səhifə (*ing.* "home page") adlanır. Səhifə üçün pulsuz yer verən serverlər də mövcuddur. Başlanğıc səhifənin ünvanı, məsələn, bu cür olur: www.mekteb.az/~bahar.

WWW sisteminin eyni serverdə yerləşmiş və mənaca görə qruplaşdırılmış səhifələri sayt (*ing.* site – "yer", "sahə") təşkil edir. Çox vaxt saytlar şirkətlərin İnternetdəki nümayəndəlikləridir. Həmçinin dövlət strukturları, ictimai təşkilatlar, bu və ya digər informasiya layihələrinin də saytları mövcuddur. İnternetin fərdi istifadəçisi də öz şəxsi saytını yarada bilər. Sayt ünvanları başlanğıc səhifələrdən fərqli olaraq www.mekteb.az kimi görünür.

BRAUZERLƏRİN MÜHARİBƏSİ

Uzaqdan irişim (TELNET) və faylların ötürülməsi (FTP) sistemlərində olduğu kimi, World Wide Web-də istifadəçinin kompüterində işləyən müştəri-proqramları mövcuddur. Bunlar hipermətn sənədlərini nəzərdən keçirən proqramlar və ya *brauzerlərdir* (*ing.* to browze – "nəzərdən keçirmək"). İlk brauzerlər DOS komanda sətrinə bənzər mətn interfeysinə malik idi. 1993-cü ilin əvvəllərində İllinoys Universitetinin Milli superkompüter proqramları mərkəzində (NCSA) Mark Andrissenin rəhbərliyi altında bir qrup proqramçı Mosaic brauzerini yaratdı. O zamanlar üçün bu, müxtəlif kompüter platformalarında işləyən, istifadəçilər üçün əlverişli qrafik interfeysə malik olan və peşəkar səviyyədə yazılmış yeganə proqram idi. Mosaic tez bir zamanda böyük uğur qazandı və öz yaradıcılarına dünya şöhrəti gətirdi.

1994-cü ildə Mosaic proqramının yaradıcıları Netscape Communications

İLK SAYT



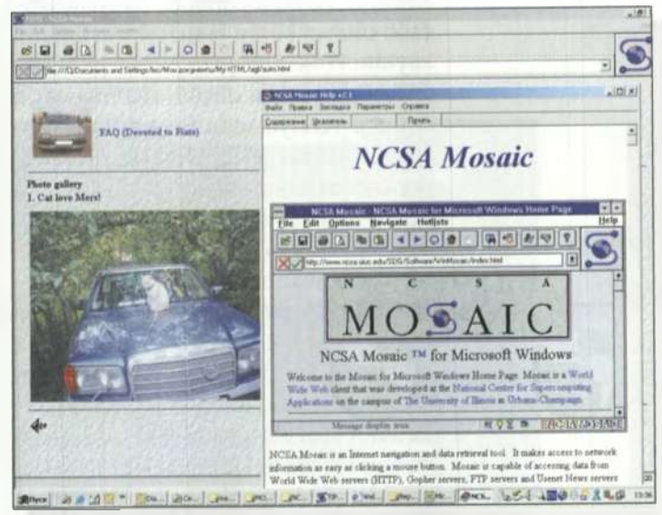
Pol Kunz.

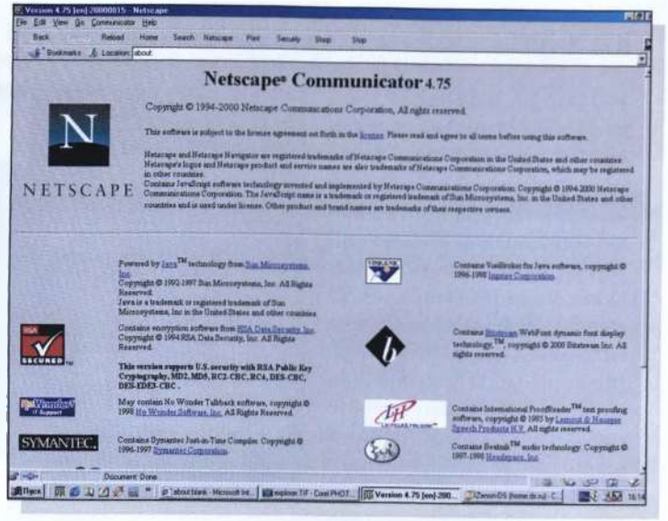
1991-ci ilin dekabrında Stanford Universitetindən amerikalı fizik Pol Kunz (Paul Kunz) World Wide Web sistemi üçün ilk saytı yaratdı. Bu sayt dünyanın istənilən yerindən olan alimlərə universitet mərkəzinin elmi əsərlərinə əlyetərliliyi təmin etdi. O zamanlar universitetin verilənlər bazasında məqalələrə 200 min istinad saxlanılırdı və ona dünyanın 40 ölkəsindən minlərlə istifadəçi müraciət edirdi. Lakin lazımi informasiyanın axtarışı və alınması prosesini tezləşdirəcək və sadələşdirəcək bir texnologiyaya ehtiyac var idi. Kunzun saytı məhz bu işdə yaradı.

1992-ci ilin yanvarında Tim Berners-Li Fransada dünyanın müxtəlif ölkəsindən 200 fizikin iştirak etdiyi konfransda bu saytı nümayiş etdirdi. Əvvəlcə o, WWW sistemi barədə danışdı, sonra isə onun köməyilə okeanın o tayında, Kunzun ABŞ-dakı saytı ilə necə əlaqə yaradıldığını göstərdi. Müvəffəqiyyət gözlənilməndən də yüksək oldu. Lazımi informasiyanı əldə etməyin nə qədər asan və sadə olması konfrans iştirakçılarını heyran etmişdi.

Kunzun saytının tərtibatı üç sətirlik mətndən və iki istinaddan ibarət idi. İstinadlardan biri telefon nömrələri və elektron poçt ünvanlarından ibarət "telefon kitabçasına", digəri isə elmi məqalələr bazasına idi.

Mosaic brauzerinin pəncərəsi.





Netscape Navigator brauzerinin pəncərəsi.

şirkətini yaratdılar və tezliklə fantastik populyarlıq qazanmış ilk ticari brauzer olan Netscape Navigator-u satışa buraxdılar. Bu sahədə öz liderliyini möhkəmlətmək və eyni zamanda yeni istifadəçilər cəlb etmək məqsədilə Netscape HTML-ə sənədin xarici görünüşünü yaxşılaşdırmaq və formatlaşdırma imkanlarını genişləndirmək üçün yeni təkmilləşdirilmələr daxil edirdi. Dilin yeni versiyası yalnız Navigator brauzeri tərəfindən dəstəklənirdi. Belə siyasət Netscape-ə daha böyük uğur gətirdi. Müəyyən müddət ərzində dünyada istifadə olunan bütün brauzerlərin ümumi sayının 90%-dən çoxu Netscape Navigator-un payına düşdü. Bu brauzerin üçüncü versiyasında Java dili dəstəkləndi, JavaScript daxili ssenarilər dili, pəncərəni hissələrə bölmək imkanı və digər yeni qaydalar təqdim olundu.

1995-ci ildə Netscape-in hökmranlığının rəqiblərə heç bir ümid yeri qoymadığı bir vaxtda brauzerlər bazarına Microsoft şirkəti daxil oldu. Bu sektorun təkbaşına sahibi olmağa adət etmiş bu şirkət uzun müddət İnternetin perspektivlərini lazımi qədər qiymətləndirmədi və onun inkişaf etdirilməsində hər hansı şəkildə iştirak etmək niyyətində deyildi. Lakin Netscape-in qazandığı inanılmaz uğur şirkət rəhbərliyini öz fikrini dəyişməyə vadar etdi. Microsoft-un o zaman heç bir üstünlüyü ilə seçilməyən Internet Explorer brauzerinin Netscape-nin rəqibi ola biləcəyinə əvvəlcə inanmaq çətin idi. Lakin Internet Explorer-in 1996-cı ilin yayında çıxmış üçüncü versiyası orijinal, cəlbedici interfeysə malik idi və yavaş-yavaş Netscape Navigator-u bazarda sıxışdırmağa başladı.

Brauzerlərin müharibəsində həlledici an İnternetə çıxış vasitələrinin və hazırda dünyada ən çox istifadə olunan Internet Explorer brauzerinin növbəti versiyasının daxil olduğu Microsoft Windows 98 əməliyyat sisteminin satışa buraxılması oldu.

VEB-DİZAYN

WWW-nin reytinqinin yüksəlməsi qrafik brauzerlərin hər yerdə yayılması ilə eyni vaxta təsadüf etmişdir. Sayt yaradıcıları lap əvvəldən təqdim olunan informasiyanın tərtibatı üçün mövcud olan bütün imkanlardan istifadə etməyə başlamışdılar. Nəticədə yeni fəaliyyət sahəsi – *veb-dizayn* meydana gəldi.

Geniş mənada veb-dizayn veb-səhifələrin tərtibatıdır. O, sayt üçün kitab nəşrində səhifələmə və poliqrafik dizaynın oynadığı rolu oynayır. Veb-dizayn dedikdə, sadəcə, sayt üçün qrafik elementlərin yaradılması deyil, həm də onun bütün strukturunun layihələşdirilməsi, yəni bütövlükdə saytın yaradılması başa düşülür. Veb-dizayn mütəxəssisinin HTML standartını bilməsi zəruridir.

Hipermətn sənədi hər biri "teq"lərlə əhatə olunmuş elementlər toplusudur. Öz əhəmiyyətinə görə teqlər proqramlaşdırma dilində komandaların təsir sahəsini təyin edən "begin – end" ("başlanğıc – son") mötərizələrinə yaxındır. Teqlər sənədin mətn elementlərinin izahı üzrə qaydaların təsir sahəsini təyin edir. Məsələn: <I> teqi kursiv (*ing.* "italic") sahəsini təyin edir. HTML dilində mətn:

Bu mətn <I>kursivlə yığılıb.</ I>

Bu mətnin baxış proqramında (brauzerdə) görünüşü:

Bu mətn *kursivlə yığılıb.*

Göstərilən misalda kursivlə yazılacaq mətn bu üslub üçün <I> başlanğıc teqi və </ I> son teqilə qeyd olunub seçilməlidir. Hər-hansı mətn elementinin məzmunundan əvvəl gələn konstruksiya elementin başlanğıc teqi adlanır və bu şəkildə yazılır: <elementin_adı parametrlərin_siyahısı>. Elementin məzmunundan sonra yerləşən konstruksiya isə elementin son teqi adlanır və bu cür yazılır: </elementin_adı>.

WWW sistemində hipermətn istinadlarının yazılışı üçün URL-dən istifadə olunur. Məsələn,

<A HREF="http://www.ayna.az/idman.html" İdman </ A>

Bu misalda A (*ing.* "anchor" – "lövbər") elementi URL formasında qeyd üçün hipermətn istinadını ifadə edən "HREF" parametrindən istifadə edir. Verilən istinad HTTP hipermətn ötürmə protokolu üzrə müraciət olunacaq www.ayna.az serverində saxlanılan idman.html faylını göstərir. Bir qayda olaraq, istinadlar altından xətt çəkilməklə seçdirilir, göstərici onların üzərinə qoyulduqda isə şəklini dəyişir (məs., ox əvəzinə barmaqla göstərmə).

Sənədə şəkillər oxşar teq vasitəsilə daxil edilir:

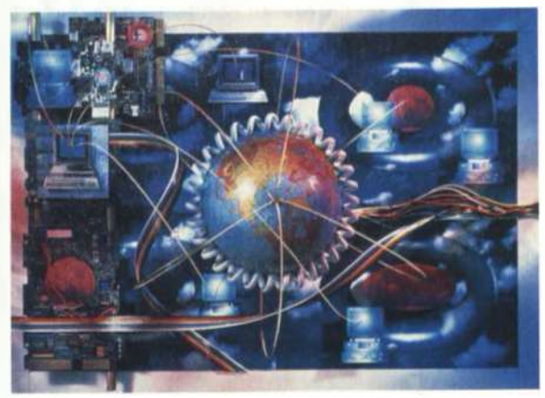
<IMG SRC = URL_şəkil>.

Şəklin eni (weight=), hündürlüyü (height=) və təsviri (alt=), həmçinin onun yerləşməsi (align.) kimi parametrləri tətbiq etmək olar. Avtomatik olaraq şəkillər daxil edildiyi mətnin aşağı küncündə yerləşdirilir.

Hipermətn sənədinin özünü "HTML" adlanan bir böyük element kimi təsəvvür etmək olar. O, iki hissədən – sənədin başlığından və ekranda əks olunan mətnin yerləşdiyi sənədin əsas gövdəsindən ibarətdir.

```
<HTML>
<HEAD>
Sənədin başlığı
</HEAD>
<BODY>
Sənədin gövdəsi
</BODY>
</HTML>
```

Formatlaşdırma sənədin ekrana çıxması prosesində avtomatik yerinə yetirilir. Sətirlərə bölünmə çıxış pəncərəsinin enindən, şriftin hazırkı ölçüsündən, ekranın imkanlarından və s. amillərdən asılıdır, lakin yeni sətrə məcburi keçid üçün
 teqi var. Yaxud, məsələn, <HR> komandasından istifadə etməklə, mətni şaquli xətlə bölmək olar. Baxış proqramında formatlaşdırmanı dayandıraraq əvvəlcədən hazırlanmış mətni ekranda təsvir etmək üçün xüsusi <pre>...</pre> teqlər cütündən istifadə edirlər. Mətni qalın şriftlə, kursivlə, altından xətt çəkməklə qeyd etmək üçün də teqlər var. Daha mürəkkəb



Əgər xanımınızı www.mylady.home.wife kimi təqdim edirsinizsə, istirahət barədə ciddi düşünməlisiniz.

formatlaşdırma üçün adi və nizamlanmış siyahılar, cədvəllər, kadrlar tətbiq olunur. Ayrı-ayrı sözlər, cümlələr və ya bütövlükdə sənəd üçün üslub yaratmaq imkanı mövcuddur və bu üslubun dəyişdirilməsi, dərhal onun istifadə olunduğu bütün sənədlərdə dəyişikliyə gətirib çıxaracaq.

HTML 4 dilinin (CSS-üslubları daxil olmaqla) tam təsviri – çoxsəhifəli sənəddir. Lakin hipermətn dilinin mənimsənilməsi üçün daha asan üsul da var. Bütün baxış proqramları WWW- səhifələrinin ilkin kodunu göstərmək imkanına malikdir, buna görə də misallar üzərində öyrənmək mümkündür.

İNTERNET ŞƏBƏKƏSİNİN MİQYASI



1969-cu ildə kompüterləri İnternet şəbəkəsinin əcdadı olan qlobal şəbəkədə birləşdirmək üçün ilk təcrübələr aparılarkən, ona sadəcə üç kompüter daxil idi. Lakin İnternet durmadan inkişaf edir. Onun ölçülərini şəbəkəyə qoşulmuş kompüterlərin sayını hesablayaraq təyin edirlər. Qrafikdə Internet Software Consortium təşkilatının məlumatı (rəqəmlərlə) göstərilmişdir.

Kompüterlərin sayını hesablamaq olduqca asan olsa da, İnternet istifadəçilərinin sayını yalnız təxmini söyləmək olar. Həqiqətən də, bir nəfərin İnternetə bir neçə kompüterdən çıxışı ola bilər və əksinə, bir kompüter bir neçə istifadəçiyə şəbəkəyə çıxış imkanı verə bilər.

1996-cı ildə şəbəkəyə orta hesabla 50 milyona yaxın insanın çıxışı var idi. 1999-cu ildə onların sayı 200 milyona qədər artdı, 2005-ci ildə isə 440 milyona çatdı. Onların əksəriyyəti (185 mln.) ABŞ-dan (ölkə əhalisinin 60%-i) olan istifadəçilər idi. Böyük Britaniyada – 33 milyon, Yaponiyada – 78 milyon, Çində – 99 milyon, Almaniyada – 42 milyon, Kanadada isə – 20 milyon insan İnternetə qoşulur.

İctimai fikir fondunun məlumatına görə, 2005-ci ildə Rusiyada İnternet istifadəçisinin sayı 17 milyondan artıq idi ki, onlardan 10,3 milyonu həftədə bir dəfədən az olmayaraq İnternetə qoşulur. İnternetin populyarlığının kəskin artması Ümumdünya hörümçək toru – WWW-nin meydana gəlməsilə əlaqədar idi. 1993-cü ildə 130-a yaxın WWW-sayt qeydə alınmışdı, 1994-cü ildə onların sayı 9 mini (o zamanlar onları hələ saymaq mümkün idi), 1996-cı ildə artıq 100 mini, 1997-ci ildə isə 200 mini aşmışdı.

2005-ci ilə qədər yaradılmış saytların dəqiq sayını hesablamaq mümkün deyil. Müxtəlif hesablamalara görə, onların sayı təxminən bir neçə on milyondan



Dünya üzrə İnternetə qoşulan kompüterlərin sayının artımı.

milyarda qədərdir. Onların səhifələrinin sayı isə 8 milyard ətrafındadır.

Mütəxəssislərin proqnozlarına görə, onların sayı tədricən telefon, televizor və ya su xəttindən istifadə edənlərin sayına yaxınlaşacaq...

Həmçinin İnternetə qoşulanların sayı da artacaq. Bu yalnız kompüterlərin deyil, İnternet vasitəsilə idarə oluna biləcək soyuducu, mikrodalğalı soba və ya kondisioner kimi digər məişət qurğularının da hesabına artacaq.

İNTERNETDƏ KOMMERSİYA

Bəşər tarixi boyunca hansı məqsədlər üçün yaradılmasından asılı olmayaraq, yeni ixtira olunmuş texnologiyalardan demək olar ki, hər zaman biznesə xidmət göstərmək üçün istifadə olunub. Dəmir yolu və teleqrafla vəziyyət də belə olmuşdur. Elmi və hərbi məsələlərin həlli üçün işlənib hazırlanmış İnternetlə də belə oldu. İnternetin yayılması müəyyən səviyyəyə çatdıqda o, ticarətçilərin diqqətini çəkdi.

Artıq XX əsrin 80-ci illərinin ortalarında elektron poçtdan pullu istifadə xidməti göstərən, WWW sisteminin inkişafı ilə İnternetə çıxışı təmin edən şirkətlər meydana gəldi. Beləliklə, İnterneti dəstəkləyərək işlək vəziyyətdə saxlayan və istifadəçilərin İnternetə çıxışını təmin edən müstəqil iqtisadiyyat sahəsi yarandı.

İnterneti dəstəkləyən və istifadəçilərə müxtəlif xidmətlər göstərən biznes alətləri İnternetlə qarşılıqlı əlaqədədir. Artıq gündəlik həyatımıza *elektron kommersiya* (e-commerce) anlayışı daxil olub.

Elektron kommersiya İnternetdən tərəfdaşlar, banklar, mal tədarükçüləri və istehlakçıları ilə qarşılıqlı əlaqə vasitəsi kimi istifadə etməklə, sazişlər bağlanması və kommersiya fəaliyyətinin texniki və təşkilati formalarının məcmusudur.

ELEKTRON KOMMERSİYANIN NÖVLƏRİ

İştirakçıların birbaşa təmas yolu ilə deyil, elektron sistemlər və İnternet vasitəsilə qarşılıqlı əlaqədə olduğu istənilən işgüzar əməlləri, heç şübhəsiz, elektron kommersiya sahəsinə aid etmək olar.

Bir qayda olaraq, elektron kommersiya sistemlərində sazişin bağlanmasının bütün mərhələləri: tələb olunan məhsulun və ya xidmətin axtarışı, sazişin təfsilatlarının dəqiqləşdirilməsi, sifarişlərin ödənişi, alınması (çatdırılması) mövcuddur.

Elektron kommersiyanı sövdələşmə prosesində yaranan münasibətlərin növünə görə fərqləndirirlər.

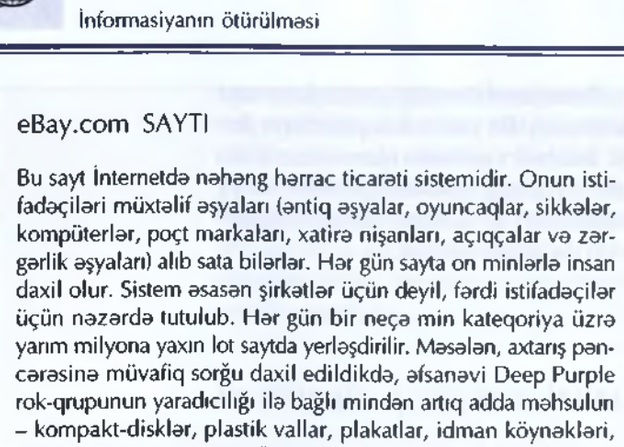
Hal-hazırda İnternetdə ticarət əməliyyatlarının böyük hissəsi münasibətlərin *business-to-business* (B2B) münasibətləri sahəsində həyata keçirilir. B2B texnologiyasının tipik nümunəsi kimi iri şirkətlərin tədarükçülər ilə qarşılıqlı əlaqəsini göstərmək olar. Yüksək texnologiyalar sahəsində fəaliyyət göstərən bəzi şirkətlər nəinki lazımi hissələrin alınmasını, həm də demək olar ki, satış əməliyyatlarının əksəriyyətini İnternet vasitəsilə həyata keçirirlər. B2B texnologiyaları son dərəcə dinamik şəkildə inkişaf edir. Dünyanın bir çox iri korporasiyaları gələcəkdə tədarükçülərlə bütün münasibətləri tamamilə elektron formaya keçirməyi planlaşdırırlar.

Lakin elektron biznesin ən çox yayılmış modeli *business-to-consumer* (B2C) modelidir. Bu, hamının çox yaxşı tanıdığı

eBay.com SAYTI

Bu sayt İnternetdə nəhəng hərrac ticarəti sistemidir. Onun istifadəçiləri müxtəlif əşyaları (əntiq əşyalar, oyuncaqlar, sikkələr, kompüterlər, poçt markaları, xatirə nişanları, açıqçalar və zərgərlik əşyaları) alıb sata bilərlər. Hər gün sayta on minlərlə insan daxil olur. Sistem əsasən şirkətlər üçün deyil, fərdi istifadəçilər üçün nəzərdə tutulub. Hər gün bir neçə min kateqoriya üzrə yarım milyona yaxın lot saytda yerləşdirilir. Məsələn, axtarış pəncərəsinə müvafiq sorğu daxil edildikdə, əfsanəvi Deep Purple rok-qrupunun yaradıcılığı ilə bağlı mindən artıq adda məhsulun – kompakt-disklər, plastik vallar, plakatlar, idman köynəkləri, musiqiçilərin öz avtoqrafları və s. siyahısını görmək olar.

Bəs eBay.com saytının sahibləri nəyin hesabına gəlir əldə edirlər? Satıcılar öz məshulları haqqında məlumatları saytda yerləşdirmək üçün pul ödəyirlər, alıcılar isə onları maraqlandıran əşyaları almaq üçün sayta müraciət edirlər. Adətən, hər bir əmtəə bir neçə gün ərzində göstərilir. Sövdələşmə baş tutduğu halda, eBay.com saytının sahibləri satıcıya satış qiymətinin müəyyən faizi üçün hesab təqdim edirlər. Bu zaman nə malların saxlanılması üçün anbar, nə adi hərraclarda olduğu kimi ticarət yerləri kirayələmək, nə də hərracı aparmaq üçün adam tutmaq lazım gəlmir. Əslində eBay – əsas üstün cəhəti etibarlılıq və populyarlıq olan nəhəng bir vasitəçidir. Əgər nadir tapılan bir əşyanı almaq və ya satmaq lazım gələrsə, o zaman ilk öncə eBay.com saytına daxil olmağa dəyər. Deyilənlərə görə, Pyer Omidyar (Pier Omidiyar) bu saytı özünün nadir oyuncaqlardan ibarət kolleksiyasını genişləndirmək istəyən həyat yoldaşının xahişilə yaratmışdı.



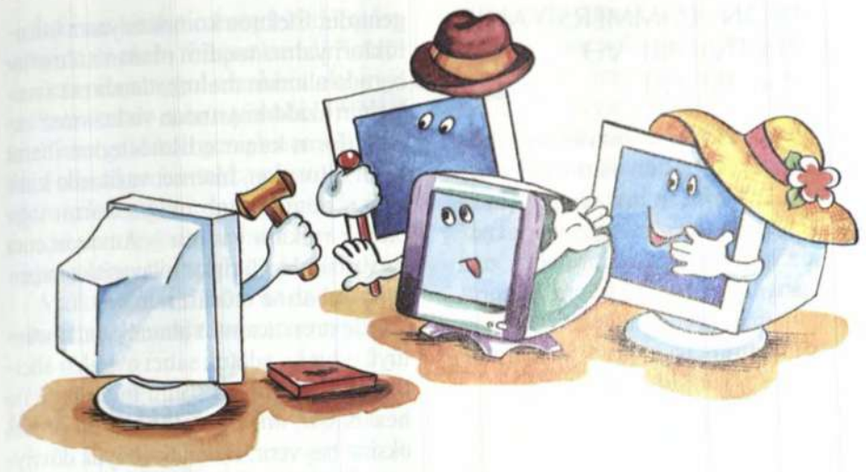
İnternet-mağazalar, elektron kataloqlar, sifariş masaları və şirkətlərin İnternetdəki nümayəndəlikləridir. İlk İnternet-mağazalarda (onlardan ən çox tanınanı Amazon.com-dur) kitablar, proqram təminatı, kompüter hissələri, audioyazılar (CD) və videofilmlər (VHS və DVD) satılırdı. Bir müddət sonra istifadəçilər daha bahalı malları da, məsələn, əntiq əşyaları, incəsənət əsərlərini, zərgərlik və zinət əşyalarını, mebeli əldə etməyə başladılar. Müştərilər xüsusiyyətləri onlara əvvəlcədən məlum olan mallara daha çox üstünlük verir. Çox vaxt hansısa bir əşyanı almaq barədə qərar qəbul etmək üçün istehsalçının adını bilmək kifayət edir. Lakin delikates yeməklər sifariş etmək, geyim və ya ayaqqabını geyinib baxmadan almaq alıcıdan müəyyən cəsarət tələb edir.

Digər tərəfdən İnternet-mağazanın saytında bazarlıq, insana real mağazada ticarət zallarını gəzərkən və malları seçərkən keçirdiyi hisslərin yerini vermir. İnternet-mağaza, həmçinin insana əşyanı alarkən bir anlıq keçirdiyi məmnunluq hissini də yaşada bilmir.

Fiziki şəxslər arasında elektron kommersiya münasibətləri *consumer-to-consumer* (C2C) adlanır. Buna tipik nümunə fiziki şəxslərin digərləri tərəfindən alınması üçün nəzərdə tutulmuş müxtəlif əşyaları satışa çıxardığı elektron hərracdır. eBay.com ən çox tanınan və daha çox müraciət olunan elektron auksionlardan biridir.

İnternetdə təklif edilən əmtəə və xidmətlərə görə ödəniş müxtəlif yollarla – nağd pulla, əmtəə poçt vasitəsilə alınarkən ödəniş tapşırığı əsasında, bank köçürmələri və ya poçtla köçürmə ilə, kredit kartı vasitəsilə həyata keçirilir. Sonuncu variant daha sürətlidir, lakin kredit kartları barədə məlumatların bədniyyətlilər tərəfindən şəbəkədən oğurlana biləcəyindən haqlı olaraq ehtiyat edən alıcılar çox vaxt öz kredit kartı rekvizitlərini İnternetlə göndərməkdən çəkinirlər. Bu isə elektron ticarətin həyata keçirilməsi qarşısında duran maneələrdən biridir. Onları aradan qaldırmaq üçün kredit kartlarındakı

İnternet-mağaza öz istehlakçıları ilə təklif olunan məmulat və ya xidmətlərin təqdim olunduğu və bilavasitə şəbəkədə olan istifadəçi tərəfindən sifariş edilməsini nəzərdə tutan sayt vasitəsilə əlaqə yaradan ticari təşkilatdır.



hesabla heç bir əlaqəsi olmayan xüsusi "elektron" hesablar açılır. Elektron ödənişlərin təhlükəsizliyinin artırılması da olduqca vacib şərtdir.

ŞƏBƏKƏ VASİTƏSİLƏHANSI MALLARIN SATIŞI DAHA MƏQSƏDƏUYĞUNDUR

Tədqiqatlar göstərir ki, İnternet vasitəsilə daha çox gündəlik istehlak malları alınır. Bahalı və uzun müddət planlaşdırılan əşyaların alışı isə burada daha nadir hallarda baş verir. Sifarişlərin çatdırılma xüsusiyyətlərini nəzərə aldıqda, aydın olur ki, şəbəkə vasitəsilə daha ağır, çox yer tutan və çatdırılmanı çətinləşdirən mallardan fərqli olaraq kompakt, rahat daşınan mallar daha asan satılır.

İnternet vasitəsilə alıcıya çatdırılması mümkün olan mallar ilə şəbəkə vasitəsilə ticarətin imkanları arasında ideal uyğunluq var. Bu cür əmtəəyə nümunə olaraq proqram təminatını göstərmək olar. Kitab və qəzetlər də sifarişçilərə elektron formada çatdırıla bilər. Bir sözlə, rəqəmləşdirilə bilən nə varsa, o cümlədən musiqi, qrafik təsvirlər, videomateriallar şəbəkədə satılmaq üçün əlverişlidir. XX əsrin 90-cı illərinin sonundan başlayaraq İnternet vasitəsilə aviabiletlər, turist, birja, sığorta və bank xidmətlərinin satış populyarlığı artır.

Buna baxmayaraq, şəbəkənin təşəbbüskar alverçilərindən çoxu iddia edir ki, elə bir əmtəə yoxdu ki, onu İnternet vasitəsilə satmaq mümkün olmasın.



Əgər İnternetdə dostlarınız real həyatda olduğundan çoxdursa, təcili olaraq bir qonaqlığa getməyiniz məsləhətdir.

ELEKTRON KOMMERSİYANIN PERSPEKTİVLƏRİ VƏ ÜSTÜNLÜKLƏRİ

Gələcəkdə elektron biznes ənənəvi əmtəə satışı kanallarına nəinki ciddi rəqabət yarada bilər, hətta onları tamamilə sıxışdıra bilər. İnformasiya texnologiyalarının sürətli inkişafı daima yeni imkanlar yaradır. Bunlardan ən əhəmiyyətlisi İnternetdə verilənlərin ötürülməsi sürətinin artırılması və iri həcmli informasiyanın ötürülməsi imkanıdır.

Geniş zolaqlı rabitə kanallarının meydana gəlməsi və XXI əsrin ilk onilliyində daha çox fərdi istifadəçinin onlara qoşulması ilə əlaqədar elektron kommersiyanın daha sürətli inkişafı gözlənilir.

Bundan başqa, mobil telefonların və digər elektron məişət qurğularının (fərdi kompüterlərin monopoliyasını pozaraq) İnternetə qoşulma vasitələrilə təchiz olunması da elektron kommersiyanın inkişafına böyük təsir göstərir.

Qərb ekspertlərinin hesablamalarına görə televiziya və radio üzrə reklamlar hələlik İnternetdəki reklamlara nisbətən daha güclü təsirə malikdir, lakin İnternet həmin göstəricilər üzrə informasiyanın çap vasitələri olan qəzet və jurnallardan artıq "irəlidədir".

Elektron kommersiyanın ilk baxışdan iqtisadi imkanları göründüyündən xeyli genişdir. Elektron kommersiyanın üstünlükləri yalnız təqdim olunan xidmətlər barədə alıcıları məlumatlandırma xərclərinin azaldılmasından və lazımsız vasitəçilərin kənar edilməsindən ibarət deyil. Məsələn, İnternet vasitəsilə kitab satışı, böyük kitab mağazalarını ləğv etmək imkanı yaradır – Amazon.com serverindəki sifarişlər bilavasitə topdansatış anbarına ötürülür.



Elektron ticarətin əhəmiyyətli üstünlüyü odur ki, adətən satıcı əvvəlcə alıcıdan pulları alır və sonra tədarükçü ilə hesablaşır, adi ticarətdə isə bu proses əksinə baş verir. Beləliklə, böyük dövriyyəyə kapitalı cəlb etməyə ehtiyac qalmır.

Elektron kommersiyanın digər iqtisadi üstünlükləri də var. Məsələn, alıcı ona daha uyğun olan qiyməti seçir. Qiymətləri "siçanın"un düyməsini basmaqla müqayisə etmək, mağazaları uzun-uzadı piyada gəzməkdən daha asandır. Təxminlərə görə İnternet bununla qiymətlərin süni artırılmasının qarşısını alacaq.

İnternet biznesin inkişafı baxımından da əlverişlidir. Yalnız bir saytın köməyilə bütün dünya üzrə əməliyyatları həyata keçirmək olar: sadəcə hər an saytı ziyarət edənlərin sifarişlərini yerinə yetirə biləcək etibarlı işləyən bir sistem qurmaq lazımdır.

Müştərilərin uzaq məsafədə olmasına, saat qurşaqlarının fərqinə baxmayaraq biznesi istənilən miqyasda genişləndirmək olar. İnternet həmçinin istifadəçi birlikləri yaratmağa, onlar üçün şəbəkəsiz mümkün olmayan xüsusi hərraclar keçirməyə imkan verir.

İnternet turist şirkətlərinin ənənəvi maraq sahəsinə də son dərəcə ciddi şəkildə daxil olmuşdur. İstirahət edənlər təyyarə bileti almaq və ya saxlatmaq üçün turizm şirkətlərinə müraciət etmək əvəzinə İnternet vasitəsilə birbaşa aviaşirkətlərə müraciət edir, eyni qayda ilə mehmanxanalarda otaq sifariş edir və ödəniş aparırlar.

İnternet kitab ticarətçilərinin mövqelərinə də aktiv "hücumlar edir": topdan satış ilə məşğul olanların əksəriyyəti parakəndə satışa yönəliblər. Yeni kitabların getdikcə genişlənən elektron nəşri praktikası isə təkcə müxtəlif xırda ticarətçilərin deyil, eyni zamanda topdansatışla məşğul olanların, hətta nəşriyyatçıların belə işini çətinləşdirir.

Audio-yazıların satışı ilə də vəziyyət eynidir: melomanlar ən son musiqi yeniliklərinin yüksək keyfiyyətli kopiyasını şəbəkə vasitəsilə əldə edə bilər.

İnternet, istənilən texnoloji inqilab kimi, cəmiyyətə o zaman maksimum təsir edəcək ki, onun köməyilə tamamilə yeni mallar və xidmətlər yaranıb geniş yayılacaq, mövcud olanlar isə keyfiyyətcə yeni şəkildə təqdim olunacaq. Kommersiyanın ənənəvi və elektron formaları arasında müharibə hələ yenicə başlayır.

İNTERNETDƏ AXTARIŞ

İNTERNETDƏ NƏ TAPMAQ OLAR?

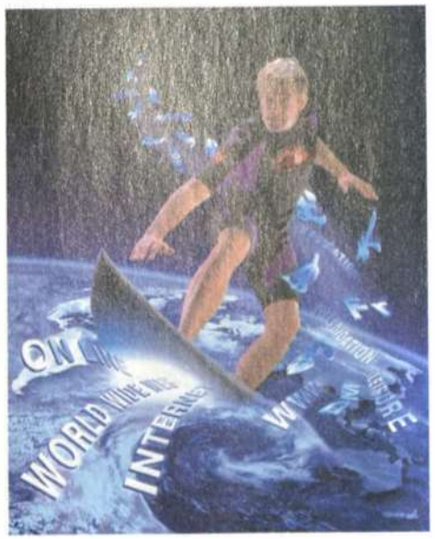
İnternetdə, xüsusi halda onun əsas təcəssümü olan WWW sistemində, saxlanılan informasiyanın həcmini ölçmək qeyri-mümkündür. Onun ancaq tərtibini qiymətləndirmək olar. Ehtimal olunduğuna görə bu, onlarla və ya hətta yüzlərlə terabaytdır (milyon dəfə milyon bayt). Bunlar çox müxtəlif xarakterli və istiqamətli məlumatlardır.

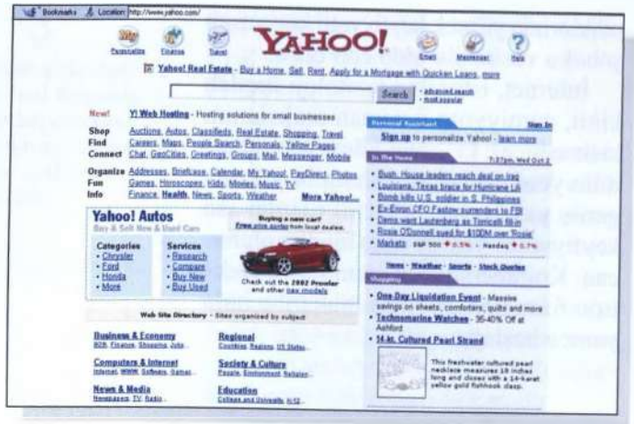
İnternetdə ən son siyasi, mədəni, iqtisadi xəbərləri; hər cür elmi, texniki, tədris və sorğu məlumatlarını; müxtəlif mal və xidmətlərin reklamlarını tapmaq mümkündür. Bura həm də mədəniyyət və incəsənətə – ədəbiyyat, rəssamlıq, teatr, kino, televiziyaya həsr olunmuş resurslar da aiddir.

Bundan başqa İnternetdə yüz minlərlə müxtəlif mövzulu əyləncəli saytlar tapmaq olar: yumor, oyunlar, səyahət, idman. Hər gün İnternetdə təqribən bir milyon yeni WWW-səhifə meydana gəlir, lakin bununla yanaşı bəzi səhifələr birdəfəlik yox olur.

İnternetdə sanki hər şey var. İstifadəçi qarşısında duran məsələ isə istədiyi məlumatı oradan tapmaqdan ibarətdir. Müasir İnterneti elə bir kitabxanaya bənzətmək olar ki, orada kitablar son dərəcə nizamsız şəkildə yerləşdirilmişdir: orada heç bir vahid kataloq, təsnifat sistemi və heç bir kitabxanaçı yoxdur. "Kitabxana" ziyarətçiləri ora vaxtaşırı yeni kitablar əlavə edir və ya onları həmişəlik götürüb aparır.

İnternetdən faydalı məlumatları əldə etmək üçün axtarışı necə və harada aparmağı bilmək lazımdır. Bu məsələ insanları, demək olar ki, WWW-nin geniş yayılması ilə eyni vaxtda düşündürməyə başladı. O zaman ilk *axtarış sistemləri* və *resurs kataloqları* meydana gəldi.





Yahoo! pəncərəsi.

RESURS KATALOQLARI

XX əsrin 90-cı illərində, WWW serverlərini hələ saymaq mümkün olan bir zamanda, *başlanğıc nöqtə* (starting point) adlanan səhifələr mövcud idi. Bunlar mövzu və ya coğrafi əlamətlərinə görə qruplaşdırılmış bütün WWW-serverlərinə istinadların sadalandığı səhifələr idi. Məsələn, "biokimya üzrə bütün WWW-serverləri" və ya "Finlandiyanın bütün WWW-serverləri" (indi çox vaxt "WWW-serveri" termini əvəzinə "sayt" sözü işlənir). Beləliklə, lazımi başlanğıc nöqtəni açaraq bütün istinadları ardıcıl nəzərdən keçirmək mümkün idi. Başlanğıc nöqtələr İnternetin müasir resurs kataloqları üçün ilkin nümunə oldu.

Resurs kataloqları saytlara sistemləşdirilmiş istinadlardan ibarət sorğu kitabçalarına bənzəyir. İstinadlar müəyyən əlamətlərə görə, adətən mövzular üzrə qruplaşdırılır. Hər bir qrupun bir neçə səviyyəsi ola bilər, yəni kataloqlar ağacvari struktura malik olur. Bundan başqa, kataloqlar müxtəlif əlavə xidmətlər təqdim edir: açar sözlərə görə axtarış, son daxilolmaların siyahısı, onlardan daha maraqlı olanların siyahısı. İstinadları kataloqlara, öz kolleksiyalarını hər mövzuda bütün mövcud resurslarla mümkün qədər dolğun etməyə çalışan inzibatçılar daxil edirlər. Sayt sahibləri də öz resurslarına istinadları təklif edir. Kataloq inzibatçıları istinadı yoxlayır və uyğun bölməyə əlavə edir.

Ən "yaşlı" resurs kataloqu Yahoo! (www.yahoo.com) 14 əsas bölmədən ibarətdir:

- Arts & Humanities (incəsənət və humanitar elmlər);
- Business & Economy (biznes və iqtisadiyyat);
- Computers & Internet (kompüterlər və İnternet);
- Education (təhsil);
- Entertainment (əyləncə);
- Goverment (ABŞ hökumətinin resursları);
- Health (səhiyyə);
- News & Media (xəbərlər və kütləvi informasiya vasitələri);
- Recreation & Sport (istirahət və idman);
- Reference (sorğu məlumatları);
- Regional (regionlar üzrə resurslar);
- Science (təbiət elmləri);
- Social Science (ictimai elmlər);
- Society & Culture (cəmiyyət və mədəniyyət).

Resurs kataloqu vasitəsilə axtarış bu qaydada baş verir: istifadəçi axtarılan informasiyanın hansı mövzuya aid olduğunu təyin edir; ağac üzrə dərinə hərəkət edərək kataloqda uyğun bölməni tapır; bölmədə sadalanan bütün səhifələri gözdən keçirir.

Məsələn, Korniş Reks cinsindən olan pişiklər haqqında informasiyanı tapmaq lazımdır. Onda tədricən dərinə gedərək, Science > Biology > Zoology > Animals, Insects, and Pets > Mammals > Cats > Breeds > Cornish Rex bölməsinə keçmək olar. Bu cinsdən olan pişikləri kimin satdığını tapmaq lazımdırsa, onda Business and Economy > Shopping and Services > Animals > Cats > Breeders > Cornish Rex bölməsi kömək edə bilər.

Resurs kataloqundan istifadə etməyin rahatlığı ondan ibarətdir ki, əgər istifadəçiyə axtarılan sənədin mövzusu məlumdursa, o, diqqətini işə aid olmayan kənar sənədlərə yayındırmadan kataloqun müvafiq qolunu araşdıracaq.

Lakin kataloqun həcmi inzibatçılarının imkanları və onların materialı seçmə subyektivliyilə məhdudlaşır. Bundan əlavə axtarılan sənədin mövzusunu hər zaman kataloqun təsnifatı çərçivəsində ifadə etmək olmur. Bu halda axtarış sistemləri köməyə gəlir.

Çox vaxt saytın ünvanı şirkətin adına və ya mövzuya uyğun gəlir. Əgər www.şirkətin_adı.com ünvanına baş çəksəniz, böyük ehtimalla məhz həmin şirkətin saytı açılacaq. Məsələn, www.adobe.com Adobe korporasiyasının saytı, www.cnn.com isə CNN media şirkətinin saytıdır. Əgər baxılan şirkət Amerika şirkəti deyilsə, milli domen sonluqlarını əlavə etmək olar. Məsələn, www.science.az Azərbaycan Milli Elmlər Akademiyasının saytıdır, www.ferrari.it İtaliyanın "Ferrari" konserninə məxsusdur.



AXTARIŞ SİSTEMLƏRİ

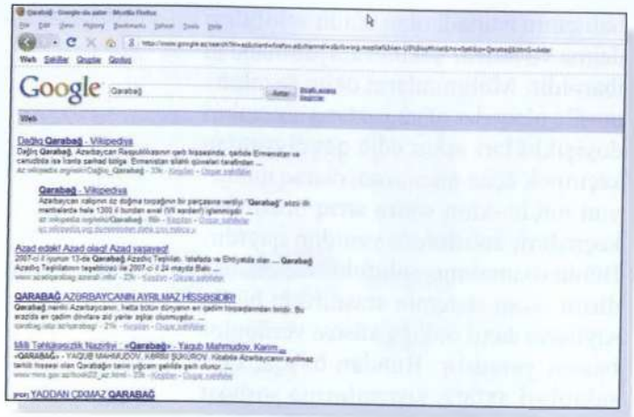
Axtarış sistemlərinin işi, müraciət edə bildikləri bütün İnternet saytlarını, həmçinin istinadı olan bütün səhifələri daima və ardıcıl şəkildə araşdırmaqdan ibarətdir. Məlumatların daim təzələnməsilə əlaqədar olaraq axtarış sistemləri dəyişiklikləri aşkar edib qeydiyyatdan keçirmək üçün müntəzəm olaraq müəyyən müddətdən sonra artıq nəzərdən keçirilmiş səhifələrə yenidən qayıdır. Bütün oxunulmuş səhifələr indeksləşdirilir, yəni sistemin araşdırdığı bütün saytların daxil olduğu xüsusi verilənlər bazası yaradılır. Bundan başqa, sayt sahibləri axtarış sistemlərinə sərbəst şəkildə öz saytları üçün istinadlar əlavə edə bilər.

İstifadəçinin sorğusu daxil olduqda, axtarış maşını öz indeksini araşdırır və İnternetdə axtarış meyarlarına uyğun olan səhifələrin siyahısını göstərir. Tapılmış sənədlər, bir qayda olaraq, mətndə açar sözlərin yerindən (başlıqda, mətnin əvvəlində), rast gəlmə tezliyindən və başqa xüsusiyyətlərdən asılı olaraq sıralanır.

Axtarış üçün çoxlu sayda saytlar mövcuddur. Onlardan bəziləri öz indekslərindən istifadə edir, metaaxtarış sistemləri adlanan digərləri isə öz indeksləri olmadığından başqa axtarış qovşaqlarından nəticələri toplayaraq nizamlayır. İş prinsipində olan oxşarlığa baxmayaraq, axtarış sistemləri sorğu

AltaVista axtarış sisteminin pəncərəsi.





Google axtarış sisteminin pəncərəsi.

dili, axtarış zonası, sənəddaxili axtarışın dərinliyi, nizamlama üsulları və prioritetlərinə görə bir-birindən fərqlənir. Buna görə də müxtəlif axtarış sistemlərindən istifadə etdikdə müxtəlif nəticələr alınır. Lycos (www.lycos.com) və AltaVista (www.altavista.com) ən "yaşlı" axtarış sistemləridir.

Böyük həcmdə informasiyanın araşdırılması və məlumatların vaxtaşırı yenilənməsini axtarış sistemlərinin üstünlüklərinə aid etmək olar. Lakin, bu zaman tərkibində açar sözlər olmayan sənədlər nəzərə alınmır, digər tərəfdən isə siyahıda ələnməsi xeyli vaxt aparan mövzudan kənar çoxlu informasiya olur.

Axtarışa başlayarkən istifadəçi bir və ya bir neçə açar sözü daxil edərək axtarışın tipini seçir. Axtarış sistemlərinin əksəriyyətində üç əsas axtarış tipi mövcuddur: sözlərdən istənilən birinə görə, bütün sözlərə görə və ifadəyə görə dəqiq axtarış. Bundan asılı olaraq nəticələr olduqca fərqlənə bilər.

Axtarış sisteminə sorğunu adi danışıq dilində sual şəklində verməyə cəhd etmək də olar: Məsələn, "Daşınmaz əmlakın sığortalanması xidmətinin qiyməti neçədir?".

NECƏ AXTARMALI

Sözlərdən istənilən birinə görə axtarış
Axtarış nəticəsində tərkibində açar sözlərdən istənilən biri olan bütün səhifələrin siyahısı tərtib olunur. Çox vaxt bu cür axtarış zamanı təsadüflərin sayı olduqca çox olur. Lakin axtarış qovşağı nəticələri mövzuya uyğun yaxşıca nizamladıqda lazım olan səhifəni siyahının başlanğıc hissəsində tapmaq mümkündür. Sözlərdən istənilən birinə görə axtarış istifadəçilərin açar sözlərdə əmin olmadığı halda əlverişlidir.

Bütün sözlərə görə axtarış
Bu axtarış rejimində açar sözlərin hamısının istənilən ardıcıllıqda daxil olduğu bütün səhifələrin siyahısı formalaşır. Burada da mövzuya aidiyyəti olmayan nəticələrin alınması ehtimalı qalır.

Dəqiq ifadəyə görə axtarış
Bu axtarış rejimində tərkibində açar ifadə ilə üst-üstə düşən ifadə olan bütün səhifələrin siyahısı verilir; durğu işarələri nəzərə alınmır. Başqa ifadələr istifadə edilməklə lazım olan mövzuya həsr olunmuş səhifələr siyahıya düşmür. Hətta bu axtarış rejimində belə yanlış nəticələr ola bilər.

Mürəkkəb axtarış
Bir çox axtarış sistemi saytları əlavə axtarış funksiyalarına malikdir. Onlardan bəziləri bunlardır:

Sorğu dilindən istifadə etməklə axtarış. Sorğunu VƏ, VƏ YA, DEYİL məntiqi operatorlarından istifadə etməklə qurmaq, söz və ifadələri əlavə etmək və ya kənarlaşdırmaq imkanı verir.

Kateqoriyalardan istifadə etməklə axtarış. İndeksləşdirilmiş səhifələr çoxsəviyyəli kateqoriya kataloqunda nizamlı şəkildə saxlanır. Lazımi material üçün bütün kateqoriya kataloqunu gözdən keçirmək, yaxud müəyyən səviyyəyə çatdıqdan sonra axtarışı seçilmiş kateqoriya daxilində davam etmək olar. Beləliklə, axtarış sistemi resurs kataloqu ilə birləşdirilir.

Tapılmış səhifələr arasında axtarış. Həddən çox səhifə tapıldıqda daha bir açar sözü daxil edərək axtarışı təkrarlamaq olar. Bunun üçün əksər axtarış sistemlərində tapılan məlumatlar içərisində sürətli axtarışı təmin edən funksiya olur.

Axtarış parametrlərinin dəqiq köklənməsi. Dili seçmək, axtarışı müəyyən domenin saytları ilə məhdudlaşdırmaq, ancaq başlıqlar üzrə axtarış aparmaq, yalnız təqdim etdiyi informasiya müəyyən müddət ərzində təzələnən saytları nəzərdən keçirmək və ya verilən səhifəyə istinadlar olan səhifələri göstərmək imkanları mövcuddur.



AXTARIŞIN GƏLƏCƏYİ

İldən-ilə toplanmış informasiyanın həcmi artır. Uzun müddət ərzində bu artım həndəsi silsilə ilə getsə də, İnternetdə istifadəçilərin istifadə edə biləcəyi faydalı informasiyanın həcmi və orada yerləşdirilməsi mümkün olan informasiyanın həcmi, çox guman ki, məhduddur. Buna görə də axtarış mexanizmlərindən daim effektivliyin artırılması və hesablama resurslarının gücləndirilməsi tələb olunur.

HARADA AXTARMALI

Google (www.google.com)
Ən sürətli və ən böyük axtarış sistemidir. Yarım milyarddan artıq səhifə indeksləşdirilmişdir (onlardan təxminən yarısı tam şəkildə, qalanları isə ünvan və istinad mətni şəklində). İnterfeys dilini seçmə imkanı var. Müəyyən sayt və ya domenlərdən alınmış nəticələri əlavə etmək və ya kənarlaşdırmaq olar. Axtarış sistemlərinin çoxundan fərqli olaraq, Google həmçinin başqa səhifələrdən olan istinadların sayına görə də resursun populyarlığını qiymətləndirir. Bundan başqa, burada son 20 il ərzində toplanmış USENET sisteminin bütün telekonfransları üzrə axtarış aparmağa imkan verən arxiv də mövcuddur.

Yandex (www.yandex.ru)
Yəqin ki, Rusiya istehsalı olan axtarış sistemləri arasında ən yaxşısıdır. Rus dilinin söz formalarını nəzərə alaraq keyfiyyətli axtarışı həyata keçirməsilə məşhurlaşıb. Əsasən rusdilli resursları siyahıya alır. Bu zaman imkanlarına görə xarici sistemlərdən heç də geri qalmır. Axtarışı dəqiq söz və ya onun istənilən forması, tarix üzrə, həmçinin göstərilən sayt və ya onun alt-direktoriyası üzrə aparmaq olar. Sitat indeksini nəzərə almaqla axtarış aparmaq, sənədin dilini də göstərmək mümkündür.

AltaVista (www.altavista.com)
Ən "yaşlı" axtarış sistemlərindən biridir. Geniş əhatəli axtarış meyarları təqdim edir: məsələn, Advanced search bölməsində resursun yaradılma və ya dəyişdirilmə tarixini göstərmək olar, 25 dil dəstəklənir; hər sayt üçün bir nəticənin çıxarılması imkanı var (bu keyfiyyətə zərər vermədən axtarış sahəsini məhdudlaşdırır).

Yahoo! (www.yahoo.com)
İnternetdə ilk resurs kataloqlarından biridir. Yahoo!-da böyük bir strukturlaşdırılmış kateqoriya kataloqu tərtib edilmişdir. Axtarış əvvəlcə orada, sonra xüsusi arxivdə, sonra isə Google axtarış sisteminə müraciət etməklə aparılır. Kateqoriyalar üzrə axtarış yaxşı nəticələr verir: onların sayı çox olmadığından meyarlara uyğunluq daha tam olur. Standart funksiyalar dəstindən başqa, resursları tarixə görə seçmək, açar sözlərdə simvolların istənilən ardıcıllığı əvəzinə "*" işarəsini qoymaq imkanı var.

Rambler (www.rambler.ru)
Rusiyanın ilk axtarış sistemlərindən biridir. Standart axtarış imkanlarından əlavə saytda resursların reytinq-kataloqu mövcuddur.

Lakin dünən tapılmış informasiyanın bu gün də həmin yerdə olacağına və məhv edilməyəcəyinə heç bir zəmanət verilmir. Vaxtilə şəbəkədə yerləşən bir çox məlumatlara indi müraciət etmək olmur. Bu, axtarış üçün yeni üfüqlər açır – İnternetdə arxeologiya (itirilmiş informasiyanın tapılması üçün "qazıntılar") meydana çıxır. Ayrı-ayrı şəxslər və təşkilatlar az qala hər şeyin İnternetdə saxlanmasına cəhd göstərsələr də, texnikanın hazırkı inkişaf səviyyəsi nəzəri olaraq buna imkan versə də, hələ ki, həyata keçirilməsi çox bahadır. Nanotexnologiyaların daha da inkişaf etməsilə bu məsələnin həll olunacağı gözlənilir. Hələlik isə sizi maraqlandıra biləcək hər nə varsa, hamısını dərhal kompüterinizə köçürmək məsləhətdir.

Daha bir problem ondan ibarətdir ki, istifadəçilər nəyi axtarmaq istədiklərini ifadə etməkdə çətinlik çəkirlər. Beləliklə, gələcək axtarış sistemlərinin xüsusiyyətlərindən biri də güman etmək olar ki, dəqiq ifadə olunmamış sorğular üzrə məlumatların verilməsi imkanı olacaq. Məsələn, şəkillərin axtarışını götürək. Müasir axtarış sistemləri şəkli onun sözlə təsvirinə görə tapa bilmir, axtarışı kontekstə görə və ya sadəcə faylın adına görə aparmağa çalışırlar. Gələcəyin axtarış sisteminə isə "Rafaelin Madonnasına bənzəyən qızın şəkilini tap" demək kifayət olacaq.

"ELEKTRON HÖKUMƏT"

"Elektron hökumət" qurmaq ideyası 1999-cu ildə, Goverment Online adlanan ümumavropa layihəsinin başlaması ilə meydana gəldi. O, 20-dən çox ölkənin hökumətlərini birləşdirdi. Bu layihənin əsas məqsədi hökumət, vətəndaşlar və biznes arasında birbaşa elektron xidmətlərin (*ing.* On-Line) qurulması istiqamətində birgə əməkdaşlıqdır.

İnternetin yaranması ilə meydana gəlmiş texnologiyalar insan həyatının heç bir sahəsindən yan keçmədi və XXI əsrin astanasında dövlət strukturlarına da daxil olmağa başladı.

Xoşbəxtlikdən "Elektron hökumət" (*ing.* "E-Government") yazıçı-fantastların sevimli mövzusu – "Terminator" filmində baş verdiyi kimi kompüter superbeyninin yer kürəsini zəbt edərək bəşəriyyəti zülm altında əzib planlı şəkildə məhv edən robot tiraniyası qurduğu bir gələcəyə heç bənzəmir.



"Elektron hökumət" avtomatlaşdırılmış idarəetmə üçün kompüter sistemi deyil. "Elektron hökumət" dövlət orqanları və hökumətin İnternet saytları və hamı üçün açıq verilənlər bazalarının toplusudur. Bu, əhaliyə xidmətin yaxşılaşdırılmasına yönəldilmiş dövlət proqramlarıdır. Məsələn, sosial elektron kartlar, distant təhsil və s.

"Elektron hökumət" ilk növbədə vətəndaşlara kömək məqsədilə yaradılır. İnternetin verdiyi geniş rabitə imkanları vasitəsilə "Elektron hökumət" siyasi aşkarlığı təmin etməyə, məmur süründürməçiliyi və bürokratizmi, dövlət orqanlarına çox vaxt aparan və bəzən mənasız olan "səfərləri" minimuma endirməyə çalışır. "Elektron hökumət"in təqdim etdiyi bütün xidmətlərin əsas

hissəsi əhalini məlumatlandırmaqdır: gündəlik xəbərlər və hüquq məsləhətləri, qanunvericilik üzrə verilənlər bazaları.

Hökumət orqanlarının saytlarında telekonfrans qrupları yaradılır. Bu, əhalini siyasi diskussiyalara, ölkənin siyasi həyatında iştiraka cəlb etməyin bir üsuludur. Belə ki, vətəndaşların konfranslarda bildirdiyi fikirlər müvafiq hökumət xidmətləri tərəfindən diqqətlə öyrənilir və təhlil edilir. Demək olar ki, vətəndaşlar evdən çıxmadan siyasi həyatda iştirak edirlər.

Elektron ərizə və vergi bəyannaməsi planetin istənilən nöqtəsindən şəbəkə ilə göndərilə bilər. Şübhəsiz, poçtla məktub göndərmək məmurların, vergi müfəttişlərinin şəxsən yanına getməkdən xeyli rahatdır.

Doğrudur, ənənəvi olaraq məktublar elektron şəkildə deyil, çap və ya əlyazma şəklində göndərilir: "qələm yazanı qılınc kəsməz". Vətəndaş kağıza, poçta, məktubun çatdırılması barədə bildirişə inanır. Göndərilmiş kağızı tullaya bilməzsən! Lakin elektron poçtun çatdırılma vasitələri də, adi poçt və ya məktubları qeydiyyatdan keçirən katiblik kimi və hətta ondan da etibarlı və operativ olmaqla elektron məktubların çatdırılmasına və qeydiyyatına zəmanət verə bilər. Qanunvericilik isə elektron məktubu vətəndaşın adi müraciətilə bir tutur. Bundan başqa, işlərinin necə getməsini öyrənmək üçün vətəndaşın müxtəlif instansiyalara zəng etməsi lazım gəlməyəcək. Bunu kompüter vasitəsilə İnternetdən "görmək" olar.

Elektron yazışmaların çatdırılması vasitələri kənardan müdaxilə olunma baxımından daha zəif görünür – məxfi məlumatları asanlıqla oğurlamaq olar. Telefon xəttində olduğu kimi şəbəkəyə istənilən adam qoşulub xətti "dinləyə", informasiyanı oğurlaya bilər, məsələn, vətəndaşın gəlirləri barədə məlumatları öyrənə bilər. Lakin məxfi məlumatlar ötürülərkən şifrlənmə alqoritmləri tətbiq olunur. Bu zaman cinayətkar ötürmə xəttinə "gizlicə" qoşula bilsə belə, o, şifrlənmiş məlumatdan heç bir faydalı informasiya əldə edə bilməyəcək. Əldə etdiyi isə sadəcə cəfəngiyat olacaq və vergi bəyannaməsi qərəzkarların əlinə keçməyəcək.

Gələcəkdə elektron xidmətlərin əsas prinsipi olan "One-Stop Service" müxtəlif bürokratik səviyyələrdə vətəndaşın xahişini, bəzi hallarda onun özünün iştirakı olmadan, müzakirə etməyə imkan verəcək. Ərizələr üçün imzalar toplamaq, müxtəlif instansiyaları gəzmək lazım olmayacaq. Sadəcə e-mail ilə elektron ərizə vermək kifayətdir ki, "sənəd" özü idarədə bütün səviyyələrdən keçsin. İdeal vəziyyətdə vətəndaş sənədin instansiyalardan keçməsinə evdən nəzarət edə biləcək. Hər bir müraciət unikal nömrə alır. Bu nömrə-kod vətəndaşa məktubun qeydiyyatdan keçərək baxılmaq üçün qəbul olunduğunu bildirən məlumatla birgə göndərilir. Bu koddan istifadə edərək ərizənin bütün "tarixçəsini" asanlıqla izləmək olar.



AVROPADA VİRTUAL TƏHSİL

2001-2002-ci tədris ilindən başlayaraq Avstriyada məktəblərin bütün səviyyələrdə (ibtidai və orta məktəblər, gimnaziyalar, peşə təmayüllü məktəblər) sürətlə kompüterləşdirməsi başlandı. Vyana bələdiyyəsi İnternetdən istifadə, proqram təminatı və fərdi kompüterlərin alınması üçün 400 məktəbə 17 milyon dollara yaxın vəsait ayırdı.

Avstriyanın Təhsil Nazirliyi məktəblərdə yaxın gələcəkdə dərslikləri, dəftərləri, qələmləri və başqa köməkçi tədris vasitələrini əvəz edən daşınan kompüterlərin geniş tətbiqini planlaşdırır.

Avstriya məktəblərində kompüterlərin tam yeniləşdirilməsi, müxtəlif avadanlıqların alınması və İnternetə azad çıxışı təmin etmək, həmçinin müəllimlərin ixtisaslarının artırılması üçün ümumilikdə təxminən 70 mln dollar xərclənəcək. Universitetlərdə bütün tələbələrin mühazirə və seminarlarda işləmək üçün şəxsi kompüterlə təmin olunması nəzərdə tutulur. 2001-ci ilin payızında aparıcı İnternet provayderləri evdə İnternetə azad bağlantı quraşdırmaq istəyən tələbələr üçün xüsusi qiymətlər təklif etdilər.

Artıq bir normaya çevrilmiş virtual aləm tədrisdə bürokratik prosedurları asanlaşdırmağa və hər cür maneələrdən xilas olmağa imkan verir: kağızsız idarəetmə, seminarlara yazılma, yoxlama işləri, imtahanlar, video-konfranslar, professor və müəllimlərlə On-Line məsləhətlər.

Avstriyada yeni informasiya texnologiyaları sahəsinə ilk cığırı açanlar Vyana İqtisadiyyat Universiteti və Linse şəhərində yerləşən Kepler Universiteti oldu. Vyana Universitetində tələbələrin 30%-dən çoxu məşğələlərə və seminarlara İnternet vasitəsilə yazıldı. (Avstriyada gimnaziyanı bitirəndən sonra ali təhsil müəssisələrinə qəbul, qəbul imtahanı verilmədən aparılır.)

Kataloniya Açıq Universitetində vebtələbələrin sayı 1995-ci ildə 400 nəfərdən 2000-ci ildə 20 minədək artmışdır. Hagen şəhərində (Almaniya) yerləşən universitet 26 ildir ki, virtual tədrisdə birinciliyi qoruyub saxlayır və hazırda orada 60 min tələbədən 10 mini İnternet vasitəsilə təhsil alır.

Distant təhsilə yaxşı bir nümunə kimi Böyük Britaniyada Milton Keyns Açıq Universitetini göstərmək olar. Universitetin dünyanın dörd bir yanından 200 min vebtələbəsi var, 1992-ci ildə isə Vyanada filialı açılmışdır. Açıq universitetin tələbələri müəllimlərlə İnternet vasitəsilə ünsiyyət qurur, bundan başqa CD ROM-da təqdim olunan tədris materiallarını öyrənirlər. Hər altı həftədən bir tələbələr birgə məşğələlər üçün auditoriyalarda görüşür, ildə bir dəfə isə birgə praktika keçirlər.

Milton Keyns Universitetində ənənəvi tədris üsullarını əvəz edəcək "T171" tədris kursu işlənib hazırlanaraq tətbiq olunur. Bu kurs çərçivəsində tədris İnternet vasitəsilə həyata keçirilir: hər bir tələbə öz müəllimilə E-mail vasitəsilə daimi əlaqə saxlayır.

Çox vaxt bir məmurun "vizasını" aldıqdan sonra, başqa bir dövlət orqanına yollanmaq lazım gəlir. One-Stop Service bu problemi də həll edir. Vətəndaşların məktubları məmurlar və hətta müxtəlif şöbələr arasında elektron şəkildə ötürülür. Vətəndaş möhürlü kağız sənəd kimi eyni qüvvəyə malik rəsmi cavabı E-mail ilə alacaq.

XX əsrin son ili "Elektron hökumət"in yaranması ili kimi qeyd edildi. Tam uğura hələ çox var. Müvafiq qanunvericilik aktlarının qəbul edilməsi lazımdır. Nəzərə almaq lazımdır ki, müxtəlif ölkələrdə kommunikasiya (İnternet) vasitələrinin inkişafı, hökumət idarələrinin uyğun proqram sistemləri (uçot, kargüzarlıq və verilənlər bazası sistemləri) və kompüterlərlə təchiz olunması müxtəlif səviyyədədir. Bu sahədə aparıcı ölkələrdən biri Avstriyadır.

"AVROPADA E-AUSTRİA"

14 aprel 2000-ci il tarixində Avstriya hökuməti 5 il müddətində həyata keçirilməsi nəzərdə tutulan "E-Austria in E-Europe" proqramını dərc etdi. Layihənin əsasını elektron təhsil və "Elektron hökumət" təşkil edirdi. "Elektron hökumət" xidmətlərindən yararlanmaq üçün təkcə yazıb-oxumağı deyil, həm də kompüterdən istifadə etməyi bacarmaq lazımdır. Buna görə də Avstriya hökuməti yaşlı nəslin təhsili məsələlərinə ciddi diqqət yetirir.

26 sentyabr 2000-ci il tarixində hökumət "Avstriyanın virtual bazarı" layihəsini təqdim etdi. Artıq İnternetdə siyasət, idarəetmə, hüquq, iqtisadiyyat, texnika, ətraf mühit, təhsil kimi sahələrdə informasiya almaq üçün azad giriş yaradıldı.

2003-cü ilin ortalarında hər bir avstriyalı sosial və tibbi sığorta vərəqəsini əvəz edəcək plastik kart (vətəndaşın elektron kartı, *ing.* "chipcard") aldı. Bu kartların işlənib hazırlanması layihənin əsas bəndlərindən birini təşkil edirdi. Bu işdə Avstriya tək deyil. Əksər Avropa ölkələrində də vətəndaşlar üçün elektron kartlar hazırlanır.

"Elektron hökumət" xidmətlərinin tam miqyasda tətbiqi, müvafiq dövlət idarələrinin işçiləri üçün kifayət qədər yüksək tələblərin qoyulmasını nəzərdə tutur. İşçi öz sahəsində mütəxəssis olmaqla bərabər kompüter və departamentdə istifadə olunan proqram təminatı ilə də işləməyi mükəmməl səviyyədə bacarmalıdır. Dövlət idarələrinin bütün işçiləri kompüterdə iş bacarığını təsdiqləyən ECDL (*ing.* "European Computer Driving Licence") sertifikatı almalıdırlar. Bu məqsədlə imtahanlar keçirilir; bu imtahanları keçməyənlər tutduqları vəzifəyə buraxılmayacaqlar. Hər kəsin kompüter savadı olmalıdır. Buna görə hər bir vətəndaş istədiyi zaman ECDL sertifikatı ala bilər.

2001-ci ilin sonunda On-Line rejimində vergi bəyannamələrinin təqdim edilməsi və vergilərin çıxılmasının pilot-layihəsi başlandı. Avstriyanın Maliyyə Nazirliyi 2003-cü ildən gec olmamaq şərtilə bütün Avstriya vətəndaşlarına sistemə sərbəst müraciəti təmin etməyi nəzərdə tuturdu. Artıq 2004-cü ildə müxtəlif sənədlərin böyük bir hissəsi (pasport, sürücülük vəsiqəsi və s.) şəbəkə vasitəsilə qeydiyyatdan keçirilirdi. 2005-ci ildə əhaliyə istənilən xidmətlər şəbəkə vasitəsilə göstərilirdi.

One Stop Service xidmətlərindən istifadə etmək üçün ölkənin bütün vətəndaşları (evdə və iş yerində) İnternetə geniş və sərbəst çıxışla təmin olunmalıdır. İnternetə çıxışı olmayanlar üçün isə açıq kompüter terminalları yaradılacaq. Paralel olaraq əlillər üçün də proqramlar inkişaf etdirilir. Məsələn, hazırda korlar üçün ECDL üzrə layihə işlənib hazırlanır.

ELEKTRON KİTABXANALAR

Adi çap nəşrlərini onların elektron forması ilə əvəz etmək ideyası çoxdan yaranıb. Yazıçı-fantastlar bunu dəfələrlə təsvir ediblər. Apple şirkətinin prezidenti Con Skallinin (John Skally) 90-cı illərdə Moskvaya ilk səfərində Moskva Dövlət Universitetində oxuduğu mühazirə zamanı o vaxtlar hələ mövcud olmayan üçölçülü qrafika ilə yanaşı oxumağı bacarmayanların təlim keçməsi üçün nəzərdə tutulmuş danışan elektron kitablar nümayiş etdirilən futuristik bir reklam çarxı da göstərilmişdi.

Lakin elektron kitabxana ideyasının tətbiqi üçün elektron kitabların harda və necə əldə ediləcəyini bilmək lazımdır.

Elektron kitabxanalara necə abunə olmaq olar? Bunun qiyməti nədir? Necə oxumaq olar? Sonuncu suala cavab vermək daha asandır – kompüterin monitorundan. Lakin çoxları iş gününün sonuna kompüterdən cana gəlir. Bundan başqa,



Alan Key.

Alan Key 1968-ci ildə kitabı əvəz edən ilk portativ interaktiv kompüter – DynaBook-u işləyib hazırladı. 1981-ci ildə isə ilk dəfə olaraq ədəbiyyatla, yəni elektron kitabların satışı ilə məşğul olmağı planlaşdıran Xanadu işçi qrupu yaradıldı.

belə qiraət rahat deyil; birincisi, gözlər yorulur, ikincisi isə metro və ya avtobusda oxumaq olmur.

Buna görə alternativ kimi PocketPC, PDA (personal digital assistant – "fərdi rəqəmli köməkçi") – cib ölçülərində cəmi 200 qr çəkisi olan elektron yazı kitabçalarını göstərmək olar (Düzdür, bahalı olduqlarından elektron yazı kitabçaları o qədər də geniş yayılmayıblar). Bu cırtdanın yaddaşında onlarla kitab, telefon kitabçası, elektron poçt, kiçik verilənlər bazaları, elektron cədvəllər, oyunlar və s. saxlana bilər. Təbii ki, bu "əl" kompüterlərindən böyük əksəriyyətini İnternetə qoşmaq olar.

XX əsrin sonunda artıq elektron-kitab adlandırıla biləcək qurğular istehsal olunurdu. ABŞ-da elektron kitabların oxunması üçün xüsusi qurğu işlənib hazırlanarkən onlarda elektron kitab mağazası ilə pullu bağlantı da nəzərdə tutulub. SoftBook Reader qurğusu (SoftBook Press şirkəti) üzlüyü təbii dəridən olan noutbuka bənzəyir (çəkisi 1,3 kq, ekranın ölçüsü – 20x15 sm).

Elektron-kitabın qiyməti SoftBookstore kitab mağazasının 2 illik abunə xidmətilə (ayda 20 dollar) birlikdə bir



neçə yüz dollara çatırdı. Kitabın yaddaşı 5000 səhifəyə kimi mətn saxlaya bilir, batareyaları isə 5 saat qiraət üçün kifayət edirdi. Hər alıcıya 100 klassik bədii əsər pulsuz verilirdi. Qalan kitablar üçün isə o, pul ödəyirdi (orta qiymət 10–25 dollar idi).

Rocket eBook-un (NuvoMedia şirkəti; çəkisi 600 qr, displeyi 11,5x7,5 sm) qiyməti SoftBook Reader-dən daha ucuzdur. Batareyası bir sutkadan artıq müddətdə işlək vəziyyətdə qalır, yaddaşının tutumu isə 3 min mətn səhifəsindən bir qədər çoxdur. Rocket eBook dəstində alıcı hədiyyə kimi "Alisa Möcüzələr ölkəsində" və 75 min söz həcmində olan Random



House Webster's eDictionary kitablarını alırdı.

XXI əsrdə elektron kitablar (oxuma qurğuları) demək olar ki, hər gün daha da təkmilləşir. Təbii ki, İnternet-mağazadan elektron kitabı yüklomək adi kitabı almaqdan daha ucuz başa gəlir. Şəbəkədən alınmış kitabların, adətən, sürət çıxarmaya qarşı müdafiəsi olur. Bəzən bu müdafiə o qədər sərt olur ki, kitabın oxunmasını çətinləşdirir və ya ümumiyyətlə qeyri-mümkün edir.

1971-ci ildə İllinoys Universitetinin (ABŞ) professoru Maykl Hart (Michael Hart) "Qutenberq layihəsi"ni işləyib hazırladı.

Xerox Sigma V (Material Research labaratoriyası) kompüterində bir saat işləməyin qiyməti 70 ABŞ dollarından baha idi. Lakin belə bahalı maşın heç də hər zaman işlə tam yüklənmirdi. Onda Maykl Hart maşında hesablamalar aparılmayanda onun "kitabxanalarında" olan məlumatın saxlanması, ötürülməsi və axtarışı üçün bir üsul təklif etdi.

"Qutenberq layihəsi"nin əsas tezisi ondan ibarət idi ki, kompüterə daxil olunmuş hər bir informasiyanın qeyri-məhdud sayda surətini çıxarmaq olar. Müəllif bu xüsusiyyəti – *replikasiya texnologiyası* (*ing.* replicator technology)



Harvard məzunu Teodor Nelson (Theodor Nelson) 1965-ci ildə Xanadu layihəsi daxilində dünya ədəbiyyatının toplanması üçün əsas olan *hipermətn* (hypertext) terminini irəli sürdü. Layihənin məqsədi elektron kitab arxivlərinə rahat qoşulmanı təşkil etmək idi. Bu zaman müəllif hüquqları pozulmur və müəllif qonorarının hesablanması sistemi işləyirdi. Əslində bu, elektron kitab mağazasının ilkin nümunəsi idi. Xanadu 1988-ci ildə Autodesk şirkətinin qurucusu Con Uokerdən (John Walker) maliyyə dəstəyi aldı. Lakin 1992-ci ildə dünya hörümcək torunun meydana gəlməsilə layihə, demək olar ki, unuduldu.

adlandırdı. İstənilən insan, harda olmasından asılı olmayaq, elektron kitabı (şəkil və ya səs daxil olmaqla maşında saxlanan hər bir məlumatı) öz kompüterinə köçürə bilər.

Buna görə də istənilən kompüter platformasında oxuna bilinməsi üçün, ən sadə elektron mətn formatı (ASCII) seçildi.

Hamıya aydın idi ki, layihənin parlaq gələcəyi var, lakin proses çox asta templərlə gedirdi. 90-cı illərin əvvəllərinə sadəcə bir neçə yüz mətn toplanmışdı. İnternetin inkişafı "layihəyə" tək oxucuları deyil, həmçinin könüllü mürəttib-operatorlar ordusunu cəlb etdi. Maqnit informasiya daşıyıcılarının qiyməti də bir xeyli ucuzlaşdı. Mətnlərin sayı sürətlə artmağa başladı.

Lakin pulsuz işçi qüvvəsinə arxalanan bu layihə yalnız klassik əsərlərin



Teodor Nelson.



Johann Qutenberq – kitab çapının alman ixtiraçısı (solda). Maykl Hart (sağda).

"Müəlliflik hüququ" (*ing.* copyright) anlayışı hərfi mənada "surət köçürmə hüququ" deməkdir. Sadə dillə desək, müəlliflik hüququ əslində müəlliflik hüquqları ilə qorunan məhsulun bir nüsxəsini almış şəxsə deyil, bu hüququn sahibinə aid olan köçürmə hüququdur.

yayılmasını nəzərdə tuturdu. Ona görə də mətnlərin orijinala tam dəqiq uyğun gəlməsi xüsusilə aktual idi.

Professor Hart yazırdı: "Bizim məqsədimiz əksər oxucuların nəzərində 99,9% dəqiq olan elektron mətnlər hazırlamaqdır."

Buna baxmayaraq, "Qutenberq layihəsi" müxtəlif ölkələrdə bir çox dövlət məktəblərinə kitabxanalarını minlərlə klassik ədəbiyyat nümunələrilə zənginləşdirmək imkanı yaratdı.

Kopirayt bütün kompüter kitabxanalarının (belə layihələr dünyanın başqa ölkələrində də mövcuddur) qarşısında duran problemlərdən biridir. Çox vaxt bu, heç də müəlliflik hüquqlarının deyil, sadəcə müəllif işinin köçürülməsi hüquqlarının başqaları tərəfindən istifadəsini təmin edir. Bu gün məhz bu məsələ Maykl Hartı narahat edir.

Müəllif əsərlərinin hər kəs üçün açıq olması müddətinin artırılması (ABŞ-da əvvəl 14, sonra 28 və nəhayət, bu yaxınlarda 70 il) qəmli düşüncələrə gətirməyə bilməz. Kopirayt şərtlərinin sərtləşdirilməsi ümumi elektron kitabxanaların mövcudluğunu şübhə altına qoyur. Belə davam edərsə, tezliklə bütün informasiya kopiraytın təsiri altına insan ömründən uzun olan bir müddətə düşə bilər.

İNTERNET VƏ CƏMİYYƏT

Böyük kəşflər və ixtiralar cəmiyyətə ciddi təsir göstərərək, onun simasını həmişəlik dəyişir. Məsələn, təkər, kitab çapı, dəmir yolu, elektrik cərəyanı, televiziya və kompüter belə ixtiralardandır. Onlardan bəzilərini bəşəriyyət dərhal qiymətləndirdi, digərlərinin dəyəri isə əsrlər keçdikdən sonra başa düşüldü.

İnternet, yəqin ki, XX əsrin ən parlaq ixtiralarından biridir. Qısa bir müddət ərzində o texnoloji hadisədən sosial bir hadisəyə çevrildi. Qlobal kompüter şəbəkəsinin qurulmasını mubaliğəsiz dünya inqilabı kimi qiymətləndirmək olar.

İnternet günbəgün deyil, sözün əsl mənasında, dəqiqəbədəqiqə inkişaf edən bir planetar strukturdur. O, istənilən tip informasiyanı – adi mətn, şəkil, audio və videoinformasiyanı yaymaq üçün ən əlverişli və ucuz vasitədir.

Əgər yaxın keçmişdə fərdi kompüter hesablamalar və sənədlərlə işləmək üçün istifadə olunurdusa, indi o, insanlar arasında kommunikasiya vasitəsi, dünyanın hər yerindən istənilən informasiyanı əldə etmək və hətta biznes fəaliyyəti göstərmək üçün də bir alətə çevrilmişdir. Şübhəsiz, gec və ya tez İnternet hər evə gələcək.

İNTERNET VƏ TELEVİZİYA

Hər bir yeni ixtira ilə müəyyən dəyişiklikləri əlaqələndirirlər. Fotoqrafiya meydana gəldikdə deyirdilər ki, o, rəssamlığı əvəz edəcək; kinematoqrafın isə teatrı sıxışdırıb çıxarılacağından ehtiyat edirdilər. Bir vaxtlar hesab edirdilər ki, televiziya radionun yerini tutacaq, lakin bu baş vermədi. İnternetlə də vəziyyət belə deyilmi?

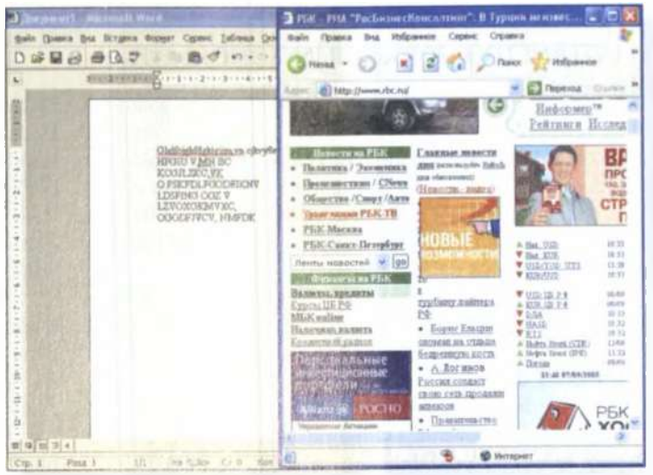
Sosioloji tədqiqatların nəticələri göstərdi ki, sorğuda iştirak edənlərdən evində İnterneti olanların təxminən üçdə biri vaxtını televizor qarşısında deyil, şəbəkədə keçirməyə üstünlük verir. Cavanların təxminən yarısı bu cür fikirləşir. İnternetdə olan və eyni zamanda televizora baxanların sayı rəyi soruşulanların 10%-ni təşkil edir.

İnternetin televiziyanı tamamilə əvəz edəcəyindən danışmaq olarmı? Buna hələ çox var. Gələcəkdə də bunun baş verəcəyi güman edilmir, çünki, nəzərə almaq lazımdır ki, şəbəkədən informasiyanın əldə olunması vaxt sərfi, müəyyən qədər səy və diqqət tələb edir. Eyni zamanda televiziyanın üstünlüklərindən biri də odur ki, insan evə gəlib kürsüdə rahat əyləşərək istirahət edir və televizorda göstərilən verilişlərə baxır. Amma indi artıq bəzi televiziya kanalları öz proqramlarını şəbəkədə təklif edir.

İNTERNET VƏ KİTABLAR

Belə fərziyyə var ki, İnternet kitabları sıxışdıraraq onların yerini tutacaq. İndi buna bir sıra faktorlar mane olur. Birincisi, kitabları kompüterin monitorundan oxumaq narahatlığının olmasıdır. Elektron mətni oxumaq üçün portativ qurğuların mövcud olmasına baxmayaraq, (heç bir şübhə yoxdur ki, onlar məhz portativ olmalıdır) onların ekranlarının çözümlülüyü (*ing.* resolution) kifayət qədər yüksək deyil. Nəşriyyat sistemlərində istifadə olunan xüsusi yüksək çözümlülüklü monitorlar isə çox yer tutur və olduqca bahadır. İkincisi, elektron nəşrdə müəllif hüquqları axıra qədər işlənməyib. İstənilən halda yalnız müəllif öz əsərinin yayılması şərtlərini təyin edə bilər. Buna görə də icazəsiz köçürməyə qarşı etibarlı müdafiə olmalıdır.

XX əsrin axırlarında məşhur yazıçılar – amerikalı Stiven Kinq (Stephen King) və ingilis Frederik Forcayt



Bu ekran görüntüsü bir pəncərənin digəri üzərində açılmasını nümayiş etdirir.

(Frederick Forsyth) öz növbəti kitablarını yalnız elektron şəkildə nəşr etdirdilər. Axı kitabı İnternetdən yükləmək üçün simvolik bir məbləğ ödəmək lazım idi. Ola bilər ki, zaman keçdikcə kağızda yalnız xüsusi hədiyyə və ya kolleksiya üçün nəzərdə tutulan kitablar çap olunsun. Elektron kitabların üstünlükləri aşkardır: çatdırılma sürəti (kitabxanaya və ya mağazaya getmək lazım deyil), kiçik həcm (çoxcildli əsərlər toplusu balaca bir diskdə yerləşir), mətndə rahat axtarış. Başqa vacib məsələlərdən biri də odur ki, kağız nəşrlərin ümumi həcminin azalması meşələri kəsilməkdən xilas etməyə imkan verəcək.

İNTERNET VƏ KÜTLƏVİ İNFORMASİYA VASİTƏLƏRİ

İnternetlə təklif olunan texnologiyalar KİV-də maksimal effektli inkişafa və tətbiqə gətirdi. Bu, bir neçə səbəblə izah olunur, onlardan biri – operativlikdir. Müəllifdən başqa televerilişin hazırlanmasında operatorlar, rejissorlar, səs rejissorları, səhnə işıqçısı, montajçı və başqaları işləyirlər. Qəzet məqalələri üzərində müəllif, redaktor, korrektor,



Stiven Kinq.



Frederik Forsayt.

İNTERNETDƏ 2010-cu İLİN SƏHƏRİ

Müasir informasiya texnologiyaları həyatımıza nə gətirdi? İlk növbədə onlar insanları İnternet adlanan bir cəmiyyətdə birləşdirdi. Bəzilərinin qlobal kompüter şəbəkəsinin xidmətlərindən hələ də istifadə edə bilməməsi eyib etməz. İnternetin hər evə, hər ailəyə gəlişi uzaqda deyil. Bu, bir neçə ilin işidir. Bəs bu necə baş verəcək? Gəlin, bir qədər xəyala dalaq.

Adi bir şəhər. Ata, ana və iki uşaqdan ibarət adi bir ailə. Ata zavodda mühəndis işləyir, ana modelyerdir. Qız universitetdə oxuyur. Oğlan isə məktəbə gedir və kompüter dəlisidir.

Səhərdir. Ailə yuxudan oyanır. Birinci, həmişə olduğu kimi, ata oyanır. Dərhal da qəzetə əl atır. Lakin poçt qutusunu yoxlamaq üçün aşağı düşmür, qəzet köşkünə də getmir. O, nazik yastı ekranı götürüb səhər xəbərlərini oxuyur. Ata barmağı ilə ekrana toxunur. Növbəti səhifəni nəzərdən keçirir. Yenə toxunur – başqa bir qəzet. Səhifələr bir-birini sürətlə əvəzləyir. Sevimli zavodu haqqında xəbərlər toplusu. Elektron qəzeti oxuduqdan sonra ata onu bükərək pijamasının cibinə qoyur.

Bu da ana, o saat 9.00-da işdə olmalıdır. O tələsir. Uşaqların otağında onun holoqrafik təsviri yaranır. O, uşaqları mətbəxə çağırır. Hamı səhər yeməyini yeyir. Eyni zamanda VEB-TV-də – İnternet-televizorda xəbərlərə baxırlar. Bütün divarı tutan ekranda əsas verilişdən başqa bütün vacib xəbərlər hərəkət edən sətir şəklində yazılır, küncdə isə hava haqqında məlumat və müxtəlif reklamlar göstərilir. Onlar artıq verilişi yarımçıq kəsmədən televizorun yuxarı hissəsində bir kənarda sakitcə "asılırlar".

Xəbərlər ürəkaçan deyil. Kibercinayətlərin sayı artmaqdadır. Proqramlar və verilənlər bazaları oğurlanır. Ekrandan diktor yenə ev kompüterlərinə soyğunçu hücumlar edərək ailə verilənlər bazalarını məhv edən cinayətkar bir qrup haqqında məlumat verir. Qonşunun bazasında mühafizə sistemi yox idi, ona görə də gecə onu "qaçırıblar". İctimaiyyət informasiya edamının tətbiqini tələb edir: quldurlar şəbəkədən uzaqlaşdırılmalı və "loginlər"dən məhrum edilməlidirlər. Ata onca il əvvəl olan sakitçiliyi yada salır: əvvəl də soyğunlar olurdu, avtomobillər oğurlanırdı, indi isə artıq kibermühitdə hücum edirlər. Bütün pulları əlinizdən alıb, faylları pozurlar. Avtomobillər və insanların özləri artıq heç kimi maraqlandırmır. Daha əvvəlki cinayətlər yoxdur. Ana narazı-narazı başını yelləyir, uşaqlar şəbəkəyə gecə tək çıxmayacaqlarına və "fayrvol"u (daxili şəbəkəni qoruyan mühafizə sistemi, *ing.* firewall) heç kimə açmayacaqlarına söz verirlər.



Ana səhər yeməyini bitirir və iş yerinin yerləşdiyi otağa keçir. Burda o, yeni geyim modellərinin layihələşdirilməsilə məşğul olur. Ana kompüteri qoşur. Moda aləmində nə yenilik var? Sifarişlərin gəldiyi poçtu yoxlayır. Məktublara cavab yazır. Sifarişçilərlə söhbət edir. Yeni fransız kolleksiyasından olan paltarın üçölçülü holoqrafik modelinə baxır. Budur öz modelinin ülgüsü ekrandadır. Ana maşına komanda verir: "Yığırıq". Modelin hazır görünüşü ekranda əmələ gəlir. Paltarın qolunu bir qədər düzəldək, ülgünü dəyişək. Ana əllərilə hərəkətlər edir, model isə öndən, arxadan göz önündə dəyi şir. Materialı, rəngi dəyişək. Hər şey hazırdı. Tikmək olar. Model istədiyi kimi alındı.

Budur sifarişçi ekranda göründü. Tələsdirir. Ana onu sakitləşdirir.

Sifarişçi Hindistandandır, öz iş otağından paltarın üçölçülü modelinə baxır. Çox razıdır. Onların hər biri öz ana dilində danışır, avtomatik tərcümə pis deyil. Ödəniş haqqında razılaşdılar. Budur pullar Şəbəkədən birbaşa olaraq ananın hesabına daxil oldu. Ana ülgünü Hindistana göndərdi. Paltar orada tikiləcək. Artıq sifariş yerinə yetirildi. Ana azaddır.

Oğul səhərdən artıq İnternetdədir. Tapşırığını yerinə yetirir. Barəsində inşa yazacağı filmi axtarır (fikir verin, kitab deyil, məhz kinofilmi). Tapdı. Atasının yanına qaçır. Deyir ki, baha deyil. Məktəbdən tapşırıblar. Ata razılaşır. Filmin pulunu birbaşa İnternetdən ödəyir. Oğul sevincək onu şəbəkədən yükləyir, sonra oturub baxır.

Onun 18 yaşlı bacısı öz kompüterindən İnternetlə səsli məktublar göndərir. Eyni zamanda üç pərəstişkarına məktub göndərmək lazım olanda bu çox əlverişlidir. Məktublara yeni şəkillərini də əlavə edir... Roma hüququnu öyrənmək elə darıxdırıcıdır ki. Nə maraqlı şəkil, nə də filmlər var. Yeni kosmetika kataloqlarına baxmaq əlbəttə ki, daha maraqlıdır. Onu maraqlandıranları qeyd edib şəbəkədən sifariş edir. Tezliklə poçtla göndərəcəklər. Eyni zamanda o, özü kimi gənclərlə anonim gapda söhbət edir, amma özünü 60-yaşlı bir qadın kimi təqdim edib bədbəxt sevgi yaşayan oğlanları sakitləşdirir. Bu onu çox əyləndirir. Lakin, o bilmir ki, sevgi haqqında söhbətdə iştirak edən 5 nəfərdən yalnız birinin 18 yaşı var. İkisinin yaşı çoxdan 50-ni keçib, digər iki nəfərin isə hələ heç 14 yaşı yoxdur. Gap çox məzəlidir. Burda hər kəs özünü bu gün necə istəyirsə elə təqdim edə bilər.

Ata kompüter arxasındadır. Çertyojlar üzərində fikirləşir. O məzuniyyətdədir, lakin işsiz dura bilmir. Çertyoju düzəltdi. Modelin hesablanmasını sonraya saxladı. Deyəsən hər şey yaxşıdır. Birdən zavoddan İnternet-telefonla təcili zəng gəlir. Zəng edən direktordur. Təcili məsləhət lazımdır. Ata zavod müdiriyyətinin keçirdiyi telekonfransa qoşularaq digər iştirakçılarla birlikdə vacib bir məsələni müzakirə edir. Yarım saat keçdi. Problem həll olundu.

Ata oğlunun yanına gedir və onu oyun arxasında tutur. Sən demə kinofilmə pulsuz oyun da əlavə olunurmuş. Uşağın əsl məqsədi elə bu idi. Ata çox narazıdır. Oğlunu cəzalandırır. Onu şəbəkədən uzaqlaşdırma şərtilə bir həftə təbiət qoynuna göndərilmək gözləyir.

Ana razı halda otağa daxil olur. O, sifarişi yerinə yetirdiyini və bir həftəliyinə həyat yoldaşı və uşaqları ilə birlikdə dəniz kənarında dincəlmək istədiyini deyir. Ailə kompüter qarşısındadır. Təcili bilet, mehmanxanada otaq, taksi sifariş edirlər. Ailə tez yır-yığış edir. Taksi gəlir, hamı artıq küçədədir. Maşına əyləşən kimi öz portativ kompüterlərini qoşurlar. Onlar yenidən şəbəkədədirlər.

Günortadır. Gün davam edir – İnternetdə 2010-cu ilin bir günü.

İNTERNET VƏ QANUN

Telekommunikasiya və kompüter texnologiyalarının, xüsusilə də İnternetin inkişafı ilə kibercinayət adlanan məvhum meydana gəldi: kompüterdəki informasiyaya icazəsiz müraciət, vebsəhifələrin işini qəsdən pozma, İnternetdə maliyyə fırıldaqları, kompüter quldurluğu və digər müəlliflik hüququ pozuntuları, virusların yayılması, insanlara nifrət aşılayan ideologiyaların təbliği.

Praktiki olaraq istənilən dövlətin müasir cinayət qanunvericiliyində cinayətin texniki vasitələrdən (şəbəkə və kompüter məhz texniki vasitələrdir) istifadə etməklə törədilməsi ağırlaşdırıcı hal hesab olunur və cəzanı artırır.

rəssam çalışır və onu çap etmək və oxuculara çatdırmaq lazımdır. Buna xeyli vaxt sərf olunur. İnternetdə isə informasiya bir neçə dəqiqə ərzində yerləşdirilərək birbaşa oxucunun iş yerinə çatdırılır. Qəzet və jurnallardan fərqli olaraq, İnternetdəki materiallar həmişə aktualdır və real vaxtda yenilənə bilir (qəzetdə informasiyanın minimal yenilənmə müddəti bir gündür). İnternetdə qəzetin elektron versiyasının yerləşdirilmə qiyməti, onun kağız analoqunun çap olunma qiymətindən azdır (kağıza, poliqrafiya və nəqliyyat xərclərində xeyli qənaət olunur).

İnternet gələcəkdə interaktivliyi (oxucularla sürətli və daimi əks əlaqə imkanları), operativliyi və çevikliyi nəticəsində ənənəvi informasiya vasitələrini xeyli sıxışdıracaq.



İNTERNET ASILILIĞI

İnternet – əlaqələrin genişləndirilməsi üçün unikal vasitədir, lakin o, tam mənada insan ünsiyyətini əvəz edə bilməz. İnternetlə artıq dərəcədə maraqlanan insanlar bəzən reallığın sərhədlərini itirirlər. Bu İnternet asılılığına və ya sağlamlığa (fiziki və psixoloji) ciddi təhlükə yaradan addiksiyaya gətirir. İnsan öz aludəliyi qarşısında aciz olur: iradə zəifləyir və asılılığa müqavimət göstərməyə imkan vermir. Şəxsi səviyyədə münasibətlərə ciddi ziyan dəyir. Belə asılılıq özünü onların davranışında, diqqətini daim İnternetdə cəmləşdirməklə reallıqdan uzaqlaşmaqda göstərir. Bu, intensiv emosiyaların artması ilə müşayiət olunur. Göstərilən proses insanı o dərəcədə əsir edir ki, onun həyatını idarə etməyə başlayır. İnternet zaman, güc, enerji və emosiyaları bütünlüklə ələ almaqla dostluğu, məhəbbəti əvəz edir.

PROQRAMÇI ÖZ SOYADINI SAYTININ ADI İLƏ ƏVƏZ ETDİ

İsrail proqramçısı Tomer Krissi öz soyadından narazı idi. Onun fikrincə soyadları bir arxaizmdir və yalnız tarixi əhəmiyyət kəsb edir. Tomer istəyirdi ki, onun necə adlanmasında bir məna olsun. Bir qədər səy göstərərək, o, Tomer.com adına yeni pasport ala bildi.

İndi proqramçı iddia edir ki, həmsöhbətinin ona aid bütün lazımi informasiyanı tapa biləcəyi şəxsi saytının olduğunu başa düşməsi üçün sadəcə özünü təqdim etmək kifayətdir. Belə soyad ona son dərəcə yaraşır: bir müddət əvvəl o, İnternet asılılığından əziyyət çəkirdi və sutkanın 12 saatını Şəbəkədə keçirirdi.

İNTERNET VƏ SİYASİ HƏYAT

İnternet bir neçə il ərzində dar ixtisaslı kompüter mühitindən qüdrətli informasiya fəzasına çevrildi. Şəbəkə informasiya mübadiləsi və ünsiyyət vasitəsi kimi çıxış edərək siyasi təbliğat üçün özünəməxsus tribuna rolunu oynayır. Rusiyanın siyasi partiyalarına məxsus ilk sayt 1996-cı ildə "Demokratik ittifaq" partiyasının informasiya xidməti tərəfindən açılan DS.ru saytı oldu.

1999-cu ilin payızında parlament seçkiləri İnternetdə siyasi fəaliyyətin əhəmiyyətli dərəcədə aktivləşməsinə səbəb oldu. Praktiki olaraq bütün partiya və hərəkatlar Şəbəkədə öz səhifələrini açmağa çalışırdılar. Elə o zaman, ilk dəfə seçkiqabağı kompaniyanın gedişatını və seçkilərin nəticələrini işıqlandıran xüsusi informasiya resursları meydana gəldi. Onlardan bəzilərində bu və ya digər siyasi qüvvə haqqında sensasiyalı materiallar yerləşdirilirdi. Televiziya ekranları və müxtəlif nəşrlərin səhifələrdən ölkəni sarsıdan siyasi qalmaqallar İnternet səhifələrində davam edirdi.

Öz saytının olması siyasi imicin vacib atributuna çevrildi. Misal üçün, RF hökuməti – www.goverment.gov.ru, Dövlət Duması – www.duma.ru, Rusiya Federasiyası prezidentinin xüsusi saytı var – president.kremlin.ru.

İNTERNETDƏ SÖZ AZADLIĞI VƏ SENZURA

İNTERNET VƏ SÖZ AZADLIĞI

> Hər bir insanın etiqad azadlığı və bunu azad şəkildə ifadə etmək hüququ var; bu hüquq maneəsiz şəkildə öz etiqadına sadiq qalmaq, istənilən üsullarla və dövlət sərhədlərindən asılı olmayaraq informasiya və ideyaları axtarmaq, əldə etmək və yaymaq azadlığını nəzərdə tutur.
>
> *İnsan haqlarının ümumi bəyannaməsi, maddə 19.*

Söz azadlığı – demokratik cəmiyyətin əsaslarından biri olub, insanların əsas hüququ və bütün azadlıqların təməlidir. İnternetin meydana çıxması ilə informasiyanın əldə olunması və yayılması üsulları radikal şəkildə dəyişdi. İnformasiyanın ötürülməsi üçün əvvəlki vasitələrdən fərqli olaraq, İnternet kompüteri və şəbəkəyə qoşulma imkanı olan hər kəsə bütün dünyada olan insanlarla dərhal əlaqə yaratmağa imkan verir. İnternet dövlət sərhədlərini aşır və azad informasiya axınının yolunda duran maneələri aradan qaldırır. Bu texnologiya təkcə qloballığı ilə seçilmir, o, informasiyanın yaradılması və yayılması üzərində nəzarəti ayrı-ayrı şəxslərə verir.

İnternetdə söz azadlığı dedikdə "heç bir dövlət orqanı informasiya resurslarının məzmununu nizamlamamalıdır" – anlayışı başa düşülməlidir.



İnsanın əsas hüquqlarından biri də göstərilən biliklərə, intellektual fəallığa və yaradıcı fikrə müraciət etmək, həmçinin öz fikirlərini açıq söyləməkdir. Hər bir şəxs öz kompüterinə əsassız informasiya yerləşdirə və arzu edən hər kəsə ondan istifadə etməyə icazə verə bilər, bir şərtlə ki, belə informasiyanın yayılmasına beynəlxalq qanunvericilik və ya həmin ölkənin qanunvericiliyi tərəfindən qadağa qoyulmasın (qanunlar dövlət sirrini qoruyur, sosial, irqi, milli və dini zəmində ziddiyyətin yaradılmasını qadağan edir). Bunun üçün radio və televiziya yayımından fərqli olaraq icazə və ya bildiriş sənədləri lazım deyil. Bu hüquqlara başqa insanlar, təşkilat və ya dövlət tərəfindən müdaxilə edilməsi – vətəndaşların yaradıcılıq azadlığı və şəxsi həyatına qarışmaq deməkdir.

İnformasiyaya kütləvi və maneəsiz müraciət faydalıdır, çünki informasiyaya malik olub ondan düzgün istifadə etmək bacarığı iqtisadi, sosial və mədəni inteqrasiyanı inkişaf etdirir, demokratik dəyərlərin bərqərar olmasına kömək edir.

"Kim ki, ani təhlükəsizlik naminə əbədi azadlığını qurban verməyə hazırdır, o, nə azadlığa, nə də təhlükəsizliyə layiqdir".

Bencamin Franklin





Bencamin Franklin.

SENZURA VƏ İNTERNET

Qədim zamanlardan bəri mövcud olan senzura Qərbi Avropada XV əsrdə yaranmışdır. Dövlət formalaşmağa başlayarkən azad olmağa, öz fikirlərini söyləməyə, hakimiyyətin siyasətini tənqid etməyə cəhd göstərən ictimaiyyətlə hakimiyyət arasında mübarizə gedir. Müxtəlif tarixi dövrlərdə senzuradan əvvəlcə kilsə, sonra hökmdarlar, dövlət hakimiyyəti istifadə edib. İnsan hüquqlarına dəyər verilən demokratik dövlətlərdə azadlıqların məhdudlaşdırılmasına cəhdlər hər zaman vətəndaşların etirazı ilə qarşılanır. Anarxiyaya bənzər strukturu ilə İnternetin meydana gəlməsi senzuranın problemlərini çətinləşdirdi. İnternetin mərkəzləşdirilməmiş xarakteri sayəsində dövlətin bir çox nəzarət vasitələri səmərəsiz və hətta faydasız olur, çünki istifadəçilərin onlardan yayınmaq üçün çoxlu sayda üsulları var.

İnternetdə senzuranın yaranmasına səbəb hakerlərin milli təhlükəsizliyə və böyük korporasiyaların təhlükəsizliyinə qarşı yönələn hədələri etik normadan kənar informasiyanın yayılması, intellektual mülkiyyətin mənimsənilməsi, ekstremist siyasi qruplaşmaların öz ideologiyalarını yaymaq məqsədilə Şəbəkədən istifadə etməsi, həmçinin (narkotiklər, özünə qəsd üsulları barədə) təhlükəli məlumatların açıqlığı və ya yayılması oldu. İctimai dəyərlər baxımından (əksər hallarında həm də qanunvericilik nöqteyi-nəzərindən) İnternetdə bu cür materialların yayılması bədniyyətlilərin bəd əməli kimi qiymətləndirilir.

Hər kəsin nəşr işinə rəhbərlik edə bildiyi vaxtda informasiya yaymaq hüququ yeni bir məna kəsb edir. İnternetin özünəməxsus imkanları insan haqları haqqında ən mühüm beynəlxalq sənədlərdə yer tutan "dövlət sərhədlərindən asılı olmayaraq" anlayışına uyğundur. Hakimiyyətin İnternetə nəzarət etmək üçün tədbirləri beynəlxalq hüquq normalarına ziddir. Bir ölkədə həyata keçirilən senzura digər ölkədə şəxsiyyətin informasiya yaymaq hüququnun birbaşa pozulmasına gətirib çıxara bilər.

İnternetdə senzuranın yaranma təhlükəsinə cavab olaraq şəbəkə birlikləri yaranmağa başladı. "İnternetdə azadlıq uğrunda beynəlxalq hərəkat" (Global İnternet Liberty Campaign) bütün dünya üzrə söz azadlığı və informasiya əlyetərliyini təbliğ edən 45 təşkilatı

Senzura (*lat.* censeo – "qiymətləndirirəm") – mətbuat və kütləvi informasiya vasitələri üzərində dövlət nəzarəti sistemidir.

özündə birləşdirən ən böyük birlikdir. Başqa bir misal – İnternet resurslarından istifadə edilməsinin istənilən şəkildə məhdudlaşdırılmasına qarşı çıxış edən "Sərhədsiz reportyorlar" (Reporteurs sans frontiers) beynəlxalq şəbəkə birliyidir.

Bu cür təşkilatların üzvləri hesab edirlər ki, mənəviyyat nöqteyi-nəzərindən mühakimə oluna bilənləri qanunla mühakimə etmək vacib deyil. Onlar ictimaiyyətə belə fikri çatdırmağa çalışırlar ki, cinayətlə İnternetə senzura tətbiq etmədən cinayətlərin bilavasitə mənbəyini məhv etməklə mübarizə aparmaq lazımdır. İnternet müasir dünyada ən mühüm kommunikasiya vasitələrindən biridir və prinsipcə senzuraya məruz qalmamalıdır.

İnternetə azad çıxış məsələsi etik normalara görə yetərli qədər mürəkkəb məsələdir. Hər bir insan özünü və uşaqlarını arzuolunmaz hesab etdiyi informasiyadan qorumaq hüququna malikdir və o, yalnız öz şəxsi kompüterində istədiyi hər şeyi süzgəcdən keçirə bilər. Əgər hansısa sayt kiminsə xoşuna gəlmirsə, uşaqlarına da bu sayta baxmağa icazə verməyə bilər, amma heç kəsdən bu sayta baxmamağı tələb edə bilməz.

Demokratik ölkələrdə heç bir qanunverici senzura tətbiq edə bilməz. Lakin provayderlər şəxsi kommersiya strukturları olaraq özlərinə məxsus olan disk sahələri və rabitə kanallarından istədikləri kimi istifadə edə, yəni öz informasiya kanalları ilə ötürülən materiallara nəzarət edə bilərlər. Narazı müştərilər isə özlərinə başqa provayder axtara bilərlər. Dövlət ictimai maraqları əsas gətirərək, buna haqq qazandırmasından asılı olmayaraq, hər-hansı bir üsulla şəxsi informasiyanı süzgəcdən keçirməməlidir, çünki söz azadlığı sivil cəmiyyətin ən ali dəyəridir.

İNTERNETİN İDARƏ EDİLMƏSİ

İnternetin məzmununu ilk nizamlama cəhdləri onun cəmiyyətin həyatına sürətlə nüfuz etməsilə eyni vaxta təsadüf etdi. Artıq 1995-ci ildə ABŞ kommunikasiyalarının ədəb qaydalarına uyğunluğu haqqında Akt ictimaiyyətə açıqlandı. Bu qanunvericilik aktı İnternetlə ötürülən etik normadan kənar materialları qanuna zidd elan edərək cərimədən (10 min dollara qədər) tutmuş azadlıqdan məhrum edilməyə qədər (iki iləcən) cəza təyin edirdi. Əsasən də 18 yaşından kiçik olanlara istənilən təhqiredici materialların göstərilməsi və ötürülməsi üçün interaktiv kompüter xidmətindən şüurlu istifadə xüsusilə qadağan olunurdu. Akt 1 fevral 1996-cı il tarixdə qəbul edilmiş və 8 fevral tarixdə prezident Klinton tərəfindən imzalanmışdır. Elə həmin gün vətəndaşların İnternet hüquqları üzrə koalisiyası adını almış Amerika vətəndaş haqlarının müdafiəsi birliyinin başçılıq etdiyi bir qrup təşkilat bu aktın konstitusiyaya zidd olması barədə iddia ilə ABŞ Ali məhkəməsinə müraciət etmişdir. İnternetdə senzuraya qarşı söz azadlığının müdafiəsi məqsədilə geniş miqyaslı kampaniya təşkil olunmuşdu. Məhkəmə aktın ABŞ Konstitusiyasının söz azadlığını məhdudlaşdıran qanunların qəbul edilməsini qadağan edən birinci düzəlişinə zidd olması barədə qərar qəbul etdi.

İNTERNETƏ QADAĞA

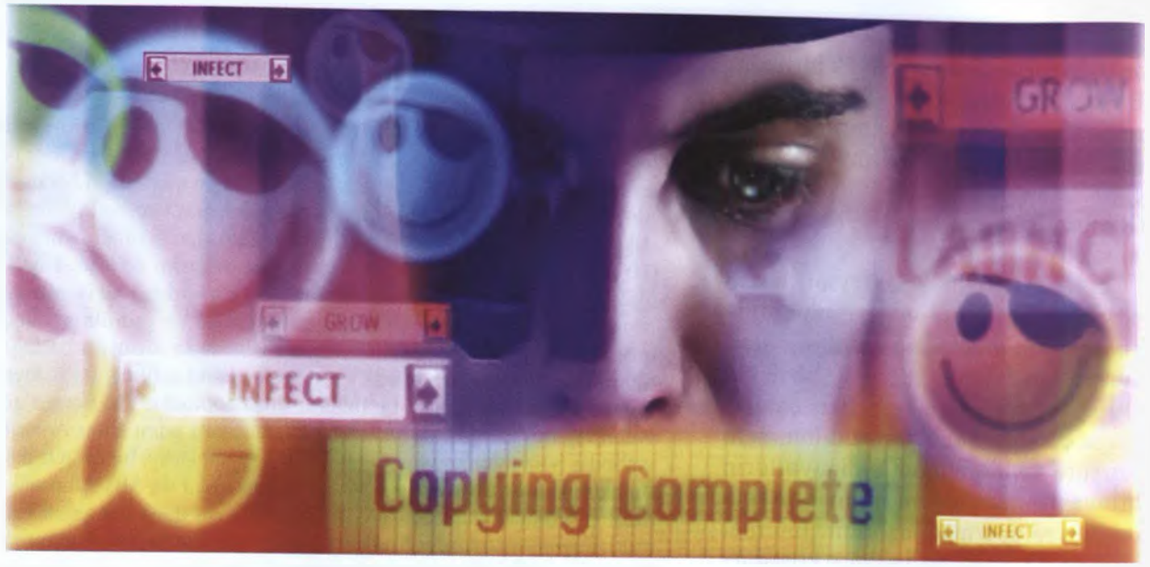
"Sərhədsiz reportyorlar" beynəlxalq təşkilatı İnternetə "düşmən" ölkələrin siyahısını yayıb. Bu ölkələrdə məhvedici ideyaların yayılması, müdafiə və milli birliyin təhlükəsizliyinə zəmanət verilməsi adı altında vətəndaşların İnternetə azad müraciəti xeyli məhdudlaşdırılıb və ya tamamilə qadağan olunub. Bu siyahıya Belorus, Vyetnam, İraq, İran, Çin, Kuba, Liviya, Səudiyyə Ərəbistanı, Şimali Koreya, Suriya, Sudan, Tacikistan, Türkmənistan, Tunis və Özbəkistan kimi ölkələr daxildir.

Şimali Koreya prezidenti Kim Çen İr ölkəsinin bütün dünyaya qapalı qalması üçün İnternetin qadağan edilməsi həqqında qərar qəbul etmişdir. Rejimin rəsmi təbliğat apardığı yeganə sayt isə Yaponiyada qeydiyyatdan keçib.

Kuba lideri Fidel Kastro hesab edir ki, İnternet kapitalizmin ictimai fikri idarə etmək üçün istifadə etdiyi bir alətdir. Kubada yalnız xarici turistlər və ya kommunist partiyasının funksionerləri İnternetdən istifadə etmək hüququna malikdirlər.

Qədim Çin bütün dünyadan Böyük Çin Səddilə ayrıldığı kimi, sosialist ölkəsi olan müasir Çinin kompüter şəbəkələri də ümumdünya şəbəkəsindən ayrılıb, qoşulmalar və trafik isə hökumət tərəfindən nəzarətdə saxlanılır. Kommunizm və hökumət əleyhinə İnternetdə dərc olunmuş məqalələrin müəlliflərini təqib edən xüsusi polis bölmələri yaradılıb. Şəbəkədə fərqli düşüncə uzunmüddətli həbslə cəzalandırılır.

Belorusiyada İnternetə qoşulma vasitələrini (modem) daxili işlər orqanlarında qeydiyyatdan keçirmək tələb olunur.



İNFORMASİYANIN QORUNMASI

RƏQƏMLİ ƏSRDƏ MÜƏLLİFLİK HÜQUQU

MÜƏLLİFLİK HÜQUQU VƏ KOMPÜTERLƏŞDİRMƏ

Azərbaycanda müəlliflik hüququ Azərbaycan Respublikasının "Müəlliflik hüququ və əlaqəli hüquqlar haqqında" 8 oktyabr 1996-cı il tarixli qanunu ilə tənzimlənir.

Kompüterlərin insan həyatına sürətlə daxil olması, informasiya texnologiyalarının dinamik inkişafı ilk vaxtlarda qanunvericilərin buna hazır olmadıqlarını göstərdi. Bir müddət kompüter proqramlarına göstəriləcək hüquqi müdafiənin müəyyən edilməsi məsələsi açıq qaldı. Lakin kompüter sənayesinin lazımi səviyyədə inkişafını təmin etmək üçün müdafiənin zəruri olduğu aydın idi.

Hal-hazırda kompüter proqramları (redaktorlar, kompilyatorlar, verilənlər bazası) əmtəə məhsulu statusu almışdır. Məsələn, verilənlər bazası EHM vasitəsilə əldə oluna və emal edilə bilməsi üçün sistemləşdirilmiş verilənlər yığınının (məqalələr, hesablamalar və s.) təqdim edilməsi və təşkilinin obyektiv formasıdır. Hansı formada təsvir edilməsindən asılı olmayaraq, proqramlar və verilənlər bazası ədəbi əsərlər kimi qorunurlar. Bundan əlavə kompüter proqramlarında müəlliflik hüququnun digər obyektlərindən – ədəbi əsərlər, təsviri incəsənət, kinematoqrafiya, musiqi əsərlərindən istifadə olunur.

Alıcılar müəllifin təyin etdiyi xüsusi şərtlər daxilində maddi mülkiyyətdən fərqli olaraq, intellektual mülkiyyət obyektlərindən, o cümlədən kompüter proqramlarından istifadə edə bilərlər. Bu zaman yalnız informasiya daşıyıcısı, məsələn, proqramın yazıldığı kompakt disk alıcının mülkiyyətinə keçir. Alıcının yalnız arxivləşdirmə məqsədilə proqramın sahibilə razılaşdırılmadan və ona əlavə haqq ödəmədən bu proqramın surətlərini çıxarmaq hüququ var.

MÜƏLLİFLİK HÜQUQU VƏ İNTERNET

Yeni informasiya texnologiyalarının inkişafı bütün informasiyanın rəqəmli şəklə keçirilməsi prosesinin başlanğıcını qoydu. İnternetin yaranması və kütləvi istifadəsi isə informasiyanın

Qanunla proqram elektron-hesablama maşınlarının (EHM) və digər kompüter qurğularının fəaliyyəti üçün proqramın hazırlanması zamanı alınan köməkçi materiallar və proqram vasitəsilə alınan audiovizual təsvirlər də daxil olmaqla müəyyən bir nəticənin alınması üçün nəzərdə tutulmuş məlumatlar yığınının və komandaların təsvirinin obyektiv forması kimi təyin edilir.

MÜƏLLİFLİK HÜQUQU NƏDİR

Müəlliflik hüququ – elm, ədəbiyyat, və incəsənət əsərlərindən istifadə qaydalarını tənzimləyən vətəndaş hüquqlarının bir hissəsidir. Kompüter proqramları və elektron verilənlər bazası da müəlliflik hüququ obyektləri hesab olunur.

Qorunan əsər yaradıcılıq fəaliyyətinin nəticəsi, insanın intellektual fəaliyyətinin məhsulu olmalı, müəllifin fərdi yaradıcılığını özündə əks etdirməlidir. İntellektual fəaliyyətin nəticələrini *intellektual mülkiyyət* adlandırırlar.

Müəlliflik hüquqlarının subyektləri, yəni əsər üzərində müstəsna hüquqları olan şəxslər hər şeydən əvvəl əsərlərin müəllifləri hesab olunurlar. Bu hüququn daşıyıcıları həm də əsərdən kommersiya məqsədlə istifadə etmək hüququnu almış müxtəlif şirkətlər və ya müəllifin varisləri ola bilərlər.

Müəlliflik hüquqları şəxsi qeyri-əmlak və əmlak hüquqlarına bölünür. Qeyri-əmlak hüquqlarından müəllifə aşağıdakılar aiddir:

- müəlliflik hüququ – əsərin müəllifi kimi tanınmaq hüququ;
- ad hüququ – əsərdən müəllifin adı və ya onun təxəllüsü ilə istifadə etmək və ya istifadə etməyə razılıq vermək hüququ;
- açıqlanma hüququ – əsəri istənilən formada açıqlamaq və ya açıqlanmasına icazə vermək hüququ;
- müəllifin ləyaqətinin qorunması hüququ – müəllifin şərəf və ləyaqətinə xələl gətirə biləcək istənilən təhrifdən və ya digər müdaxilələrdən əsərin adı da daxil olmaqla, qorunması hüququ.

Şəxsi qeyri-əmlak hüquqları müəllifin əmlak hüquqlarından asılı olmayaraq hətta əsərin nəşrinə istisna olaraq hüquqlar verildiyi halda belə, onun özünə aiddir və bu hüquqlar yalnız ona məxsusdur.

Müəllifin əmlak hüquqlarına aiddir:

- yenidən istehsal etmək hüququ;
- yenidən işləmək hüququ;
- idxal hüququ;
- tərcümə etmək hüququ;
- açıq şəkildə nümayiş etdirmək hüququ;
- açıq şəkildə ifa etmək hüququ;
- efirə vermək hüququ;
- kabellə ictimaiyyətin nəzərinə məlumat çatdırmaq (yəni, nümayiş etdirilmə, ifa etmə və ya efirə vermə də daxil olmaqla kabellə, naqillə və ya digər analoji vasitələrlə əsəri ictimaiyyətin nəzərinə çatdırmaq) hüququ;
- ictimaiyyətin nəzərinə çatdırmaq (yəni, əsəri elə vermək lazımdır ki, istənilən şəxs üçün interaktiv rejimdə istənilən yerdə və vaxtda əlçatan olsun) hüququ.

Müəlliflik hüququ müəllifin bütün həyatı boyu və onun ölümündən keçən 70 il ərzində qüvvədə qalır. *Qorunma* müddəti başa çatdıqdan sonra əsər ictimai mülkiyyətin ixtiyarına keçir, bu zaman heç bir icazə istəmədən ondan istifadə etmək olar. Müəlliflik hüququ, ad hüququ və müəllifin ləyaqətinin qorunması hüququ müddətsiz qorunur.



MÜƏLLİFLİK HÜQUQUNUN TARİXİNDƏN

Müəlliflik hüququnun tarixi kitab çapının ixtirasından başlayıb, bunun sayəsində isə əlyazma əvəzinə əsərin mexaniki şəkildə çoxaldılması mümkün olub. İlk vaxtlarda müəlliflik hüquqlarının qorunması krallar tərəfindən verilən imtiyazlar şəklində təqdim olunurdu. 1557-ci ildə ingilis kraliçası Mariya Tüdorun fərmanı ilə Britaniya nəşriyyat gildiyalarından biri ölkədə kitab çapı və kitab ticarəti üzrə inhisarçılığı ələ almışdı. Bu fərmana görə kitabı onun müəllifindən satın alaraq onu xüsusi reyestrdə qeyd edən və bu gildiyaya mənsub olan kitab alverçisi kitabın çap etdirilməsi hüququna malik olurdu.

11 yanvar 1709-cu ildə Britaniya Nümayəndələr Palatasında "Müəllif və ya çap olunmuş kitabların surətlərini alan şəxslərə bu kitablara hüquqların bundan sonra alınması yolu ilə maarifləndirilmənin həvəsləndirilməsi barədə" qanun layihəsi dinlənildi. 10 aprel 1710-cu ildə bu layihə müəllif hüquqları haqqında ilk qanunvericilik aktı oldu. Bu qanun tarixə "Kraliça Annanın əsas qanunu" adı ilə daxil oldu.



Kraliça I Mariya Tüdor.



Kraliça Anna Stüart.

Qanun müəlliflərə onların əsərlərinin 14 il müddətinə hüquqi mühafizəsini təmin edirdi, müəllif sağ ikən mühafizə daha 14 il uzadıla bilərdi.

Müəlliflik hüquqlarına aid olan ilk çoxtərəfli beynəlxalq razılaşma 1886-cı ildə qəbul olunan ədəbi və bədii əsərlərin qorunması haqqında daimi Bern konvensiyası oldu.

1952-ci ildə ikinci əsas beynəlxalq konvensiya – "Müəllif hüququ haqqında ümumdünya konvensiya"sı qəbul olundu.

MÜƏLLİFLİK HÜQUQUNUN POZULMASINA GÖRƏ MƏSULİYYƏT

Azərbaycan qanunvericiliyi müəlliflik hüququ haqqında bütün beynəlxalq razılaşma prinsiplərinə cavab verir. Qanunvericilikdə bu hüququn müxtəlif formalarda mühafizəsi əks olunmuşdur. Müəlliflik hüquqlarının pozulmasına görə mülki, cinayət və inzibati qaydada məsuliyyət tətbiq olunur.

Hazırlanması və yayımlanması müəlliflik hüququnun pozulmasına gətirib çıxaran əsər və ya proqramların nüsxələri qanunla kontrafakt (*fran.* "contrefaçon" – "saxta") adlandırılır. Məişətdə işə daha çox "pirat surəti" (*ing.* "piracy" – "müəlliflik hüququn pozulması" deməkdir) terminindən istifadə edilir. Müəlliflik hüququnun sahibləri müdafiə üçün məhkəməyə müraciət edə bilərlər və bu hüququ pozan şəxsdən müəllifin hüquqlarının tanınmasını; hüququn pozulmasına qədər olan vəziyyətin bərpasını, hüquqları pozan və ya onun pozulması üçün təhlükə yaradan hərəkətlərə son qoyulmasını; əldən çıxan gəlir də daxil olmaqla ziyanın ödənilməsini; müəlliflik və əlaqəli hüquqların pozulması nəticəsində qanun pozucusunun əldə etdiyi gəlirin geri alınmasını təmin edə bilərlər.

əldə olunmasını və yayılmasını asanlaşdırdı, müəlliflik hüquqlarının qorunması problemini daha da kəskinləşdirdi. Müasir qanunvericilik bu dəyişiklikləri özündə əks etdirir və ixtiyari formada müəlliflik hüququ obyektlərini qoruyur. Belə ki, İnternetdə yerləşdirilən əsər müəlliflik hüququ obyektinin bütün xüsusiyyətlərinə cavab verir: birincisi, əsərin təyinatından və dəyərindən, həm də onun ifadə olunma üsulundan asılı olmayaraq o, yaradıcılıq fəaliyyətinin məhsuludur, ikincisi, obyektiv formada – yazı (mətn faylları və veb-səhifələr) və təsvirlər şəklində təşkil olunmuşdur. Bununla bərabər, İnternetdən geniş istifadə World Wide Web sistemində yerləşdirilən materiallar üzərində müəlliflik hüququ barədə bir sıra yanlış fikirlər yaratmışdır.

• Əgər müəlliflik hüququnu göstərən işarə (çevrənin içərisində C hərfi – ©) qoyulmayıbsa, o zaman əsər qorunmur. Bu belə deyil. Əsərə və ya proqrama müəlliflik hüququ onların yaradılması ilə qüvvəyə minir. © işarəsinin qoyulması isə o qədər də vacib deyil.

• Əgər müəlliflik hüququnu pozan bundan gəlir əldə etmirsə, o zaman müəlliflik hüququ pozulmuş hesab olunmur. Əslində bu da düzgün deyil, çünki müəllifdən icazə almadan əsərdən istifadə etmək istənilən halda hüququn pozulması deməkdir.

• İnternetdə yerləşdirilən əsərlər ictimai mülkiyyətdir. Lakin əsər onun qorunma müddəti bitdikdə və yaxud əsərin müəllifi buna etiraz etmədikdə, ictimai mülkiyyətin ixtiyarına keçir.

• Özgənin əsərinin İnternetdə yerləşdirilməsi ondan şəxsi istifadə olunması deməkdir. Bu isə belə deyil. Əsərin ictimaiyyətin nəzərinə çatdırılması yalnız müəllifin hüququdur. Müəllifin istənilən əsərinin surətinin asanlıqla çıxarılması qanunvericiləri onların qorunmasını gücləndirməyə məcbur edir. 1998-ci ildə ABŞ Konqresi "Rəqəmli minillikdə müəlliflik hüququnun qorunması haqqında" (*ing.* Digital Millennium Copyright Act – DMCA) qanun qəbul etmişdir. DMCA, həmçinin İnternetdə də müəlliflik hüquqlarını pozan subyektlərin mülki qaydada və cinayətkar kimi təqib olunmasını nəzərdə tutur. Bu qanuna görə müəlliflik hüququnun pozulması üsulları

THE DIGITAL MILLENNIUM COPYRIGHT ACT OF 1998
U.S. Copyright Office Summary

December 1998

INTRODUCTION

The Digital Millenium Copyright Act (DMCA)[1] was signed into law by President Clinton on October 28, 1998. The legislation implements two 1996 World Intellectual Property Organization (WIPO) treaties: the WIPO Copyright Treaty and the WIPO Performances and Phonograms Treaty. The DMCA also addresses a number of other significant copyright-related issues.

The DMCA is divided into five titles:

haqqında informasiya verən, həm də müəlliflik hüququnun qorunmasını sındırmaq üçün alətləri layihələndirən və hazırlayan şəxslər müəlliflik hüququ ilə qorunan materiallardan qeyri-qanuni istifadə edənlər kimi saxlanıla və məhkəməyə verilə bilər. Surətçıxarmaya qarşı texniki mühafizə vasitələrinin işinin pozulmasına görə məsuliyyət daşıyan DMCA-ya bənzər qanunlar həm Avropa Birliyi ölkələrində, həm də ölkəmizdə hazırlanır.

COPYRIGHT və "COPYLEFT"



Linus Tornvald.

Bu gün işi kompüterdən keçən hər bir insan azad şəkildə yayılan Linux əməliyyat sistemi haqqında eşidib. Çoxlarının onun yaradıcısı olan Linus Tornvald barədə məlumatı da var. Lakin Linus Tornvaldın məşhur sələfi Riçard Stolmeni (Richard Stallman) isə hamı tanımır. Halbuki, o, uzun illər bundan əvvəl Linux əməliyyat sisteminin uğur qazandığı fəlsəfi, hüquqi və sanballı texniki bazanın əsasını qoymuşdur.

XX əsrin 70-ci illərində Stolmen məşhur Massaçusets Texnologiya İnstitutunda müəllim və proqramçı kimi işləyərkən yalnız azad proqram təminatından (PT) istifadə edən mütəxəssislər qrupunun tərkibində idi. Stolmenin özü fəal şəkildə bunları təbliğ edirdi: kompüter proqramlarının ilkin mətnləri də daxil olmaqla, istənilən intellektual nailiyyətlər istifadə olunma, köçürülmə, səhvlərin düzəldilməsi, inkişaf etdirilmə, tam şəkildə yenidən işlənmə və s. üçün bütün bəşəriyyətə əlçatan olmalıdır.

Həmin illərdə elə indi olduğu kimi, bir çox proqramlar ayrı-ayrı şirkətlərin mülkiyyəti idi. Belə proqramlardan qanuni şəkildə istifadə etmək üçün lisenziya almaq lazım idi. Adi fərdi lisenziya alıcının hüquqlarını ciddi şəkildə məhdudlaşdırır:

• verilən proqramı yalnız bir kompüterdə yükləməyə imkan verir;

• proqramın yalnız bir ehtiyat nüsxəsini köçürməyə imkan verir;

• proqramın sahibi olmayan şəxsə hətta proqramın qanuni şəkildə yüklədiyi kompüterdə işləməyi qadağan edir;

• proqramda səhvləri düzəltməyi və ya səhvlərin mənbəyini tapmaq məqsədilə proqramın ilkin kodunu öyrənməyi qadağan edir.

Riçard Stolmen öz manifestində yazırdı: "GNU layihəsi" (GNU's Not UNIX-in rekursiv qısaltması – *red.*) qapalı PT sahibləri tərəfindən yaradılmış maneələri aradan qaldıraraq əməkdaşlığa yenidən imkan yaratmaq, əvvəllər kompüter istifadəçilərinin arasında üstünlük təşkil edən əməkdaşlıq ruhunu geri qaytarmaq vasitəsi kimi 1983-cü ildə yaradılmışdır. Massaçusets Texnologiya İnstitutunun hamiliyilə o, Free Software Foundation (Azad şəkildə yayılan proqram təminatına dəstək cəmiyyəti) adlanan qeyri-kommersiya birliyini təşkil etdi.

Stolmenin ideyalarını gerçəkləşdirmək üçün ilk növbədə elə əməliyyat sistemi tələb olunurdu ki, onun idarəsi altında bu proqram təminatı işləməyə başlasın. Stolmen bu əməliyyat sisteminin UNIX-ə uyğun olacağına heç şübhə belə etmirdi.

"Copyright" hərfi mənada "müəlliflik hüququ" və ya "surətin çıxarılması hüququ" anlamını verir.

Bundan başqa, əməliyyat sistemi və bütün alətlər (kompüter proqramları) azad şəkildə yayılmalıdır. GNU layihəsinin ilk proqramı GNU Emacs adlı mətn redaktoru oldu ki, bu da təbii idi, çünki proqramı yazmaq üçün mətn redaktoru lazımdır. Proqram tez bir zamanda məşhurlaşdı və layihənin tərəfdarları meydana çıxdı.

1990-cı ildə GNU proqramlar dəsti əməliyyat sisteminin nüvədən başqa bütün əsas komponentlərinə malik idi. GNU proqramlar dəstinin idarəsilə fəaliyyət göstərən əsas nüvə Linux-un nüvəsi oldu. Qeyd edək ki, HURD nüvəsi üzərində işlər hələ də davam etdirilir.

Əksər proqramçılar bu gün onunla razılaşırlar ki, Free Software Foundation-in nailiyyətlərindən istifadə edilmədən Linux-un yaradılması mümkün olmazdı. Nəticələrdən hansının əsas olduğuna gəlincə, burada rəylər müxtəlifdir. Bəziləri "Əlbəttə əsas kompilyatordur", digərləri isə etiraz edərək "Yox, əsas Stolmen tərəfindən hazırlanmış lisenziyadır" – deyəcək. Lisenziya-GNU Public License, qısa şəkildə GPL adlanır və onun mənası çox sadədir. Tutaq ki, X müəllifi Y proqramını yaratmışdır və onun müstəqil yayımını təmin edərək proqramı bütün bəşəriyyətə hədiyyə etmək istəyir. Əgər X deyirsə ki, onun bu proqramından hamı həm istifadə edə, həm də nə istəsə edə bilər, bu halda növbəti vəziyyət yarana bilər: hansısa bir bədniyyətli Y proqramını cüzi dəyişir və onu Z proqramına çevirərək özünün mülkiyyəti elan edir. GNU lisenziyası isə bunu qadağan edir. Bu lisenziyaya əsasən, GNU proqramında dəyişikliklərin edilməsinə yalnız o şərtlə icazə verilir ki, dəyişdirilmə nəticəsində yenidən GNU proqramı alınsın.

"GNU layihəsinin məqsədi bütün istifadəçilərə GNU PT-ni sərbəst şəkildə yaymaq və dəyişdirmək imkanını verməkdir. Əgər vasitəçi sərbəstliyi aradan götürə bilsəydi, bizim yəqin ki, çoxlu istifadəçilərimiz olardı, lakin onların sərbəstliyi olmazdı. Buna görə də, GNU proqram təminatını ictimai məhsul etmək əvəzinə, biz onu "Copyleft" qaydaları ilə yayırıq... Copyleft elə qaydadır ki, proqramların bu qayda ilə yayılması zamanı siz onlara insanları əsas sərbəstlikdən məhrum edən məhdudiyyətlər əlavə edə bilməzsiniz" – Stolmen yazır.

Sərbəst PT problemi təkcə proqram təminatının özünün sərbəst olmasında deyil, həm də bu proqram təminatının texniki təlimat sənədlərinin sərbəst olmasındadır.

"Sərbəst təlimat kriteriyası bir çox cəhətdən sərbəst proqram kriteriyası kimidir... PT-nin hər bir nüsxəsində üzərindəki interaktiv formada və ya kağız təlimatın olması üçün təkrar yayılmaya (o cümlədən kommersiya məqsədilə) icazə verilməlidir. Dəyişikliklərin edilməsinə icazə də həlledici rol oynayır".

Bir qayda olaraq, sərbəst şəkildə yayılan PT çox vaxt surətinin çıxarılmasına, dəyişdirilməsinə və yayılmasına qadağa qoyulan qapalı təlimat sənədlərinə malikdir. Sərbəst yayılan proqram təminatı üçün təlimat sənədlərinin də "GNU Public License" lisenziyası altında yayılması ideal olardı.

Sərbəst şəkildə yayılan PT-nın surətinin həmişə pulsuz çıxarıldığını düşünmək düzgün deyil. Sərbəstlik elə o deməkdir ki, nüsxəni həm pulsuz, həm də pulla almaq olar. Sadəcə olaraq, sərbəst şəkildə yayılan proqram təminatının alınması üçün min cür üsul vardır, onların arasında PT-ni pulsuz alma üsulu da göstərilmişdir. Bu, elə proqram təminatının sənədlərinə də aiddir.

Lakin məsələ bundadır ki, əgər PT-ni pulsuz almaq mümkündürsə, o zaman nə üçün pul ödəməliyik?

HURD – rekursiv qısaltmadır: HIRD of UNIX-Replacing Daemons, burada HIRD – HURD of Interfaces Representing Depth açılışının qısaldılmış formasıdır. Bu rekursiv qısa yazılışı (GNU ənənələrinə uyğun olaraq) dilimizə tərcümə etmək praktik olaraq mümkün deyil. İngilis dilini bilənlər bunu özləri edə bilərlər (ipucu: "daemons" sözü ilə UNIX dünyasında server proqramlarını adlandırırlar).



Riçard Stolmən.

İNFORMASİYA TƏHLÜKƏSİZLİYİ



Kompüter və informasiya texnologiyalarının intensiv inkişafı ilə informasiya təhlükəsizliyinin vacibliyi də artdı. Məlumdur ki, informasiya əsl xəzinə ola bilər. Məhz buna görə də onun həm qorunmasına, həm də əldə olunmasına tez-tez böyük qüvvə sərf olunur. İnformasiyanı o zaman qorumaq lazımdır ki, onun kənar adamların əlinə düşmək təhlükəsi olsun. Burada söhbət məxfi (*ing.* "confidencial") informasiya anlayışından gedir.

Müasir bir şirkətin ofisi elektronika ilə doldurulmuş otaqlardan ibarətdir. Burada məxfi informasiya kağız üzərində, kompüterlərdə yerləşir, telefonla, faksla ötürülür, maqnitofon, diktofon, videomaqnitofonla emal olunur, yazılır və oxunur, nəhayət, danışıqlarda mövcud olur. İnformasiya emalı sistemlərinə zərər yetirə biləcək çoxlu sayda təhlükələr mövcuddur. İlk növbədə bu təhlükə insandan asılı olmayan təbii fəlakət və hadisələr nəticəsində (yanğın, daşqın, zəlzələ və ildırım çaxması) yarana bilər.



İnsanlar tərəfindən vurulan ziyan daha böyük və təhlükəli ola bilər:

• layihələndirmə zamanı nəzərə alınmayan, xidmətçi personalın və proqram təminatının fəaliyyətilə bağlı olan səhvlərin, hesablama texnikasının, rabitə xətlərinin və enerji təminatının işində yaranan təsadüfi qəzaların, istifadəçilərin səhvləri, aparaturaya elektromaqnit sahələrinin təsirinin nəticəsi kimi bilmədən dəyən ziyan;

• xidmətçi personalın icazəsiz fəaliyyəti və kənar şəxslərin informasiya ya çıxışı kimi bilərəkdən vurulan ziyan.

Bunun nəticəsində informasiyanın məhv edilməsi, dağıdılması, həmçinin təhrifi, saxtalaşdırılması, sızması, surətinin çıxarılması və s. baş verə bilər.

Bütün ciddi tədbirlər kimi informasiyanın qorunması da kompleks şəkildə həyata keçirilməlidir, yəni ən yaxşı nəticələr almaq üçün informasiyanın bütün mühafizə üsulları birləşdirilməlidir. İnformasiyanın kompleks mühafizə sisteminə aşağıdakılar daxildir:

• təşkilati mühafizə, yəni iclaslardan tutmuş planların hazırlanmasına və informasiyanın (icazəsiz müdaxilədən) qorunması üzrə şöbələrin yaradılmasına qədər xüsusi tədbirlər;

• proqram-aparat mühafizəsi, kompüter sistemləri və proqramları;

• mühəndis-texniki mühafizə: videomüşahidə kameraları, kənar şəxslərin məxfi otaqlara (məxfi informasiyaya) girişini məhdudlaşdıran intellektual qıfıllar;

• qanunverici mühafizə.

Azərbaycanda "İnformasiya, informasiyalaşdırma və informasiyanın mühafizəsi haqqında" qanun qüvvədədir. Qanun informasiyanı (təqdim olunma formalarından asılı olmayaraq) şəxslər, əşyalar, faktlar, hadisələr və proseslər haqqında məlumatlar kimi təyin edir. O, informasiyanın yaradılması, toplanılması, emalı, yığımı, saxlanılması, axtarışı və yayımı əsasında informasiya resurslarının formalaşması və istifadə olunması; informasiya texnologiyalarının və onların təminatı vasitələrinin yaradılması və istifadə olunması; informasiya proseslərində və informasiyalaşdırmada iştirak edən subyektlərin hüquqlarının qorunması zamanı meydana çıxan münasibətləri tənzimləyir. Qanuna görə istənilən sənədləşdirilmiş informasiya qorunmalıdır. Bu informasiyadan qeyri-qanuni istifadə olunması onun mülkiyyətçisinə, sahibinə, istifadəçisinə və ya hər hansı digər şəxsə ziyan vura bilər. Qanun vətəndaşların razılığı olmadan onların şəxsi həyatını və ailə sirlərini açıqlayan informasiyanın, yazışma sirrinin, telefon danışıqlarının, poçt məlumatlarının, məhkəmə qərarı istisna olmaqla, toplanılması, saxlanılması, istifadə olunması və yayılmasını qadağan edir. Bu qanunla həm də açıq informasiyadan istifadə hüququ qorunur.

AR cinayət məcəlləsində kompüter informasiyası sahəsində cinayətə görə



SƏHVLƏR VƏ QƏZALAR



Proqram təminatında yol verilən səhvlər və nasazlıqlar kompüter cinayətkarlığı qədər təhlükəlidir. Hər dəfə növbəti qalmaqaldan xəbər tutarkən bunun səhv və ya təxribat olması ilə bağlı sual yaranır.

Venesuelanın paytaxtı Karakasda Ali Məhkəmə kompüter sistemindəki səhvə görə 2000-ci ilin mayında keçiriləcək seçkiləri təxirə salmışdı.

17 iyun 2000-ci ildə Böyük Britaniyada aviauçuş sistemində böyük bir nasazlıq baş vermişdir. Gün ərzində nasazlıq aradan qaldırılsa da, minlərlə adam aeroportlarda gecələməli olmuş, uçuşların vaxtı dəyişdirilmişdir. Buna baxmayaraq, nasazlığın vaxtında aradan qaldırılması və insanların salamat qalması sevindirici haldır.

Avstraliyanın Telstra Telefon şirkətinin kompüter əməliyyat sistemində baş verən nasazlıq ucbatından isə 76 min abunəçi öz telefonlarından istifadə edə bilməmişdi.

ABŞ-ın Northeastern Universitetinə (Şimal-şərqi universitet) kifayət qədər yoxlanmamış kompüter sistemi quraşdırıldıqdan sonra nəzərdə tutulduğundan 600 nəfər çox tələbə (25%) qəbul olunmuşdur. Universitet rəhbərliyi əlavə qəbul olunmuş tələbələrlə nə edəcəklərini bilmirdi.

Berkli Bankından bildirilmişdir ki, daha mükəmməl proqram təminatı quraşdırıldıqdan sonra bankın İnternet vasitəsilə işi pozulmuşdur, belə ki, bəzi müştərilər özgələrinin hesabına daxil olmaq imkanı əldə etmişlər.

Avstraliyanın Ehtiyat Bankı 64 müştərisinə faiz dərəcəsinin plan üzrə 0,5% artması haqqında məlumatı nəzərdə tutulduğundan 6 dəqiqə tez göndərmişdi və həmin vaxt ərzində birja brokerləri bu səhvdən istifadə edərək 3 milyard avstraliya dolları həcmində borc öhdəliklərini və vekselləri sataraq külli miqdarda gəlir əldə etmişlər. Maraqlı hadisələr də olmuşdu. Eyni adı daşıyan iki qadın (Belinda Li Perri) elə eyni gündə 7 yanvar 1969-cu ildə anadan olmuşdular. Belə təsadüf onlara daim dolaşıqlıq vəd edirdi. Belindalardan biri vaxtaşırı öz sığorta vəsiqəsində və bank kartlarında məlumatların başqa məlumatlarla əvəz edildiyini aşkara çıxarırdı. Bu əhavalatda yeganə müsbət cəhət hər iki qadının dostlaşması oldu. Göründüyü kimi, ad və təvəllüd tarixi unikal identifikator roluna yaramır.

məsuliyyətə cəlb olunma maddəsi nəzərdə tutulmuşdur. Bu cinayətlərə aiddir:

• qanunla qorunan kompüter informasiyasına qeyri-qanuni müraciət, əgər bu, informasiyanın məhvi, blokadası, dəyişdirilməsi və ya köçürülməsinə, EHM-in və ya şəbəkənin işinin pozulmasına gətirib çıxarmışsa;

• ziyan verən proqramların yaradılması, istifadəsi, yayılması, yaxud da mövcud proqramlarda qəsdən informasiyanın icazəsiz məhvinə, blokadasına, dəyişdirilməsi və ya köçürülməsinə, EHM-in və ya şəbəkənin işinin pozulmasına gətirib çıxaran dəyişikliklərin edilməsi;

• qanunla qorunan informasiyanın məhv olunmasına, blokadasına və ya dəyişdirilməsinə gətirib çıxaran EHM və şəbəkənin istismar qaydalarının pozulması.

VİRUSLAR

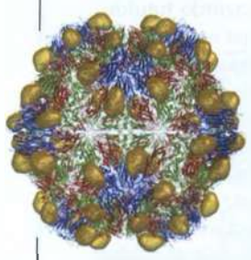
Rəqəmsal əsrin gəlməsilə insanlar böyük həcmli hesablamalardan azad oldu, avtomatlaşdırılması mümkün olan hər şey avtomatlaşdırıldı və bununla bərabər yeni problem və qayğılar da ortaya çıxdı. Dövrümüzün belə problemlərindən biri də *kompüter viruslarıdır*. Burada hər şey həyatda olduğu kimidir – özünəməxsus "xəstəliklər"i və "həkimlər"i var.

Kompüter virusu EHM-də istifadəçidən asılı olmayaraq fəaliyyət göstərən, adi virusun insana ziyan verdiyi kimi kompüterə zərər yetirən və öz surətlərini əlçatan informasiya daşıyıcılarında (disklərdə, disketlərdə) yaratmaq qabiliyyətinə malik olan proqramdır.

ÖZÜMÜZ SEÇDİYİMİZ VİRUSLAR

Qeyd etmək maraqlı olardı ki, bu tərif kompüter virusunu kifayət qədər dəqiq təsvir edir, lakin bu tərifə bir sıra digər proqramlar da uyğundur.

Məsələn, PC DOS yaxud MS DOS və eləcə də onlara bənzər əməliyyat sistemləri öz surətlərini yarada bilərlər. Windows-da isə sistemin istifadəçinin bilmədiyi bir çox hissələri fəaliyyət göstərir. Ziyana gəldikdə, öz kodunda səhvi olan hər bir proqram kompüterə və burada saxlanılan informasiyaya böyük zərər yetirə bilər.



Bioloji hüceyrəyə soxulmuş mozaik virusun strukturu.

Viruslar təxminən XX əsrin 70-80-ci illərində kompüter üçün ciddi təhlükə yaratmağa başlamışlar. Virus müəllifləri yeni "elektron salamlar"ın yaradılması üçün kompüter əməliyyat sistemlərində mövcud olan müxtəlif çatışmazlıqlardan istifadə edərək çoxlu sayda müxtəlif "yoluxucu xəstəliklər" yayıblar. Bəzən sağlam düşüncəli insanın ağlına belə gəlməz ki, bu işin hərəkətverici qüvvəsi nədir. Ola bilər ki, kimsə sadəcə olaraq təcrübə aparırmış, kimsə Herostrat kimi məşhurlaşmaq istəyirmiş, kimisi isə sadəcə olaraq proqramlaşdırmanı öyrənirmiş.

Kompüter viruslarının tarixçəsi EHM-in təhsil müəssisələrində peyda olması ilə başlayır. Məhz orada istedadlı tələbə-proqramçılar kompüter resursları (yaddaş, disk yaddaşı və hətta *prosessor vaxtı*), digər tapşırığa keçməmişdən öncə məsələnin həlli üçün prosessora ayrılmış millisaniyələr kimi faktor uğrunda bir-birilə "vuruşan" proqramlar yazırdılar.

Hay-küyə səbəb olan ilk virus Morris virusu oldu. Bu virus ABŞ-da yazılmış və dövlət müəssisələrindəki çoxlu sayda kompüteri uzun müddətə iflic vəziyyətinə salmışdı.

Fərdi kompüterlərin meydana gəlməsilə virusların sayı yağışdan sonra çıxan göbələklər kimi artdı. Buna fərdi

EHM-lərin artan populyarlığı və onlar üçün proqram vasitələrinin natamamlığı şərait yaradırdı.

"Virusyazma" coğrafiyasının inkişafını zamana görə asanlıqla izləmək olar:

• ABŞ, Avropa – 80-ci illərin əvvəlləri;

• Bolqarıstan, Hindistan – 80-ci illərin ortaları;

• SSRİ – 80-ci illərin axırları;

• Çin, Tayvan, Asiyanın digər ölkələri – 90-cı illərin əvvəli və ortaları.

Beləliklə, viruslar cəmiyyətin informasiyalaşdırılması üzrə növbəti "bum"un meydana gəldiyi yerlərdə peyda olmağa başladı. Öncə Amerika, sonra Avropa, Asiya... Hadisələr XX əsrin 80-90-cı illərində cərəyan edirdi.

Rusiyaya ilk viruslar iqtisadiyyatı inkişaf etmiş ölkələrdən köhnə kompüterlərlə birlikdə gəlmişdi. Sonradan kompüterlər Bolqarıstan və Hindistandan alınmağa başlandı və onlar bəzən zavodlarda "yoluxdurulurdu" (görünür, bu, bilməyərəkdən baş verirdi).

Tezliklə Rusiyanın öz virus "müəllifləri" ortaya çıxmağa başladı. Bir çox insanların proqram kodlarını bir-birindən sadəcə olaraq "çırpışdırmalarına" baxmayaraq, rus virusyazma "bum"u dünyanın qalan hissəsi üçün böyük qorxu mənbəyi idi. Rus proqramçılarının lazımi səviyyədə ənənəvi hazırlığı, həm də özünəməxsus yumor hissi virus "tutmaqda" ixtisaslaşan Qərb şirkətlərini yüngülvari çaşqınlığa salmışdı. Eyni zamanda Rusiyada antivirus proqramları yazılmağa başlandı. Şübhəsiz ki, onlar bəzi parametrlərinə görə xarici analoqlarından üstün idilər, belə ki, məhz rus proqramçıları dünyada ən istedadlı proqramçılardan hesab olunurlar. Onların işləri bir sıra ölkələrdə diqqəti cəlb etmiş və satış sayına görə lider olmuşdur. Sonradan virus istehsalı mərkəzi yerini Asiya istiqamətində dəyişdi, daha sonra mərkəz anlayışı ümumiyyətlə yoxa çıxdı. Bunun səbəbi, faktiki olaraq sərhədsiz olan İnternetin inkişafıdır.

Virusları müxtəlif cür təsnif etmək olar:

• *çoxalma üsuluna görə*. Viruslar ilə faylların, poçt proqramlarını, hətta sənədlərin yerləşdiyi mətn fayllarını məhv edə bilər. Müasir mətn redaktorları köməkçi əlavələr yaratmaq üçün daxili vasitələrə malikdirlər. Bu vasitələrdən yalnız işdə əlavə imkanlar yaratmaq üçün deyil, həm də ziyanverici proqramlar yazmaq üçün istifadə etmək olar.

• *fəaliyyət üsuluna görə*. Virus özünü müxtəlif cür göstərə bilər, onun fəaliyyət spektri səs və vizual effektlərin nümayişindən başlamış kompüterdə olan informasiyanın sakitcə silinməsinə qədər müxtəlif işləri əhatə edir;

• *vurulan ziyanın həcminə görə*. Viruslar həm zərərsiz, həm də olduqca təhlükəli ola bilərlər.

Əgər Windows ekrana "Əlləriniz yanlış komandanı daxil etdiyinə görə kəsiləcəkdir" kimi bir məlumat çıxarıbsa, çox güman ki, kompüterinizə virus düşüb.

BİR VİRUSUN TARİXÇƏSİ



Kiçik Robert Morris.

2 noyabr 1988-ci ildə Kornell Universitetinin informatika fakültəsinin aspirantı kiçik Robert Morris (Robert Morris Junior) yazdığı virusla İnternetə qoşulmuş çoxlu sayda kompüteri yoluxdurdu.

Yayılması üçün virus bir çox sistemlərdə quraşdırılmış standart UNIX əməliyyat sisteminin bəzi nöqsanlarından və lokal şəbəkələrdən uzaqda yerləşən kompüterlərə müraciət mexanizmindən istifadə etmişdir.

Virus iki hissədən – əsas proqram və onun yayılmasını təmin edən proqramdan ibarət idi. Əsas proqram növbəti kompüterə yüklәndikdən sonra şəbəkədə bu kompüterlə əlaqəsi olan digər kompüterlər barəsində informasiya toplayırdı. Sonra yayılma proqramının bu kompüterlərə göndərilməsi həyata keçirilirdi ki, burada o, virusun yerdə qalan hissəsinin göndərilməsi və kompilyasiyasını təmin edirdi. Bundan sonra bütün prosess yenidən təkrar olunurdu.

Virusun yayılması zamanı ən nəzərəçarpan effekt virusa yoluxmuş kompüterin yüklənməsinin durmadan artması idi. Müəyyən vaxt ötəndən sonra bəzi kompüterlər virusun surətlərinin yayılması nəticəsində artıq heç bir faydalı işi yerinə yetirmək qabiliyyətinə malik deyildilər. Digər maşınların isə yaddaşı həddən artıq yüklәndiyinə görə onları yenidən yükləmək lazım gəlirdi.



Kompüter proqramı EHM üçün təlimatlar yığınından ibarətdir. O daim informasiya daşıyıcısında saxlanılır, icra olunarkən kompüterin operativ yaddaşında yerləşir. Artıq deyildiyi kimi, virus da proqramdır. Virusun işi ilk növbədə EHM-də mümkün qədər çox proqramı elə dəyişməkdir ki, onun daxilində virusun özü şəklində bir əlavə yerləşsin.

Yoluxma təxminən belə baş verir: istifadəçi öz kompüterində artıq virus daşıyıcısı olan proqramı başladır. Bu, İnternetdən "yüklənmiş" və ya tanışlardan köçürülmüş bir proqram ola bilər. Proqramın yüklənməsindən əvvəl və ya sonra virus aktivləşərək fəaliyyətə başlayır. Onun fəaliyyət ssenarisi təxminən belədir:

1) EHM-də yoluxdurulması mümkün olan bütün proqramları tapmaq;

2) Özünü proqramın əvvəlinə və ya sonuna yazmaq;

3) Əgər "kritik" tarix (virusun "zərbə" vurduğu gün) yetişmişsə, dağıdıcı işlər görmək;

4) Əgər həmin tarix yetişməmişsə, o zaman hər hansı bir "xırda" zərər yetirmək, məsələn, EHM-in sərt diskində hər-hansı bir kiçik sahəni şifrləmək.

Yoluxmaya qədər istifadəçinin kompüterində yerləşən və artıq özündə virus saxlayan proqram yüklənərkən bütün ssenari təkrarlanır.

Bundan başqa, virus özünü diskdə xüsusi bir sahəyə – yüklənmə sektoruna yaza bilər. Bu ona istifadəçinin öz kompüterini yandırdıqdan sonra sistemin sərt diskdən yüklənməsi başlayarkən aktivləşməyə imkan verir. Belə üsul kompüterləri disketdən yükləmək lazım gəldiyi zamanlar populyar idi. Ümumiyyətlə, *yüklənmə* virusları *fayl* virusları qədər geniş yayılmadı.

Proqram fayllarını yoluxduran viruslar *fayl virusları*, özlərini kompüterin yüklənmə sektoruna yazan viruslar isə yüklənmə virusları adlanır. Həm faylları, həm də *yüklənmə sektorunu* ləğv edən viruslar *fayl-yüklənmə* virusları adlanır.

Bir qayda olaraq, viruslar çox kiçik ölçüyə malik olur və əməliyyat sisteminin proqramların təlimatlarında və xüsusi ədəbiyyatda göstərilməmiş imkanlarından istifadə edirlər. Bu onlara faylları sürətlə və gizli yoluxdurmaq imkanı verir. Bəzi viruslar "maskalana", yəni əksər antivirus proqramlarının başa düşmədiyi, iş prosesində öz-özünü dəyişdirə bilən dolaşıq və mənasız kod şəklini ala bilirlər. Belə "məmulatlar" *polimorflar* adlanır.

İnternetin inkişafı virusların yayılma sürətinə güclü təsir göstərdi. Bundan əlavə, viruslar keyfiyyətcə də dəyişiblər, onlar özlərini başqa maşınlara keçirmək və öz "yaradıcıları" ilə əlaqə saxlamaq üçün İnternet və lokal kompüter şəbəkələrindən istifadə etmək imkanı qazanıblar. Virus yayıcıları kompüterləri yoluxdurmaq üçün çox vaxt orijinal fəndlərdən istifadə edirlər. Bu fəndlərdə insanların zəif damarı, yəni onların "qadağa qoyulmuş meyvəyə", məsələn, pulsuz pirat proqramlara, "krək"lərə və Ümumdünya şəbəkəsinin bu kimi cazibələrinə marağı nəzərə alınır. Belə proqramların fəaliyyət sxemi aşağıdakı kimidir: istifadəçi İnternet mühitində saytlara pulsuz giriş vəd edən yaxud məşhur proqram məhsulunun mühafizəsini sındırmaq üçün hər hansı bir proqram tapır. O, həmin proqramı öz kompüterinə yükləyir və işə başladır. Bununla artıq iş görülmüş olur. Proqram məlumat verir ki, bu və ya digər səbəbdən icra oluna bilmir, özü isə istifadəçinin kompüterinə virusu "otuzdurur". Və yaxud da, məsələn, sizinlə tanış olmaq istəyən tanımadığınız bir adamdan məktub gəlir və o, sizə məktuba əlavə olunmuş fotoşəkillərə baxmağı təklif edir. Lakin nəyə görəsə fotoşəkillər icra olunan fayllar olur... Bu cür fənd bədniyyətliyə və ya arzu edən hər bir kəsə hədəf seçilmiş şəxsin kompüterinə müraciət etmək və oradan istənilən informasiyanı əldə etmək imkanı verir. Bir qayda olaraq, bu proqramlardan İnternetə çıxış üçün parolları ələ keçirmək məqsədilə istifadə olunur. Bu tip viruslar "*troyalı*" və ya "*Troya atı*" adlanır.

"Troyalı" proqramlardan başqa "*soxulcan*" adlanan virus növü də mövcuddur. Bu proqramların da "troyalı"lar kimi İnternet vasitəsilə fəaliyyət göstərməsi nəzərdə tutulmuşdur. "Soxulcan"lar kompüterdən kompüterə şəbəkə vasitəsilə keçirlər və onların vurduğu əsas ziyan kompüter resursları və İnternet kanallarının həddən artıq yüklənməsi olur. Kompüter "soxulcan"ına misal olaraq Morris virusunu göstərmək olar.

Viruslar müasir fərdi kompüterlər üçün real təhlükə daşıyır. Əgər 5-7 il öncə virus müəllifinin əsas məqsədi kompüteri sıradan çıxarmaq idisə, XXI əsrin əvvəllərində viruslar əsasən kompüterdən hər hansı bir informasiyanı oğurlamağa və kompüterinizə kənar şəxslərin müraciətini təmin etməyə yönəlib. Əlbəttə, kompüter viruslarının tarixində İnternetin inkişaf etməsinin rolu heç də



ƏN MƏŞHUR "TROYALI"

Yəqin ki, ən çox səs-küyə səbəb olan "troyalı" 1997-1998-ci illərdə daha çox "uğur" qazanan BackOrifice oldu. Bu adı "Arxa qapı" kimi tərcümə etmək olar. Bu proqram kompüteri uzaqdan idarə etmək üçün bir sistem kimi düşünülmüşdü və hərfi mənada istədiyiniz HƏR ŞEYİ etməyə imkan verirdi (CD ROM disk sürənin altlığınıhərəkət etdirməkdən başlamış hədəf-kompüterə qoşulmuş videokameradan informasiyanın götürülməsi və ya səsin mikrofon vasitəsilə yazılmasına qədər). BackOrifice kodunda kiçik dəyişikliklərin edilməsi hədəf-kompüterə virusu gizlicə oturdaraq bu proqramdan çirkin məqsədlər üçün istifadə etmək imkanı yaradırdı.

Proqramın müəllifləri əvvəl heç fikirləşmirdilər ki, yazdıqları proqram bu qədər hay-küy yaradacaq. Onlar yalnız kompüterin uzaqdan idarə olunması üçün alət yaratmışdılar, lakin çox keçmədən bu proqram İnternet istifadəçiləri üçün əsl bəlaya çevrildi. İstifadəçilər bir-birlərinə BackOrifice "yapışdırılmış" fayllar göndərirdilər və məktuba əlavə olunmuş proqram işə salınarkən onunla birlikdə BackOrifice proqramı da aktivləşirdi. O, sistemdə zərərsiz bir prosess kimi maskalanırdı və istifadəçi onu xüsusi antivirus proqramları olmadan aşkar edə bilmirdi.

Siz yalnız əməliyyat sisteminin pulunu ödəyirsiniz! Səhvlər, qüsurlar və "boşluqlar" tamamilə pulsuzdur, yəni müftədir!

Başqasının kompüterini uzaqdan idarə etmək üçün heç də aşkar olunmaq qorxusu ilə bu kompüterə "troya atı" otuzdurmaq vacib deyil. Müasir əməliyyat sistemləri həddən artıq iri, etibarsız və hücumlara açıqdır. Təəssüflər olsun ki, səhvlərin və mühafizədəki "boşluq"ların sayı versiyadan versiyaya keçdikcə azalmır. Windows ƏS-in tərkibinə daxil olan məşhur Internet Explorer brauzerinin son versiyaları istifadəçinin xəbəri olmadan baxdığı veb-səhifələr vasitəsilə başqalarına onun kompüterinə girişi təqdim edir. Yəni istifadəçi veb-sayta baxarkən kompüterə xüsusi proqram kodu ötürülə bilər. Bu kod yerinə yetirildikdə İSTƏNİLƏN proqramı onun kompüterində yükləmək imkanı yaranacaq.

az deyil. Virusların yayılma kanallarının sayı xeyli artmışdır. Bunlar həm elektron poçtu, həm pirat saytları və hətta sadə sənədlərdir. İnformasiyanı oğurlayan virus bir kommersiya şirkətinin kənar şəxslər üçün nəzərdə tutulmamış sənədlərini açıqlamaqla onun nüfuzuna ciddi ziyan vura bilər. Əgər belə virus, məsələn, hərbçilərin və ya dövlət müəssisəsinin fəaliyyətilə bağlı olan kompüterə düşsə necə olar?

Qərbdə kompüter viruslarının vurduğu ziyan yüz milyon dollarlarla ölçülür.

Bəs kompüter virusundan necə qorunmaq olar? Bunun üçün *antivirus* adlanan kompüter proqramları mövcuddur.



Antivirus proqramı belə fəaliyyət göstərir: proqram "bilir ki" virusun "xarici görünüşü" necədir. Daha doğrusu, virus-proqramların fraqmentlərini öz verilənlər bazasında saxlayır. Verilənlər bazasını vaxtaşırı yeni virus "nümunələri" ilə yeniləmək mümkündür. Antivirus kompüterdəki fayllarda oxşar fraqmentləri axtarır və əgər taparsa, "müalicə" prosedurunu həyata keçirməyə çalışır.

"Müalicə" isə bu şəkildə həyata keçirilir: antivirus proqramı fayllardan virusun "kritik" hissələrini silməklə onları təmizləməyə çalışır, sonra isə virusun fəaliyyətinin nəticələrini aradan qaldırır, məsələn, korlanmış informasiyanı bərpa edir.

Bundan başqa, daim EHM-in yaddaşında yerləşib bütün kritik vəziyyətləri və şübhəli əməliyyatları (proqramların başladılması, son dərəcə vacib olan sistem fayllarına yazma cəhdləri və s.) izləyən monitor proqramlar da mövcuddur.

Lakin antivirus proqramlarının mükəmməlliyinə ümid etmək olmaz. Hər hansı bir "infeksiyaya" yoluxmamaq üçün sadə qaydalara əməl etmək kifayətdir:

- kənar şəxslərin kompüterinizdə işləməsinə icazə verməyin;
- leqal proqram təminatından istifadə edin. Bəzən pirat nüsxələrində çoxlu sayda "troya" virusları aşkar edilir;
- yad adamlardan məktub alarkən məktubla birlikdə icra faylı şəklində hər-hansı bir "əlavə"nin gəlmədiyinə əmin olun;
- hətta kompüterinizin iş qabiliyyətinin 300% artacağını vəd etsələr belə, araşdırmadan İnternetdən rastınıza çıxan hər proqramı yükləməyin. Yaxşı olardı ki, təklif olunan proqramlar haqqında məlumatı xüsusi icmal saytlarında axtarasınız.

Bəs virusların müəllifi kimlərdir? Bütün dünyanı fəth etmək arzusunda olan bədniyyətlilərmi? Öz rəqiblərilə mübarizə aparan şirkətlətin xüsusi şöbələri və yaxud təxribatçılarmı? Adətən, bu belə deyil. Çox vaxt bunlar ali məktəb tələbələri, bəzən hətta yuxarı sinif şagirdləri olur. Bəs bu insanları belə bir addım atmağa vadar edən nədir? Birinci növbədə elmi maraq. Onlar üçün kompüterlə bağlı olan hər şey maraqlıdır, amma onlar hərəkətlərilə vuracaqları ziyanı heç də hər zaman başa düşmürlər...

Bəzən onlar heç güman etmirlər ki, "törəmə"ləri artıq azadlığa çıxıb öz işini görür.

Azərbaycan Respublikasının Cinayət Məcəlləsində EHM üçün ziyan verən proqramların yaradılması və yayılmasına dair virus müəlliflərinin fəaliyyəti haqqında maddə mövcuddur. Əgər müəllif ələ keçərsə, onu azadlıqdan məhrumetmə cəzası gözləyir.

Bəs yaxın zamanlarda kompüter virusları və antivirus proqramlarını hansı perspektivlər gözləyir?

Antiviruslarla vəziyyət aydındır. Onlar bundan sonra da təkmilləşdiriləcək alqoritmlər də təkmilləşəcək, virusların axtarışı və "müalicə" isə daha da effektli olacaqdır.



Viruslara gəldikdə isə, vəziyyət bir az mürəkkəbdir, buna baxmayaraq, müəyyən proqnozlar vermək olar. Birincisi, dünya hələ bir neçə "virusyazılma" dalğasına məruz qalacaqdır. İkincisi, viruslar üçün fəaliyyət sahəsi genişlənir və artıq onların hədəfi ancaq ev və ofis EHM-ləri deyil, həm də cib kompüterləri, mobil telefonlar və prosessorlu mürəkkəb elektron avadanlıqlar olacaqdır.

SINAQ MÜDAXİLƏSİ

"Mən ən vacib məhsulun hazırlanma texnologiyasını şirkət prezidentinin qarşısına qoydum. Şirkətin bütün məhsullarının hazırlanması üçün ətraflı plan prezidentin masası üzərinə qoyulanda, o, hələ də susurdu. Bu kağızlara çoxmilyonluq məhkəmə prosesi barədə danışıqların sənədləri də əlavə olunanda, prezident əyləşdiyi kürsüdə arxaya söykəndi. Nəhayət o, dedi: "Allaha şükür etməliyik ki, siz rəqiblərimizə işləmirsiniz".

Mən bütün bunları və daha bəzi şeyləri şirkətdə müvəqqəti işçi olduğum zaman oğurlamışdım. Fikirləşirsiniz ki, söhbət təhlükəsizlik xidməti zəif təşkil edilən bir şirkətdən gedir? Əsla yox. Xarici mühafizə çox yüksək səviyyədə təşkil olunmuşdu, lakin rəhbərlikdə şübhələnirdilər ki, şirkətin daxili mühafizəsi zəif ola bilər. Məqsədyönlü şəkildə hərəkət edən informasiya oğrusunun nələri oğurlaya biləcəyini yoxlamaq üçün mənə müraciət etdilər.

Şirkətdə bunun üçün mənə üç gün vaxt verdilər. Mən hər şeyi oğurladım..."

"Korporativ casusluq" kitabının müəllifi, informasiya təhlükəsizliyi sahəsində ən məşhur ekspertlərdən biri, Internet Security Advisors Group (ISAG, "İnternetdə təhlükəsizlik üzrə ekspertlər qrupu") şirkətinin yaradıcısı və prezidenti, sınaq müdaxilələri üzrə əvəzedilməz mütəxəssis Ayre Vinklerin (Ira Winkler) hekayəsi belə başlayır. O, təhlükəsizlik sistemində boşluğu aşkar edən casusluq üsullarını təsvir edir.

İstənilən korporasiyada informasiya ciddi şəkildə qorunur. Bu zaman istifadə olunan mühafizə vasitələri bəzən Fort-Noksda tətbiq olunan vasitələrlə müqayisə oluna bilər. Mühafizənin kinolarda



Ayre Vinkler.



göstərildiyi kimi birbaşa sındırılması demək olar ki, mümkün deyil. Korporasiyanın İnternetə çıxışı olan kompüterlərində Windows sisteminin sonuncu versiyasından istifadə etmək heç bir ciddi təhlükəsizlik xidmətinin ağlına belə gələ bilməz. Bu kimi qeyri-ciddi əməliyyat sistemlərində açıq qalan "arxa qapıları" sındırmaq üçün heç bir xüsusi vasitə tələb olunmur. Sadəcə içəri daxil olmaq lazımdır. "Boşluq"ların siyahısı isə daim təzələnir və arzu edən hər bir kəs bu siyahını İnternetdən götürə bilər.

Xüsusi komissiyalar tərəfindən müəyyən təhlükəsizlik dərəcələri üzrə sertifikatlaşdırılmış əməliyyat sistemləri isə başqa məsələdir. ABŞ, Avropa və Rusiyada onlar müxtəlif cür adlanır, lakin onlara qoyulan tələblər eyni dərəcədə yüksəkdir.

Amma daha təkmilləşdirilmiş bir üsul da var ki, bu da daxildən sındırmadır. Bunun üçün müəyyən işlər aparmaq lazımdır.

Casusluqda bütün mümkün texniki və qeyri-texniki üsullar və ya *hücumlardan* istifadə olunur.

Beləliklə, Vinkler korporasiyaya qarşı beş hücum növündən istifadə etmişdir:

- açıq informasiya mənbələri üzrə axtarış;
- maskarad;
- daxilolma səlahiyyətlərini aşmaq;
- ştatda olan işçinin hakerliyi;
- xarici ortaqların hərəkətlərinin daxildən koordinasiyası.

AÇIQ İNFORMASİYA MƏNBƏLƏRİ ÜZRƏ AXTARIŞ

Hər şeydən əvvəl hücum obyektini ətraflı şəkildə tədqiq etmək lazımdır. Təəccüblüdür, çoxlu sayda informasiyanı İnternetdən necə də asanlıqla əldə etmək olar. Şirkətin layihələri barəsində məlumat əldə edib, bu tədqiqatlara aid elmi məqalələri oxumaq olar. Bir qayda olaraq, şirkətin saytında rəhbərlik üzvlərinin adları, onun maliyyə vəziyyəti, şirkət və onun korporativ siyasəti haqqında çoxlu ümumi məlumat verilir. Bəzən aparıcı mütəxəssislərin adlarını da müəyyən etmək mümkün olur. İnternetin xəbərlər qrupunda şirkətin adına görə axtarış apararkən heyrətləndirici sürprizlərlə rastlaşmaq olar. Ayre Vinkler nəinki yuxarıda göstərilən bütün məlumatları əldə etmiş, hətta şirkətin bir çox əməkdaşının adlarını da öyrənmişdi. Əməkdaşların texniki xəbərlər qrupuna məktublarından şirkətin kompüter təchizatı və proqram təminatı haqqında informasiya əldə olundu, digər qruplara göndərilmiş məktublar isə məktub müəlliflərinin şəxsi maraqları haqqında məlumat əldə etməyə kömək etdi.

MASKARAD

Casus artist olmalıdır, peşəkar casus isə ikiqat artist olmalıdır ki, ifşa olunduqda illərlə vaxtını həbsxanada keçirməli olmasın. Özgələrinə şübhə ilə yanaşmaq, onunla eyni korporasiyada çalışan insanları isə mehribanlıqla qarşılamaq çoxlarına xas olan xüsusiyyətdir.

ABŞ-ın ciddi şəkildə mühafizə olunan Fort-Noks qızıl ehtiyatları anbarı eyniadlı hərbi şəhərciyin demək olar ki mərkəzində yerləşir.



Maskarad vizit kartlarının hazırlanmasından başladı (nümunə qismində bir şirkət işçisinin həqiqi vizit kartı istifadə olunmuşdur). Doğrudur, onun vəzifəsi dəyişdirilmişdi: casus sıravi, müvəqqəti işçidən informasiya təhlükəsizliyi üzrə əməkdaşa çevrilmişdi.

Yalnız müvəqqəti işə qəbul olunma maskaradsız ötüşdü. Lakin burada da müvəqqəti işçinin blankları yalnış informasiya ilə doldurulmuşdu. Sosial sığorta nömrəsi, ünvan və telefon saxta idi.

Hər bir casusun öz üslubu var. Lakin düzgün seçilmiş vəzifə möcüzələr yaradır. Vinkler şirkətin informasiyasının qorunması haqqında bir əfsanə fikirləşdi. O, çox əhəmiyyətli bir layihə üzərində işləyən tədqiqatçıya dedi ki, guya ştata bu yaxınlarda götürülüb və ona tapşırılıb ki, layihənin daha çox mühafizəyə ehtiyacı olan hissəsini və müvafiq informasiyanın saxlanıldığı yeri müəyyən etsin. Tədqiqatçı layihə rəhbərinə müraciət etməyi məsləhət bildi. Ayre Vinkler belə də etdi. O, layihə rəhbərilə görüşüb ona vizit kartını təqdim edərək xahiş etdi ki, hansı informasiyanın vacibliyinə görə kritik olduğunu və kimlərin bu informasiyaya daxilolma səlahiyyətinə malik olduğunu ona izah etsin. Ona cavab verdilər ki, ən vacib informasiya şirkətin məhsullarının hazırlanması haqqında məlumatdır. Və ona bu məlumatların əsas mənbəyi olan layihə qrupunun iclas protokollarının surətlərini və bu protokolları alan əməkdaşların siyahısını göstərdilər. Layihə rəhbəri casus üçün nəinki bütün sənədlərin surətini çıxartdı, hətta onun adını protokolların paylanması siyahısına əlavə etdi.

Şirkətin digər aparıcı əməkdaşları ilə bu kimi söhbətlər informasiyanı bir qədər də dolğun etdi. Maskarad həqiqətən də bol məhsul gətirir, lakin təsadüfləri də unutmaq olmaz. Alınmış məxfi məlumatlar arasında direktoriya və giriş üçün parolu göstərilmiş məxfi bir sənəd də var idi. Həmin kataloqda dəyəri milyard dollardan artıq olan məxfi sənədlə yanaşı şirkətin daha iki əhəmiyyətli layihəsini təsvir edən digər sənədlər də vardı. Təbii ki, sənədlərin elə həmin an surəti çıxarıldı.

DAXİLOLMA SƏLAHİYYƏTLƏRİNİ AŞMAQ

Bir qayda olaraq, şirkətin ofisində kompüterlər xaricdən sındırılmaya qarşı istifadə olunan mühafizəyə nəzərən daha zəif qorunur. Məsələn, məsafədən şirkətin məxfi şəbəkəsinə daxil olmaq mümkün deyil, belə ki, şəbəkə sadəcə olaraq xarici rabitə kanallarına birləşdirilməyə bilər. Lakin casus şirkətin daxilində parolu tapa bilsə, əməkdaşlardan birinin kompüterində mühafizəni



"Canıbərk" (Die Hard) filmindən kadr.

sındıra bilər. Çox vaxt şirkətin işçiləri mürəkkəb parollar fikirləşmək üçün özlərinə əziyyət vermirlər, onlar əsasən xüsusi isimlərdən istifadə edirlər: Barbie, Ken, Starwars və s.

Parol bəlli olarsa yüksək vəzifəli əməkdaşın kompüterində saxlanan məxfi informasiyanı da asanlıqla əldə etmək olar.

Casusluq fəaliyyətində hər bir xırda detal əhəmiyyətlidir. Aparıcı əməkdaşlardan birinin qapısında qıfılın olmaması və ya üzərində "Şirkətin hesabatları" yazılan masanın üstünə səpələnmiş disketlər primitiv oğurluq planının hazırlanması üçün imkan yaradır.

Casus süpürgəçi kimi fəaliyyət göstərir, açıq qalmış otaqları, şkaf və yeşikləri axtarır. Onu zibil yeşikləri (orada sistem parolları qeyd olunmuş kağız parçaları ola bilər), seansın başa çatdırılması proseduru yerinə yetirilməmiş kompüterlər (bəzən sistem blokunu işlək vəziyyətdə saxlayaraq, sadəcə monitoru söndürürlər) maraqlandırır.

Vinklerin bəxti gətirdi. Yaddaşında elektron məktublar olan kompüterin açıq qalması işlərin əsas qrafikinin – şirkətin ən məxfi sənədlərindən birinin surətini çıxarmağa imkan verdi.

ŞTATDA OLAN İŞÇİNİN HAKERLİYİ

Hesablama texnikasının hazırkı inkişaf səviyyəsində daşınan kompüterlər həqiqətən də miniatür ölçüdədirlər. Bu isə casusa əvvəlcədən hakerlik üçün hazırladığı noutbukundan istifadə etmək imkanı verir. İnternetdə həddən artıq haker aləti mövcuddur, bu alətlər skaner (haker proqramı) vasitəsilə aşkar edilmiş boşluqlardan istifadə etməklə sistemləri "sındırmaq"üçün əsas vasitədir. Casusa öz iş otağında ofis kompüterini Ethernet şəbəkəsindən ayırıb, şəxsi noutbukuna qoşmaq kifayətdir.

Şirkətin kompüterlərində olan məlumatlar skanerdən keçirilir. Əgər mühafizə vasitələrinin yenisilə əvəz edilməsi azacıq da olsa gecikibsə, sistemdə üzə çıxarılmış və haqqında İnternetdə açıq şəkildə məlumat verilən boşluqlar varsa, belə maşınların taleyi artıq həll olunub. Bəzən casus super-istifadəçi səlahiyyətlərini ələ keçirə bilir. İstənilən halda boşluq sındırılmış maşınlardan alınmış məxfi informasiyanı hakerin noutbukuna köçürməyə imkan verir. Bundan sonra parolları seçmək məqamı yetişir.

XARİCİ ORTAQLARIN HƏRƏKƏTLƏRİNİN DAXİLDƏN KOORDİNASİYASI

Casus xaricdən dəstək aldıqda, müvafiq komanda şirkətin xarici şəbəkəsini də sındırmaq qabiliyyətinə malik olur. Bunun üçün istifadəçinin adı, parolu və modemlə giriş üçün telefon nömrəsi tələb olunur. Adətən şirkətin əməkdaşları belə məlumata malik olurlar. Bu növ informasiyanı, misal üçün, telefonla ötürmək və ya hətta kuryerlə göndərmək olar.

Casus üstəlik şirkətin işçilərinin sistemə giriş adları və parollarının siyahısını əldə edərsə, onun şərikləri şirkətin praktiki olaraq bütün kompüter sistemlərinə soxula bilərlər. İcazəsiz girişdən müdafiə məqsədilə layihələndirilmiş uzaqdan qoşulma vasitələri qeydiyyatdan keçmiş istifadəçilərin qarşısını ala bilməz.

Özünü ələ verməmək üçün casus şirkəti tərk etmir. Buna görə də haker-ortağın hərəkətləri şirkətin əməkdaşlarının dürüstlüyünə kölgə salır.

Ayre Vinkler üç gün ərzində 300 Mbayt-dan çox məxfi verilənləri ələ keçirdi. Şirkətin beş ən vacib məhsulunun istehsal texnologiyası barədə müfəssəl informasiya onun əlində idi. Bundan əlavə, o, praktiki olaraq, bütün yeni tədqiqat işləri barəsində informasiya almışdı. Bu informasiya rəqiblərin əlinə düşsəydi, şirkət müflis ola bilərdi.

Casus fəaliyyət göstərdiyi zaman kobud üsullardan istifadə etsə də, heç kəsdə şübhə yaratmadı. Peşəkar sənaye casuslarının arsenalı olduqca genişdir. Məsələn, "böcək"lər quraşdırıb telefon xətlərinə qulaq asmaq və ya işçiləri öz tərəfinə cəlb etmək olar. Casuslar hücumun hazırlanması və planlaşdırılmasına real hücumun həyata keçirilməsindən daha çox vaxt sərf edirlər.

Mükəmməl xarici müdafiə xəttinə malik olan şirkət ştatda olan belə işçinin casusluq fəaliyyəti qarşısında acizdir. Bu isə onunla izah olunur ki, məxfi korporativ informasiya açıq olur, xarici müdafiə isə qüvvəsini itirir.

HAKERLƏR

"Bu bizim dünyamızdır! Elektronika və kommunikasiya dünyası, verilənlərin ötürülməsi sürətinin gözəllik aləmi. Bizim üçün millət anlayışı, dərinin rəngi və dini mövhumat yoxdur. Siz, bütün bunlar bizim mənafeyimiz üçündür deyərək müharibələr aparır, qətllər törədir, aldadır və beyinlərimizi yumağa çalışırsınız, son nəticədə isə cinayətkar biz oluruq! Bəli, mən cinayətkaram! Məni cinayətə hər şeyi bilmək istəyim sövq edir. Mən HAKERƏM və bu mənim manifestimdir! Siz məni tuta bilərsiniz, lakin BİZİM hamımızı tuta bilməzsiniz" ("Hakerin manifesti"ndən).

Axı haker kimdir? Bu söz vur-tut 10 il bundan əvvəl əsaslı şəkildə məişətimizə daxil olmuşdur.

Lüğətdə ingilis dilində "to hack" feili belə izah edilir:

- baltalamaq, baltalayıb ayırmaq, didib tökmək, hissələrə ayırmaq;
- yonmaq, yonub hazırlamaq, çapmaq, diş-diş etmək;
- cansıxıcı, ağır işdə istifadə etmək;
- ədəbiyyat sahəsində muzdlu kimi istifadə etmək.

Kompüter texnologiyalarının mahir bilicisi, peşəkar proqramçı olaraq, kompüter şəbəkələri və əməliyyat sistemlərinin sındırılması üzrə mütəxəssis olan şəxsləri haker adlandırırlar. Ümümiyyətlə, bu sözdə bütöv bir nəslin ideologiyası, onun həyat tərzi və fəlsəfəsi öz əksini tapmışdır.

Tipik hakerin portretini nəzərdən keçirsək görərik ki, bu, kompütərə ciddi şəkildə, hətta dəlicəsinə aludə olan, qeyri-standart düşünən, qapalı həyat tərzi keçirən, 15 yaşından 35 yaşınadək adamayovuşmazın biridir. Ətrafdakılar onu sakit, fağır hesab edir. O, riyaziyyatı və fizikanı yaxşı bilir, nadir hallarda idmanla məşğul olur və onun bütün həyatı kompüterdir – məktəbdə, universitetdə, işdə və evdə.

Kinematoqrafiya bəşəriyyətə hakerlər haqqında ən azı iki yadda qalan film bəxş etmişdir: rejissor Con Bədhəmin (John Badham) "Müharibə oyunları" (War Games) və rejissor Ayeyn Softlinin (Iain Softley) "Hakerlər" (Hakers) filmi.

Birinci filmdə fərdi kompüterdə "Ümumdünya nüvə müharibəsi" videooyunu oynayan Amerika məktəblisi gözlənilmədən Pentaqonun (ABŞ-ın hərbi idarəsi) kompüter sisteminə qoşulur. Ekrandakı müharibə az qala gerçəkliyə çevrilmişdi...

"Hakerlər" filmində hadisələr Nyu Yorkda baş verir. Min beş yüz lokal



"Hakerlər" filmindən kadr.

Britaniya Mülki Aviasiya nazirliyinin məlumatına görə radiohakerlərin uçuşa nəzarət sisteminə soxularaq səhv komandalar verməsi halları qeydə alınmışdır. Belə halların sayı artmaqdadır: 1988-ci ildə – 3, 1999-cu ildə – 18, 2000-ci ildə – 20.



kompüter şəbəkəsinə virus yaydığına və Uoll-Strit birjasında çaxnaşma yaratdığına görə on iki yaşlı kompüter dahisinin valideynləri böyük məbləğdə cərimə olunurlar. Onun özünə isə şərti olaraq, yəni azadlıqdan məhrum edilmədən, yeddi il iş kəsilir, 18 yaşına qədər kompüter və telefondan istifadə etmiş qadağan olunur. Təzə məktəbdə o, özü kimi hakerlərin dəstəsilə tanış olur. Onlardan biri təsadüfən böyük bir korporasiyanın şəbəkəsinə daxil olur və milyonlarla dolları oğurlamaq üçün "soxulcan" virusundan istifadə edərək kompüter təhlükəsizliyi xidmətinin başçısını ifşa edən faylları oradan köçürür. Məsələdən xəbər tutan oğru hakerləri günahlandırmaq qərarına gəlir, lakin dünyanın bütün hakerləri İnternet vasitəsilə birləşirlər və ədəbsiz cinayətkarla mübarizəyə başlayırlar...

Real həyatda hakerlər əsasən proqramçı və ya şirkətlərdə kompüter texnologiyaları üzrə mütəxəssis kimi işləyirlər. Hakerlik peşəsi yoxdur. Bu hər şeydən əvvəl psixika və düşüncə tərzidir. Uşaqlıqda çoxları oyuncaqları sındırırlar ki, içərisində nə olduğunu bilsinlər. Lakin uşaqlar böyüyürlər, oyuncaq əvəzinə onların bəzilərinin əlinə kompüter və yeni əməliyyat sistemləri düşür. Maraq onları proqram və kompüterin özünü "söküb" öyrənməyə, nəyi isə təkmilləşdirməyə sövq edir. Bəzən bu işdən həqiqətən də faydalı bir şey alınır.

Müasir əməliyyat sistemləri olduqca mürəkkəb və böyükdür. Sistem isə mürəkkəb olduqca işin təhlükəsizliyi və stabilliyi baxımından daha zərif olur. Belə səhvləri və çatışmazlıqları ilk olaraq hakerlər tapırlar və bir qayda olaraq tapılan qüsurlar barəsində istehsalçılara məlumat verirlər.

Bu, həmçinin veb-saytlara da aiddir. İnternetdə gündə yüzlərlə sayt mühafizədə boşluqları aşkara çıxarmaq məqsədilə hakerlər tərəfindən tədqiq olunur.

Zarafatla deyirlər ki, haker lazım olan düymələri müəyyən ardıcıllıqla basmağı bacaran adamdır. Bəzən hakerləri, sındırmaq məqsədilə proqramlardan istifadə etməklə (belə proqramlar İnternetdə çoxdur) universitetlərin, xeyriyyə təşkilatlarının, məktəblərin saytlarına girərək, əyləncə üçün onları korlayan adi tələbələrlə dəyişik salırlar. Belə saytlar təhlükəsizlik baxımından ən zərif saytlardır, belə ki, onlar məxfi materiallar daşımır və pulsuz dəstəklənir. Təəssüf ki, dələduzlar tərəfindən saytların sındırılmasına İnternetdə olduqca tez-tez rast gəlinir. Kompüter dələduzları özlərini fəxrlə haker adlandırsalar da, belə vandalizmin hakerlərlə əlaqəsi yoxdur. Maraq üçün yoldan keçənlərə atəş açmaqla supermen olmaq qeyri-mümkündür.

Ümumiyyətlə, qanun nöqteyi-nəzərindən hakerlik fəaliyyəti həmişə birmənalı şərh edilmir. Bir tərəfdən, beynəlxalq qanunvericilikdə kompüter sistemlərinə və verilənlər bazasına qeyri-qanuni müdaxiləyə görə heç də az olmayan müddətə azadlıqdan məhrum edilmə cəzası nəzərdə tutulan maddələr var. Əlbəttə, məxfi informasiyaya hakimiyyət orqanlarından və ya



"AÇIQ QAPILAR"

Hakerlərlə çoxlarını qorxudurlar, lakin bəzən oğurluğa səbəb gənc proqramçıların super intellekti yox, sadəcə "açıq qapılar" olur.

1999-cu ilin sonunda 19 yaşlı Rusiyalı haker musiqi disklərinin satışı ilə məşğul olan "CD Universe" İnternet-mağazasının veb-saytına daxil olub, bu şirkətin müştərilərinə məxsus 300 min kredit kartı haqqında informasiyanı ələ keçirməyə müvəffəq olmuşdu. O, şirkətdən 100 min dollar pul tələb etmişdi, rədd cavabı aldıqda isə öz saytında 25 min kredit kartı haqqında informasiyanı dərc etməyə başlamışdı. Təbii ki, həmin elektron səhifə bağlandı, lakin buna baxmayaraq, kənar adamlar bu informasiyadan istifadə etməyə imkan tapmışdılar.

2000-ci ildə FTB (FBI – ABŞ Federal Təhqiqatlar Bürosu) 18 yaşlı Rafael Qreyi həbs etdi. O, doqquz İnternet mağazanı sındıraraq ABŞ-da, Kanadada, Taylandda, Yaponiyada və Britaniyada 26 min kredit kartı haqqında informasiya oğurlamışdı (yeri gəlmişkən, bu kredit kartlarından biri Microsoft-un kralı Bill Geytsə məxsus idi). Haker öz hərəkətilə saytların hansı səviyyədə qorunduğunu göstərmək istəyirdi.

Rusiyada hakerlər "Qazprom"un kompüter sistemlərinə daxil olmuşdular. Rusiya Daxili İşlər Nazirliyinin məlumatına görə onlar qazın boru kəmərlərilə axınını idarə edən sistemi nəzarət altına ala bilmişdilər.



sahibindən xəbərsiz daxil olan haker cinayətkardır. Lakin, digər tərəfdən, əgər bu informasiya "üzdədirsə", yəni əməliyyat sistemi və veb-serverin quruluşu ilə azacıq tanış olan hər bir kəs onu asanlıqla əldə edə bilirsə, buna görə insanı mühakimə etmək olarmı? Yaxud başqa bir misal: proqramın mühafizəsini sındırdıqdan sonra haker onun sahiblərini mövcud olan "boşluqlar" barədə xəbərdar edir, bununla da, faktiki olaraq, həmin boşluğu "bağlayır". Bu halda, yəqin ki, haker cinayətdə günahlandırmaq olmaz.

Əlbəttə, real həyatda cinayətkarlar olduğu kimi, kompüter dünyasının da öz qanun pozucuları vardır. Onların hücum obyektləri müxtəlif ölçülü banklar, birjalar, İnternet-mağazalar olur. Qanun hamı üçün bir olduğundan, hətta dahilər tərəfindən də pozula bilməz. Hakerlərin istənilən qanuna zidd hərəkətlərinin üstü açılacaq və cinayətkar cəzalandırılacaqdır. Ən azı buna inanmaq istərdik.

KOMPÜTER CİNAYƏTKARLIĞI

Kompüterdə yol verilən səhvlərin və cinayətlərin vurduğu ziyan çox böyükdür.

Kompüter cinayətkarlığı üzrə Amerika Milli İnformasiya Mərkəzinin məlumatına görə təkcə bir il ərzində cinayətkarlar Amerika firmalarına 500 milyon dollar həcmində ziyan vurmuşlar. Almaniyada kompüter mafiyası ildə 2 milyard avro vəsait oğurlayır. XX əsrin axırlarında Fransada analoji itkilər ildə 1 milyard franka çatmışdır. Oxşar cinayətlərin sayı hər il 30-40% artır.

İnformasiya cinayətləri haqqında məlumatlar tam deyil. Bu gün dünyada heç kəs informasiya təhlükəsizliyinin pozulması barəsində tam təsəvvürə malik deyil. Avropa və Amerikada kompüter cinayətkarlığı nəticəsində illik ümumi itkilər (sadə hesablamalara görə) on milyardlarla dollar təşkil edir. Aydındır ki, zərərçəkmiş dövlət və kommersiya strukturları öz müdafiə sistemlərinin "səmərəliliyini" nümayiş etdirməyə o qədər də həvəsli deyillər. Ona görə də cinayət halları heç də həmişə ictimaiyyətə məlum olmur. Lakin məlum olan faktlar güclü təəssürat yaratmaq üçün kifayətdir. BMT-nin cinayətlər üzrə

Bir dəfə Almaniyada bankda işləyən bir haker ələ keçmişdi. O, bankın əmanətçilərinə faiz dərəcələri hesablandıqdan sonra alınan pfenniq (dövriyyəyə avro daxil olana qədər istifadə olunan markanın yüzdə birinə bərabər alman pulu) hissələrini öz hesabına köçürən proqram tərtib etmişdi. Bu minvalla il ərzində onun hesabında olduqca böyük məbləğdə vəsait toplanmışdı.

komitəsi bu problemi əbəs yerə terrorizm və narkotiklərlə bir cərgəyə qoymayıb.

Görəsən inkişaf etmiş ölkələr XXI əsrin bu bəlasını dəf edə biləcəklərmi?

ABŞ prezidentinin administrasiyası 2000-ci ildə kiber-cinayətkarlıqla mübarizə aparmaq məqsədilə milli plan təklif etdi: İnformasiya infrastrukturunu müdafiə etmək üçün yeni institut yaradıldı və bu sahədə olan elmi tədqiqatların maliyyələşdirilməsinə 600 milyon dollardan çox pul ayrıldı. Həmin il İnternetdə cinayətkarlığın qarşısını almaq

QƏSDƏN ZİYANVERİCİLİK

Kompüter cinayətkarlığı dövlət strukturları və şirkətlərə bəzən aradan qaldırılması qeyri–mümkün olan ziyan vurur. İşdən qovulan və ya acıqlı olan sistem inzibatçıları və proqramçılar professional casuslardan daha qorxulu olurlar.

35 yaşlı kompüter operatoru onu işlə təmin etmiş şəxsi yarım milyon funt sterlinqdən məhrum etdi. O, şəraitdən və təyin olunmuş məbləğdən razı qalmadığı üçün şirkətin mərkəzi serverindən bir neçə kabeli açır. Bundan sonra kompüter sistemi periodik olaraq sıradan çıxır və ancaq dəvət olunmuş ixtisaslı mütəxəssis diversiyanı aşkar edə bilir. Günahkarın vəkili onun işləməkdən yorulduğunu sübut etmək istəsə də, məhkəmə ziyankarı həbsxana ilə cəzalandırdı.

Rəhbərlikdən narazı əməkdaşların bədniyyətli hərəkəti nəticəsində şirkətin dilerlərinin hesabları barədə məxfi informasiya, bank hesabı və kredit kartlarının nömrələri İnternetdə açılmış bir sayta düşür. Bu məlumatlar bir neçə saatdan sonra bağlanmışdı. Kim bilir, neçə cinayətkar bu informasiyadan istifadə etməyə macal tapdı.

Çox vaxt nəinki ayrı-ayrı insanlar, hətta şirkətlər belə bu hərəkətlərə yol verirlər. Bu qeyri-sağlam rəqabətə gətirib çıxara bilər.



2000-ci ildə Avropa İttifaqı Microsoft şirkətini Windows 2000 seriyasından olan əməliyyat sistemlərini ancaq Microsoft-un öz məhsulları ilə dayanıqlı işləyəcək şəkildə hazırlamaqda ittiham etdi. Bu isə, həm digər şirkətlərin marağına xələl gətirir (Microsoft illik gəlirlərinin dörddə birini Avropa bazarlarından qazanır), həm də istifadəçiləri bir şirkətin məhsullarını almağa məcbur etməklə onlara ziyan vururdu.



SƏNAYE CASUSLUĞU

XXI yüzillikdə Avropa Parlamenti ABŞ, Böyük Britaniya, Avstraliya, Kanada və Yeni Zellandiyanın xüsusi xidmət orqanları tərəfindən idarə olunan ECHELON Amerika peykinə yenidən diqqət yetirdi. Bu peyk bütün dünyada telefon, faks, elektron poçt siqnallarını tutmaq iqtidarındadır, O, ABŞ-ı və müttəfiqlərini hədələyən terror və digər təhlükələrə qarşı mübarizə aparmaq üçün nəzərdə tutulmuşdur.

Lakin şübhə var ki, peyk həm də vahid Avropa ticarətinə qarşı ABŞ xeyrinə kommersiya casusluğu məqsədilə istifadə edilir. Avropa Parlamenti bunun nə dərəcədə doğru olmasını aydınlaşdırmaq niyyətindədir.

Daha bir misal. Johnson & Johnson şirkətinin şəkər xəstələri üçün ərzaq istehsal edən şöbəsi sənaye casuslarını təşviq etməkdə günahlandırıldı. Rəqib şirkətlər barədə ən çox məlumat toplayan əməkdaşlara "Inspector Clouseau" və "Columbo" mükafatlarının təqdim edilməsi buna əsas kimi göstərilmişdi.



məqsədilə FTB işçilərinin sayı kompüter mütəxəssisləri və hüquqşünaslardan ibarət olmaqla 150 nəfər artırıldı. Bu xərcləri ödəmək üçün konqres FTB-yə 38 milyon dollar ayırdı.

FTB elektron poçtda kriminal və ya terror xarakterli informasiyaların mövcudluğunu üzə çıxaran müasir avtomatlaşdırılmış sistemdən istifadə edir. Bu sistem yüzə yaxın cinayətin aşkar olunub açılmasına kömək etmişdir.

Böyük Britaniya hökuməti bildirmişdir ki, Kriminal cinayətlər üzrə milli xidmət orqanı fırıldaqçılıq, çirkli pulların yuyulması, qeyri-qanuni qumar oyunları, pornoqrafiya kimi internet-cinayətlərlə mübarizə aparmaq üçün xüsusi komanda yaratmalıdır.

Kompüter cinayətkarlığının nə milliyyəti, nə də sərhəddi olmadığına görə Avropa Komissiyası bu işə İnterpolun cəlb olunmasını təklif edir.

KOMPÜTER PİRATLARI

Məlum olduğu kimi, proqram təminatını iki yolla əldə etmək olar.

Birincisi – istehsalçı şirkətin nümayəndəsindən lisenziyalı versiya almaq və texniki dəstək, yenilənmələr kimi bütün əlavə imkanlardan istifadə etmək. Bundan əlavə, proqramın qanuni nüsxəsinə malik olduqda istifadəçi məmulatın keyfiyyətinə də əmin olur.

İkinci üsul – proqram diskləri satılan ən yaxın bazara gedərək hərəsi iki dollar olmaqla bu disklərdən istədiyiniz sayda almaq.

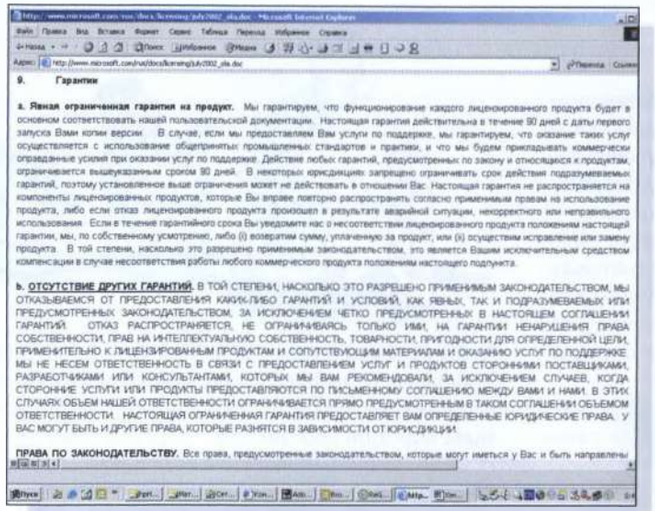
Hər iki üsulun öz üstünlükləri var. Bu üstünlük birinci halda proqramların istehsalçısına dəstək, ikinci halda isə ucuz qiymətə başa gəlməsilə izah olunur.

Proqramın leqal surətinin üstünlüklərindən çox danışmaq olar, lakin burada bəzi məqamlara diqqət yetirmək lazımdır.

Hər şeyə konkret istifadəçi nöqteyi-nəzərindən baxaq.

Tutaq ki, kimsə içində proqramların yazıldığı bir və ya bir neçə disk, təlimatlar, digər proqramların reklamı və s. yerləşən "qutucuğu" evə gətirdi. O, əmindir ki, xüsusilə də proqram baha olduqda, satıcı məhsulun işlək olmasına zəmanət verir. Ancaq diski kompüterə daxil etmək və yükləmə proqramını başlatmaq qalır...

Alıcının ilk gördüyü şey lisenziya razılaşmasıdır. Bu razılaşmada proqramın "As is" – "necə var elə" prinsipilə bütün səhvlər və çatışmazlıqlarla birlikdə yayıldığı qeyd olunur. Yəni, əgər proqramın işinin nəticəsində istifadəçinin



Microsoft şirkətinin proqram təminatının lisenziyasından çıxarış (ekrandan alınan foto).

verilənləri itərsə və ya nə isə korlanarsa, dəyən ziyanı məhkəmə və ya digər üsulla bərpa etmək çox çətindir. Əla! Bəs onda bu qədər pul vermək nəyə lazımdır?

Tutaq ki, istifadəçi ümid edir ki, müştəri xidməti ona kömək əlini uzadacaq.

Budur, artıq proqram kompüterə yüklənmişdir. Ancaq məlum olur ki, nə isə düz yerinə yetirilmir. Müştəri xidmətinə müraciət etməyin əsl vaxtıdır... Oradan isə cavab verirlər (növbəti variantlardan birini seçin):

• "Filan Service Pack-ı (yamağı) yükləyin". Yaxşı ki, telefonla da olsa İnternetə qoşulmaq imkanı var. Əlbəttə, üç saat ərzində bu "yamağın" yüklənməsi bir şey deyil – telefon uzun müddət məşğul olduğuna görə valideynlər, bacı və ya qardaş azacıq narazı qalacaq. Əgər rabitə pis olarsa, buna daha iki saat, əgər fayl səhvlə "gələrsə" daha üç saat və s. əlavə edək.

• "Sizin "dəmir" bizim proqrama uyğun deyil". Məsələn, enerji bloku. Bəlkə də kompüterin qapağı. Yaxud siçan altlığı, nə isə, lazım olan rəngdə deyil. Qəribədir ki, satıcı bu barədə heç bir məlumat verməmişdi və qutuda da heç nə yazılmamışdı... Qutu isə artıq açılıb və onu geri qaytarmaq mümkün deyil.

Əlbəttə, şirkətin nümayəndələrinə müraciət etmək olar, lakin onları istifadəçinin şəxsi problemləri maraqlandırmadığından onların köməyinə, demək olar ki, ümid yoxdur.

• "Bizim gözəl proqramımızın yeni versiyasını gözləyin". Nə qədər gözləmək olar? Və yenilənmə üçün nə qədər ödəmək lazımdır? Onların gözəl proqramının yeni versiyasının lazımi şəkildə işləməsinə zəmanət varmı?

Qanuni nüsxələrin əsas "silahı" – onların istehsalçı tərəfindən dəstəklənməsidir (bu istehsalçıların özləri istəyirlər ki, hamı belə fikirləşsin). Lakin bəzən reallıq bundan istifadə etməyə imkan vermir, ona görə ki:

• sadəcə olaraq o, yoxdur, belə ki, firmanın nümayəndəliyi ancaq məmulatı satır, onun dəstəklənməsilə isə haradasa uzaq Dublində yerləşən hansısa Avropa bürosu məşğul olur;

• xidmət pulludur. Daha dəqiq desək, əvvəlcə o, əlbəttə ki, ödənişsizdir, sonra isə hər şey ola bilər. Yəni müştəri xidmətinə ilk iki zəng pulsuz olacaq, sonrakı zənglərə isə böyük məbləğ tələb olunur.

Belə çıxır ki, proqram təminatının hər yerdə reklam olunan texniki dəstəklənməsi bəzən ancaq bir əfsanə və alıcını cəlb etmək üçün bir fənddir.

İddia olunur ki, proqramların qanuni şəkildə alınması proqram təminatı sənayesinə yardım edir. Bu belədirmi?

Təsəvvür edək ki, proqram sındıranlar, piratlar, qeyri-qanuni nüsxələri yayanların heç biri yoxdur. Çox gözəl! İndi fərz edək ki, alıcı 800 dollara kompüter alır və proqram almağa onun pulu qalmır. Daha doğrusu, bir qədər pulu qalır, lakin məlum olur ki, kompüterdə normal işləmək imkanı verən proqramlar çox bahadır.

Çox adam kompüter almaq arzusundadır və bütün hallarda məlum olur ki, əgər ödənişsiz pirat surətləri olmasaydı, onda dünyanın Rusiya kimi ağır

keçmişi və mürəkkəb bu günü olan bir çox ölkələrində hal-hazırda yaşanan informasiya bumu baş tutmazdı. Hadisələr yəqin ki, tamamilə başqa cür inkişaf edərdi. Kompüter adi müasir insanın həyatının bir hissəsinə çevrilməzdi, ancaq mütəxəssislər üçün alət olaraq qalardı. Qanuni proqram təminatı satıcıları isə, əlbəttə ki, başqa şeylər satardılar. Məsələn, tozsoran və ya santexnika.

Qəribədir ki, Rusiya proqramların pirat surətlərinin sayına görə heç də birinci yerdə deyil. Bu sahədə Çin Xalq Respublikası liderdir (bazarın 70%-i).

Qapalı dairə alınır. Şirkət məhsulunun potensial alıcısı başlanğıc mərhələdə proqramdan əsasən piratların köməyilə istifadə edir, iri şirkətlər buna mane olmur. İstehsalçıya dəyən ziyanın səbəbkarları da piratlardır.

Qanuni surətlərin satılması üçün əlverişli şəraiti yaratmaqla bu problemi həll etmək olar. Belə ki, əgər lisenziyalaşmış proqramın qiyməti 120 dollar deyil, 20 dollar olarsa, onda bəlkə də piratlara bunları "əl altdan" satmaq sərf etməzdi. Hazırda proqramların ucuzlaşması işini elə piratlar yerinə yetirirlər. İri firmalar isə informasiya bumu yaratmaq üçün proqram təminatının qiymətini aşağı salmaq əvəzinə, piratlarla mübarizə üsulları barəsində fikirləşirlər.

Kompüter piratlığı məsələsi kütləvi informasiya vasitələrində çoxdan – təxminən kompüterin evlərdə peyda olduğu və daşıyıcıların kifayət qədər ucuzlaşdığı vaxtlardan, yəni 80-ci illərin əvvəllərindən qaldırılmışdı. Uşaqlar üçün "Haker" jurnalından yeni nəsil üçün "Göyərçin"in oxşarı) tutmuş intellektual elita üçün nəzərdə tutulmuş "Kompüter" jurnalına qədər bir çox nəşrlərdə bu mövzuya toxunulub. Bəzi nəşrlərdə piratlar hakerlər haqqında kinolarda göstərilən igid gənclər, bəzilərində isə kainatın bəlası kimi təsvir



olunurlar. Lakin hər şey bir yana, onlar da bizim kimi insanlardır. Pirat fəaliyyəti bütöv bir dünyadır. Öz qaydaqanunları olan bir dünya. Bu dünyanın sakinlərinin özünəməxsus həyat tərzi var, onlar da doğulur və ölürlər, reallıqla üzləşir və öz sənətlərilə məşğul olmağa davam edirlər.

Rusiya bu işdən kənarda qalmır. Burada kompüter piratlığı fərqli formada fəaliyyət göstərir. Belə ki, qeyri-leqal proqram təminatının satışına nəhəng qüvvələr cəlb edilmişdir. Kədərləndirici haldır ki, onların fəaliyyəti müxtəlif səviyyələrdə kriminal strukturların təsiri



PİRAT LÜĞƏTİ

Warez. Software sözündən yaranmışdır. Bütün pirat "həyatı" bunun ətrafında gedir. Qeyri-qanuni üsulla ələ keçirilib sonradan sındırılaraq hər hansı şəkildə dəyişiklik və ya əlavələr edilmiş istənilən kommersiya xarakterli, qapalı, çətin əldə olunan proqram təminatı belə adlandırılır.
Scene. Səhnə. Bu kompüter andəqraundu ilə bu və ya digər şəkildə əlaqəsi olan bütün adamlar üçün sehrli bir sözdür. Warez-scene, cracking-scene, demo-scene və s. mövcuddur. Səhnəni adamların, hadisələrin, kompüter və proqramların hər hansı qarşılıqlı əlaqəsi kimi xarakterizə etmək olar. Səhnə bu qəbildən olan insanların cəlb olunaraq fəaliyyət göstərdiyi yerdir. Bir sözlə, səhnə – səhnədir.
Crack. Lazım olan effekti almaq üçün digər proqramda dəyişiklik aparan proqramdır.
Key, Keygen. Açar. Bu unikal simvollar ardıcıllığıdır. Bu simvollar ardıcıllığı hər hansı proqram təminatından istifadə etməyə və ya tam həcmdə istifadə etməyə imkan verir. Keygen – açarları generasiya etmək üçün proqramdır. Çox əlverişli bir şeydir.
NFO. Qrupun hər hansı məhsulunun təsvirindən ibarət olan müəyyən qayda ilə tərtib olunmuş fayldır.
Release. Reliz. Qrupun ümumi işinin son məhsuludur. Əvvəlcə qapalı, indi isə hamı üçün açıq olan hər hansı proqram təminatıdır.



altına düşür, axı kompüter andəqraundu (*ing.* underground) ilə əlaqəli adamlar öz mədəniyyəti və ideologiyası olan bütöv bir cəmiyyətdir.

Bəs normal vəziyyətdə bəzən külli miqdarda pulla alınan proqramların sındırılmış və ya ödənişsiz açarla olan versiyaları necə meydana gəlir? Bu halda xeyriyyəçi rolunda, piratların barışmaz düşmənlərinin sözlərilə desək, pirat qrupları və ya qruplaşmaları çıxış edirlər.

Qrup adətən gənc və istedadlı adamlardan ibarət olan bir qeyri-formal cəmiyyətdir. Bəs bura kimlər daxildir?

Organiser (məhz belə yazılır, Organizer yazılmır). Qrup bu adamdan başlayır. O, qalan üzvləri tapır, qrupun məqsəd və vəzifələrini müəyyən edir. Menecer və ya rəisin bənzəri olan bu adam qrupun üzvləri üçün "döğma ata" rolunu oynayır, onsuz yoldaşlar arasında razılıq olmur və iş getmir.

Cracker. Qrupda vəzifəsinə görə ikinci adamdır. O, reverse engineering (*ing.* "tərs layihələndirmə") texnologiyasını bilir və kommersiya proqramlarının mühafizəsini ləğv edə və ya dəyişə bilir.

Supplier. Qrupu materialla – mühafizəsinin qırılması mümkün olan proqramların yeni versiyaları ilə, yeni oyunlarla və s. təmin edir. Supplier proqram təminatı satıcısı kimi işləyə və ya istehsalçı, yaxud testerlərlə əlaqələri ola bilər.

Courier. Kuryer, çapar. Səhnədə qrupun imicinə cavab verənlərdən biridir. O, sonradan yayılması üçün relizin vaxtlı-vaxtında serverlərdə olmasına cavabdehdir.

Trader. Digər qruplar və ya şəxslərlə yenicə çıxmış "Soft"un alverini aparır.

Eyni funksiyaları bir neçə nəfər yerinə yetirə bilər və ya bir adamın bir neçə funksiyası ola bilər. Bunlar qrupun ölçülərindən asılıdır.

Qrupda qarşılıqlı münasibətlər adətən qeyri-formaldır. Elə hallar olur ki, qrupun üzvləri bir-birini şəxsən tanımırlar, ancaq elektron poçt və ya IRC (Internet Relay Chat) vasitəsilə ünsiyyətdə olurlar. Çox vaxt qrupun özünün xüsusi IRC kanalı olur. Bu cür ünsiyyət heç də işin ümumi məhsuldarlığını aşağı salmır.

Reliz necə meydana gəlir? Təşkilatçı fəaliyyət istiqamətini seçir (oyunların, multimedia proqramları və ya, məsələn, ofis proqramlarının relizi). Misal üçün,

CLASS qrupu ancaq oyunlar üzrə, RADIUM qrupu isə musiqi proqramları üzrə ixtisaslaşır. Supplier proqramın yeni versiyasını əldə edir, cracker isə mühafizəni aradan qaldırır. Sonra proqram testləşdirilir, müəyyən qayda ilə paketləşdirilir, məsələn, İnternetdən yükləmək rahat olsun deyə müəyyən hissələri arxivləşdirmək lazım gəlir. Mütləq NFO-fayl yazılır. Burada bəzən qrupun ASCII Art (kompüterin populyarlaşması nəticəsində əmələ gələn yeni incəsənət növü – mətnin simvollarının köməyilə EHM-in ekranında şəkil çəkilməsi) üslubunda verilmiş loqotipi, həmçinin proqramın qısa təsviri, relizi hazırlayanların təxəllüsləri, yükləmə təlimatları, qrup haqqında informasiya və müxtəlif yeniliklər olur. Çox vaxt reliz demo janrında kiçik bir filmlə müşayiət olunur – canlı musiqi və effektlər, qrup haqqında bəzi məlumatlar, salamlar və relizin adı. Bundan sonra treyderlər və kuryerlər relizi hamı üçün açıq ümumi serverlərdə yayırlar.

Bir dəqiqə! Bəs mənfəət harada qaldı? Axı burada bir gəlir yoxdur. Bəs komanda nə əldə edir? Yəqin ki, birinci növbədə mənəvi məmnunluq hissi. Səhnədə ad-san və şöhrət. Daha mürəkkəb müdafiənin ləğv edilməsi (daha dəqiq, onun düzgün şəkildə ləğv edilməsi) qrupa daha çox xal qazandırır. Bu isə o deməkdir ki, qrupun relizləri olduqca populyar olacaq və o, yaxşı ad qazanacaq.

Qrupun diqqətini hansı proqramlar cəlb edir? Populyar olanlar. Məhz populyar olanlar. Onların əhəmiyyəti pulla deyil, iş üçün yararlı olması ilə müəyyən edilir.

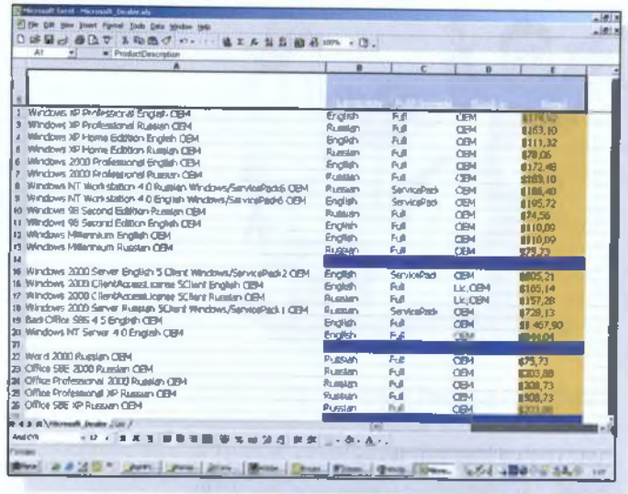
Piratların bu və ya digər məhsula verdiyi qiymət çox şeyə dəyər. Piratlar bir qayda olaraq kompüter texnologiyaları sahəsində peşəkar olurlar və onların fikri ən mötəbər rəylərdən biri hesab olunur. Əgər proqram piratların əlinə düşübsə, deməli onun bir dəyəri var.



Bundan başqa, əsl yaxşı proqram NFO faylında belə bir sətirdə qiymətini alacaq: If you like this programm, buy it! – "Əgər bu proqram xoşuna gəlirsə, onda onu al!" Kompüter piratlığı tarixində hətta belə bir görünməmiş hal olmuşdur: LAXITY qrupu Bleem Sony Playstation emulyatorunun yeni versiyalarını sındırmaqdan imtina etmişdir! Qrup bu proqramı çox əla qiymətəndirərək müəllifin qonorardan məhrum edilməsini düzgün hesab etməmişdir. Bunun özündə bir məna var. Ən yaxşılar gəlir əldə etməlidir. Autsayderlər isə daha yaxşı olmaq haqqında düşünməlidirlər.

Nə üçün piratlar bəzən istifadəçilərə proqramın orijinalını almağı məsləhət görürlər? Birincisi, hesab edirlər ki, müəllif həvəsləndirilməyə layiqdir. İkincisi, başa düşürlər ki, müəllif ona çatası pulu almasa, bəlkə də proqramın yeni versiyaları üzərində apardığı işi tam dayandırar. Ən qəribəsi odur ki, burada həm piratlıq, həm də alicənablıq yanaşı mövcud olub fəaliyyət göstərir.

Qərb piratları Rusiya piratlarından nə ilə fərqlənirlər? Aşağıdakı səbəblərə görə. Birincilərinin güclü ənənələri var – onlar proqramları elə sındırmaq xətrinə sındırırlar. Qərbdə həyat tərzi yüksək olduğuna görə istifadəçilər özlərinə lazım olan proqram təminatını ala bilərlər. Rusiya səhnəsi, daha dəqiq desək, onun ənənələri hələ gəncdir və proqramlar məhz onları sonradan pulsuz



Microsoft şirkətinin proqram təminatı üçün qiymətlər cədvəlindən çıxarış. (ekrandan alınan foto).

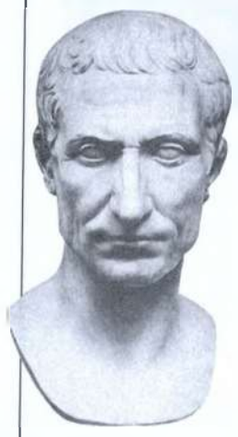
istifadə etmək üçün sındırılır. Lakin nəticələr eynidir – proqram təminatının mühafizəsində etibarlılığın artması, yaxşı proqramların pulsuz reklamı, kütləvi istifadəçilərdə zövqün formalaşması.

Piratlığa mütləq şər kimi baxmaq lazım deyil. Medalın iki tərəfi var. İri şirkətlər Rusiya bazarını real qiymətləndirməlidirlər. Məsələn, oyun istehsalçıları proqramların xüsusi, daha ucuz versiyalarını Rusiyada satmaqla artıq bunu edirlər. Ümid etmək olar ki, digər proqram istehsalçıları da onlardan nümunə götürərək üzlərini satış üzrə menecerlərin hesabatlarında adı çəkilən "iri müştərilərə" deyil, sadə istifadəçilərə çevirərlər.

İNFORMASIYANIN ŞİFRLƏNMƏSİ

İnformasiyanın gizli ötürülməsi və saxlanması üçün etibarlı üsulların axtarılmasının kökləri uzaq keçmişə təsadüf edir. Bəşəriyyət tarixində mətnlərin şifrlənməsi sevimli oyunlardan biri olmuşdur. Şifrlər hərbi məqsədlərdə, dostlar arasından gizli məlumatların ötürülməsində, biliklərin gizli saxlanılmasında və bu kimi yüzlərlə hallarda istifadə edilmişdir.

İlk şifrlənmiş məlumat hələ qədim Misirdə istifadə olunmuşdur. Şifrləmə üsulu çox sadə idi və hazırda *sadə əvəzetmə ilə şifrləmə* adlanır. Şifrlənən məlumatda ilkin informasiyanın hər bir heroqlifi başqa heroqliflə əvəz olunurdu. Heroqliflərin uyğunluğu birmənalı idi, belə ki, şifrlənmiş məlumatı oxumaq üçün tərs əvəzetmənin aparılması tələb olunurdu.



Qay Yuli Sezar.

SEZAR ŞİFRİ

Ən sadə şifrə misal olaraq sadə əvəzetmə şifrlər qrupuna aid olan Sezar şifrini göstərmək olar. Qədim Roma tarixçisi Svetoninin dediyinə görə Qay Yuli Sezar (e.ə. 102 və ya 100–44) ondan gizli məktublaşma üçün istifadə edirdi. Sezar şifrində ilkin məlumatın hər bir hərfi əlifbada irəli mövqe sayı ilə təsbit edilərək hərəkət edir, lazım gəldikdə dövrü olaraq əlifbanın başlanğıcına keçilirdi. Sezarın özü üç mövqe üzərində hərəkətdən istifadə edirdi. Bu halda

ROMAYA GERİ QAYIDIN

məlumatı aşağıdakı kimi şifrlənir:

TRÖÇBÇ XGTQ NÇBKFKP

Burada R hərfi ondan üç mövqe sağda yerləşən T kimi, O hərfi R kimi və s. şifrlənmişdi. Bu üsulla əlifbanın axırıncı Z hərfi C kimi şifrlənir. Bu məlumatı açmaq üçün hər bir hərfi əlifbada üç mövqe geriyə sürüşdürmək lazımdır.

Sezar şifri sürüşmənin kəmiyyətilə müəyyən olunur. Müxtəlif sürüşmələrin sayı əlifbada olan hərflərin sayından bir vahid az olduğu üçün Sezar şifrini açmaq o qədər də çətin deyil. Azərbaycan əlifbasında 1-dən 32-yə qədər bütün mümkün olan sürüşmələrin qiymətini götürmək kifayətdir. Anlaşıqlı bir mətn alındıqda məlumatın şifri açılır.

SADƏ ƏVƏZETMƏ İLƏ ŞİFRLƏMƏ

Sadə əvəzetmə üsulu bir qədər mürəkkəbdir. Burada ilkin məlumatın hər bir hərfi kodlaşdırma cədvəlində verilən işarələr vasitəsilə kodlaşdırılır.

Cədvəlin yuxarı sətrində əlifbanın bütün hərfləri yerləşir. Cədvəlin aşağı sətrində hərflərin istənilən qaydada yerdəyişməsi aparılır. Hər bir belə yerdəyişmə müəyyən bir şifrə uyğun gəlir.

Sadə şifrləmə üsulu Artur Konan Doylun (Arthur Conan Doyle) "Rəqs edən adamcığazlar" hekayəsində təsvir olunmuşdur: "Burada məlumatın hər bir hərfi rəqs edən adamcığazların müəyyən fiquru şəklində təsvir olunur. İşarələri sadə uşaq rəsmləri kimi qələmə verməklə, bu sistemi kəşf edənlərin məqsədi yəqin ki, onların yazı olduğunu gizlətmək olmuşdur. Lakin bu işarələrin hərflərə uyğun olduğunu başa düşən hər kəs çətinlik çəkmədən şifrlərin açılması üçün adi qaydalardan istifadə etməklə onları oxuya bilər". Bəs Şerlok Holms hansı qaydaları nəzərdə tuturdu?

"Rəqs edən adamcığazlar" şifrini ixtira edən çikaqolu banditlər sözün əvvəlində və axırında duran uşaqları bayraqla təchiz etməklə şifrlərin oxunması məsələsini xeyli asanlaşdırmışdılar. Beləliklə, şifrlənmiş məlumatın sözlərə bölünməsi o dəqiqə anlaşıldı. Həm də istənilən dildə bir və ya iki hərfli sözlərə çox az təsadüf olunduğuna görə onları seçmək və düzgün hərfi tapmaq olar. Tapılmış hərfi digər sözlərdə yerinə qoymaqla düzgün olmayan variantları tez kənar etmək olar. Hər bir müəyyən olunmuş söz mətndəki uyğun hərfləri aydınlaşdıraraq bütün şifrin açılmasını sürətlə yaxınlaşdırır. Şerlok Holmsun əlavə informasiyaya malik olması şifrin tapılmasını asanlaşdırdı. O bilirdi ki, məktublar İlsi adlı bir qadına göndərilir. Məktublar isə çox zaman kimə göndərilirsə onun adı ilə də başlayır. Beləliklə, o ən qısa zamanda üç hərfin şifrini tapa bildi. Sonra Şerlok Holms "gəl" sözünü müəyyən etdi ki, bununla da işin qalan hissəsi xüsusi zəhmət tələb etmədi.



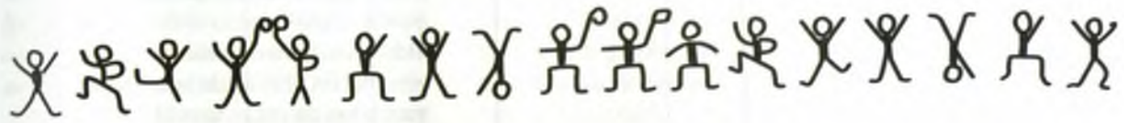
"Şerlok Holms və doktor Vatsonun macəraları" kinofilmindən kadr.

MƏTNİN TEZLİKLİ ANALİZİ

"Rəqs edən adamcığazlar" şifrini açmaq üçün Serlok Holmsun tətbiq etdiyi üsul kompüter üçün o qədər də müvafiq deyil. Ehtimal nəzəriyyəsinə əsaslanan və mətndə hər bir hərfin rast gəldiyi başqa bir səmərəli üsul vardır. Bu üsul *tezlikli analiz üsulu* adlanır. Bu üsulun yeganə çatışmayan cəhəti şifrlənmiş məlumatda mətnin iri həcmli olmasıdır.

Böyük həcmli müxtəlif mətnləri analiz etməklə əlifbanın hər bir hərfinin istənilən mətndə rast gəlmə tezliyini təqribi olaraq hesablamaq mümkündür. Əgər sözlərin arasındakı aralıqlar nəzərə alınarsa, onda məlum olur ki, aralıq 0,1558 ehtimalı ilə ən çox rast gəlinən simvoldur (yəni 1000 işarədən ibarət olan mətndə orta hesabla 155 aralıq iştirak edir). Şifrlənmiş mətnlərdə sözlər arasında ayırıcıların olması onun tapılmasını asanlaşdırır. Ona görə də şifrləmə zamanı aralıq nəzərə alınmır.

Əgər aralıq nəzərə alınmazsa Azərbaycan əlifbasının hərflərinin tezlikləri əsasən aşağıdakı kimi olar (göstərilənlər ümumi həcmi min simvoldan ibarət olan bir neçə bədii mətnin analizinin nəticəsidir):



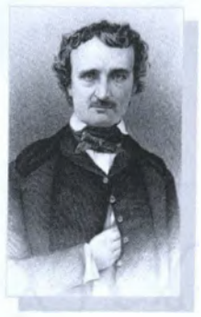
A 0,0910	Ə 0,0788	İ 0,0708	N 0,0599
D 0,0538	R 0,0523	L 0,0448	I 0,0390
M 0,0355	B 0,0277	Y 0,0261	S 0,0237
U 0,0209	Ü 0,0200	O 0,0190	T 0,0188
K 0,0183	E 0,0180	Ş 0,0160	Z 0,0148
Q 0,0129	G 0,0118	H 0,0103	Ğ 0,0101
X 0,0091	V 0,0088	Ç 0,0087	Ö 0,0084
C 0,0077	F 0,0037	P 0,0035	J 0,0001

Beləliklə, istənilən mətndə hər min hərfdən orta hesabla 91 hərf – A, 78 hərf – Ə, 70 hərf – İ, 59 hərf – N və s. olacaqdır. Ən az rast gəlinən hərf – P və J hərfləridir: hər min hərfdən orta hesabla 35 hərf – P və cəmi 1 hərf – J olacaq.

Mətnin digər əhəmiyyətli xarakteristikası isə mətndə müxtəlif iki hərfli birləşmələrin işlənmə tezliyidir. Kriptoqrafiyada belə cütlər *biqram* və ya *diqram* (diaqram ilə qarışıq salmayın. "Di" – *yun*. "ikili" deməkdir) adlanır. Növbəti siyahıda Azərbaycan dilində yazılmış mətnlərdə ən çox rast gəlinən biqramlar ehtimallarının azalması sırası ilə göstərilmişdir:

Dİ İR AN AR DƏ LƏ DA İN
MƏ ƏR Kİ IN RD Bİ DI ND

Əvəzetmə qrupundan olan istənilən şifrlərin açılması üsulu mətnin statistik xarakteristikalarına əsaslanır. Şifrlənmiş məlumatda bütün hərflərin və biqramların tezliklərini hesablamaqla ən böyük tezliyə malik olan hərfin əlifbada ən çox rast gəlinən hərf olduğunu fərz etmək olar. Oxşar olaraq, şifrlənmiş mətndə ən böyük tezliyə malik olan biqramın şifrlənməmiş mətndə tez-tez rast gəlinən biqrama uyğun olması nəzərdə tutulur. Əlbəttə, şifrlənmiş mətnin statistik analizində kompüterin köməyi əvəzsizdir, lakin prinsipcə bu hesablamaları kompütersiz də aparmaq olar.



Edqar Allan Po.

Məşhur Amerika yazıçısı Edqar Allan Po (Edgar Allan Poe) da kriptoqrafiya ilə maraqlanmışdır. Onun "Qızıl böcək" hekayəsinin süjeti gizli məlumatların açılması ətrafında qurulmuşdur. Edqar Po şifrlərin və digər gizli məlumatların açılmasında ancaq məntiq və əqli tətbiq etməyin yetərli olduğunu düşünürdü. O, 1893-cü ildə "Alexander's Weekly Messenger" jurnalında oxuculara müraciət edərək ona göndərdikləri istənilən əvəzetmə üsulundan istifadə edən kriptoqramı açа biləcəyini iddia etdi.

"Anlamaq gücü olmayan insan təfəkkürünün açmağa qadir olmadığı bir şifri tərtib etməyə də gücü çatmaz". Bu sözlər Edqar Poya məxsusdur. Lakin müasir kriptoqrafiya məşhur yazıçının bu fikrini təkzib edir. Açıq açarlı müasir şifrləri ("Müasir kriptoqrafiya" məqaləsinə bax) ən güclü kompüterlərin köməyilə də açmaq olmur.

VİJENER ŞİFRİ

XVI əsrin axırlarında Fransa kralı III Henrinin saray əyanı Blez de Vijener (Blaise de Vigener) maraqlı bir şifrləmə üsulu təklif etdi. Bəzi hallarda bu şifr *mətni gizlətməklə şifrləmə* adlanır. Şifrləmə üçün gizli söz və ya ifadədən istifadə olunur. Bu gizli sözü ilkin mətnin yuxarısında yazaraq məlumat qurtarana qədər təkrarlamaq lazımdır. İlkin mətnin hər bir hərfi əlifbada ondan bir neçə mövqe sonra yerləşən hərflə əvəz edilir. Sürüşmənin sayını isə gizli açar sözünün ilkin mətndəki hərfin yuxarısında yerləşən hərfi müəyyən edir. A hərfi üçün sürüşmə ümumiyyətlə yoxdur, B hərfi bir mövqe, C hərfi iki mövqe irəli və s. sürüşdürməyə uyğun gəlir. Axırıncı Z hərfi isə 31 mövqe sürüşdürməyə uyğundur (Azərbaycan əlifbasında 32 hərf olduğuna görə). Yəni sürüşmənin ölçüsü hərfin əlifbadakı sıra nömrəsindən bir çıxmaqla müəyyən olunur.

Misalda açar söz olaraq HOLMS sözü istifadə edilir. Tutaq ki, aşağıdakı məlumatı şifrləmək lazımdır:

TƏCİLİ OLARAQ GƏL

Bunun üçün açar söz şifrlənən ifadənin üzərində yazılır:

HOLMSH OLMSHO LMS
TƏCİLİ OLARAQ GƏL

İndi məlumatda hər bir hərfi üstündə duran açar sözün hərfinə uyğun olaraq əlifba üzrə irəliyə doğru sürüşdürmək lazımdır. Məsələn, H – hərfi əlifbada 11-ci hərfdir və 10 mövqe irəli sürüşməni təyin edir. Beləliklə, ilkin mətndəki T hərfinin əvəzinə şifrlənmiş məlumatda D hərfi yazılır. İlkin mətnin ikinci Ə hərfi açar sözün O hərfinə uyğun olaraq 20 mövqe irəli sürüşdürülərək T hərfilə əvəz edilir və s. Beləliklə, aşağıdakı şifrlənmiş mətn alınır:

DTNZĞR GCMKHD ŞSĞ

VİJENER ŞİFRİNİN AÇILMASI

Təxminən üç yüz il ərzində Vijener şifrinin açılması qeyri-mümkün hesab olunmuşdur. İlk dəfə Vijener şifrini açmaq üsulunu 1863-cü ildə Prussiya ordusunun mayoru Fridrix Kasitski təklif etmişdir. Bu üsul açar sözün uzunluğunun təyin edilməsinə əsaslanırdı. Açar sözün uzunluğu məlum olduqda, mətni bir neçə fraqmentə bölmək və onlara Sezar şifrini tətbiq etmək olar. Göstərilən misalda açar sözün ("HOLMS") uzunluğu beşdir. Bu o deməkdir ki, məlumatın birinci, altıncı, on birinci, on altıncı və s. hərfləri Sezar şifrinin eyni bir hərfilə kodlaşır. H hərfi əlifbada 11-ci hərf oduğuna görə 10 mövqe irəli sürüşməklə alınan hərf götürülür. Analoji olaraq, ilkin məlumatın ikinci, yeddinci, on ikinci və s. hərflərinə açar sözün ikinci O hərfinə uyğun olan 20 mövqe sürüşmə ilə alınan Sezar şifri tətbiq olunur. Bütün mətn beş fraqmentə bölünür. Hər bir fraqmentə tezlikli analiz üsulunu tətbiq etmək olar. Beləliklə, Vijener şifrinin tapılması 90% açar ifadənin uzunluğunun tapılmasına uyğunlaşdırılır. Əlbəttə, mətn çox böyük, açar söz isə ona nisbətən çox qısa olduqda bu üsul tətbiq oluna bilər.



Blez de Vijener.

Kasitski üsulu şifrlənmiş mətndə təkrarlanan biqramlar arasındakı məsafələrin təyin edilməsinə əsaslanır. İlkin mətndə eyni bir biqram açar sözün uzunluğu qədər olan məsafədə təkrarlanarsa, o, şifrlənmiş mətndə də həmin məsafədə təkrarlanacaqdır. Təkrarlanan biqramlar arasındakı məsafəni hesablayaraq və bu ədədin bütün bölənlərini taparaq açar sözün uzunluğuna namizəd olan ədədlər çoxluğunu almaq olar.

Tutaq ki, BARTO açar sözündən istifadə edərək aşağıdakı bəndi şifrləmək lazımdır:

Əvvəlcə aralıqları, durğu işarələrini kənara atmaqla və mövqeləri saymağın asan olması üçün alınan hərflər ardıcıllığını beşliklərə bölməklə şəri aşağıdakı kimi yazaq:

DƏRSƏ GEDƏN BİRUŞ AQÇIX DIBUZ
ÜSTƏQ OÇAQS ÜRÜŞD ÜBİRD ƏNBİR
ƏDƏYD İÜZÜS TƏYER Ə

Vijener şifrləmə üsulunu tətbiq etməklə alarıq:

EƏJMT ĞEUAF CİJÖİ BQTƏZ EİSÖN
VSLAD ÖÇRHI VRNNS VBDLS FNSFX
FDVSS JÜPPI UƏÖZX F

OAİTABNPXÖPMCAGMAZÇAFRÖƏ
ÜMATVUŞKQÖNŞİCDOOƏVTIJÇĞ
TIJPITEGNXEAPNXDRSEZCUNƏZ

Bu mətndə biqramlar müəyyənləşdirilir və onların yerləşməsində qanauyğunluq axtarılır. Məsələn, əgər hər hansı biqram 5 mövqedən bir təkrarlanarsa, güman edilir ki, açar sözün uzun-



Fridrix Kasitski.



luğu 5-ə bərabərdir. Uzunluğu 5 olan açar söz düz gəlmədikdə başqa biqram yoxlanılır.

Açarın uzunluğunun böyük ehtimalla beş olduğunu təyin edərək Vijener şifrinin açılmasında ən mühüm addımı atmış olduq.

Kasitskinin üsulunda müəyyən qədər bəxtin gətirməsi tələb olunur. İlkin mətndə biqram açar sözün uzunluğuna bölünən məsafədə təkrarlanmalıdır. Mətn iri həcmli deyilsə, bu proses baş verməyə bilər. Bundan əlavə, şifrlənmiş mətndə müəyyən məsafədə yerləşən, açar sözün uzunluğuna bölünməyən, təsadüfi təkrarlanan biqramlar ola bilər. "Requlyar" biqramlara nisbətən onların ehtimalı azdır və qısa mətnlərdə onlar maneçilik törədə bilər. 1920-ci ildə amerikalı Vilyam F.Fridman (William F.Friedman) açar sözün uzunluğunu müəyyən etmək üçün daha sadə və səmərəli üsul təklif etdi. Bu üsul klassik kriptoqrafiyanın ən parlaq nailiyyətlərindən biri hesab olunur.

Fridman üsulu "üst-üstə düşmə indeksi"nin hesablanmasına əsaslanır. Şifrlənmiş məlumat mətnin bir neçə mövqe dövri sürüşməsilə yenidən yazılır. Bu qayda ilə alınan məlumatın biri digərinin üzərində yazılır, sonra aşağı və yuxarı sətirlərdə üst-üstə düşən hərflərin sayı hesablanır. Sonra üst-üstə düşmələrin sayının məlumatın uzunluğuna olan nisbəti hesablanır. Bu nisbət üst-üstə düşmə indeksi adlanır.

Azərbaycan dilində olan mətnlər üçün üst-üstə düşmə indeksi orta hesabla 6%-dir. Ancaq Azərbaycan hərflərinin tamamilə təsadüfi ardıcıllığında üst-üstə düşmə indeksi xeyli az olacaq. Azərbaycan əlifbasında 32 hərf olduğuna görə üst-üstə düşmə ehtimalı 1/32 və ya 0,031, yəni 3%-dən bir qədər çox olur.



Vilyam F.Fridman.

Fridman üsulu məhz bu mülahizəyə əsaslanır. Şifrlənmiş mətn 2, 3, 4 və s. sürüşmə ilə yazılır və hər sürüşmədə üst-üstə düşmə indeksi hesablanır. Əgər bu indeks 0 və 5% arasında dəyişərsə, deməli sürüşmənin ölçüsü açar sözün uzunluğuna uyğun deyildir. Üst-üstə düşmə indeksi sıçrayışla 6% və daha çox artdıqda açar sözün uzunluğu tapılır.

Bizim misalda şifrlənmiş mətn aşağıdakı kimidir:

```
OAİTABNPXÖPMCAGMAZÇAFRÖƏ
ÜMATVUŞKQÖNŞİCDOOƏVTIJÇĞ
TIJPITEGNXEAPNXDRSEZCUNƏZ
```

Mətni dövri olaraq sürüşdürməklə üst-üstə düşmə indeksini hesablayıb aşağıdakı nəticəni alarıq:

Sürüşmə	Üst-üstə düşmələrin sayı	İndeks
2	0	0.000
3	5	0.068
4	2	0.027
5	8	0.110 – bu, açarın uzunluğudur!
6	1	0.014
7	1	0.014
8	2	0.027

Sürüşmənin qiyməti 5 olduqda üst-üstə düşmələrin sayı istənilən digər sürüşmələrdən daha çox olur. Açarın uzunluğu isə böyük ehtimalla 5-ə bərabərdir.

Sürüşmənin qiyməti açar sözün uzunluğuna bölünən bir ədəd olduqda üst-üstə düşmə indeksinin nə üçün birdən-birə artdığını başa düşmək çətin deyil. İstənilən əvəzetmə şifrində, xüsusilə də Sezar şifrində şifrlənmiş və ilkin mətnin üst-üstə düşmə indeksi eyni olur. Vijener üsulu ilə şifrlənmiş mətndə açar sözün uzunluğuna bərabər sürüşmə zamanı biz eyni bir Sezar şifrilə şifrlənmiş hərfləri müqayisə edirik. Beləliklə, şifrlənməmiş mətndə üst-üstə düşmə indeksi açar sözün uzunluğu qədər sürüşmə aparıldıqdan sonra alınan üst-üstə düşmə indeksilə eyni olur.

"ENİQMA" ALMAN ŞİFRLƏMƏ MAŞINI

Birinci dünya müharibəsi dövründə və sonralar da mexaniki və elektromexaniki şifrləmə maşınları geniş yayılmağa başladı. Onlardan biri – "Eniqma" adlı alman şifrləmə maşını praktiki kriptoqrafiyanın ən böyük nailiyyətlərindən hesab olunur.

"Eniqma" Almaniyada 1928-ci ildən başlayaraq, Hitlerin hakimiyyətdə olduğu dövrdə və İkinci dünya müharibəsi illərində istifadə olunurdu. 1932-ci ildən başlayaraq Varşavanın şifrləmə bürosunda "Eniqma"nın sirrini açmaq istiqamətində işlər aparılırdı. Qrupun rəhbəri Poznan Universitetinin riyaziyyat fakültəsinin məzunu gənc Polşa riyaziyyatçısı Marian Reşevski idi. Almaniyadan alınmış köhnə şifrləmə maşını qrupun ixtiyarına verilmişdi. Bu model o dövrün tələbinə cavab vermədiyi üçün xeyri də az oldu. Ona görə də "Eniqma" məsələsinin həlli üçün alimlərin işində əsas məqam riyaziyyatın tətbiq edilməsi idi (qruplar nəzəriyyəsi və yerdoyişmələr nəzəriyyəsi).

1933-cü ilin yanvarında maşının nəzəri iş prinsipi tapıldı ki, bu da sonradan onun real modelini hazırlamağa imkan verdi. Həmçinin elə keçirilmiş məlumatların əsasında şifr açarlarının bərpa olunması üsulu işlənib hazırlandı.

MÜASİR KRİPTOQRAFİYA

Kompüterlərin yaranması ilə kriptoqrafiya tamamilə dəyişdi. Şifrləmənin klassik üsulları kompüterlərin sınağından keçmədi və artıq bu üsullar istifadə olunmur.

Müasir kodlaşdırma üsullarının əsasını təşkil edən başlıca ideyalardan biri şifrləmədə klassik şifrlərdə olduğu kimi hərflərdən deyil, ikilik kodlardan istifadə olunmasıdır. Bu ancaq mətnləri deyil, həm də istənilən faylları (təsvirləri, kompüter proqramlarını və s.) şifrləməyə imkan verir.

Hər bayt 256 qiymətə malikdir, bu isə etibarlı şifrləmə üçün kifayət etmir. Ona görə də ayrı-ayrı baytlar deyil, bir neçə ardıcıl baytdan ibarət olan bloklar şifrlənir. Şifrləmə sxemlərinin çoxunda hər bir blok digərlərindən asılı olmadan şifrlənir. Blokun ölçüsü nə qədər böyük olarsa, şifr də bir o qədər etibarlı olar.

Bir qədər sonra, 1939-cu ildə "Eniqma" haqqında materiallar Fransaya və İngiltərəyə ötürüldü. İngilislər ən yeni alman maşınlarının konstruksiyası və kodlar sistemində aparılmış təkmilləşdirmələrin sirrini açdılar. Londondan 70 km məsafədə yerləşən Bletçli məntəqəsində bir qrup alim tərəfindən aparılan bu işdə məşhur riyaziyyatçı, xəyali "Türinq maşını"nın müəllifi kimi tanınmış Alan Türinq (Alan Mathison Turing) də iştirak edirdi. İş çox gizli saxlanılırdı və almanların heç ağlına gəlmirdi ki, onların bütün məxfi məlumatları artıq anti-Hitler koalisiyası üçün açıqdır. İkinci dünya müharibəsini tədqiq edən tarixçilərin çoxu belə qənaətə gəliblər ki, bu, faşist Almaniyasının süqutunu sürətləndirmiş və minlərlə insanın həyatını xilas etmişdir. "Eniqma"nın tarixçəsi müharibədən sonra da çox gizli saxlanılmış və ancaq 30 il sonra artıq hərbi sirr kimi qüvvədən düşdükdə nəşr olunmuşdur.



Marian Reşevski.

Həqiqətən də XX əsrin ikinci yarısında kriptoqrafiyada əsl inqilab baş verdi. Açıq açarla müasir kodlaşdırma sxemində nə kodlaşdırma maşını, nə də şifrləmənin açarı artıq məxfi deyil! Yəni məlumatı şifrləmək üçün heç bir sirri bilmək lazım deyil (gizli açar ancaq deşifrləmə üçün lazımdır) və şifri praktik olaraq sındırmaq mümkün deyil. Beləliklə, "Eniqma"nın sirlərinin açılmasına oxşar bir əhvalat bizim dövrdə çətin ki, baş verə bilər.

"ENİQMA"NIN QURULUŞU

"Eniqma" elektromexaniki şifrləmə maşınları qrupuna daxil idi. Onun konstruksiyası latın əlifbasının 26 hərfini dəyişdirən üç rotordan ibarət bir sistemə əsaslanırdı. Hər rotorun bir səthində 26 giriş kontaktı, digər səthində isə 26 çıxış kontaktı var idi. Hər bir rotorun daxilində giriş və çıxış kontraktlarını öz aralarında birləşdirən naqillər yerləşirdi. Birinci rotorun çıxış kontaktı ikinci rotorun giriş kontaktı ilə birləşdirilirdi. Operator klaviaturanın hər hansı hərfini basdıqda elektrik cərəyanı birinci rotorun bu hərfə uyğun olan kontaktına verilirdi. Cərəyan birinci rotordan keçərək hər hansı digər hərfə uyğun olan çıxış kontaktına düşürdü. Sonra cərəyan ardıcıl olaraq ikinci və üçüncü rotordan keçib tərpənməz reflektora (*lat.* "reflecto" – "əks etdirirəm", "geriyə göndərirəm") verilirdi. Reflektorun konstruksiyasında 26 kontakt cütlərə bölünmüş və hər bir cüt öz aralarında birləşdirilmişdi. Beləliklə, reflektor hər bir hərfi onunla bir cütdə yerləşən digər hərflə əvəz edirdi.



"Eniqma" şifrləmə maşını.

Reflektordan keçən cərəyan geriyə – rotorlar sisteminə verilirdi. O yenidən tərs ardıcıllıqla üç rotordan keçirdi. Nəticədə "Eniqma"nın işıq tablosunda şifrlənmiş hərfə uyğun olan 26 lampadan biri yanırdı.

"Eniqma"nın ən vacib xüsusiyyəti rotorların fırlanması idi. Birinci rotor hər şifrlənmiş hərfdən sonra bir mövqe dönürdü. Birinci rotor tam bir dövr vurandan, yəni 26 hərf çevirməsindən sonra ikinci rotor bir mövqe dönürdü. İkinci rotorun tam dövriyyəsindən sonra, yəni 26 x 26 = 676 hərf şifrləndikdən sonra isə üçüncü rotor bir mövqe dönürdü.

"Eniqma"nın reflektoru hesabına bir cütdə yerləşən hərflər hər addımda yerini dəyişirdi. Məsələn, N hərfi S-lə əvəz olunurdusa, rotorların eyni vəziyyətində S hərfi də N-lə əvəz olunurdu (cərəyan eyni naqildən, sadəcə əks istiqamətdə keçirdi). "Eniqma"nın özünəməxsus cəhəti bununla izah olunurdu: məlumatın şifrini açmaq üçün rotorların ilkin vəziyyətini bərpa edərək onu yenidən maşından keçirmək lazım idi. Beləliklə, rotorların başlanğıc vəziyyəti şifrləmənin açarı rolunu oynayırdı. Rotorların başlanğıc vəziyyəti bu günkü tarixə uyğun olaraq qoyulurdu. Hər bir operatorun rotorların hərgünkü vəziyyətini göstərən xüsusi kitabı var idi. Bu isə şifrləmə sisteminin aşkar zəif cəhəti idi: bütün sirləri açmaq üçün kitabı və maşını ələ keçirmək kifayət idi.

Şifrləmənin müasir üsulları iki qrupa ayrılır: *simmetrik kodlaşdırma*, yaxud gizli açarla kodlaşdırma və *asimmetrik kodlaşdırma*, yaxud açıq açarla kodlaşdırma.

DES SXEMİ. GİZLİ AÇARLA KODLAŞDIRMA

Gizli açarla kodlaşdırmada eyni açar məlumatın həm şifrlənməsində, həm də deşifrlənməsində istifadə olunur. Belə kodlaşdırma *simmetrik kodlaşdırma* adlanır.

Qeyd edilmiş sayda ardıcıl baytlardan ibarət mətn blokları kodlaşdırılır. Açar da bir neçə baytdan ibarət ola bilər. Məsələn, DES (Data Encryption Standard – "verilənlərin şifrlənməsinin standartı") sxemində blokun ölçüsü 8 bayt və ya 64 bitdir. Açarın uzunluğu da 8 baytdır. Doğrudur, bunlarda ancaq 56 bit istifadə olunur: hər baytın böyük biti cüt olmanı yoxlayır və şifrləmədə iştirak etmir.

Şifrləmədə 64 ikilik mərtəbədə yerləşən giriş söz həmin uzunluqda olan digər şifrlənmiş giriş sözlə əvəz olunur. DES-in tam təsviri çox mürəkkəbdir və onlarla səhifə tutur. Onun əsas ideyaları bunlardır:

1. Proqramlaşdırmanın müxtəlif sahələrində, xüsusilə də şifrləmədə bitlər üzərində əməldən – "istisna edən VƏ YA" əməlindən istifadə edilir (bu əməl XOR kimi işarə olunur). Riyaziyyatçılar bu əməli "modul ikiyə görə toplama" adlandırırlar. Əməl aşağıdakı qaydada təyin olunur:

$$0 \oplus 0 = 0,\ 0 \oplus 1 = 1,$$
$$1 \oplus 0 = 1,\ 1 \oplus 1 = 0.$$

XOR əməli $\oplus$ simvolu ilə işarə olunur. Bu əməl iki ikilik sözə tətbiq edildikdə, o, uyğun mövqelərdə yerləşən hər bit cütü üçün ayrıca yerinə yetirilir. XOR əməli çox yaxşı geriyə dönmə xassəsinə malikdir:

$$(X \oplus Y) \oplus Y = X$$

X-ə Y əlavə etməklə X məlumatı şifrlənir və deşifrləmə üçün həmin əməli bir də yerinə yetirmək kifayətdir.

2. DES sxemində çevirmələr raund adlanan elementar addımlara bölünür. Belə raundların sayı 16-dır. Hər bir raundda ilkin açardan müəyyən qayda ilə təşkil olunmuş 48-bitlik K açarından istifadə edilir. İlkin 64-bitlik söz 32-bitlik L (sol) və R (sağ) yarım sözə ayrılır. Bunlara aşağıdakı Feynstel çevirməsi tətbiq olunur. Bu çevirmə IBM şirkəti üçün ilk simmetrik kodlaşdırma üsullarından biri olan "Lütsifer"i işləyib hazırlamış Horst Feystelin (Horst Feistel) şərəfinə belə adlandırılmışdır.

$$L' = R;$$
$$R' = L \oplus f(R, K),$$

burada L' və R' – sözün sol və sağ yarısının yeni qiymətləridir.

Feynstelin əsas ideyası ondan ibarətdir ki, belə çevirmə f funksiyasının növündən asılı olmayaraq geriyə çevrilə bilər:

$$R' \oplus f(R, K) = L;$$

və

$$R = L';$$
$$L = R' \oplus f(L', K)$$

çevirməsi Feynstel çevirməsinin tərs çevirməsidir və ona görə də o, deşifrləmədə istifadə olunur.

3. DES sxemində ən vacib məqam f funksiyasının qurulmasıdır. Bu funksiya hər raundda K açarına və şifrlənən sözün sağ yarım hissəsinə tətbiq olunur. f funksiyasının qurulması Klod Şennonun (Claude Shannon) 1949-cu ildə "Bell System Technical Journal" jurnalında dərc olunan ideyalarına əsaslanır. Burada f funksiyasının arqumentilə nəticəsinin bir-birilə mümkün qədər az əlaqəli olması tələb olunur. Arqumentin kiçik dəyişməsi nəticənin ciddi dəyişməsinə səbəb olmalıdır ki, bu tamamilə təsadüfi görünsün.

Klod Şennon belə bir funksiyanı qurmaq üçün bir neçə dəfə "qarışdırma" və "çaşdırma" (diffusing və confusing) tipli çevirmələrin ardıcıl yerinə yetirilməsini təklif etmişdir. XX əsrin 60-cı illərində Horst Feynstel tərəfindən bu ideyanın həyata keçirilməsi təklif olundu. "Qarışdırma" sadəcə olaraq sözün bitlərinin yerdəyişməsindən ibarətdir. "Çaşdırma" üçün S-qutular adlanan (*ing.* "substitution" – "əvəzləmə") anlayış tətbiq olunur. Uzun ikilik söz dörd bitdən ibarət qruplara bölünür. Bu dörd bitin qiyməti 0-dan 15-ə qədər bir ikilik ədəd kimi nəzərdən keçirilir və bu ədəd S-qutunun girişinə verilir. S-qutu bu ədədi digər ədədlə əvəz edərək çıxışa verir. Qaytarılan ədəd də dörd bit şəklində təsvir olunur. Beləliklə, hər S-qutu dörd bitli söz üzərində müəyyən qeyd olunmuş çevirməni yerinə yetirir. Bu çevirmə imkan daxilində təsadüfiliyə yaxın olmalıdır. Deşifrləmənin mürəkkəbliyi əsasən S-qutu tərəfindən yerinə yetirilən çevirmənin mürəkkəbliyinə əsaslanır.

DES sxemində cari raundun K açarı və sözün R sağ yarım hissəsinin bitləri əsasında seçilən bir neçə S-qutudan istifadə edilir.



Feynstel çevirməsinə bənzər olan çevirmələr təkcə DES sxemində deyil, demək olar ki, bütün müasir simmetrik kodlaşdırma sxemlərində tətbiq olunur.

DES sxemi son 20 ildə praktikada müvəffəqiyyətlə tətbiq olunmuş və kifayət qədər etibarlı sxem kimi tanınmışdır. Onu sındırmaq üçün hələ ki, bütün açarların tam nəzərdən keçirilməsi üsulundan ciddi şəkildə üstün olan heç bir üsul tapılmamışdır. Bununla bərabər DES sxemində açarın uzunluğu cəmi 56 bit olmaqla kifayət qədər qısadır və bütün açarları seçib ayırmaq mümkündür. Müasir fərdi kompüterdə bunun üçün təxminən 1000 il tələb olunur ki, bu da alqoritmlər nəzəriyyəsində ciddi vaxt hesab olunmur! Xüsusi olaraq hazırlanmış çoxprosessorlu hesablama kompleksi isə DES şifrini bir neçə saata sındırır. Buna görə də DES sxeminin geniş istifadəsi dövrü sona çatmaq üzrədir. Hazırda onun modifikasiyası olan 3DES sxemi istifadə olunur. 3DES sxemi DES sxeminin üçqat tətbiqindən ibarətdir. 3DES sxemində açar iki dəfə uzundur, ona görə də tam seçmə yolu ilə onu təyin etmək real deyil. Bununla yanaşı, simmetrik şifrələmənin DES-i tamamilə əvəz edəcək yeni standartının axtarışı fəal şəkildə davam edir.



AÇIQ AÇARLA KODLAŞDIRMA

Asimmetrik şifrləmə və ya açıq açarla şifrləmə sxeminin işlənib hazırlanması kriptoqrafiyada əsl inqilab oldu. Əgər müasir simmetrik kodlaşdırma alqoritmləri bu və ya digər dərəcədə klassik şifrləri, əsasən də XX əsrin əvvəllərində istifadə olunan rotorlu şifrləmə maşınlarını xatırladırsa, açıq açarla kodlaşdırma sxeminin bənzəri tarixdə mövcud olmamışdır.

Açıq açarla kodlaşdırma ideyası çox sadədir: məlumatın şifrlənməsi və deşifrlənməsi üçün iki müxtəlif açar istifadə edilir. Məlumatın şifrlənməsi üçün istifadə olunan açar – *açıq açar* (public key) adlanır. Deşifrləmə üçün istifadə olunan açar *gizli* və ya *xüsusi açar* (private key) adlanır. Açıq açarı bilməklə gizli açarı bərpa etmək mümkün deyil.

Təsəvvür edin ki, qapı kilidinin iki müxtəlif açarı vardır. Biri qapını bağlamaq, digəri isə açmaq üçün istifadə olunur. Açıq açarla olan şifrləmə sxemi də oxşar qaydada qurulur. Açıq açar hamıya məlum olur və istənilən adam məlumatı şifrləyə bilər ("qapını bağlamaq"). Məlumatı deşifrləmək imkanını ancaq gizli açara malik olanlar əldə edirlər ("qapını açmaq").

Ceyms Bond filmlərindən biri olan "Rusiyadan məhəbbətlə" filmində "Casus 007"nin əsas məqsədi "Lektor" adlı sovet şifrləmə maşınını əldə etməkdir. Hal-hazırda buna bənzər süjet gülünc görünərdi: şifrləmə maşını, yəni kompüter üstə gəl şifrləmə alqoritmi, burada məxfi heç nə yoxdur. Şifrləmənin açıq açarını hətta telefon kitablarında da dərc etmək olar. İngilis əkskəşfiyyatı üçün gizli açarı bilmək daha maraqlı olardı, ancaq şifrlənmiş məlumatları Moskvaya göndərən agentin özü gizli açarı bilmir (gizli açar Moskvadadır).



Hazırda ən geniş yayılmış açıq açarla kodlaşdırma sxemi olan RSA sxeminin çox vacib əlavə xüsusiyyəti var: şifrləmə və deşifrləmə əməliyyatları qarşılıqlı tərsdir. Bu o deməkdir ki, açıq açarla kodlaşdırma sxemini əks istiqamətdə də tətbiq etmək olar: məlumatı gizli açarla şifrləmək, açıq açarla isə deşifrləmək olar. Beləliklə, məlumatı göndərənin şəxsiyyətini təsdiq etmək mümkündür, axı məlumatı ancaq gizli açara malik olan şəxs göndərə bilər. Bu halda açıq açarı deşifrləmə üçün tətbiq etməklə istənilən şəxs məlumatı oxuya bilər. Məlumatı göndərənin şəxsiyyətini təsdiq edən elektron imzanın qurulması məhz bu xassəyə əsaslanır.

RSA AÇIQ AÇARLA ŞİFRLƏMƏ SXEMİ

1977-ci ildə dərc olunmuş RSA açıq açarla şifrləmə sxemi öz yaradıcıları – Ronald Rivest, Adi Şamir və Leonard Adlemanın şərəfinə adlandırılmışdır. Bu sxem elmi dairələrdə hazırlandığı üçün bir çox başqa şifrləmə alqoritmlərindən fərqli olaraq heç zaman dövlət sirri statusu almamışdır. Bu sxem qədim yunanların (genişləndirilmiş Evklid alqoritmi), çinlilərin (qalıq haqqında çin teoremi) və XVII-XVIII əsrlərdə yaşamış Avropa alimlərinin (Ferma teoremi, Eyler teoremi) elementar ədədlər nəzəriyyəsində aldıqları sadə və gözəl nəticələrə əsaslanır.

RSA sxeminin əsas ideyasını bir neçə ardıcıl blokdan ibarət olan mətnin modul m-ə görə çıxıqlar halqasının elementləri kimi nəzərdən keçirilməsi təşkil edir. Blokun bütün baytlarının ikilik yazılışını birləşdirərək tam ədədin ikilik yazılışı kimi qəbul edilən uzun ikilik sözlər almaq olar. Öz növbəsində bu ədəd modul m-ə görə çıxıqlar halqasının elementi kimi nəzərdən keçirilir. Beləliklə, mətn blokları $\mathbf{Z}_m$ halqasının elementlərilə eyniləşdirilir. Şifrləmə isə $\mathbf{Z}_m$ elementlərinin çevrilməsindən ibarət olur.

RSA sxemində m ədədi kifayət qədər böyükdür, ən azı 200 onluq işarədən ibarətdir. O, iki böyük sadə ədədin hasilinə bərabərdir:

$$m = pq$$

m ədədi açıq olur, lakin onun iki sadə ədədin hasili şəklində təsviri gizli saxlanılır. m-in vuruqlara ayrılışını bilərək Eyler funksiyasını hesablamaq olar:

$$\phi(m) = (p-1)(q-1)$$

Sonra $\phi(m)$ ilə qarşılıqlı sadə ədəd olan ixtiyari e natural ədədi seçilir. Bu ədəd üçün genişləndirilmiş Evklid alqoritminin köməyilə modul $\phi(m)$-ə görə çıxıqlar halqasında d tərs elementi hesablanır:

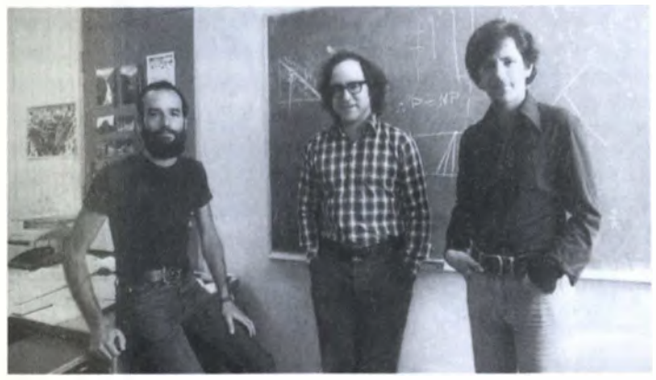
$$de \equiv 1 \pmod{\phi(m)}$$

◂ Ceyms Bond "Lektor" maşını ilə.



Pyer Ferma.

Ronald Rivest, Adi Şamir və Leonard Adleman.





Açıq açar (m, e) cütü, gizli açar isə (m, d) cütü olur.

E şifrləmə üsulu ilkin s ədədini modul m-ə görə çıxıqlar halqasında e dərəcədən qüvvətə yüksəltməkdən ibarətdir:

$$E(s) \equiv s^e \pmod m$$

Burada $\mathbf{Z}_m$ halqasının elementi kimi nəzərdən keçirilən ilkin məlumat bloku s ilə işarə edilmişdir.

D deşifrləmə üsulu modul m-ə görə çıxıqlar halqasında şifrlənmiş bloku d dərəcədən qüvvətə yüksəltməkdən ibarətdir:

$$D(t) \equiv t^d \pmod m$$

Burada t – şifrlənmiş məlumat blokudur.

E və D qarşılıqlı tərs üsullardır.



Şifrlənmiş

$$t \equiv s^e \pmod m$$

məlumatına deşifrləmə üsulunu tətbiq edək:

$$D(t) \equiv t^d\,(s^e)^d \equiv s^{ed} \pmod m.$$

e və d ədədləri modul $\phi(m)$-ə görə çıxıqlar halqasında qarşılıqlı tərs ədədlər kimi təyin olunduğu üçün yazmaq olar:

$$ed \equiv 1 \pmod{(m)}.$$

Buradan alınır ki, $ed-1$ fərqi $\phi(m)$-ə bölünür.

Hər-hansı k natural ədədi üçün:

$$ed - 1 = k\,(m),$$
$$ed = k\,(m) + 1.$$

Buradan

$$s^{ed} = s^{k(m)+1}\ s \pmod m.$$

Axırıncı bərabərlik Eyler teoreminə görə doğrudur. Beləliklə, ilkin s məlumatına əvvəlcə şifrləmə üsulunu, sonra isə deşifrləmə üsulunu tətbiq etməklə yenidən s ilkin məlumatını almaq olar.

Göründüyü kimi həm şifrləmə, həm də deşifrləmə çıxıqlar halqasında qüvvətə yüksəltmədən ibarətdir. Şifrləmədə iştirak edən ədədlərin uzunluğu 200 onluq işarədən az olmayan kifayət qədər böyük ədədlər olmalıdır.

Kiçik ədədlər üçün növbəti misala baxaq.

Tutaq ki, m=11 · 13=143. Eyler funksiyasını hesablayaq: (m) = 10 · 12 = 120

e =113 seçərək $\mathbf{Z}_{120}$ halqasında e ədədinə tərs olan d = 17 elementini alarıq:

$$113 \cdot 17 = 1921 = 120 \cdot 16 + 1 =$$
$$= 1 \pmod{120}$$

(143, 113) cütlüyü açıq açarı, (143, 17) cütlüyü isə gizli açarı təşkil edir. E şifrləmə üsulu 143 moduluna görə 113 dərəcədən qüvvətə yüksəltməkdən, D deşifrləmə üsulu isə 143 moduluna görə 17 dərəcədən qüvvətə yüksəltməkdən ibarətdir. Məsələn, s=123 məlumatını şifrləyək.

$$E(123) \equiv 123^{113} \pmod{143} \equiv 41$$

Burada 41 kodlaşdırılmış məlumatdır. Buna dekodlaşdırma üsulunu tətbiq etsək, alarıq:

$$D\,(41) \equiv 41^{17} \pmod{143} \equiv 123$$

123 isə ilkin məlumatdır.

Gizli açarın açıq açara görə bərpa olunması məsələsi nə qədər mürəkkəbdir? Açıq açar (m, e), gizli açar isə (m, d) ədədlər cütüdür, burada d modul m-ə görə çıxıqlar halqasında tərs elementdir. Deməli, d-ni hesablamaq üçün (m) Eyler funksiyasını bilmək lazımdır. Eyler funksiyasının hesablanması məsələsi m ədədinin vuruqlara ayrılması məsələsinə ekvivalentdir. Başqa sözlə desək, açıq açarı müəyyən etmək üçün m ədədini vuruqlara ayırmaq lazımdır.

Qeyd edək ki, ədədin vuruqlara ayrılması çox mürəkkəb məsələdir. Bu məsələ Eylerdən başlayaraq riyaziyyatçılardan çoxunun diqqətini cəlb etmişdir. Eyler Fermanın beşinci ədədini vuruqlara ayırmışdır:

$$F_5 = 2^{2^5} + 1 = 4\,294\,967\,297.$$

O vaxta qədər bu ədəd səhv olaraq sadə ədəd hesab olunmuşdur. Eyler gözəl bir ideyadan istifadə etmişdi – m moduluna görə üst-üstə düşən iki kvadratı tapmalı. Bu ideya müasir dövrdə də bir çox vuruqlara ayrılma alqoritmlərinin əsasını təşkil edir. Lakin bütün məlum alqoritmlərdə m ədədinin yazılışının uzunluğu artdıqca, onların yerinə yetirilməsi vaxtı da sürətlə artdığına görə onların təbii tətbiq olunma məhdudiyyəti mövcuddur. Hazırda bu məhdudiyyət 100 onluq işarədən ibarətdir. Bu zaman kompüterin sürətinin artması, demək olar

MODULA GÖRƏ ÇIXIQLAR HALQASI

Elementar ədədlər nəzəriyyəsinin əsas obyekti $\mathbf{Z}_m$ kimi işarə olunan modul m-ə görə çıxıqlar halqasıdır. Cəbrdə halqa elə çoxluğa deyilir ki, onun elementlərini toplamaq və vurmaq mümkün olsun. Məsələn, Z tam ədədlər çoxluğu halqadır. Tutaq ki, m hər hansı natural ədəddir. $\mathbf{Z}_m$ halqası aşağıdakı şəkildə qurulur: bütün Z tam ədədlər çoxluğu m sayda elə altçoxluğa bölünür ki, bu çoxluqların hər birinə daxil olan ədədlərin fərqi m-ə bölünsün. Sıfırıncı çoxluğa {...–m, 0, m, 2m, ...} ədədləri, birinci çoxluğa {...–m+1, 1, m+1, 2m+1, ...} ədədləri, (m-1)-ci çoxluğa {...–1, m–1, 2m–1, 3m–1, ...} ədədləri daxildir. Riyaziyyatda bu altçoxluqlar sinif adlandırılır. Göstərilən siniflər $\mathbf{Z}_m$ halqasının m sayda elementləridir. Siniflər arasında vurma və toplama əməlləri hər sinifdən biri olmaqla iki "nümayəndə" element arasında aparılır.

Tərifə görə cəm (hasil) hansı sinfə daxildirsə, həmin sinif iki ilkin sinfin cəmi (hasili) adlanır.

x və y ədədlərinin eyni sinfə daxil olması faktı (yəni, onların fərqinin m-ə bölünən olması) riyaziyyatda $x \equiv y$ (mod m) kimi yazılır. Bu belə oxunur: "x y-lə modul m-ə görə müqayisə olunandır".

Hər sinfin nümayəndələri sisteminə qalıqlar sistemi deyilir. Adətən riyaziyyatçılar mənfi olmayan {0, 1, 2, ..., m–1} qalıqlar sistemini qiymətləndirirlər. Lakin proqramlaşmada həm mənfi, həm də müsbət ədədlərdən ibarət olan mütləq qiymətcə minimal simmetrik qalıqlar sistemilə işləmək əlverişlidir. Məsələn, $\mathbf{Z}_5$ halqası üçün simmetrik qalıqlar sistemi {-2, -1, 0, 1, 2} ədədlərindən ibarətdir.

Kompüter tam ədədlərlə deyil, modul m=232-yə görə çıxıqlar halqasının elementlərilə işləyir. Proqramçıların müsbət və mənfi adlandırdıqları ədədlər əslində bu halqanın simmetrik qalıqlar sisteminin elementləridir.

$\mathbf{Z}_m$ halqalarının əhəmiyyəti praktikada ondan ibarətdir ki, bu halqaların elementləri sonsuz deyil və elə bu səbəbdən hesablama zamanı bu ədədlər sonsuzluğa qədər artmır. "Həqiqi" tam ədədlərlə çox nadir hallarda hesablama aparılır. Bu onunla izah olunur ki, tam ədədlərin hasilində nəticə çox sürətlə artır.

Məsələn, 100 rəqəmli iki onluq ədədin vurulması zamanı hasil 199 və ya 200 rəqəmdən ibarət olur, 100 rəqəmli ədədin 100 rəqəmli qüvvətə yüksəldilməsi isə ümumiyyətlə real deyildir: nəticə bütün kainatdakı elementar zərrəciklərin sayından çox ola bilər!



Açıq açarı təyin etmək üçün m ədədinin vuruqlara ayrılmasının kifayət olub-olmaması hələ də isbat edilməmişdir (RSA şifrinin sındırılmasının ədədin sadə vuruqlara ayrılması məsələsinə ekvivalent olması məlum deyil).

EVKLİD ALQORİTMİ

Ədədlər nəzəriyyəsinin təməl daşlarından biri də Evklid alqoritmidir: n və m tam ədədləri verilmişdir, onların ən böyük ortaq böləni d = ƏBOB(n, m) ədədini tapmaq lazımdır. Bunun üçün (n, m) ədədlər cütü ardıcıl olaraq (m, r) cütü ilə əvəz edilir. Burada r n-in m-ə bölünməsindən alınan qalıqdır, yəni $n = qm + r$. (n, m) və (m, r) cütlərinin ümumi bölənləri üst-üstə düşdüyünə görə: ƏBOB(n, m) = ƏBOB(m, r). Digər tərəfdən ƏBOB(m, 0) = m olacaqdır. (n, m) cütünü (m, r) cütü ilə əvəz etdikdə ikinci ədəd azalır və sonlu sayda addımdan sonra sıfra bərabər olur. Ona görə də $r = 0$ olan halda alınan cütün birinci ədədi ilkin cütün ƏBOB-u olacaq.

Evklid alqoritmi çox sürətlə işləyən alqoritmdir. Bu alqoritmdə addımların sayı təqribi olaraq ədədin ikilik yazılışında bitlərin sayına bərabərdir. Yəni 100 rəqəmli onluq ədədlər üçün Evklid alqoritmi təxminən 330 addımdan sonra yekunlaşır. Praktikada d = ƏBOB(m, n) ədədini n və m ədədlərinin xətti kombinasiyası şəklində göstərmək imkanı verən genişlənmiş Evklid alqoritmi xüsusi əhəmiyyət kəsb edir. Bu halda

$$d = un + vm,$$

olacaq ki, burada u və v hər hansı tam ədədlərdir. Genişlənmiş Evklid alqoritmi modul m-ə görə çıxıqlar halqasında n-ə tərs olan elementi hesablamaq imkanı verir. $\mathbf{Z}_m$ halqasında u və n elementlərinin hasili vahid olarsa, u elementi n-in tərsi adlanır. Məsələn, $\mathbf{Z}_5$ halqasında

$$2 \cdot 3 = 6 \equiv 1 \pmod 5$$

olduğundan $\mathbf{Z}_5$ halqasında 2 ədədi 3-ün tərsi olur. Modul m-ə görə çıxıqlar halqasında n-ədədinin yalnız və yalnız o vaxt tərsi var ki, n və m ədədləri qarşılıqlı sadə ədədlər olsunlar, yəni ƏBOB (n, m) = 1 Genişlənmiş Evklid alqoritmini tətbiq edərək elə u və v ədədlərini tapmaq olar ki,

$$1 \equiv un + vn \text{ olsun.}$$

Lakin $vm \equiv 0 \pmod m$ olduğuna görə

$$1 \equiv un \pmod m,$$

yəni $\mathbf{Z}_m$ çıxıqlar halqasında u elementi n elementinin tərsidir.



SÜRƏTLİ QÜVVƏTƏ YÜKSƏLTMƏ

Praktikada hər zaman istifadə olunan alqoritmlərdən biri də sürətli qüvvətə yüksəltmə alqoritmidir. Bir qayda olaraq, bu alqoritm tam ədədlərə deyil, çıxıqlar halqasının elementlərinə tətbiq olunur (tam ədədləri böyük qüvvətə yüksəltmək mümkün deyil, çünki nəticə çox sürətlə artır). Alqoritmin ideyası ondan ibarətdir ki, qüvvətə yüksəltmə əməli kvadrata yüksəltmə və vurma əməlinə gətirilir. Bu əməllər $\mathbf{Z}_m$ halqasında yerinə yetirilir. Tutaq ki, s elementini e dərəcədən qüvvətə yüksəltmək lazımdır. Aşağıdakı alqoritm istifadə olunur:

```
alq tam sürətli qüvvətə yüksəltmə
(tam: s, e)
verilib: s ədədi, e qüvvət dərəcəsi
gərəkdir: çıxıqlar halqasında
s ədədini e dərəcədən qüvvətə yüksəltmək
baş tam p
| p := 1
| db nə qədər ki e > 0
| | əgər e/2*2 = e | e cütdür
| | | onda e := e/2; s := s*s
| | | əks halda e := e – 1; p := p*s
| | tamam
| ds
| qiym := p
son
```

Alqoritmin düzgünlüyü ondan alınır ki, hər dəfə dövr yerinə yetirildikdən sonra ps^e kəmiyyəti dəyişməz qalır. Başlanğıcda $p = 1$ olduğu üçün $ps^e = s^e$ olur. Alqoritm sona çatdıqda $e = 0$ və $ps^e = p$ olur, yəni p dəyişəninə s-in ilkin qiymətinin axtarılan dərəcədən qüvvəti mənimsədilir. Alqoritm həmçinin çox sürətlə işləyir. Addımların sayı e ədədinin ikilik yazılışındakı bitlərin sayının iki mislini aşmır. Xüsusilə, əgər e 100 rəqəmli onluq ədəd olarsa, alqoritmdə addımların sayı orta hesabla 600 olar.

FERMANIN KİÇİK TEOREMİ VƏ EYLER TEOREMİ

Fermanın kiçik teoremi ədədlər nəzəriyyəsinin əsl incisi hesab olunur.

Tutaq ki, m sadə ədəddir (yəni qalıqsız olaraq ancaq özünə və vahidə bölünür). Onda m-ə bölünməyən istənilən tam b ədədi üçün aşağıdakı müqayisə doğrudur:

$$b^{m-1} \equiv 1 \pmod m$$

Məsələn, tutaq ki, $m = 7$, $b = 2$, onda

$$2^6 = 64 = 9 \cdot 7 + 1 \equiv 1 \pmod 7.$$

Fermanın kiçik teoreminin digər forması da var. Tutaq ki, m – sadə ədəddir. Onda istənilən b tam ədədi üçün aşağıdakı eynilik doğrudur:

$$b^m \equiv b \pmod m.$$

XVIII əsrdə Leonard Eylenin (Leonhard Euler) m ixtiyari ədəd olan hal üçün Ferma kiçik teoremini ümumiləşdirdi. Tərifə görə, m-dən asılı Eyler funksiyası Z_m halqasında tərsi olan elementlərin sayıdır.

Eyler teoremi: Tutaq ki, m vahiddən fərqli ixtiyari natural ədəddir, b isə m ilə qarşılıqlı sadə ədəddir, onda

$$b\phi^{(m)} \equiv 1 \pmod m.$$

Eyler funksiyası asanlıqla digər klassik teoremin – qalıqlar haqqında Çin teoreminin köməyilə hesablanır (teorem burada verilmir). RSA sxemində $m = pq$ üçün Eyler teoreminin xüsusi halı istifadə olunur ki, burada p və q müxtəlif qarşılıqlı sadə ədədlərdir. Bu halda

$$(m) = (p–1)(q–1)$$

Eyler teoreminin aşağıdakı gücləndirilmiş variantı mövcuddur: Tutaq ki, $m = pq$, burada p və q müxtəlif sadə ədədlərdir. Onda istənilən b tam ədədi və istənilən k natural ədədi üçün aşağıdakı eynilik doğrudur:

$$b^{k(m)+1} \equiv b \pmod m$$

Bu halda artıq b və m ədədlərinin qarşılıqlı sadə ədədlər olması tələb olunmur.

ki, heç nə vermir. Sadə dillə desək, kompüter milyon dəfə sürətli işlədikdə belə, 100 onluq işarəli ədəd əvəzinə 120 onluq işarəli ədəd emal edə bilər, lakin 140 onluq işarəli ədədin ayrılmasını heç bir zaman yerinə yetirə bilməz. Riyaziyyatçıların çoxu ədədlərin vuruqlara ayrılması məsələsi üçün sürətli alqoritmin mövcud olmadığına əmindirlər. Baxmayaraq ki, bu fakt ciddi şəkildə isbat olunmamışdır.

Ədədlərdən fərqli olaraq, çoxhədlilər üçün vuruqlara ayırmanın sürətli alqoritmi vardır. Bu bir daha tam ədədlərin mürəkkəb təbiətə malik olduğunu təsdiq edir. Vuruqlara ayırmadan fərqli olaraq, ədədlərin sadə olmasının yoxlanılması və böyük sadə ədədlərin alınması məsələləri XX əsrin 70-ci illərində tapılan alqoritmik həllərə malikdir. Bu isə RSA sxemində kifayət qədər uzun açar generasiya etməyə imkan verir.

RSA sxemində şifrləmə və deşifrləmə simmetrik şifrləmə alqoritmlərində istifadə olunan əməliyyatlara nisbətən daha ləng yerinə yetirilir. Ona görə də açıq açarla şifrləmə ən mühüm hallarda tətbiq olunur. Hər şeydən əvvəl bu şəbəkəyə giriş və istifadəçinin identifikasiyası üsuludur ki, burada açıq açarla kodlaşdırma sxemi parolun primitiv ötürülməsini əvəz edir. İstifadəçinin şəxsiyyətini və elektron imzasını təsdiq etmək üçün bu imkandan başqa gizli açarların açıq əlaqə kanalı ilə mübadiləsi üsulu mövcuddur. Adətən istifadəçi şəbəkəyə daxil olduqdan sonra sistem ona simmetrik şifrləmə üçün (3DES və s.) bir dəfə istifadə olunan açar ayırır və şəbəkə ilə ötürülən bütün informasiya simmetrik açar vasitəsilə şifrlənir.

> Açıq açarla kodlaşdırma zamanı hakerlərdən çəkinməyə ehtiyac yoxdur. Axı açıq açarı təyin etmək üçün sırf riyazi bir məsələni – ədədin vuruqlara ayrılması məsələsini həll etmək lazımdır. Bunu isə adi haker bacaran deyil.



Leonard Eyler.



MÜASİR İNFORMASİYA TEXNOLOGİYALARI

İNFORMASİYA VƏ POSTİNFORMASİYA CƏMİYYƏTİ

Hazırda bəşəriyyət bir cəmiyyətdən digərinə keçid dövrünü yaşayır. Sənaye cəmiyyəti informasiya cəmiyyəti ilə əvəz olunur. İnformasiya və Postinformasiya Cəmiyyətində informasiyanın əldə olunması daha asan olur, məlumatların saxlanması, emal olunması və ötürülməsi üçün kompüter universal cihaz kimi istifadə olunur. Gələcəkdə hansı hadisələrin baş verəcəyini, İnformasiya və Postinformasiya Cəmiyyətinin hansı cəmiyyətlə əvəz olunacağını bilmək çox maraqlıdır.

Qənaətlərə görə, informasiya cəmiyyəti dövrü praktiki olaraq başlamamış artıq sona çatmışdır. Kommunikasiya vasitələri və yeni texnologiyaların sürətli inkişafı nəticəsində elə bir şərait yaranıb ki, informasiya istifadə olunmağa macal tapmadan aktuallığını itirir. İnkişaf etmiş ölkələrdən birinin kəşfiyyat xidmətinin rəhbəri qeyd etmişdir ki, əgər yaxın zamanlaradək əsas problem məlumatların əldə edilməsi idisə, indi birinci yerdə məlumatların emal olunması durur. İnformasiya cəmiyyətinin başa çatması ilə dünya maliyyə sisteminin dağılması təhlükəsi yaranır, belə ki pulu onun daha çox haradan əldə edildyini bilən, yəni daha çox informasiyaya malik olan qazanır. Qlobal xaosu şərtləndirən digər faktor isə idarəetmə texnologiyasının böhranıdır. Ənənəvi "tədqiqat – nəticələrin analizi – idarəetmə qərarlarının sintezi" sxemi üzrə fəaliyyət göstərmək üçün sadəcə vaxt çatmaya bilər və bu səbəbdən qlobal xarakterli qərarlarda xəta və səhvlərə yol verilə bilər. Vəziyyətə nəzarətin itirilməsi nəticəsində bəşəriyyətin böyük bir hissəsinin əzilməsi

və məhv edilməsilə bütün dünyada tiranlıq bərqərar olar. Heç də ürəkaçan perspektiv deyil!

Belə yanaşma da özünü doğrultmur. Bəşəriyyət kompüterlərdən fəal şəkildə istifadə edərək hələ ki daim artan informasiya kütlələrini "həzm edə" bilir, qlobal idarəetmə qərarları isə əvvəlkitək insanlar tərəfindən qəbul edilir. Təbii ki, böyük həcmdə etibarlı informasiyanın mövcudluğu şəraitində qəbul olunan qərarlar daha doğru olacaq.

Bir başqa nöqteyi-nəzərdən baxsaq görərik ki, hazırda informasiyanın bolluğu deyil, qıtlığı müşahidə olunur. Məlumatların həcmi və informasiya axını sürətinin qar uçqunu kimi artmasına baxmayaraq, son nəticədə bütün bunlar insanlara cəmi iki eşitmə və görmə kanalları vasitəsilə çatdırılır. Məsələ ondadır ki, informasiya cəmiyyətində bu kanallar ifrat dərəcədə yüklənmiş olur.

Gələcək postinformasiya cəmiyyəti informasiyadan imtina etməyi deyil, digər kanallara da üstünlük verilməsini nəzərdə tutur. Hər halda insanın duyğu orqanları ikidən çoxdur (hərçənd bu halda sual ortaya çıxır onda bu cəmiyyəti nəyə görə meqainformasiya deyil, postinformasiya cəmiyyəti adlandırırlar?).

Sabahın mütəxəssisləri arasında əsas fiqur kimi intellektual – informasiya istehsal edən adam çıxış edəcək və informasiya istehsalı isə sənayedə ən vacib yeri tutacaq. Dünya qloballaşır: onun vahid maliyyə sistemi, informasiya sahəsi, mənəvi dəyərləri, yaşayış stili və s. formalaşır. Bununla bərabər hər ölkə özünəməxsusluğunu qoruyaraq müstəqil həyat sürəcək. Yeni minilliyin məhz bu paradoksu postinformasiya cəmiyyətinin formalaşmasına gətirib çıxaracaq. Bu isə daxili impulsların və mövcud olan bütün texnologiyaların insanın cari vəziyyətə uyğunlaşmasına kömək edəcəyi şəraitdə baş verəcək.

> Ostankino teleqülləsində 2001-ci ilin yayında baş verən faciə və sonrakı günlərin hadisələrini postinformasiya cəmiyyətində mümkün olan yollardan biri kimi nəzərdən keçirmək olar. Milyonlarla moskvalının evində mavi ekranlar söndükdə bir çoxları həqiqətən də informasiya aclığı keçirdi. Hərçənd ki, bu daha çox "televiziya" aclığı idi, çünki radiostansiyalar məlumat axınında çatışmamazlığı uğurla kompensasiya edirdi.

Ümumilikdə postinformasiya cəmiyyəti barədə fərziyyələrin əksəriyyəti (dünyanın axırının çatacağını bildirən az bir qismi nəzərə almasaq) informasiya cəmiyyəti daxilində bəzi, çox vaxt əhəmiyyətsiz dəyişiklikləri proqnozlaşdırır. Gənclik və yetkinlik arasında dəqiq bir sərhəd qoymaq mümkün olmadığı kimi, bu keçidi də dəqiq təyin etmək qeyri-mümkündür. 15 yaşında insan hələ gəncdir, 40 yaş isə, şübhəsiz, yetkinlik yaşıdır. Cəmiyyət də belədir, XV əsrdə o, sənaye cəmiyyəti olmayıb, XIX əsrdə isə informasiya cəmiyyəti deyildi. İnformasiya cəmiyyəti bəşəriyyətin yetkinlik çağıdır. İnsanın özü kimi bu cəmiyyəti də tənəzzül və ölüm, yaxud da xəstəliklərə düçar olmuş qocalıq gözlədiyini deyirlər. Lakin optimist fərziyyə də səslənir: yetkinlik dövrü tənəzzüllə əvəz olunmayacaq, yəni informasiya cəmiyyəti heç vaxt qocalmayacaq. Lap nağılların sonluğu kimi: "Onlar birlikdə uzun və xoşbəxt həyat sürdülər".



KÜTLƏVİ İNFORMASİYA İSTEHSALÇILARI

Kütləvi informasiya vasitələrinin (KİV) sadəcə xəbərləri çatdırması və ya araşdırmalar aparması, yaxud da dünyanı tanınmayacaq dərəcədə dəyişən informasiyanın məhz KİV nümayəndələri tərəfindən istehsal edilməsilə bağlı çox vaxt mübahisələr yaranır. Təəccüblü deyil ki, onlardan ehtiyat edir və çəkinirlər, axı KİV dördüncü hakimiyyətdir.

Kütləvi informasiya vasitələri nədir? Bu, ilk növbədə ətraf aləmlə bağlı bizə məlumatlar verən nəşrlərdir. Bu nəşrlərdə ən əsas yeri həftə, gün, saat, bir neçə dəqiqə kimi kiçik zaman kəsiyində dünyada, cəmiyyətdə, konkret insanlar və məşhur personajlar ətrafında, mədəniyyət və elmdə baş verən yeniliklər tutur. Buna görə bu günün və gələcəyin KİV-i üçün informasiyanın operativ çatdırılması əsasdır.

Getdikcə KİV sistemi və texnologiyaları daha sürətlə dəyişir. Dünya bütövləşərək daha açıq olur, bu səbəbdən həmin proses *informasiya sahəsinin* inkişafına təsirsiz ötüşmür. Hətta ən güclü fantaziyalar belə gələcəyi qabaqcadan bilə bilmir.

Kütləvi informasiya vasitələri şərti olaraq *ənənəvi* və *sintetik* vasitələrə ayrılır. Ənənəvi qrupa bunları aid etmək olar:

• oflayn və ya "kağız" nəşrlər – qəzet və jurnallar, digər çap məhsulları;

• elektron vasitələr – televiziya və radio;

• İnternet-nəşrlər.

Sintetik KİV, adından da göründüyü kimi, bütün bu texnologiyaları bir araya gətirir.

Zaman keçsə də qəzet və jurnallara münasibət dəyişmir. Onlar əvvəllər olduğu kimi, yenə mövcud olacaqlar, çünki qəzet səhifələrində çap olunmuş mətni oxumaq rahat və xoşdur. Bu, hətta ciddi iş adamları və mötəbər görünmək istəyən gənclər üçün nüfuz və konservatizmin elementi kimi qəbul olunacaq. Bundan əlavə, qəzet və jurnalın bir şəksiz üstünlüyü öz oxucusunu gözləməsidir. Hər hansı informasiyanı, hadisələrin şərhini elə həmin anda oxumaq məcburiyyətilə üzləşmirsən. Bu barədə başqa bir gün, asudə vaxtda, istirahət zamanı oxuyub düşünmək olar. Qəzet oxuyarkən ətraf aləmdən təcrid olunmaq, əldə bir fincan qəhvə onunla təkbətək qalmaq imkanı var.

Digər tərəfdən, daycestlər – müəyyən mövzu barədə toplu şəklində nəşrlər meydana gələcək. Eyni bir hadisə müxtəlif nəşrlərin jurnalistləri tərəfindən fərqli nöqteyi-nəzərdən veriləcək. Məsələn, yeni raketin yaradılması barədə həm elmi nəşrin, ədəbi məcmuənin, müxtəlif baxışlı siyasi nəşrlərin, həm də gənclər üçün qəzetin müəllifləri yazacaq. Beləliklə, oxucu eyni bir məvhuma verilən hərtərəfli qiymətləndirmələri əldə edəcək. Nəşrlərin mövzusu dəyişəcək. Əgər indi qəzetlər əsasən siyasət və sağlamlıq barədə yazırsa, gələcəkdə əsas mövzular elm və mədəniyyətlə bağlı olacaq. Planetimizin xəritəsi dəyişəcək, dünya birləşəcək. Siyasət artıq bütün cəmiyyəti maraqlandırmayacaq, lakin yerin dərinliklərinə nüfuz etmə problemi, yüksək texnologiyaların inkişafı, ən aktual sahələrdə biliklər, mədəniyyət, mənəviyyat cəmiyyəti hər zaman narahat edən məsələlərdir. Əlbəttə, qəzet xəbərlərin birbirini sürətlə əvəz etməsilə ayaqlaşa bilmir, lakin qeyd olunan mövzularla bağlı informasiyanın səhifələrdə daim yenilənməsi də tələb olunmur.

Stiven Spilberqin 2005-ci ildə işıq üzü görmüş "Dünyaların müharibəsi" blokbasterində belə bir epizod var. KİV-in böyük əksəriyyətinin yad planetlilər tərəfindən məhv edildiyi bir şəraitdə mühasirədə möcüzə olaraq sağ qalan telekanallardan birinin müxbiri sensasiyalı reportajın alınmamasından təəssüflənir. Həqiqətən də jurnalist üçün xəbər həyatından qiymətlidir.



Qəzetlərin taleyilə bağlı nikbin olmaq olar, lakin qalın ədəbi jurnallar barəsində də eyni şeyi söyləmək qeyrimümkündür. Yeni romanlar, povest və hekayələr kitab şəklində nəşr olunaraq İnternetdə satılacağından onlar ixtisar olunacaq. Bu iki format qalın jurnalların ailəvi oxunması ənənəsinin tamamilə axırına çıxacaq.

Fotoşəkillərə nisbətən mətnin az olduğu parlaq şəkilli jurnallar da öz yerində qalacaq, çünki işdən sonra (məsələn, kompüterlə) hər bir insanın daxilində gözəl illüstrasiyalara baxaraq yüngül oxunan sadə mətnli səhifələri vərəqləmək ehtiyacı var.

Radio və televiziyaya münasibət dəyişəcək, belə ki, kanalları çevirərək pultun düymələrini çıqqıldatmağa az adamın həvəsi olacaq. İnformasiyanın istehlakçısı onun bilavasitə sifarişçisinə çevriləcək. O, öz radio və tele-gününü artıq özü proqramlaşdıracaq. Məsələn, bir radiostansiyanın iqtisadiyyat xəbərlərini, digərinin idman, üçüncüsünün isə mədəniyyət xəbərlərini və s. sifariş etmək üçün texnologiya tətbiq olunacaq.

Musiqini də zövqə uyğun seçəcəklər. Qəbuledici müəyyən proqram əsasında başqa-başqa radiodalğalardan musiqi kompozisiyalarını "çəkib gətirəcək". Dinləyici, onu maraqlandıran tok-şouda iştirak etmək istədikdə səslə uyğun komandanı vermək kifayət edəcək ki, qəbuledici müəyyən dalğaya köklənərək onu həmin verilişin iştirakçısı etsin.



Bundan əlavə, qəbuledicini elə kökləmək olar ki, o, ancaq pozitiv (və ya neqativ) informasiyanı qəbul etsin, ancaq müəyyən tembrə malik aparıcıların səsini versin. Təbiidir ki, belə vəziyyətdə radiostansiyalar və dinləyicilər arasında "hərbi əməliyyatlar" güclənəcək. İnsanlar öz seçim haqqını təkidlə tələb edəcək, radiostansiyalar isə hər cür doğruyalan üsullarla müştəri uğrunda mübarizə aparacaq. Buna görə də bir "dalğaya düşən" qeyd olunacaq və oraya "yapışdırılacaq". Xüsusi elektron "qarmaqlar" onun başqa radiostansiyaya köklənməsinə imkan verməyəcək. "Qarmağı"





daha güclü olan qalib gələcək. Bir çox stansiyalar elə bu məqsədlə qeyri-qanuni olaraq dinləyicilərin *neyroleptik proqramlaşdırılmasından* istifadə edəcək. Bunun qarşısını almağa çalışacaqlar, lakin hətta stansiyanın bağlanma təhlükəsi bu problemi həll etməyəcək.

Eyni situasiya televiziyada da yaranacaq, tamaşaçı özü bəyəndiyi diktoru seçir və sifariş edir ki, müxtəlif kanallardan bütün xəbərləri məhz o oxusun. Kompüter texnologiyaları bir proqramda bütün televiziya kanallarının videotekalarında olan film və verilişlərə baxmaq imkanı verəcək. Və əlbəttə ki, filmin sonluğunu öz istəyinizə uyğun seçə biləcəksiniz.

Televiziya verilişlərinin cədvəli çox dəyişkən olacaq. Vacib və fövqəladə hadisələr zamanı verilişlər kəsiləcək və görüntü avtomatik olaraq studiyaya deyil, hadisə yerinə keçəcək. Tamaşaçı detallaşdırma texnologiyasından istifadə edə biləcək. O özü kadrda maraqlandığı fraqmenti seçərək onu böyüdəcək, növbəti kadrla, aparıcının sifəti və s. ilə montaj edəcək. Tamaşaçı studiyaya rəqəmli kanal vasitəsilə bağlanaraq öz virtual təsvirini İnternetlə göndərib toksşounun iştirakçısı ola biləcək.

İnternet-nəşrlərin gələcəkdə oflayn və elektron KİV arasında tayı-bərabəri olmayacaq. Onların arasında oxucu uğrunda amansız mübarizə alovlanacaq. Mövzuların genişləndirilməsi, bağlantıların rəngarəngliyi və jurnalistlərin ələ alınması populyarlıq uğrunda mübarizənin ancaq bir neçə formasıdır. Hər hansı sayta daxil olmuş bir kəs lazımi informasiya axtarışında oranı tərk etməməlidir. O, həmin KİV-in şəbəkələrində ilişib qalmalı və onu daim startda saxlamalıdır. Orada nəinki bu və ya digər jurnalist materialları, həmçinin müəyyən mövzuda İnternetdə yerləşdirilmiş bütün KİV və digər mənbələrə avtomatik bağlantılar olacaq.

İnternet-nəşrlər sintetik olaraq özündə mətn, səs və video kimi oflayn və elektron texnologiyaları birləşdirəcək. Bir çox materialları həm görmək, həm də eşitmək mümkün olacaq. İndinin özündə artıq elə radio-stansiyalar var ki, onları nəinki dinləmək, hətta İnternetdə görmək də mümkündür, belə ki, xüsusi layihələr rəqəmsal video-kameralarla çəkilir. Çox tezliklə qonaqların müsahibələri, diccyin işi, videokliplər, bir sözlə, bütün efir çəkilib ümumdünya şəbəkəsində translyasiya olunacaq.

İnternetdə, həmçinin bütün telekanalların verilişlərinə və filmlərinə tamaşa etmək mümkün olacaq. Buna görə də istənilən süjeti saytda oxumaq-görmək-



İNFORMASİYA MÜHARİBƏLƏRİ

Gələcəkdə elektron KİV-də ən qəddar qaydasız döyüşlər informasiya müharibələri olacaq.

"İnformasiya kimin əlindədirsə, o, dünyanı idarə edir" ifadəsi aksioma çeviriləcək. İnformasiya, onun alınması və emalının sürəti biznes sektorunun uğurları ilə birbaşa əlaqəli olacaq. Bütün işgüzar fəallıq müəssisə aksiyalarının alqı-satqısı və çatdırılması sferasında cəmləşəcək. Buna görə rəqib təşkilatlar bütün mümkün informasiya kanallarından istifadə edəcəklər. Bu işdə informasiyanı həm səlis və ətraflı şəkildə, həm də analitik materiallar, digər kontekstlərdə istinad, eyham və təsadüfi sitatlar formasında saxlayan bütün KİV-in mühüm rolu olacaq. Bundan başqa KİV-lər öndə olmaq, daha çox dinləyici-tamaşaçı qazanmaq üçün eksklüziv, xüsusi, rəqiblər üçün əlçatmaz informasiya ardınca qaçacaqlar. Bu zaman siyasətçilər, hakimiyyət nümayəndələri xüsusi fəallıq göstərərək KİV-i həyasız şəkildə öz məqsədləri üçün istifadə edəcəklər. Təbii ki, insanların ətraf aləm barədə təsəvvür və biliklərini jurnalistika verir, bu barədə məlumatları biz ancaq KİV-dən ala bilirik. Çox vaxt bizə elə gəlir ki, bu və ya digər fakt barədə fikri özümüz analitik qabiliyyətimiz əsasında formalaşdırırıq. Lakin bu heç də belə deyil. Biz kiminsə çatdırdığı informasiyadan istifadə edirik, o isə həm subyektiv, həm də obyektiv ola bilər, həm də ölçülüb-biçilmiş olub informator üçün lazımi vurğu və məlumatları daşıya bilər. Məsələn, KİV-lər ancaq Avropa və Amerikada həyat səviyyəsinin artımı barədə məlumatları verə bilər. Lakin heç bir məlumat verilmədiyi üçün heç kim Afrikadakı aclıqdan xəbər tutmayacaq. Bu nəhəng dünyada adi insan nəinki dünyanın o başında, hətta qonşu şəhərdə, əgər orada tanışları yoxdursa, nə baş verdiyini bilmir.

Müharibələrdə informasiya vasitələrini öz məqsədləri üçün istifadə edərək ictimai fikri formalaşdıranlar qalib gələcəklər. Əgər bir X ölkəsi Z ölkəsini bombalayacaqsa, tərəflərdən biri üçün hərbi əməliyyatlar deyil, qəzet və jurnal səhifələrində, tele-ekranlarda və radio-efirdə gedən döyüşlər daha vacib olacaq. Əgər həqiqi müharibə haqqında xəbər vermək mümkün deyilsə, onu "rəqabət aparan bandit qrupları arasında xırda toqquşmalar" da adlandırmaq olar. Tam miqyaslı hərbi toqquşmalar barədə məlumat vermək məcburiyyəti olduqda isə fərqli informasiyalar, fikirlər yayılacaq. Təcavüzkar ölkə planetimizdə həyati təhlükəni aradan qaldırmaq uğrunda mübarizə aparan dövlət adlandırılsa belə, heç kim həqiqəti bilməyəcək. Millətçi şüarlar arxasında gizlənən quldur, narkotik maddə və insan alverçilərini azadlıq və müstəqillik uğrunda fədai adlandırmaq, onlara müqavimət göstərməyə çalışanları isə genosid təşkilatçıları adı ilə damğalamaq olar. Digər tərəfdən, əhalini artıq baş vermiş və ya gözlənilən terror aktları ilə qorxudaraq, hərbi və ya polis qüvvələrinin sadə insanlara qarşı qəddarlığını bandit qrupları və onların təəssübkeşlərilə mübarizə barədə xəbərlərin arxasında gizlətmək olar. Siyasətçilər, iri biznesmenlər və bütövlükdə dövlətlər jurnalistikanın informasiya və təbliğat resurslarından getdikcə daha çox istifadə edəcəklər. Bunların fonunda müharibələr lap uşaq bağçasındakı dava-dalaş kimi görünəcək.



eşitmək və forumda müzakirə etmək (həm onlayn, həm də arxiv-istinad rejimində) imkanı yaranacaq. Bir çox tele və radio-ulduzlar İnternetə keçəcək, burada da əvvəlki kimi səs və canlı təsvirlə işləyərək auditoriyalarını genişləndirə biləcəklər. Müəllifin öz materialını oxumasını görmək mümkün olacaq. Bundan başqa redaksiyanın onlayn rejimdə iş prosesini izləyə biləcəyik. Bu həm adi oxucular, həm də jurnalistika fakültələrinin tələbələri üçün maraqlı olacaq. Onlar peşəkarların ustad dərslərindən yararlanacaqlar.

XXI ƏSRDƏ AUDİO

YENİDƏN YAZMA = = KÖÇÜRMƏ

Kaset və vinil qramofon vallarından CD və mini-disklərə keçid səsin keyfiyyətini yaxşılaşdırmış, nəticədə nəzəri olaraq köçürməni səsləndirmə sürətindən qat-qat yüksək sürətlə həyata keçirmək imkanı meydana gəlmişdir. Audiokasetləri hətta ikiqat sürətlə yenidən yazdıqda səs keyfiyyətinin aşağı düşməsi qaçılmazdır. Səsin rəqəmli təsviri köçürüldükdə isə bu baş vermir. Köçürmə sürətinin artırılması isə ancaq istifadə olunan qurğulardan asılıdır. Məsələn, kompüterdə quraşdırılmış CD nəzəri olaraq adi səsləndirmə sürətindən 50 dəfə (əslində təqribən on dəfə) sürətlə oxumaya nail ola bilir, yazma zamanı isə bu göstərici on dəfələrlə çoxdur. Yəni 74 dəqiqəlik CD bir neçə dəqiqəyə yenidən yazıla bilər. MP3 formatında (256 kbit/san axın) 6 saatlıq musiqi parçası praktik olaraq 6 dəqiqəyə köçürülür.

Mini-disk pleyerinin mexanizmi.



Mini-disk qurğularında kütləvi köçürməyə məhdudiyyət qoyulub. Avadanlıq səsi həm rəqəmli (optik), həm də adi analoq kanalla yazmaq imkanı verir. Əgər bir disk rəqəmli kanalla yazılıbsa və ya istehsalçı fonoqramını daşıyırsa, onu ancaq oxşar kanalla yazıb (yazıcı deka nəyi yazdığını "bilir"), rəqəmli kanalla oxutmaq olar. Bu zaman yenidən yazıldıqdan sonra səsin keyfiyyəti aşağı düşür. Əksinə, əgər disk oxşar kanalla yazılıbsa, onu rəqəmli kanalla köçürmək olar. Kompüterin köməyilə asanlıqla bu qadağadan yan keçmək mümkündür.

Beləliklə, kompüterlərin köməyilə səs yazılarının köçürülməsi olduqca sadələşmişdir. İndi hər bir kəs öz evində kompakt və mini-disklərin istehsalını təşkil edə bilər. Bunun üçün tələb olunan avadanlıq da çox baha deyil. Bəs müəllif hüquqları necə olsun? Qanunvericilikdə "vəhşi" köçürmənin qadağan olunması çətin ki, kiminsə qarşısını alsın. Ümumiyyətlə, qeyri-qanuni səsyazmanın qarşısı alınmalıdır.

Praktikada isə, hər evdə İnternetə qoşulmuş kompüterin olduğu bir şəraitdə, bunu etmək hələ qeyri-mümkündür.

"NAPSTER" ƏHVALATI

1999-cu ilin yayında 19 yaşlı Şon Feninq (Sean Fanning) ümumdünya hörümçək şəbəkəsi vasitəsilə MP3 formatında musiqi fayllarını bir-birinə göndərmək imkanı verən "Napster" proqramını yazdı. Bu sistem İnternetdə xüsusi olaraq ayrılmış maşınlar – qoşulmuş istifadəçilərin kompüterlərində yerləşən MP3 fayllarının axtarışını təmin edən "Napster" serverlərindən istifadə edirdi, fayl mübadiləsinin özü isə birbaşa baş verirdi. "Napster" mühitində dövriyyədə olan yazıların mühüm bir hissəsi müəllif haqları barədə qanun nəzərə alınmadan yayılırdı. Bir neçə günə "Napster" Amerika universitetlərinin tələbələri arasında böyük populyarlıq qazandı, lakin faylların azad mübadiləsi yarım ildən də az bir müddətdə başa çatdı.

7 dekabr tarixində Amerika Səsyazma Sənayesi Assosiasiyası (RIAA) "Napster" şirkətini "müəllif haqlarının birbaşa və bilavasitə pozulmasında" ittiham edərək məhkəməyə verdi. Bu xəbərlə "Napster" qəzetlərin birinci səhifəsinə çıxdı.

İddiaya baxılarkən vəkillər öz mövqelərini onunla müdafiə edirdilər ki, "Napster" serverlərində heç bir pirat

materialları saxlanmır. 2000-ci ilin aprelində "Metallica" qrupu "Napster"i onun müəlliflik hüququnu pozduğuna görə məhkəməyə verdi. Elə həmin vaxt repçi Andre Yanq (Andre Young – Dr.Dre) həm "Napster", həm də onun mahnılarının pirat yazılarını yayan bütün istifadəçilərə qarşı iddia ərizəsilə məhkəməyə müraciət etdi. "Napster" iddiaçıların sifarişilə IP-ünvanlarını müəyyən etmiş yarım milyondan çox istifadəçinin ("Metallica" sevənlər və "Dr.Dre" azarkeşləri) öz serverlərinə girişini dayandırmaq məcburiyyətində qaldı. Hiddətlənmiş istifadəçilər cavab olaraq iddiaçıların vəkillərinə şikayətlər göndərdilər. Təkcə bir həftə ərzində 30 mindən çox belə şikayət ərizəsi verildi.

Doğrudur, "Napster"in tərəfdarları da ortaya çıxdı: reperlər "Chuck D" və "Ice-T" bu sistemi "iyirmi birinci əsrin radiosu" adlandırdılar. O, musiqiçilərə birbaşa öz auditoriyaları ilə ünsiyyətdə olmaq imkanı verir ki, bu da onların yazılarının satış həcmini artırır.

12 iyun 2000-ci ildə RIAA məhkəmə qərarı çıxana qədər "Napster"i bağlamağı tələb etdi.

OXUMAQ, DİNLƏMƏK, TAMAŞA ETMƏK HAQQI

Əslində "Napster" RIAA-da düşündükləri kimi böyük təhlükə mənbəyi deyil. "Napster"in qadağan olunması İnternetdə faylların nəzarətsiz mübadiləsinin qarşısını almadı. "Napster"lə paralel olaraq mövcud olan onlarla "Gnutella" variantı və oxşar proqramlar orada öz bildiklərini edirlər. Bu səbəbdən "Napster" və Amerika səsyazma sənayesi nəhəngləri arasında münaqişə heç də sadəcə müəllif haqlarının pozulması problemindən ibarət deyil.

GNU layihəsinin atası Riçard Stolmen (Richard Stallman) "Napster" əhvalatından bir neçə il əvvəl eyni problem-

> "Məktəbdə birinci sinifdən şagirdlərə təlqin edirdilər ki, kitabını bir başqası ilə bölüşmək dəniz quldurluğu ilə müqayisə oluna biləcək dəhşətli cinayətdir. SPA-nın (Software Protection Authority – Proqram Təminatının Mühafizəsi Xidməti) ayıq-sayıq nəzərlərindən yayınmaq şansı praktik olaraq yox idi. Den bilirdi ki, hər kitabın onu kimin və harada oxuduğunu Lisenziyalaşdırma Mərkəzinə (Central Licensing) xəbər verən nəzarət monitoru var. Kompüter Şəbəkəyə daxil olan kimi, Mərkəz bunu həmən aşkar edə bilər, Den isə kompüterin sahibi kimi cinayətin qarşısını almadığına görə ən ağır cəza alacaq".
>
> *R.Stolmən. "Oxuma haqqı".*



lərin qaldırıldığı "Oxuma haqqı" adlı çox uğurlu fantastik hekayə yazmışdır.

"Napster" kimi sistemlərin mövcudluğu artıq danılmaz faktdır və aydındır ki, onları qadağan etmək qeyri-realdır. Əlbəttə, daha da mürəkkəb qoruma üsullarını təşkil etmək, ciddi qanunlar qəbul etmək olar, amma bunlar istənilən nəticəni verməyəcək. Böyük ehtimalla satış həcmləri aşağı düşəcək, qoruma isə sındırılacaq. Əgər bir avadanlıq vətəndaşların baxmaq, dinləmək və ya oxumaq azadlığına məhdudiyyət qoyursa,



Vinil qramofon valı, CD və mini-disk.

Bəzi universitetlər trafikin 20%-nin MP3 fayllardan ibarət olduğunu aşkar etdikdə "Napster"ə çıxışı bağladılar. Hiddətlənmiş tələbələr etiraz kampaniyası və "Napster"ə çıxışın bağlanması əleyhinə imza toplanmasını təşkil etdilər.

"Den bilirdi ki, onun ailəsi təhsil haqqını güclə ödəyir, kitab almağa pulları heç çatmır. Özgəsinin kitablarını oxumaq onun üçün təhsilini başa vurmağın yeganə yolu ola bilərdi. Məlum məsələ idi, lazımi məqalələri oxumaq üçün o özü borca girmişdi (bu məbləğlərin 10%-ini müəlliflər alırdı; Den özü akademik karyera barədə xəyallar qurduğundan ümid edirdi ki, onun tədqiqatlarına kifayət qədər çox istinad edəcəklər və o, borcunu qaytara biləcək).
Sonradan Den öyrəndi ki, vaxtilə hər kəs kitabxanaya gedib ödənişsiz jurnal məqaləsi və hətta kitab əldə edə bilirmiş. O vaxtlar hökumətin kitabxana qrantlarından istifadə etmədən minlərlə səhifə oxumaq imkanı olan azad tələbələr var idi".

Tezliklə analoji proqram – "Winamp" MP3 pleyerinin müəllifi "Nullsoft" şirkətinin buraxdığı "Gnutella" meydana gəldi. GNU idealogiyasına uyğun olaraq proqramla bərabər onun ilkin kodunun mətni də yayılırdı. "Napster"dən fərqli olaraq "Gnutella" serverlərə qoşulmadan məxsusi şəbəkələr yaradaraq bu şəbəkə daxilində faylları bir-birinə göndərmək imkanı verirdi. "Gnutella"dan istifadə edərək istənilən faylların mübadiləsini aparmaq olar. Bu proqramdan əlavə bir sıra silsilə məhsullar da buraxıldı.



Artıq bu gün oxumaq haqqı uğrunda döyüşlər gedir. Özü də mübarizə hansısa abstrakt azadlıqlar uğrunda deyil, bir neçə yüzillik ərzində formalaşmış mövcud qanun, ənənə və normaların İnternetlə bərabər meydana gəlmiş yeni reallığa uyğunlaşdırılması uğrunda gedir.

Göründüyü kimi, bu yeni şəraitdə səsyazma və video sənayesini idarə edənlər, həmçinin iri kitab naşirləri auditoriya ilə ünsiyyət üsullarını, kökündən dəyişməli olacaqlar. Bəlkə də müəllif və şirkətlər ifrat gəlirlərdən əl çəkəcək və onda bir CD-nin qiyməti onun şəbəkədə yayılmasının qiymətilə müqayisə oluna biləcək, bu isə nəticədə müəllifin qonorarını artıracaq. Axı piratlarla məhz bu şəkildə, qanunauyğun məhsulu qeyri-qanunidən aşağı qiymətə sataraq, mübarizə aparırlar.

belə avadanlıqdan daha istifadə edilmir. Belə çıxır ki, müəllif hüquqlarına malik olmaq XXI əsrdə ən sərfəli işdir. Amma əgər bir şəxs narazı qaldıqda həkimi məhkəməyə verə bilirsə, oxucunu qane etməyən kitabın müəllifini məhkəməyə vermək nədən qəbahət olsun. Axı müəllif öz faizini artıqlaması ilə alıb.

MUSİQİNİN PROQRAMLAŞDIRILMASI

Akkord – adətən tersiyalar özgə yerləşmiş üç və daha çox səsin eyni zamanda ifasıdır. Not yazısında tersiya əmələ gətirən notlar ya qonşu xətlər üzərində, ya da xətlər arasında yaxın aralıqlarda yerləşir, məsələn, do-mi-sol. Tersiya notlar arasında 1, 1/2 və ya 2 tonluq (kiçik və böyük) intervaldır.

Hələ 80-ci illərin əvvəlində populyar elektron klavişli alətlərin bəzi modelləri – elektroorqanlarda bəstəkara elektron səslər əldə etməkdən başqa həm də ifa etdiyi melodiyanın notlarını dərhal çap etmək imkanı verən çox rahat bir funksiya meydana gəldi. O vaxtlar bir çox bəstəkar üçün bu arzuların son həddi idi – qələm və kağızsız notlar!

Avtomüşayiətli ilk alətlər isə böyük səs-küyə səbəb olmuşdu. Artıq praktiki olaraq hər kəs əlavə musiqiçiləri cəlb etmədən şəxsi konsertlərilə çıxış edə bilərdi. Keçmiş SSRİ-də 80-ci illərin sonları avtomüşaiyətli ev sintezatoru olan populyar musiqi ifaçılarına tələbat çox idi. Burada pop-musiqi yazan bəstəkarların qıtlığının da müəyyən qədər rolu vardı. Bəsit melodiyalar fikirləşib ifa edən bir çox qrup və psevdobəstəkarlar meydana gəlmişdi. Onlar avtomüşayiətlə üç-dörd taktlıq melodiyalar bəstələyir və hitlərini uğurla satırdılar.

Yəqin ki, ölkəmizin kütləvi musiqi mədəniyyətində çatışmazlıqlar özünü göstərmişdi, axı uzun müddət pop-musiqi faktiki olaraq qadağan olunmuşdu.

Əsərin (fortepiano üçün) orkestrdəki alətlər üzrə "bölüşdürüldüyü" orkestrləmə texnikasından fərqli olaraq aranjeman zamanı bəstəkarın əsas musiqi mövzusuna kompozisiyanın janrı üçün xarakterik çalarlar əlavə oluna bilər.

Aranjeman istənilən musiqi stilində bəstəkara məhdudiyyətlər qoyan müəyyən qaydalar toplusu üzərində qurulub. Müxtəlif üslublar üçün ümumi qaydalar da mövcuddur, misal üçün, praktiki olaraq istənilən akkordun tersiya əsası var. Amma, məsələn, hard-rokda ikili basdan (əsas melodiyaya elektrogitara iki-səslilərinin əlavə olunması) istifadə edirlər. Bir qayda olaraq müxtəlif üslublar həm də müxtəlif ritmik şəklə malik olur. Məsələn, taktda vurğunun bir qədər irəli və ya geri sürüşdürülməsi əsərə xarakterik blyuz çaları verir. Bu sürüşmə həm bas, həm də zərb alətləri qrupunda ola bilər və bu, blyuzda növbəti musiqi ifadəsinin gözlənməsi hissini verərək onun təkrarolunmaz şəklini yaradır. Bununla bərabər melodiyanın xarakteri (major, minor) və əsərin tempi üslubun xüsusiyyətlərinə aid deyil.

Həm ənənəvi, həm də kompüter aranjemanında hər bir üsul xarakterik qəlib və aranjeman üsullarının toplusundan ibarətdir. Onlar geniş yayılmışdır və gələcək bəstəkar və aranjemançılar xüsusi musiqi təhsili alarkən bu üsulu öyrənirlər. Həmin üsullar qismən formaya salınaraq avtomüşayiətli sintezatorlar və xüsusi proqramlarda istifadə olunur.

Kompüter aranjemanı sahəsində ilk təcrübələr musiqi nəzəriyyəsi baxımından düzgün kompozisiyalar versə də, hələ çox sadə olub melodiyanın xarakterini nəzərə almırdı. Populyar proqramlardan biri olan "Band-In-a-Box" ("PG Music Inc" şirkətinə məxsus) musiqiçilərə rahat, sadə interfeys və midi-klaviaturadan istifadə etmədən, alt sırada ağ, növbəti sırada isə qara düymələri



Sintezator.

əvəz etməklə birbaşa kompüterin klaviaturasında ifa etmək imkanı təqdim edirdi. Orkestrləmə beş alətdən istifadə və hətta əsəri vokalistin səsilə montaj etmək imkanı verirdi. Hazır kompozisiya səslənərkən birgə ifa etmək imkanı, bundan başqa səhv ifa olunmuş notu ən yaxın düzgün variantla əvəz edən maraqlı (yəqin ki, yeni başlayan ifaçılar üçün) üsul var idi.

Daha mükəmməl proqramlar yazılan "sempl"lərdə (alət səsləri) melodiyanın yüksəkliyi, tempi, tezlik xüsusiyyətini yolüstü dəyişmək imkanı verirdi. Beləcə populyar bir musiqi parçasını eyni yüksəklikdə olan tonla, ancaq ritmik şəklini qorumaqla oxumaq, sonra isə bu



İndi çox vaxt anlayışlar yanlış şərh olunur və elektron musiqi çox geniş mənada, elektron alətlər vasitəsilə ifa edilmiş istənilən kompozisiya kimi başa düşülür. Əslində isə əgər bir musiqi əsərini səslənmədə nəzərəçarpacaq bir fərq olmadan "canlı" alətlərdə ifa etmək mümkündürsə, çətin ki onu elektron musiqiyə aid etmək olsun. O kompozisiyalar elektron adlanır ki, onları kompüter və elektron alətlərin köməyi olmadan yazmaq və ya səsləndirmək qeyri-mümkündür.

Müasir elektron musiqi studiyalarında nəinki alışdığımız alətlər, heç midi-klaviatura da yoxdur.

"səmpl"i yeni alətin səsi kimi istifadə edərək müəyyən bir üsul seçib melodiyanı ifa etmək olardı. Məsələn, reqqi stilində "Ay qız, kimin qızısan".

Hazırda, geniş çeşiddə "səmpl"lərin mövcudluğu şəraitində, ritmik havanı asanca dəyişərək müxtəlif ifaçıların praktiki olaraq istənilən kompozisiyasını istədiyin üslubda proqramlaşdırmaq mümkündür. Vokalistlə işləyərkən kompüter aranjemanı rahat deyil, çünki kompüter ifası ritmə çox ciddi riayət edir və müğənninin səsini dinləmir. Kompüterdə bir bəstəkarın "proqramlaşdıra" bildiyini digəri də bacarar. Doğrudur, aranjeman zamanı istifadə olunan materialların iri həcmi heyranedicidir və məxsusi bir "dəst-xətt" seçməklə hətta avtoaranjemanlı kompüterdə də təkrarolunmaz bir əsər yaratmaq olar.

Sintezator-fortepiano Yamaha 2000 pro.



Misal üçün, populyar "DJ Dance" proqramı ("Ejay" şirkətinə məxsus) faktiki olaraq kompozisiyanı 32-cığırlı mikşerdə yığmaq imkanı verir. Mütərəqqi rəqs musiqisi üçün "səmpl"lərin sayı on minlərlədir və bura şəxsi "səmpl"ləri də əlavə etmək mümkündür.

Cürbəcür effektlər, zərb-maşını və daha bir çox şeylər elə öz evinizdə xüsusi musiqi studiyası quraşdırmaq imkanı verir. Sadəcə azacıq musiqini düzgün qavrama qabiliyyəti, bir qədər də musiqi duyumuna malik olmaqla heç bir musiqi savadı olmadan belə təkrarolunmaz kompozisiyalar yaratmaq olar.

Hal-hazırda bir çox bəstəkarlar öz yaradıcılıqlarında kompüterlərdən fəal istifadə edir və "bəstəkar kompüterin köməyilə nə edə bilər?" sualına "praktiki olaraq hər şey" cavabını verirlər. Lakin qeyd edək ki, kompüter bəstəkar üçün fantaziya, yaradıcılıq və təxəyüldən başqa "hər şey" verir. Xoşbəxtlikdən (və ya təəssüf ki) hələ ki kompüter özünə bu və ya digər kompozisiya yazmaq tapşırığı vermir.

Bununla bərabər kompüteri yeni bir əsərin yaradılması zamanı bütün proses boyu istifadə etmək olar. Əvvəlcə, sadəcə bəstəkar üçün "yazı makinası" kimi, sonra isə, aranjeman zamanı, əsas mövzuya müxtəlif janrların "rəngi" qatılarkən kompüter kompozisiyaya faktiki olaraq ənənəvi üslublardan tutmuş elitar elektron musiqisinə qədər yeni səslənmə verir. Bundan sonra, maşın audiomontajı yerinə yetirir (məsələn, CD hazırlanarkən).

Əgər söhbət elektron kompozisiyadan gedirsə, montajı "kənarda" etmək bir qayda olaraq müəllif üçün ağlasığmazdır, çünki montaj elektron musiqinin ilkin səslənməsini dəyişə bilər.

Bu cür inkişaf etmiş hesablama texnikasının mövcudluğu şəraitində bəstəkarın studiyada deyil, öz evində işləmək imkanı olur. Yeni bir əsər yaradılarkən

müxtəlif alətlər və sintezatorda əvvəlcədən yerləşdirilmiş "səmpl"lər istifadə olunur, ya da kompüterə CD-ROM-dan və yaxud İnternetdən yeniləri yüklənir. Peşəkarların yüzlərlə CD-ROM-da yerləşən və saksofon, alt və digər alətlərin minlərlə müxtəlif səslənməsindən ibarət geniş "səmpl" bazası olur. Məsələn, kompozisiyaya lap Karnegi-hollda "Steinway" royalı kimi səslənən klavişli alətlər partiyası əlavə etmək mümkündür.

İndi bəstəkar səyahətə çıxarkən özü ilə midi-interfeysli noutbuk götürərək dünyanın istənilən yerində özü üçün studiya düzəldə, təyyarədə uçarkən nəyi isə əlavə edə, düzəldə və hətta yenidən yaza, sonra isə kompüterini səhnədəki avadanlığa qoşaraq konsertdə səsləndirə bilər.



Müasir bəstəkarın iş otağı.

YENİ DÖVRDƏ KİNO VƏ FOTO

RƏQƏMLİ AKTYORLAR

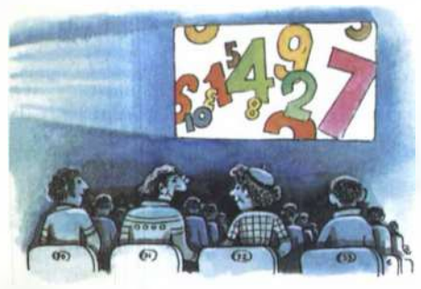
Məlum olduğu kimi, Hollivud "xəyallar fabriki" adlanır. Daim gəlir ardınca qaçdıqlarına görə Amerika kinostudiyalarını nə qədər pisləsələr də, etiraf etmək lazımdır ki, Amerika kinosu dünya kino bazarında təsadüfən liderlik etmir. ABŞ-da kino sənayesi dünyanın istənilən başqa yerindən daha yaxşı inkişaf edib.

80-ci illərin sonunda kompüter oyunları və kinofilmlərdən ibarət simbiozun meydana gəlməsi gözlənilirdi. Bir tərəfdən bu, insanın kompüter ekranında baş verən hadisələrə fəal müdaxilə edə bildiyi əvvəlki oyunlardan fərqlənmirdi. Digər tərəfdən isə nəzərdə tuturdular ki, bu yenə də əsl aktyorların oynadığı kinodur, sadəcə, praktiki olaraq hər epizodun bir neçə variantı var.

"Kvest" və "şuter"lər hamıya tanışdır, lakin EHM-in ekranında yüksək keyfiyyətli təsvir, bir də bütün bu mürəkkəb şəkildə budaqlanmış süjeti özündə saxlaya biləcək informasiya daşıyıcıları meydana gələnə qədər belə kinooyunlar barədə ancaq xəyal qurmaq olardı. Hər şey XX əsrin 90-cı illərinin ortalarında dəyişdi və kompakt-disklərdə ilk belə oyunlar çıxdı.

Aktyorlar kompüter ekranından oyunçularla və bir-birilə danışır, qaçır, atəş açır və hətta öldürdülər (kinoda isə bircə dəfə öldürlər). Beləliklə, yeni amplua – digital actor (*ing.* "rəqəmli aktyor") meydana gəldi (təbii ki, Hollivudda). Bu qəbildən olan ilk oyunlardan biri XX əsrin 90-cı illərinin ortalarında meydana çıxmış "Critical Path" ("Təhlükəli yol") idi.

Aktrisa Eylen Veysinger (Eileen Weisinger) oyunda öz ilk rolu barədə



DVD və videokaset.



deyir: "Bu rolumadək mən kompüterdən ancaq mətnlərin yazılması üçün istifadə etmişdim. İnteraktiv oyunda çəkiləcəyimi eşidəndə fikirləşdim ki, karate oyunlarındakı kimi, oyunçu mənim hərəkətlərimi idarə edəcək. Onlar mənə kompüterdə "Şerlok Holms" oyununun nümunəsində hər şeyi göstərdilər".

"Blue screen" (mavi ekran) texnologiyası həm kino, həm də oyunların çəkilişində eyni cür istifadə olunur. Aktyor səhnəni mavi ekran qarşısında oynayır, operator isə diqqətlə nəzarət edir ki, ekranda kölgələr əmələ gəlməsin. Aktyor maneələri görmədən, məsələn, əlilə səhv hərəkət edərsə, kompüter səhnəsində həmin yerdə divar yerləşə bilər və nəticədə qəhrəman onu dəlib deşər. Bundan sonra çəkilmiş material rəqəmləşdirilərək kompüterə daxil olunur, göy rəng ağla əvəz olunur və kinofilm və ya oyundan səhnələrlə doldurulur. Ənənəvi kinematoqrafiyada kombinasiya olunmuş çəkiliş səhnələrini emal edərkən çox vaxt kompüterin köməyi olmadan, qəhrəmanı mavi ekrandan "kəsib çıxardaraq" çəkilmiş materialın üzərinə köçürən avadanlıqdan istifadə edirlər. Ekranda göy (çox vaxt da yaşıl, xüsusən virtual televiziya studiyalarında) rəng ona görə istifadə olunur ki, insan təsvirinin özündə bu rəngə praktik olaraq rast gəlinmir və onu seçmək asandır.

Çəkiliş prosesində aktyorlar külli miqdarda dubldan başqa, çoxlu sayda bir-birinə bənzər səhnədə çəkilirlər, axı söhbət eyni bir epizodun müxtəlif sonluqlarından gedir. Aktyor oyun ərzində fərqli tərzdə olmaqla yüz dəfələrlə ölür. Rəqəmli aktyorların obraza girməsi daha çətindir, çünki süjet xətti yeganə deyil. Bundan əlavə, aktyor rolunu elə oynamalıdır ki, oynayanlar özlərini qəhrəmanla eyniləşdirmək istəsinlər.

Eylen Veysinger xatırlayır – "Qəhrəmanımın çox mürəkkəb xarakteri var. Mən onu mümkün qədər həqiqətə uyğun oynamaq istəyirdim ki, oyunçu onu həqiqətən xilas etmək istəsin. Ona bir qədər yumor da əlavə etdim".

Populyar aktyorların heç də hamısı interaktiv kino və ya oyunda çəkilməyə razı olmur. Bir çoxları yeni aktyor ixtisasında daha az yaradıcılıq elementinin olduğunu düşünür. Oyunlar ssenarinin çəkilişindən bir gün öncə hazırlanan, aktyorların isə öz personajlarının talelərindən çəkilişin sonuna qədər bixəbər olduqları serialları xatırladır.

Amma təxminən yüz il əvvəl çəkiliş zamanı dubllların sayının çoxluğuna görə filmlərdəki aktyor oyununu süni hesab edərək kinoaktyor peşəsinə də eyni cür münasibət bəsləyirdilər.

Faktiki olaraq ilk kompüter animasiya filmini Disney studiyası çəkmişdi və onun adı "Oyuncaq əhvalatı" idi. Bundan sonra "Qarışqa Anz" və "Böcəklərin həyatı" çıxdı. Ancaq bunlar multiplikasiya filmləri idi.

"Əlbəttə, təəssüflənirəm ki, oyunçular əsas qəhrəmanın kişi olmasını gözləyirdilər. Çəkilişlər zamanı mən oyunçuları ikicinsli varlıqlar kimi təsəvvür edirdim. Amma mən geyimimdən çox razıyam. Mənə yüngül topikli kombinezon verib çiynimdən avtomat asdılar. Bir dəfə mənə qısa şort geyindirmək istədilər, amma mən hesab etdim ki, qəhrəmanım quş deyil və razı olmadım. O, "Terminator"dan Sara Konnerlə fransız qızı Nikita arasında orta bir şeydir".

Aktrisa Eylen Veysingerin söhbətindən.

KOMPÜTER KİNEMATOQRAFİYASI

Yüz il ərzində filmlərin istehsalı və nümayişində əsas daşıyıcı kinolent olmuşdur. Lakin göründüyü kimi XXI əsrdə rəqəmsal kinematoqrafiya dövrünün başlanması ilə hər şey dəyişəcək. Məsələ



"Sonuncu fantaziya. Ruhlar" filmindən bir kadr.

bəzi kinematoqrafçıların artıq DV (digital video) formatlı mini-kameralara çəkiliş etməsində və materialın sonradan kinolentə köçürülərək kinoteatrlarda nümayiş olunmasında deyil. "Oyuncaq əhvalatı-2"nin istehsalının tamamilə rəqəmli olması və əksər hallarda nümayiş zamanı rəqəmli proyektorlardan istifadə olunması da səbəb deyil. XXI əsrdə cizgi filmi kimi deyil, əsl kinofilm kimi baxılan kompüter kinosu meydana gəldi. Burada rolları insanlar deyil, kompüterdə yaradılmış personajlar oynayır. İlk təcrübələrdən biri yapon rejissoru Hironobu Sakaquçinin (Hironobu Sakaguchi) 2001-ci ildə ekranlara çıxmış "Sonuncu fantaziya. Ruhlar" fantastik döyüş filmi oldu. "Sonuncu fantaziya..."nın personajları kompüterlərin köməyilə çəkilib, burada hətta üzün qırışlarına qədər hər şey həqiqətə bənzərdir. Təxəyyül məhsulu olan bir dünya xüsusi effektlər sahəsində ən yeni texnologiyaların köməyilə bir gerçəklik kimi canlandırılıb.

Filmdəki hadisələr uzaq 2065-ci ildə, insanların ruhlarını udan yad planetlilərin Yer kürəsini ələ keçirdiyi bir vaxtda cərəyan edir. Yerdə qalmış bəzi varlıqlar əcaiblərin işğalına müqavimət göstərə bilirlər, lakin tam qələbə çalmaq üçün "daxili ruhları" bir canlı varlıqda – doktor Aki Rossda birləşdirmək lazımdır. Bundan əlavə doktor və onun dostları Yer kürəsini sadəcə olaraq məhv etmək istəyən hərbçilərin qarşısını almalıdırlar.

Büdcəsi 150 milyon dollar olan filmin ekranlara çıxmasını animasiya filmlərinin azarkeşləri alqışlarla qarşıladılar. Bəzi qəhrəmanlar digərlərindən daha təbii alınmışdılar, amma hələ heç vaxt personajların modelləri canlı aktyorlara bu qədər oxşamamışdı. Filmə baxarkən unudursan ki, qarşındakı proqramçıların əsəridir. "Sonuncu fantaziya..." kinematoqrafiyada kompüterlə irəli sıçrayış idi.

Rəqəmli aktyorların meydana gəlməsi əsl peşəkarlarda böyük dəyişikliklərə səbəb olacaq. Aktyorlar Gildiyası

"Sonuncu fantaziya..."nın amerika versiyasını Alek Bolduin (Alec Baldwin), Donald Sazerlənd (Donald Sutherland), Ceyms Vuds (James Woods) (onun səsilə cani danışırdı) kimi ulduzlar səsləndirirdi. Kompüter artıq təsvirlərin öhdəsindən gəlir, amma eyni şeyi danışıq barədə demək olmaz. Burada hələ ki canlı aktyorlara ehtiyac duyulur.

"Sonuncu fantaziya..." filmi ekranlara çıxdıqdan bir qədər sonra qəhrəmanın vizual obrazı nümayiş etdirildi. Amma rəqəmli video vasitəsilə deyil, bilavasitə "nVidia Quadro DCC" qrafik prosessoru ilə təchiz olunmuş kompüterdə. Maşın real zamanda kadrları generasiya edirdi. Bu zaman təkcə Akinin saçlarını çəkmək etmək üçün prosessor bir saniyədə 1,5 milyon qrafik elementi emal edirdi ki, bu insan başındakı tüklərin sayından çoxdur.



"Sonuncu fantaziya. Ruhlar" filmindən bir kadr.

(Screen Actors Guild – SAG) ciddi olaraq narahatdır ki, tezliklə adi aktyorlar, xüsusən də statistlər, işsiz qala bilərlər. Rejissor Əndrü Nikol (Andrew Niccol) "Simona" filmində Al Paçinonun (Al Pachino) tərəf müqabili üçün rəqəmli aktrisa axtarırdı, belə ki, təklif olunan canlı aktrisalar onu qane etməmişdi. Sonradan, ssenarinin ideyasının rejissorun sorğusu ilə səhv salındığı məlum olduqda, səs-küy bir qədər səngidi. Baş qəhrəman, sadəcə, həqiqi olmamalı idi.

Əsil Hollivud ulduzlarının da acınacaqlı tale ilə üzləşəcəyi istisna olunmur. Belə ki, real aktyorun obrazı rəqəmli dublyorun yaradılması üçün istifadə oluna bilər. Məşhur aktyor Çarli Şin (Charlie Sheen) həmkarlarını, onları təqlid etmək istəyənlərdən qoruyacaq bir komitənin yaradılmasını təklif edib.

2002-ci ilin sonunda ekranlara növbəti kompüter filmi "Elizium" çıxdı. Ekranda baş verənlərin reallığı hissi elə böyükdür ki, bu animasiya filmini artıq cizgi filmi adlandırmaq yersizdir, hərçənd ki, müəlliflər əsl canlı ifaçıların rəqəmli aktyorla əvəz olunmasını qarşılarına məqsəd qoymamışdılar.

Müasir mobil telefonun yaddaş kartı.



Hər halda, ola bilər ki, ən yaxşı kişi və ya qadın roluna görə "Oskar"ı rəqəmli aktyorun alacağı gün çox uzaqda deyil. O zaman hamının arzuladığı heykəlciyi yəqin kompüterə verəcəklər.

AĞ-QARA FOTOQRAFİYADAN MONTAJ STUDİYALARINA

Rəqəmli texnologiyaları çox vaxt nahaq yerə canlı olmadığı üçün ittiham edirlər. XX əsrin foto rəssamları enli lentli analoq fotoaparatlardan istifadə edir, fotoşəklin süjeti, çəkiliş yeri və işıqlanmanı diqqətlə seçirdilər. Lenti aşkarlamaq üçün öz məxsusi reaktiv qarışıqlarını hazırlayırdılar. Çap prosesi isə xeyli vaxt aparırdı və xüsusi şərait tələb edirdi (qaranlıqda aparılırdı). Amma çəkilişdən sərgiyə qədər davam edən bütün bu zəncir müxtəlif dərəcədə bir yaradıcılıq elementi daşıyırdı. İndi isə texnologiya dəyişib.

Əməliyyatların əksəriyyəti kompüter vasitəsilə yerinə yetirilir, fotoqraf isə, sadəcə, süjet və çəkiliş obyektini təyin edir. Demək olar ki, qalan hər şeyi sənətkar deyil, kompüter edir. Ona ancaq birdəfəlik fotoaparatda kadrları "çıqqıldadıb" emal olunmaq üçün studiyaya vermək qalır. Hər şeyi maşın özü edəcək. Nəticədə alınan fotoşəkillərin keyfiyyəti heç də pis deyil. Köhnə texnologiya ilə zəmanətli nəticə almaq olduqca çətin idi.

Amma oxşar fotoaparatlar və köhnə texnologiyadan istifadə edən, rəqəmli fotoqrafiyada yaradıcılığın heçə endirildiyini hesab edən xeyli sayda sənətkar qalıb.

Əslində isə yeni texnologiya sənətkara daha geniş yaradıcılıq imkanları verir. Sadəcə olaraq qaranlıq otaq, kimyəvi məhlullar və fotoböyüdücünün yerini kompüter və görüntülərin emalı üçün proqramlar tutub. Rəssamın molbert və palitra, yoxsa kompüter və printerdən istifadə etməsi əsas deyil.

AVTOMOBİL VƏ İNFORMASİYA TEXNOLOGİYALARI

Bəzi proqnozlara görə daxiliyanma mühərriki öz dövrünü tezliklə başa vuracaq. Belə ki, bir neçə onillikdən sonra ABŞ-da irimiqyaslı "Azadlıq Avtomobili" layihəsinin həyata keçirilməsi nəticəsində qabaqcıl korporasiyalar "Daimler Chrysler", "General Motors" və "Ford" benzin motoru əvəzinə elektrik mühərrikindən istifadə edən avtomobil hazırlayıb onun kütləvi istehsalını təşkil etməli olacaqlar. Hidrogenin oksidləşməsi hesabına işləyən motorları quraşdırmaq nəzərdə tutulur. Bununla avtomobillərin ekoloji təmizliyi təmin olunacaq. Bundan başqa neft istehlakı və deməli, onun idxalından dövlətin asılılığı xeyli dərəcədə aşağı düşəcək.

İnformasiya texnologiyaları avtomobildə güc qurğusunun dəyişməsinə bilavasitə təsir edir. Əsas dəyişikliklər elektron rabitə qurğularının hər yerdə nəqliyyat vasitələrinə daxil edilməsilə bağlıdır.

"Daimler Chrysler" şirkəti nəinki sürücüyə yolda olan tıxaclar, bu və ya digər məntəqəyə qədər məsafə barədə məlumat verən, hətta İnternetdən məktub almaq və saytları nəzərdən keçirmək imkanı verən avadanlığa malik konseptual maşın hazırlamışdır. Sürücünün qarşısındakı paneldə səslə idarə olunan kompüter, sərnişinlər üçün isə sensor ekranına malik iki terminal yerləşdirilib. Səfər zamanı həm sürücünün, həm də sərnişinlərin saytlardakı məlumatlarla maraqlanacağı güman edilir.

"Daimler Chrysler"dən başqa Amerika avtomobil sənayesinin digər iki nəhəngi "General Motors" və "Ford" avtomobillərdə simsiz rabitə və onlayn tətbiq etmək üçün xidmət sahəsində tədqiqatlar aparır. Mütəxəssislərin rəyinə görə belə inteqrasiya olunmuş informasiya emalı və ötürülməsi sistemləri avtomobil sahiblərinin maşınlara marağını daha da artıracaq.

Rabitə ancaq ətraf aləmlə deyil, həm də avtomobil daxilində təmin oluna bilər. Belə ki, əgər mərkəzi blokda işıqlandırma cihazları üçün kodlaşdırılmış siqnal istifadə edilərsə və o, məsələn, dekodlaşdırıcı blokun yerləşdirildiyi arxa faralara bir naqillə göndərilərsə, maşının yük yerinə gedən qalın məftil dəstəsinin əvəzinə ancaq iki məftil lazım gələcək. Siqnal naqili boyu idarəedici impulslar ötürüləcək, güc naqili vasitəsilə isə fara və dekodlaşdırıcı blok qidalandırılacaq (bir qayda olaraq, avtomobildə mənfi naqil maşının gövdəsinə birləşdirilir, buna görə də üçüncü məftilə ehtiyac qalmır). Qısa impulslardan ibarət siqnallar faralarda uyğun lampanın yandırılması və söndürülməsini kodlaşdıracaq (üç impuls – stop-siqnalı yandır, dörd impuls – stop-siqnalı söndür, beş impuls – sola dönmə işığını yandır, altı – söndür və s.).

İnformasiya texnologiyalarının digər tətbiq sahəsi isə mühərrik və sürət ötürücüsünü idarəetmə sistemləri, bort hesablayıcıları – məlumat kompüterləri, avtomobilin elektron qoruma sistemləri və digərləridir.

Avtomobil qurğularının idarə olunması və diaqnostikası sistemləri hələ XX əsrdə meydana gəlmiş və "injek-

Proqnozlara görə 2010-cu ildə avtomobillərdə informasiya sistemlərindən gələn gəlir XXI əsrin əvvəlinə bu sahənin bütün dünya üzrə 4 milyard dollarlıq həcmindən ildə 50 milyard ABŞ dolları səviyyəsinə yüksələcək.

Amerika Birləşmiş Ştatları dünyada hasil olunan neftin 25%-ni istehlak edir.

Müasir "Ford" avtomobilinin salonu.





torlu" adlandırılan motorlarda geniş yayılmışdır. Avtomobil sahiblərindən bəzən maşında kompüterin "xarab olması" barədə şikayətlər eşitmək olar. Əslində bu, bir qayda olaraq, avtomobilin ən etibarlı hissəsidir. XX əsrdə motora yanacaq verilməsinin idarə olunması sistemlərində element bazası kimi mikrosxemlərdən deyil, saysız-hesabsız elementlərdən: tranzistorlardan, muqavimətlərdən və b. istifadə olunurdu. Çox vaxt kompüterlərin sıradan çıxması avtomobillərin istismarı şəraitilə bağlıdır, axı temperatur fərqi -50°C-dən +50°C-yə qədər ola bilər. Kompüterin bu cür quruluşu qənaətbəxş deyil, çünki yenidən proqramlaşdırma üçün çox vaxt elementlərin dəyişdirilməsilə blokun yenidən qurulması lazım gəlir ki, burada lehim aləti olmadan keçinmək olmur. İnteqral sxemlər istifadə olunduqda xüsusilə alışdırmanı idarə edən və iş prosesində ötmə bucağına düzəliş verən proqram bütövlüklə idarəedici prosessorun yaddaşında yerləşirdi və minlərlə verilənləri nəzərə alırdı. Yeni motor üçün belə bir blokun hazırlanması zamanı bütün blokun yenidən layihələndirilməsi lazım gəlmirdi, eyni zamanda blok qalırdı, sadəcə proqram və ya onun verilənləri dəyişdirilirdi. Beləliklə, mühərrik istehsalına maliyyə sərfi azaldılırdı.

Yuxarıda deyilənlərin hamısı digər sistemlərə də, məsələn, avtomatik ötürücü qutusunu, kondisionerə, tam ötürücü avtomobillərin transmissiyasına aiddir.

Motorun vəziyyətini də kompüter vasitəsilə qiymətləndirmək olar. Yağın mütəmadi dəyişdirilməsi üçün avtomobil sahibinə odometrin son göstəricilərini yadda saxlamaq lazım gəlməyəcək, kompüter özü bu barədə əvvəlcədən xəbərdarlıq edərək texniki xidmət stansiyasına baş çəkməyə ehtiyac olduğunu xəbər verəcək. Əgər istismar şərtləri ağır olubsa, kompüter xüsusi yağ sensorlarının göstəricisinə əsasən növbədən-



Konsept-kar – "Daimler Chrysler"in hazırladığı uzaq gələcəyin maşın nümunəsi.



"Lada" konsept-karı. "Avtosalon 99". Moskva.

kənar əvəzləmə barədə qabaqcadan xəbər verə bilər. Doğrudur, XX əsrin salon kompüterləri o qədər də informativ deyildi. Onlar, bir qayda olaraq, benzin sərfini göstərir və müəyyən məsafəni qət etmək üçün nə qədər yanacaq tələb olunduğunu qiymətləndirirdilər.

Bir avtomobilçi üçün təyinat nöqtəsinə mümkün yollar barədə məlumat verən marşrut kompüteri daha vacibdir. Bu zaman avtomobilin öz vəziyyətini bilmək faydalıdır, bu isə GPS (*ing.* Global Position System) qlobal mövqe sistemi vasitəsilə müəyyən edilir. Şəxsi elektron şturman geostasionar orbitlərdə yerləşən 24 peykin köməyilə səfər və səyahətlərdə öz yerini və hərəkət sürətini təyin etmək, marşrutu planlaşdırmaq imkanı verir. Proqram təminatı adətən xəritələrin fonunda avtomobilin mövqeyini, nəzərdə tutulan hərəkət marşrutunu, mehmanxanalara və yanacaqdoldurma məntəqələrinə qədər məsafəni və s. göstərən kiçik məlumat sistemindən ibarət olur. Belə sistemlər verilənlər bazasında küçəni, mehmanxananı axtarıb taparaq lazım olan yerə marşrutu avtomatik cızır.

Bənzər bir sistemin işləməsi üçün avtomobildə quraşdırılmış kompüter və ya noutbukun olması zəruri deyil. PDA üçün on minlərlə ünvanlıq verilənlər bazası olan optimal hərəkət marşrutunun seçilməsi, istənilən obyektin axtarışı və barəsində müfəssəl məlumatın alınmasını təmin edən elektron xəritədən ibarət mini-sistemlər mövcuddur. Belə sistemi şəhərdə piyada gəzişmək üçün avtomobildən çıxarkən GPS-qəbuledicilə bərabər özünüzlə götürə bilərsiniz.

ABŞ-ın əksər yollarının təsvirini daşıyan noutbuka qoşulan GPS qəbuledicili geoinformasiya sistemi dörd CD-ROM diskində yerləşir.



GPS-naviqator.

ELEKTRON EV



"Hər şeyi xatırlamalı" Amerika fantastik filmində baş qəhrəmanın həyat yoldaşı mənzili səslə idarə edir: işığı və televizoru yandırıb-söndürür, holoqram tennis məşqçisinin rəhbərliyilə məşq edir. "Beşinci element" kinofilmində Brüs Uillisin (Bruce Willis) qəhrəmanı sadə yaşayır, lakin elektron poçt və videotelefondan istifadə edir.

Elektron və ya çox vaxt deyildiyi kimi "rəqəmsal" ev nədir? Bir çoxları bu texniki möcüzə barədə bədii əsərlərdən eşidib. Hər bir yazıçı-fantast (və ya rejissor) gələcək barədə əsər (film) yaradarkən onların məişəti barədə söhbət açmağı özünə borc bilir. Özü də bu məişət rahatlığı və mükəmməlliyilə oxucunu (tamaşaçını) heyran etməlidir.

Elektron evdə mütləq rabitə vasitəsi – adətən, videotelefon və televizorun qarışığı olan bir şey olmalıdır. Burada təhlükəsizlik sistemi işləyir: ev sahibini, məsələn, barmaq izlərinə görə "tanıyır". Evdəki avadanlıq səs komandalarını icra edir. Və əlbəttə ki, fantastik əsərlərdə qorxulu epizodlar da göstərilir: kompüterlər sahiblərinin nəzarətindən çıxır və mühafizə sistemlərinin köməyilə insanları girov götürür.

Bəs reallıqda nə baş verir? 2001-ci il yanvarın əvvəlləri Las-Veqasda (ABŞ) keçirilən növbəti "CES 2001" məişət elektronikası sərgisində iştirakçıların əsas diqqəti rəqəmsal ev konsepsiyasına yönəlmişdi. Bu konsepsiyanın məğzi ondan ibarət idi ki, kompüterlər, televizorlar, musiqi mərkəzləri, soyuducu, mikrodalğalı sobalar və bütün digər ev cihazları ümumi, etibarlı, cəld işləyən bir sistemdə birləşməlidir.

Mikroprosessor istehsalçılarının lideri – "Intel" şirkəti məişət cihazlarını güclü fərdi ev kompüterlərinin ətrafında birləşdirmək təklifi vermişdir. "Intel" öz texnologiyasını "yeni imkanlar erası" (Extended PC Era) adlandırıb. Elektronikanın inkişafı ilə məişət cihazları fərdi kompüterlərlə bərabər vahid bir mühit yaradacaq ki, bununla həm evdəki məişət əşyaları, həm də fərdi kompüterin imkanları genişlənəcək. EHM məişət texnikasını effektiv idarə edəcək. Artıq bir çox cihazların proqram təminatını yeniləmək çətin deyil. Məsələn, paltaryuyan maşınların bəzi modellərində yeni proqramlar əlavə etmək və köhnələri düzəltmək mümkündür (xüsusən əgər kütləvi istifadə prosesində proqramda səhvlər aşkar edilibsə).



Məşhur yazıçı fantastlar Struqatski qardaşları əsərlərində gələcək dünyanın xarakterçə müxtəlif tablolarını çəkiblər. "Bazar ertəsi şənbə günündən başlayır" povestində onlar yumoristik tərzdə cadu və sehrbazlıq institutunu təsvir ediblər. "Stajçılar" və "XXII əsr günorta" əsərlərində isə Yer kürəsi sakinlərinin daha yaxşı dünya qurmaq uğrunda ümumi ideyaya uyğunlaşdırılmış məişəti təsvir olunub.

SEHRLİ SOBA

Söhbət evdə ən sevilən otaq olan mətbəxdən gedəcək. Ana və bacılarımızın ağır əməyini asanlaşdırmaq üçün alimlər əllərindən gələni əsirgəməyiblər. Bunlar həm hamıya tanış olan soyuducu və sobalar, həm də hələ ki, evlərimizdə nadir rast gəlinən qabyuyan maşın və mikrodalğalı sobalardır. Elə bu sobalar barədə də danışacağıq. Müasir mikrodalğalı sobalar müstəqil şəkildə bir çox xörəkləri bişirə bilir, sadəcə, proqramı seçib düyməni basmaq lazımdır. Bir sözlə, hər şey kompüterləşdirilib, daha nə lazımdır?

Lakin bir anlığa təsəvvür edək ki, sobamız ümumdünya şəbəkəsinə qoşulub və supermarket kassirin çek yazarkən məmulatların qiymətini öyrənmək üçün istifadə etdiyi lazer skanerinə bənzər skaner ilə təchiz olunub. Beləliklə, evdə ərzağı skanerdən keçirərək soyuducuya qoyduqda kompüterləşdirilmiş köməkçimiz həmin məmulatın adını, çəkisini və saxlama müddətini (bağlı və açıq vəziyyətdə) bilir və bütün bunları yadda saxlayır. Bax belə, soyuducumuz doludur. İndi yemək bişirməyə başlayaq.

Yeməyi seçək – soba aşpaz kitabındakı istənilən xörəyi bişirə bilər! Bu zaman sobanın şüşəsinə – kompüterin xörəyin təsvirini göstərən rəngli maye-kristal ekranına baxırıq. Yaxud "Sonra, sonra..." deyərək yeməklərin uzun siyahısını "vərəqləyirik". Ya da ki, mikrodalğalı ilə söhbətlərimizlə evdəkilərin fikrini yayındırmamaq üçün köhnə qaydada düymələri basırıq. Xırdaçılığa görə üzr istəyirik, gəlin "noxudlu küftə" bişirək. Deməli, "küftə" – "k" hərfini axtarırıq, orada isə "noxudlu"nu tapırıq, əyanilik üçün fotoşəklə baxırıq, sonra isə resepti seçirik – bundan bu qədər, ondan o qədər... Aha, bu nə sehrdir, soba reseptdəki bütün komponentlərin soyuducuda olduğunu bilir. Lazım olan ərzağı çıxarıb bişiririk. Köməkçimiz yatmır, cınq, sobadan ismarıc var: "Yadınıza salıram ki, çoxdandır soyuducuda olan süd sabah xarab olacaq, saxlama müddəti bitmək üzrədir". Nə isə etmək lazımdır – ya atmaq, ya da içmək. Südü bir stəkana süzürük. Boş qutunu skanerə yaxınlaşdırırıq. Deyirik: "Boşdur". Köməkçi onu siyahıdan silir. Hələ açılmamış paketi götürürük. Yenə skanerin yanından keçiririk. Soba deyir: "Saxlama müddəti – dörd ay". Paketi açırıq və yenə də skanerə yaxınlaşdırırıq. Deyirik: "Açdıq". Cavabında köməkçi deyir: "İstifadə müddəti soyuducuda iki həftədir".

Budur küftə də hazır oldu. Dadına baxırıq. Ləzzətlidir.



İndi isə başqa bir resept. Yeri gəlmişkən, İnternetdə nə isə ekzotik bir şey axtaraq. Oralarda nə tapmaq olar? Bəs bunun üçün soyuducumuzda nə var? Heç onu açmayacağıq, əvəzinə soba ilə danışıb ekrandakı siyahını vərəqləyərik. Ərzaqları seçib sobadan soruşuruq ki, bunlardan nə bişirmək olar. "Aha, bu da siyahı... uzundur... Bəs bundan? Heç nə alınmır, heyf, bəlkə mağazaya gedək? Bir-iki şey alaq?" Hə, indi olar. Resepti seçirik və soba özü ən yaxındakı supermarketdən çatışmayan ərzağı sifariş edir, əlbəttə ki, İnternetlə. Bu da kuryer. Soruşur: "Sifariş etmişdiniz?". "Bəli" deyə cavab veririk. Bişirməyə başlayırıq!

Sehrli mikrodalğalı soba xörəyi bişirən müddətdə isə onun ekranında İnternetdən yayımlanan kinofilmə baxırıq. Lap televizor ekranındakı kimi. Sadəcə, fasilələrdəki reklamlar mikrodalğalı sobanı istehsal edən şirkətindir, irəli fırlatmaq da olmur. Bir az vaxt keçdi və dadlı nahar masanın üzərindədir. Nahar etdik. Çirkli qabları qabyuyana qoyduq. Birdən o da İnternetə qoşulmuş olar, ha?



Microsoft korporasiyası yeni ev idarəetmə kompüterinin konsepsiyasında EHM-in hasablama imkanlarını özözlüyündə hesablayıcı gücü olmayan məişət cihazlarını idarə etmək üçün istifadə etməyi təklif etmişdir. İnsanlar sensor panelləri vasitəsilə kompüteri istənilən yerdən idarə edə biləcək.

EHM qrupları bir qayda olaraq naqillərin köməyilə şəbəkələrdə birləşdirilir. Amma bugünkü gündə, xüsusən də lokal şəbəkələrdə, bir və ya bir neçə



MDU-nun əsas binasının antena quraşdırılmış qülləsi

otaq daxilində daha çox naqilsiz birləşmələrdən istifadə olunur. Hətta belə fərziyyə də irəli sürülür ki, qədəm qoyduğumuz onillikdə belə texnologiya mühüm hesab olunacaq. Böyük ehtimalla naqilsiz rabitə vasitələri bütün ev cihazlarının daxil olduğu ev şəbəkəsinin işində əsas yeri tutacaq.

Elektron evi PDA (elektron yazı kitabçası), İnternet və hətta mobil telefon vasitəsilə istənilən yerdən idarə etmək mümkün olacaq. Daxili iqlimin (otaqların hərarətini) tənzimlənməsi (məsələn, ev sahibləri məzuniyyətə getdikdə) xeyli pula qənaət etmək imkanı verəcək. Əgər evdə heyvanlar və istiliksevən bitkilər yoxdursa, o zaman otaqlarda hərarəti adi vəziyyətdə saxlamağa ehtiyac da yoxdur. Ev sahibləri qayıdandan bir qədər əvvəl isə otaqların hərarəti avtomatik olaraq normaya gətiriləcək.

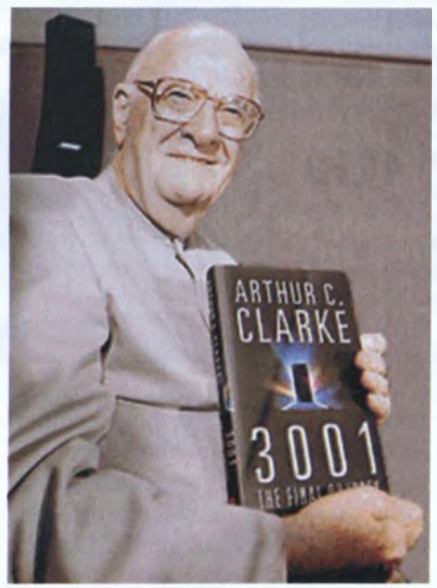
Əgər məzuniyyətdə rəqəmli video və fotokamera ilə çəkiliş aparmaq istəsəniz, material İnternetlə ev kompüterinə ötürüləcək, odur ki, yazıları saxlamaq üçün özünüzlə bir yığın yaddaş kartı daşımağa ehtiyac qalmayacaq. Çox güman ki, avtomobildəki MP3-pleyer və maqnitolanı da eyni aqibət gözləyir.

Artıq Avropada rəqəmsal evlər satışdadır, lakin onlar hələ ki müasir həyatın atributundan daha çox gələcəyin gözəl simvolu olaraq qalır. Müasir elektron evin satıcılar tərəfindən reklam olunan əsas imkanlarından biri otaqlardakı işığın məsafədən yandırılıb söndürülməsi və pərdələrin idarə olunmasıdır (bu yolla ev sahibləri potensial oğrulara guya evdə olduqlarını bildirirlər). Bu üstünlüklərə, əlbəttə ki, qonaq otağında onlayn alışveriş imkanına malik 50 düymlu plazma televizor da daxildir.

Lakin hər şey sürətlə dəyişir. 2002-ci ildə ABŞ-da informasiya texnologiyalarının məşhur entuziastı Cozef Ceykobs (Joseph Jacobs) kompüteri daha yaxşı "anlamaq" üçün dərisinin altına mikroçip yerləşdirib. İstisna deyil ki, bu cür texnologiya tezliklə insanlara rəqəmsal evi idarə etməyə imkan verəcək.

YAXIN YÜZİLLİK ÜÇÜN PROQNOZLAR

Hazırda eyni məsələləri adi avtomobil siqnalizasiyaları həll edir. Onlar sürücüyə məsafədən motoru işə salmaq, salonu qızdırmaq, maşındakı hərarətə nəzarət etmək imkanı verir. Axı, çöldə 20 dərəcə şaxta olarkən, evdən çıxıb isti avtomobilə oturmaq adama ləzzət edir.



Artur Klark öz yeni kitabının təqdimatında.

İnsan hər zaman onu gələcəkdə nə gözləyəcəyini öyrənməyə can atır. O, ulduz və kart falı ilə suallarına cavab tapacağına ümid edir. Çox vaxt yazıçı və alimlər gələcəkdən xəbər verən kahinlər qismində çıxış edirlər.

Artur Klarkın (Arthur Clarke) adı hətta elmi fantastika ilə maraqlanmayanlara da tanışdır. Ona bu şan-şöhrəti yalnız çoxsaylı romanları və məşhur nəşrlərin səhifələrindəki çıxışları deyil, həm də Klarkın ssenarisi əsasında çəkilmiş Stenli Kubrikin (Stanley Kubrick) "2001-ci il: Kosmik odisseya" filmi gətirmişdir. Yazıçı onu peyğəmbər adlandırmamağı xahiş edir, hərçənd ki, onun əvvəlcədən söylədiklərinin əksəriyyəti özünü doğrultmuşdur. Bu da onun gələcək üçün verdiyi proqnozlardan bəziləri:

...XXI əsrin 10-cu illəri. Elektron monitorinq demək olar ki, peşəkar cinayətkarlığı cəmiyyətdən sıxışdırıb çıxarır.

...XXI əsrin 20-30-cu illəri. Süni intellekt insan beyninin inkişafı səviyyəsinə gəlib çatıb. Həmin andan etibarən Yer üzündə iki növ şüurlu həyat forması birgə yaşamağa başlayır. Bu zaman nəzərə almaq lazımdır ki, ikincisi bioloji təkamül üçün əlçatmaz olan sürətlə inkişaf edir. Süni intellekt sistemilə təchiz edilmiş ilk gəmilər ən yaxın ulduzlara yola düşür. Kompüterin generasiya etdiyi DNT-dən klonlaşdırılmış dinozavrlar meydana gəlir.

...XXI əsrin 40-50-ci illəri. Avtonom, regenerasiya olunan, mobil ev tamamilə təkmilləşdirilmişdir... Nanotexnologiyalara əsaslanan "universal replikator" xammal və informasiya matrisinin mövcud olması şərtilə istənilən mürəkkəb obyekti yarada bilir. Brilyant və delikates yeməklər sözün hərfi mənasında palçıqdan hazırlana bilər. Belə lüzumsuzluq üzündən sənaye və kənd təsərrüfatı, eyni zamanda bəşər sivilizasiyasının təzə icadı – iş yerləri də yoxa çıxır! İncəsənət, əyləncə, təhsilin inkişafında partlayışlar olur. Planetin artıq ərzaq istehsalı üçün lazım olmayan geniş sahələri ilkin vəziyyətinə qayıdır, gənclər ox-yayla böyük ov yürüşlərində iştirak etməklə öz aqressiv instinktlərini biruzə verə bilirlər.

* * *

"TeleDanmark" şirkətinin inkişaf şöbəsinin rəhbəri Preben Meyer (Preben Meier) müsahibələrinin birində "düşünən texnikanın" evlərdə sürətlə yayılacağını proqnozlaşdırır. O əmindir ki, kompüterlər yaxın 20-25 il ərzində praktiki olaraq evdəki bütün işlərin qeydinə qalacaq: paltaryuyan maşınlar özləri yuyulma rejimini təyin edəcək, mikrodalğalı sobalar yemək bişirəcək, soyuducular ərzaq sifariş edəcək. Bundan əlavə, avtomobillər özləri texniki baxışın vaxtını əvvəlcədən təyin edə biləcək, ağır elektron aparatlar isə mənzillərin divarlarında kağız kimi yüngül ekranlarla əvəz ediləcək.



Müasir avtomobilin panelində təcili yardımı çağıran düymə.

* * *

900-dən artıq alim arasında sorğu aparmış alman fiziki Harold Qruppun proqnozlarına görə, XXI əsrdən başlayaraq dam və fasadlarda günəş batareyaları ilə təchiz olunmuş evlərin fəal şəkildə tikintisinə başlanacaq. Tərəqqi o səviyyəyə çatacaq ki, bütün ev işlərini kompüterlər görəcək, hətta zibil yeşiyi belə tullantıları özü çeşidləyəcək.

Artıq XXI əsrin ilk onilliyində əməliyyatların cərrah bıçağı ilə aparılmasına birdəfəlik son qoyulacaq. Cərrah-mikrorobot insanın istənilən orqanına daxil olub zəruri əməliyyatları aparacaq: trombları aradan qaldıracaq, dərmanın mikrodozalarını daxil edəcək... Mini-kompüter kimyaçı-eksperti əvəz edərək istənilən maddə barədə tam məlumat verəcək. Daha on il sonra isə Avropa vahid hökümət və vahid kompüter idarəetmə sisteminə malik birləşmiş ölkə olacaq. Ordu əsasən kompüterçilərdən ibarət olacaq... İkinci dekadanın sonunda isə Yer üzündə ilk kiborqlar – kompüterlə calaşdırılmış insanlar meydana gələcək. Bu nəinki elm və texnikada bir çevriliş olacaq, həm də yeni mənəvi problemlər yaradaraq cəmiyyətdə mübahisələr və dərin ziddiyyətlərə gətirib çıxaracaq.

* * *

Serebral iflic xəstəliyinə mübtəla olmuş, bəzilərinin ikinci Eynşteyn adlandırdığı Stiven Hokinq (Stephen Hawking) "Zamanın qısa tarixi" kitabında yazır: "Bəşəriyyət get-gedə mürəkkəbləşən dünyanın öhdəsindən gəlmək və yeni, xüsusən də kosmik səyahətlərlə bağlı problemləri həll etmək üçün əqli və fiziki səviyyəsini yüksəltməlidir". Hokinqin fikrincə insan "elektron kəllələrdən öndə

qaçmaq" üçün getdikcə daha da kamilləşmək məcburiyyətindədir. "Halhazırda kompüterlərin sürət üstünlüyü var, amma onların şüuru yoxdur". Buna baxmayaraq "kompüterlərin sürət və mürəkkəbliyinin 18 aydan bir ikiqat artdığını əsas götürən Stiven Hokinq əmindir ki, gələcəkdə onlar insan beyninin mürəkkəbliyi səviyyəsinə çata biləcəklər". Bundan başqa onun proqnozlarına görə tarixdə yeni era – planlaşdırılan təkamül dövrü başlayacaq. Embrional mərhələdə insanın inkişafındakı patoloji çatışmazlıqları təyin edib aradan qaldırmaq mümkün olacaq və nəticədə gələcəyin insanları sağlam olub 120 ilə qədər yaşayacaqlar.

* * *

Yapon alimlərinin proqnozlarına görə yeni əsrin ilk onilliyində özündə faks və video imkanlarını birləşdirən cib kompüteri buraxılacaq. Kompüterlər bir anda nəinki mətnləri, hətta danışığı dünyanın istənilən dilinə tərcümə edə biləcək. Kompüterlərin köməyilə süni qan sintez olunacaq, bəşəriyyət QİÇS (AIDS) və ateroskleroza qalib gələcək. Daha sonra insanlar süni göz və beyindən istifadə edə biləcəklər.

* * *

"Forbes ASAP" jurnalının proqnozlarına görə yaxın gələcəyin kompüteri "qoruyucu siqnalizasiyadan tutmuş soyuducuya qədər hər şeyi" işlədən bir qurğu olacaq. Kompüter sənayesinin əsası olan silisium (silikon) artıq XXI əsrin 10-cu illəri öz ölçüləri, kiçik sürəti və işləyərkən ayrılan istiliyin çoxluğuna görə kompüter istifadəçilərinin ehtiyaclarına cavab verməyəcək.

Hazırda Stanford Universitetinin əməkdaşları optik rabitənin sürəti və keçiriciliyindən istifadə edən optoelektron qurğu üzərində çalışırlar. Yeni texnika istehsalçılarının sözlərinə görə nəticədə kompüter ucuz, yığcam, daha məhsuldar olub daha az enerji sərf edəcək. Onun köməyilə evdəki bütün cihazları idarə etmək, işdə isə onu böyük interaktiv ekran kimi istifadə etmək mümkün olacaq. Kompüter əsl sahibini misal üçün, barmaq izlərinə görə tanıyacaq. Belə kompüteri idarə etmək çox asandır. Bunun üçün onunla sadəcə danışmaq kifayətdir. Klaviaturaya öyrəşmiş istifadəçilər üçün o, virtual şəkildə təqdim ediləcək və barmaqla birbaşa olaraq ekranda düymələri basmaq mümkün olacaq. Belə kompüterin yaddaşı holoqrafikdir və istənilən qədər böyük həcmli informasiyanı saxlamaq imkanına malikdir. Prosessor və yaddaş arasında verilənlərin mübadilə sürəti prosessorun sürətilə müqayisə oluna biləcək.

* * *

İstedadlı futuroloq Viktor Fellerin (Victor Feller) proqnozlarına görə ən yaxın zamanda tibbə yeni nəsil kompüter texnologiyalarının tətbiqi baş verəcək. Bunun nəticəsində biotexnologiyalar inkişaf edəcək, süni orqanlar – ürək, qara ciyər, böyrəklər meydana gələcək, virusları tapıb məhv etmək üçün orqanizmə yeridilən kompüter sistemləri yaradılacaq.

* * *

Eyndhovendəki (Niderland) "Philips" ekspert mərkəzinin "Gələcəyə baxış" layihəsinin iştirakçısı Derrik de Kerkhouv (Derrik de Kerckhove) deyir: "Hesab edirəm ki, gələcəkdə əl hərəkətlərilə idarə olunan çoxlu sayda cihazımız olacaq. Əlinizlə işarə verdikdə dərhal nəyinsə baş verməsi zamanı yaranan hissdən daha böyük hakimiyyət xülyası yoxdur. İnsanlar belədir. Səsin tanınması bu gün mövcuddur və təkmilləşdirilməkdədir. Yüksək keyfiyyətli tanıma sistemləri istifadə etdiyimiz hər bir maşında, hətta velosipeddə belə, quraşdırılacaq. Düşünürəm ki, bu mənada imza da yenidən layiq olduğu yeri tutaraq sizin şəxsi iştirakınıza zəmanət verəcək. Mənə elə gəlir ki, əlyazmanı çap hərflərinə çevirən daha yaxşı üsullar meydana gələcək".

Futuroloqun fikrincə yaxın gələcəkdə insanlar əsasən virtual olaraq səyahət edəcəklər, axı bu xeyli ucuz və fiziki olaraq belə səyahəti həyata keçirmək imkanı olmayan bir çoxları üçün əlverişlidir. Bundan əlavə maraq doğuran suallar barədə mütəmadi olaraq izahat verdiyi üçün səyahət virtual şəkildə daha da öyrədici ola bilər.



Stiven Hokinq.

Derrik de Kerkhouv hesab edir ki, texnika nə qədər mürəkkəb olursa – olsun xarici görünüşü sadə olmalıdır. İnsanların işə getməsinə, demək olar ki, ehtiyac qalmayacaq, onlar müəssisəyə həftədə iki və ya üç dəfə baş çəkəcəklər, qalan vaxtda isə tapşırıqları evdə icra edəcəklər. Elə ödənişləri də Şəbəkə ilə göndərəcəklər.

* * *

Microsoft şirkətinin prezidenti Bill Geytsin (Bill Gates) fikrincə gələcəkdə informasiya texnologiyaları insanların asudə vaxtını zənginləşdirəcək, mədəni həyatını fəallaşdıracaq, cürbəcür məlumatlar almaq imkanlarını genişləndirəcəkdir. Evdə və ya kənar ofislərdə işləmək imkanının meydana gəlməsi şəhərin yüklənməsini azaldacaq. Bir çox məmulatlar material formasında deyil, bitlər ardıcıllığı şəklində istehsal olunacaq ki, bu da təbii ehtiyatların sərfiyyatını aşağı düşürəcək. Yeni texnologiyalar müstəqil olmaq, geniş təcrübə mübadiləsi aparmaq və tələbatlara müvafiq məhsullar almaqda ən gözəl vasitəçi olacaq. Rəqəmsal texnologiyaların gücü və onların tətbiqinin universallığı şəxsi, işgüzar və dövlət sirlərinin saxlanmasında yeni çətinliklər yaradacaq. Bir çox peşələr və bütöv sənaye sahələri unudulacaq, onların əvəzinə isə başqa, əvvəllər məlum olmayan yeni ixtisaslar meydana gələcək. İnternet şəbəkəsi olduqca fəal inkişaf edəcək. Hesablama texnikası və rabitə vasitələrinin qiyməti elə ucuzlaşacaq ki, informasiyanın qiyməti xeyli aşağı olacaq. Reklamdan gələn gəlir bir çox materialları (məsələn, hökumət məlumatlarını, tibbi məsləhət xidmətlərini, elektron elan lövhələrini və bəzi tədris vasitələrini) ödənişsiz yaymaq imkanı verəcək. Şəbəkəyə xidmət göstərən şirkətlərdə yüksək gəlir səviyyəsi olan ucqar bölgələrə fiberoptik xətlər çəkmək üçün stimul yaranacaq. Elektron məlumat lövhələri və digər şəbəkə forumları "birin birə", "birin çoxa" və ya "çoxun çoxa" prinsipilə ünsiyyət qurmaq üçün əla imkanlar yaradacaq. Həmfikir insanlar asanlıqla kompüter şəbəkələrində görüşərək müzakirələr aparacaq, partiya və birliklər təsis edəcəklər. Şəbəkə insanları həm yaxınlaşdıracaq, həm də milyonlarla təşkilata ayıracaq. Əsas informasiya xətləri hər şeyi – asudə vaxtın təşkili, informasiya axtarışı, ünsiyyəti yeni şəkildə həyata keçirmək imkanı verəcək.

Amerikanın "Global Business Network" məsləhətçi şirkətinin futuroloqlarının rəyinə görə, texnikanın və informasiya texnologiyalarının inkişafı insanları məişət qayğılarından azad edərək onların həyatına ciddi şəkildə təsir edəcək. Robotlaşdırma insanın bütün fəaliyyət sahələrinə daxil olacaq. Bu hərəkətli yarışüurlu "ev heyvanları" bişdüş, paltar yumaq, səliqə-səhman və sairlə məşğul olacaqlar. Kompüterləşmə iri miqyaslara çataraq, texniki tərəqqinin əsasına çevriləcək. Miniatür ölçülü fərdi kompüterlər onlarla funksiyanı cəmləşdirərək məsləhətçi, katib vəzifələrini yerinə yetirəcək. Rabitə universal və ucuz olacaq; nəqliyyat yalnız daha rahat, sürətli və sərfəli deyil, həm də daha kompüterləşmiş olacaq. Sürücü peşəsi ləğv olunacaq. Televiziya, radio, maqnitofon, peycer, telefonun birləşdirilməsi baş verəcək. Kino və televiziya qoxuları da çatdıra biləcək. Əhalinin əsas məşğulliyəti idman və kompüterlər olacaq.

* * *

Alimlər tərəfindən aparılan tədqiqatların gələcəyin real təsvirini nə dərəcədə doğru əks etdirməsi o qədər də vacib deyil. Bu tədqiqatların məqsədi bəşəriyyətin inkişafı və təbiətin qorunub saxlanmasında xüsusi vacib rol oynayacaq texnoloji tendensiyaları vaxtında aşkar etməkdir. Derrik de Kerkhouv bu barədə belə demişdir: "Gələcəyi əvvəlcədən görmək mümkün deyil, lakin onu ekstrapolyasiya etmək realdır. Bizim proqnozlar sistemimiz sadədir: Biz bəşəriyyətin bu gün hərəkət etdiyi istiqamətləri fikrən davam etdiririk və onun bu və ya digər nöqtəyə nə vaxt çatacağı barədə nəticəyə gəlirik".

ADLAR GÖSTƏRİCİSİ

A

Amper Andre Mari (Amper Andre Marie) (1775-1836) – fransız fiziki – 60-61, 67, 438
Atanasov Con Vinsent (Atanasoff John Vincent) (1903-1995) – ilk elektron hesablama maşını "ABC"ni qurmuş Amerika mühəndisi və riyaziyyatçısı – 396, 398-400
Ayken Hovard (Aiken Hovard) (1900-1973) – ilk EHM-lərdən biri olan "Mark 1"in layihələşdirilməsinə rəhbərlik etmiş Amerika mühəndisi – 394

B

Bardin Con (Bardin John) (1908-1991) – Amerika mühəndisi, tranzistoru icad edənlərdən biri – 400, 428
Bebbic Çarlz (Babbage Charles) (1791-1871) – ingilis riyaziyyatçısı, proqramla idarə olunan ilk hesablama maşını layihəsinin müəllifi – 133, 215-217, 340, 371-379, 418
Berq Aksel İvanoviç (1898-1979) – SSRİ EA-nın akademiki, kibernetika sahəsində tədqiqatların başlanğıcını qoymuş və bu sahəyə rəhbərlik etmişdir – 62, 417
Bəkus Con (Backus John) – Amerika alimi, "FORTRAN" proqramlaşdırma dilinin yaradıcısı – 186, 400
Bərri Klifford (Barry Clifford) – ilk elektron hesablama maşını "ABC"ni qurmuş Amerika mühəndisi, Con Atanasovun tərəfdaşı – 399
Bratteyn Uolter (Brattain Walter) (1902-1987) – Amerika mühəndisi, tranzistoru icad edənlərdən biri – 400, 428
Brayl Lui (Braille Louis) (1809-1852) – fransız pedaqoqu, korlar üçün şriftin yaradıcısı – 29-30
Bul Corc (Boole George) (1815-1864) – ingilis riyaziyyatçısı, riyazi məntiqin yaradıcılarından biri – 147, 150, 152, 385

C

Cobs Stiven (Jobs Steven) – "Apple Computers" şirkətinin qurucularından biri – 432, 448-452
Coy Bill (Joy Bill) – "SUN Microsystems" şirkətinin qurucusu – 292

Ç

Çerç Alonzo (Church Alonzo) (1903-1995) – Amerika məntiqçisi – 72, 160

D

Dekart Rene (Descartes Rene) (1596-1650) – fransız filosofu, riyaziyyatçı, fizik və fizioloqu; analitik həndəsənin əsasını qoymuşdur – 139, 340, 366
Deykstra Edsger (Dijkstra Edsger) – Amerika proqramçısı, bir çox klassik alqoritmin müəllifi – 141, 189, 194
Dreksler Erik (Drexler Eric) – Amerika alimi, nanotexnologiya nəzəriyyəçilərindən biri – 465, 468

E

Edison Tomas Alva (Edison Thomas Alva) (1847-1931) – amerikalı ixtiraçı, iri elektrotexniki şirkətlərin qurucusu – 385-386, 492
Ekkert Con (Eckert John) – Amerika mühəndisi, "ENIAK" hesablama maşınının yaradıcılarından biri – 395-396
Evklid (Euclid) – qədim yunan riyaziyyatçısı – 89, 91, 176, 177, 580
Eyler Leonard (Euler Leonhard) (1707-1783) – alman riyaziyyatçısı, mexaniki və fiziki. Riyazi analiz, ədədlər nəzəriyyəsi və digər sahələrdə 800-dən çox elmi işin müəllifi – 581

Ə

Əl-Xarəzmi Məhəmməd ibn Musa (780-850) – Orta Asiya alimi, əsrlər boyu öz populyarlığını qoruyub saxlamış riyaziyyat dərsliyi olan "Kitab əl-Cəbr vəl-Mukabala" kitabının müəllifi – 94

F

Ferma Pyer (Fermat Pierre) (1601-1665) – fransız riyaziyyatçısı, ədədlər nəzəriyyəsinin banilərindən biri – 366, 577

Feynman Riçard Filips (Feynman Richard Phillips) (1918-1988) – Amerika fiziki, əsas işləri kvant mexanikası, kvant elektrodinamikası və statistik fizika sahəsində olmuşdur – 464
Fleminq Con Ambioz (Fleming John Ambrose) (1849-1945) – ingilis fiziki, diodun ixtiraçısı – 385
Fletçer Harvi (Fletcher Harvey) – Amerika fiziki, stereosəsin müəllifi – 56
Furye Jan Batist Jozef (Fourier Jean Baptiste) (1768-1830) – fransız riyaziyyatçısı və fiziki – 54, 58, 341, 372

G

Geyts III Uilyam (Bill) (Gates III William) – "Microsoft" korporasiyasının qurucusu – 452-453

H

Haffman Devid Albert (Huffman David Albert) – Amerika alimi, adını daşıyan populyar sıxma alqoritminin müəllifi – 49, 54, 55, 58
Hemminq Riçard Uesli (Hamming Richard Wesley) – Amerika alimi, EHM yaddaşında səhvlərin aşkar olunub düzəldilməsi alqoritminin müəllifi – 22-23
Hers Henri Rudolf (Hertz Heinrich Rudolf) (1857-1894) – alman fiziki, elektrodinamikanın əsasını qoyanlardan biri – 491-492
Hollerit Herman (Hollerith Herman) (1860-1929) – verilənlərin emalı üçün ilk elektromexaniki tabulyatorun ixtiraçısı – 130, 379-382, 384

J

Jakkar Jozef Mari (Jacquard Jozeph-Marie) (1752-1834) – fransız ixtiraçısı – 377, 382-384

K

Kənədey Rad (Canaday Rudd) – "UNIX" əməliyyat sistemi proqram kanallarının müəlliflərindən biri – 291
Kilbi Cək (Kilby Jack) – Amerika mühəndisi, inteqral sxemin yaradıcısı – 407, 425
Knut Donald (Knuth Donald) – hesablama üsulları sahəsində Amerika mütəxəssisi – 119, 141, 419
Kolmer Karl Ksavye Tomas de (Colmar Charles Xavier Thomas de) (1785-1870) – fransız ixtiraçısı, kütləvi istehsal olunan ilk arifmometrin yaradıcısı – 384
Kolmoqorov Andrey Nikolayeviç (1903-1987) – SSRİ EA-nın akademiki, riyaziyyatçı, informasiya nəzəriyyəsində fundamental əsərlərin müəllifi – 24-25, 68-70, 90
Krey Seymur (Cray Seymour) – Amerika mühəndisi, superkompüterlər istehsal edən "Cray Research" şirkətinin qurucusu – 415, 469
Kroneker Leopold (Kronecker Leopold) (1823-1891) – alman riyaziyyatçısı, cəbr və ədədlər nəzəriyyəsi ilə məşğul olmuşdur – 319, 328-329

Q

Qauss Karl Fridrix (Gaus Carl Friedrich) (1777-1855) – dahi alman riyaziyyatçısı – 323
Qavrilov Mixail Aleksandroviç (1903-1979) – avtomatik idarəetmə nəzəriyyəsi sahəsində sovet alimi – 152
Qluşkov Viktor Mixayloviç (1923-1982) – SSRİ EA-nın akademiki, kibernetik, sovet kompüter sənayesinin banisi, ilk fərdi kompüter prototipinin yaradıcısı – 422-423
Qutenberq İohann (Gutenberg Johannes) (təqribən 1394-1468) – alman ixtiraçısı, Avropada ilk tipoqraf – 238, 534

L

Lavleys Avqusta Ada (Lovelace Augusta Ada) (1815-1852) – dünyada ilk qadın proqramçı – 209, 214-217, 373, 377-378
Lebedev Sergey Alekseyeviç (1902-1974) – SSRİ EA-nın akademiki, sovet kompüter istehsalının banisi; ilk sovet (və Avropa) EHM-i olan "MESM"in qurulmasına başlamışdır – 420-422
Lempel Abraham – Amerika alimi, sıxılma alqoritmi müəlliflərindən biri – 50
Leonardo da Vinçi (Leonardo da Vinci) (1452-1519) – italyan rəssamı, heykəltaraşı, memar, alim və mühəndisi – 47, 360, 365
Leonardo Pizanski Pizalı (Fibonaççi) (Leonardo Pisano Fibonacci) (təxminən 1170-1228-dən sonra) italyan riyaziyyatçısı – 33, 94, 419
Leybnis Vilhelm Qotfrid (Leibniz Wilhelm Gottfried) (1646-1716) – alman filosofu, riyaziyyatçı, fizik, dilşünası – 34, 147, 340, 361-362, 368-370
Lomonosov Mixail Vasilyeviç (1711-1765) – elmin müxtəlif sahələrində kəşflər etmiş rus alimi, şair – 38
Lyapunov Aleksey Andreyeviç (1911-1973) – SSRİ EA-nın müxbir üzvü, sovet informatikasının (kibernetikasının) atası – 62, 75, 92, 416-418

M

Mak-İlroy Duq (McIlroy Doug) – "UNIX" əməliyyat sisteminin proqram kanallarının müəlliflərindən biri – 291
Mouçli Con (Mauchly John) – Amerika mühəndisi, "ENIAK" hesablama maşınının yaradıcılarından biri – 130, 395, 396
Mur Qordon (Moore Gordon) – "Intel" şirkətinin qurucularından biri – 313, 469-470, 474

N

Neyman Con fon (Neumann John von) (1903-1957) – Amerika riyaziyyatçısı və fiziki, kompüterin klassik arxitekturasını vermişdir – 65-66, 72
Noys Robert (Noyce Robert) – Amerika mühəndisi, inteqral sxemin ixtiraçısı – 407
Nyuton İsaak (Newton Isaac) (1643-1727) – ingilis fiziki və riyaziyyatçısı, mexanika və astronomiyanın nəzəri əsaslarını yaratmış, diferensial və inteqral hesabının əsasını qoymuşdur – 324-325, 370, 374

O

Odner Vilqod (Odhner Willgodt) – isveç ixtiraçısı, ilk mexaniki hesab maşınını yaratmışdır – 384-385

P

Paçoli Luka (Pacioli Luca) (təxminən 1445-1514) – fransiskan rahibi, riyaziyyatçı, Leonardo da Vinçinin tərəfdaşı – 48, 94
Paskal Blez (Pascal Blaise) (1623-1662) – fransız filosofu, yazıçı, riyaziyyatçı və fiziki – 361, 366-368, 386
Patterson Deyvid (Patterson David) – "RISC-1" və "RISC-2" maşınlarının yaradılmasında iştirak etmiş Amerika alimi – 427
Post Emil (1897-1954) – Amerika riyaziyyatçısı, məntiqçi – 90, 154, 156-158, 160

R

Ritçi Denis (Ritchie Dennis) – Amerika proqramçısı, "UNIX" əməliyyat sisteminin yaradıcılarından biri – 194-195, 291-292

S

Sobolev Sergey Lvoviç (1908-1989) – SSRİ EA-nın akademiki, "Setun" üçlük hesablama maşınının yaradılması üzrə işlərin rəhbəri – 75, 419-420
Sossür Ferdinand de (Saussure Ferdinand de) (1857-1913) – İsveçrə dilşünası, struktur linqvistikasının banisi – 13
Stolmən Riçard (Stallman Richard) – Amerika alimi, müəllim və proqramçı, "GNU" layihəsinin əsasını qoymuşdur – 546-547, 589
Suze Konrad (Zuse Konrad) (1910-1995) – alman mühəndisi, ilk elektomexaniki hesablama maşınlarının yaradıcısı – 199, 209, 386-389

Ş

Şennon Klod Elvud (Shannon Claude Elwood) (1916-2001) – Amerika alimi və mühəndisi, informasiya nəzəriyyəsinin banisi – 12, 19-21, 24-25, 68, 70, 575
Şokli Uilyam (Shockley William) (1910-1989) – Amerika mühəndisi, tranzistorun ixtiraçılarından biri – 400, 428
Şor Piter (Shor Peter) – Amerika alimi, kvant kompüterləri sahəsində novator – 473-474

T

Tanenbaum Əndrü (Tanenbaum Andrew) – Amerika alimi, əməliyyatlar sistemi sahəsində mütəxəssis – 267, 293
Tompson Ken (Thompson Ken) – Amerika proqramçısı, "UNIX" əməliyyat sisteminin yaradıcılarından biri – 291, 293
Torvalds Linus – fin proqramçısı, populyar "Linux" əməliyyat sisteminin müəllifi – 293, 546
Türinq Alan Matison (Turing Alan Mathison) (1912-1954) – ingilis riyaziyyatçısı, bioqraaflarından birinin sözlərinə görə "kompüter elminin banisi" – 71-74, 93, 156-160

U

Uelç Terri (Welch Terry) – sıxma alqoritmi müəlliflərindən biri – 52

V

Vijener Blez de (Vigener Blaise de) – Fransa kralı III Henrixin əyanı, şifri icad etmişdir – 570-571
Viner Norbert (Wiener Norbert) (1894-1964) – rabitə nəzəriyyəsinə görkəmli töhfələr vermiş, ilk hesablama maşınlarının yaradılmasında iştirak etmiş Amerika alimi, "Kibernetika" kitabının müəllifi – 16, 60-62, 67-69, 368
Vinkler Ayre (Winkler Ira) – EHM şəbəkələrində informasiyanın qorunması sahəsində qabaqcıl Amerika mütəxəssisi – 555-558
Virt Niklaus (Wirth Niklaus) – İsveçrə proqramçısı, "Pascal" proqramlaşdırma dilinin yaradıcısı – 108, 190-191, 206
Voznyak Stefan (Wozniak Stephan) – "Apple Computers" şirkətinin qurucularından biri – 432, 448, 452

Y

Yerşov Andrey Petroviç (1931-1988) – SSRİ EA-nın akademiki, nəzəri və sistem proqramlaşdırmanın banilərindən biri, Sibir informatika məktəbinin yaradıcısı – 74-77

Z

Ziv Ceykob (Ziv Jacob) – sıxılma alqoritmi müəlliflərindən biri – 50-51

TERMİNLƏR GÖSTƏRİCİSİ

A

Açar – 256, 265
Açıq sistem – 288
Alqoritm – 87
Alqoritmik dillər – 93, 96-108, 178-182
Alqoritmin addımı – 86-88, 324
Alqoritmin sonluluğu – 88
Alqoritmlər nəzəriyyəsi – 91
Altproqram – 184, 404, 411, 412
Analitik maşın – 373, 376-379
Analoq-rəqəm çeviricisi – 40
Antivirus – 554
Arifmometr – 385
Asimmetrik kodlaşdırma – 574
Aşındırma – 435
Avtomat dili – 169

B

Balans – 47
Biliklər (prosedurlu) – 336
Biliklər bazası – 338
Biliklərin təsviri modeli – 337
Binar axtarış –126, 257
Birbaşa ünvanlama – 404
Birranqlı kompüter şəbəkələri – 490
Blok-sxem – 91
Böyük inteqral sxemlər – 424, 433-437
Brauzer – 515, 516
Budaqlanma – 98-99
Budaqlar və sərhədlər üsulu – 164
Bul cəbri – 150, 151
Bul əməli – 150

C

Cəmləyici – 314

Ç

Çeşidləmə – 130-135
Çox böyük inteqral sxemlər – 424, 433-437
Çoxluq – 124-129
Çoxmaşınlı komplekslər – 413
Çoxprosessorlu komplekslər – 413

D

Dəyişən – 104, 377
Dinamik yaddaş – 122
Disketlər – 440, 441
Diskretləşdirmə – 40, 323
Dizyunksiya – 149
Doğrama – 261
Dolayı ünvanlama – 404
Dolğunluq – 39
Dolub-daşma əlaməti – 314
Dövrün gövdəsi – 102
Dövrün invariantı – 177
Drayver – 286

E

Ekspert sistemləri – 212, 336
Entropiya – 18, 20, 70
Erqonomika – 454
Evklid alqoritmi – 89, 176
Bərabərölçülü kod – 19

Ə

Əks əlaqə – 61
Əlifba – 156, 165
Əməliyyat sistemi – 267, 269
Əməliyyat sisteminin nüvəsi – 269

F

Fayl – 269, 281
Fayl sistemi – 269, 281
Faylın indeksi – 283
Fərq maşını – 372-376
Fiziki səhifə – 276
Fiziki səviyyə – 483
Fiziki ünvan – 274
Fiziki ünvan sahəsi – 274
Formal dillər – 165
Fotolitoqrafiya – 434, 435
Fraktal – 52
Freym – 337
Funksional dillər – 210
Funksiya – 98

G

Genişlənmə (miqyaslanma) – 289
Giriş verilənləri – 60
Giriş-çıxış kitabxanası – 266
Giriş-çıxışın idarə olunması – 286
Görünmə sahəsi – 188
Göstəriş – 397, 406, 408, 409, 411, 412

H

Hemminq kodu – 22
Hesablamaların paralelləşdirilməsi – 413
Hesablanan funksiyalar – 158
Hipermətn – 357, 513-516, 533

X

Xarici yaddaş – 438
Xətti cədvəllər – 107
Xətti proqramlar – 98

İ

İcraçı – 81, 88
İç-içə komandalar – 100, 102
İdarəedici qurğu – 397
İdarəedici sistem – 60
İdarəedici strukturlar – 98
İdarəetmə sistemi – 60
İkili yazı – 47
İkilik simmetrik kanal – 21
İmperativ dillər – 210
İmplikasiya – 149
İndeks – 107, 256, 257
İnformasiya – 12-18, 61, 70
İnformasiya modeli – 136-139
İnformasiya nəzəriyyəsi – 18, 21, 70
İnformasiyanın ötürülməsi kanalı – 21
İnkapsulyasiya – 202
İnteraktiv tapşırıqlar – 272
İnterfeys – 202, 287
İnteroperabellik – 289
İşarə mərtəbəsi – 38
İterasiya – 169
İterasiyalı alqoritm – 89
İyerarxik struktur – 203, 283
İyerarxiyalı model – 262

K

Kanal – 401
Kanal prosessoru – 401
Kanal səviyyəsi – 483
Kanalların kommutasiyası – 482
Keçid – 169
Keçirmə – 288, 289
Keş-yaddaş – 285, 412
Kəsilmə emalçısı – 277
Kibernetika – 60-63, 67-69, 417
Komanda – 156, 157, 397-398, 403-404, 411, 427
Kommivoyajör məsələsi – 163
Kompaktlıq hipotezi – 341
Kompandlaşdırma – 55
Kompilyator – 185
Kompüter şəbəkələri – 482-483, 488-491
Kompüter virusu – 552
Kontekstən-azad qrammatikalar – 166
Konveyer sistemləri – 414
Konyunksiya – 149, 151
Kooperativ çoxtapşırıqlıq – 270
Kritik interval – 139
Kritik bölmə – 139
Kvantlaşdırılmama əmsalları – 54

Q

Qeyri-terminallar – 166
Qiyməti olmayan sıfırlar – 37
Qovuqlu çeşidləmə – 132
Qrammatika – 166
Qrammatika qaydaları – 166

L

Layihə normaları – 471, 472
Lokal hesablama şəbəkələri – 488

M

Maqnit lentləri – 440
Massiv – 106-108, 193, 403
Maşın kodu – 183
Maşın proqramı – 396
Maşın taktı – 397
Mənimsətmə – 97
Məntiq – 146
Məntiq cəbri – 148
Məntiqi funksiya – 148
Məntiqi qiymətlər – 148
Məsələnin (alqoritmin) mürəkkəbliyi – 133, 161-164
Mikroproqramlaşdırma – 408
Mikrosxemlər – 425, 429
Mobillik – 289
Modem – 484-488, 504
Monitor – 142, 267
Multimedia – 356
Multiproqramlaşdırma – 410
Mübadilə sahəsi – 280
Mühakimə – 146, 339

N

Nəqliyyat səviyyəsi – 483
Nəsillər – 204
Nomoqram – 390
Normal alqoritm – 155
Notasiya – 43
Növbə – 110

O

Obyekt-yönlü dillər – 200
Operativ yaddaş – 438
Optik disklər – 442
Optik kanallar – 506

P

Paket emalı – 401
Paketlərin kommutasiyası – 482
Palitra – 299
Parlaqlıq – 40
Parser – 170, 174
Perfokart – 378-379, 383-384
Periferiya – 409
Pirs oxu – 149
Planar texnologiya – 407
Planlaşdırma alqoritmi – 270, 272
Polimorfizm – 204
Polinomial alqoritm – 163
Potensial həll edilə bilən alqoritmlər – 88
Predikat hesablamaları – 152
Prefiks kodları – 19
Produksiya – 337
Proqram – 87, 92, 157
Proqramçı – 217, 219
Proqramlaşdırma dilləri – 93, 165
Prosedur – 87, 99, 184
Prosedur kitabxanaları – 184
Prosedurlu proqramlaşdırma – 188
Proses – 87, 139, 269
Prosesin kontekstinin dəyişdirilməsi – 269
Prosesin prioritetliyi – 271
Prosesin ünvan sahəsi – 275
Proseslərin planlaşdırıcısı – 269
Protokol – 483

R

Rastr formatları – 299-303
Rastr massivi – 299
Redaktor – 245-248
Registr – 375, 395, 397, 403-405, 409, 411
Rekursiv prosedur – 180
Rekursiya – 155, 180, 219-226
Requlyar ifadələr – 168-169
Relyasion verilənlər bazası – 263-265
Rezident monitor – 180
Rəqəm-analoq çeviricisi – 40
Rəng keyfiyyəti – 39
Rəng dərinliyi – 298
Rənglər cədvəli – 299

S

Sabit vergüllü ədədlər – 313
Say sisteminin əsası – 34-35
Seans səviyyəsi – 483
Semaforlar – 141-142
Semantik şəbəkə – 337
Səhvdüzəldici kodlar – 24
Sıxışdırıb-çıxaran çoxtapşırıqlıq – 270
Simmetrik kodlaşdırma – 576
Simulyasiya – 348, 350
Simvol – 156
Simvollarla hesablamalar – 209-211
Sinif – 200, 203
Sintaksis – 165
Sintaksis analiz – 170
Sistem çağırışı – 288
Sistem utilitləri – 269

Siyahı – 210
Skalyar əməliyyatlar – 415
Skaner – 250, 306
Slot – 337
Sonlu avtomatlar – 168
Spektral xarakteristika – 38
Stek – 110, 112, 222
Strimer – 440
Supervizor – 267
Süni intellekt – 209, 461-475
Sürüşkən vergüllü ədədlər – 313, 320

Ş

Şeffer ştrixi – 149
Şennonun birinci teoremi – 20
Şennonun ikinci teoremi – 24
Şəbəkə modeli – 283
Şəbəkə səviyyəsi – 483
Şərt – 99
Şifrləmə – 570-575
Şin – 438

T

Tabulyator – 382
Tapşırıqlar növbəsi – 267
Teq – 302, 423, 518
Tezlik analizi üsulu – 571
Tətbiqi proqram interfeysi – 288
Tipləşdirmə – 205
Trigger – 153, 586
Türinq maşını – 157-159

V

Vaxtın bölünməsi – 270
Veb-dizayn – 516-518
Vektorlar – 415
Ventillər – 152, 153, 429
Verilənlər (baytlar) axını – 281, 301
Verilənlər bazası – 255-260, 262-265
Vinçester – 441, 443
Virtual səhifə – 276
Virtual ünvan fəzası – 274
Virtual ünvanlar – 274
Virtual yaddaş – 274-276

Y

Yaddaş səhifəsi – 276
Yaddaşın idarə olunması – 274
Yaddaşın idarəolunması dispetçeri – 275
Yazı – 256-265
Yerinə yetirilmə kvantı – 270
Yuxarıdan aşağıya araşdırma – 172
Yumşaq maqnit disklər üçün disksürən – 441
Yüksələn araşdırma – 173, 174

ALGOL – 188, 189
C, C++ – 194-197, 206-207
FORTRAN – 186-188
FTP – 509, 510
HTML – 513, 514, 517
IP-ünvanı – 501
NP-mürəkkəb məsələlər – 163, 164
Pascal – 190-194
WWW – 513-516
Z-bufer – 304

MÜNDƏRİCAT

Oxucuya 5

İNFORMASİYA VƏ İNFORMATİKA

İNFORMASİYA VƏ İNSAN

Kompüter informasiyanın işlənməsi üçün universal vasitədir 10
XIX-XX əsrlərin başlıca elmi və texnoloji nailiyyətləri 11
İnformasiya nədir 12
İnformasiyanın atomları 13
İnformasiyanın təqdim edilməsi 15
Məzmun informasiyası 17
İnformasiya nəzəriyyəsi 18
Hemminq kodları 22
Klod Şennon 24

İNFORMASİYANIN KODLAŞDIRILMASI

Rəsmdən hərfə 26
Xəzinəni axtaraq 28
Hərflərin yerinə nöqtələr 30
Say sistemləri 31
İkilik kodlaşdırma 34
Təsvirlərin kodlaşdırılması 38
Səsin və musiqinin kodlaşdırılması 40
Rənglər və səslər 42
Filmlərin kodlaşdırılması 44
Mühasibat informasiyasının kodlaşdırılması 45
Ümumiləşmiş RLE-kodlaşdırmasının alqoritmi 48
İnformasiyanın sıxılması yolları 48
LZW alqoritmi 51
Səsin sıxılması 55

İNFORMATİKA ELMİ

İnformatika 60
İnformatika SSRİ-də 62
Sənədşünaslıq 64
Con fon Neyman 65
Norbert Viner 67
Andrey Nikolayeviç Kolmoqorov 68
Alan Matison Türinq 71
Andrey Petroviç Yerşov 74
A.P.Yerşov alim və insan kimi 76
Uşaqlara proqramlaşdırmanı nə üçün və necə öyrətməli 77
Mağazaları gəzirik... 79
Proqramlaşdırma ikinci savaddır 80
İçində nə var? 82
Yanıla-yanıla öyrənirlər 83

ALQORİTMLƏŞDİRMƏ

ALQORİTM

Alqoritm və proqramlar 86
İlk alqoritmlər 89
Alqoritmin daha bir formal təsviri 90
"Alqoritm" sözünün mənşəyi haqqında 93
Alqorismiklərin və abasistlərin mübahisəsi 94

PROQRAMLAŞDIRMANIN ƏSASLARI

Sadə proqramlar 96
İdarəedici strukturlar – budaqlanmalar 98
İdarəedici strukturlar – dövrlər 102
Massivlər 106
Verilənlərin strukturu 108
Verilənlərin dinamik strukturları və istinadlı realaşdırmalar 120
Kəsilməzlik problemləri 122
Çeşidləmə 130
Nizamlılığın vacibliyi 132
İnformasiya modellərinin qurulması 135
Nəqliyyat şəbəkəsinin informasiya modeli 137
Paralel proqramlaşdırma 138
Həndəsi informasiyanın informasiya modelləri 139
Semaforlar 142

PROQRAMLAŞDIRMA NƏZƏRİYYƏSİ

Riyazi məntiq 146
Başqa bir məntiq də mümkündür 154
Alqoritmlər nəzəriyyəsi 155
Markovun normal alqoritmləri 155
Alqoritmik həll oluna bilməyən problemlər 158
Alqoritmlərin mürəkkəbliyi 160
Polinomial və eksponensial alqoritmlər 163
Formal dillər nəzəriyyəsi.
Tərcümə və kompilyasiya 165
LEX və GREP proqramları 169
Proqramların düzgünlüyünün isbatı 175
Evklid alqoritmi PASCAL dilində 177

PROQRAMLAŞDIRMA DİLLƏRİ

LOGO proqramlaşdırma dili 178
Proqramlaşdırma dillərinin tarixi 182
Strukturlaşdırılmamış kod 191
Strukturlaşdırılmış kod 192
Massivlər 193
PASCAL dilinin birinci kompilyatorunun yazılma tarixi 194
Massivlər və göstəricilər 196
C dili və UNIX əməliyyat sistemi 197
PLANKALKUL – ilk proqramlaşdırma dili 198
Obyekt-yönlü proqramlaşdırma dilləri 199
İnterfeys və reallaşdırma 202
Obyekt-varislərin valideyn obyektlər kimi istifadə olunması 204
Polimorfizm 205
Süni intellekt dilləri 209

PROQRAMÇILAR VƏ PROQRAMLAŞDIRMA

Avqusta Ada Bayron, ledi Lavleys 214
Proqramçılar 217
Rekursiya 219
Kukaraça 222
N ədədinin faktorialının hesablanması 226

Etibarlı proqramlar necə yazılır 227
Üçbucaq haqqında məsələ 228
Gözəl proqramlar necə yazılır? 230

İNFORMASİYANIN SAXLANMASI VƏ İŞLƏNMƏSİ

İNFORMASİYANIN SAXLANMASI

Kompüter və kitab çapı 238
Kağız 240
Alimlər üçün kitabxana 241
NSLOIO BXEQBXUİPİÇĞBRİ 242
Mətn redaktorları 245
Maketləşdirmə 248
Mətn redaktoru 249
Elektron cədvəllər 250
Verilənlər bazasını idarəetmə sistemləri 255
"Tərəzi" (və/və ya) alqoritmi barədə 255
Doğrama 261
Relyasion verilənlər bazası 262
Bir qiqabayt nə deməkdir 263

KOMPÜTERİN ƏMƏLİYYAT SİSTEMİ

Əməliyyat sistemlərinin meydana gəlməsi 266
Əməliyyat sistemi nədir 267
Proseslərin planlaşdırıcısı 269
Prioritetlərin hesablanması düsturu 272
Əməliyyat sistemlərinin ölçüləri 273
Yaddaşın idarə edilməsi altsistemi 274
Prosesin daxilində qorunmaq üçün
virtual yaddaşdan istifadə edilməsi 277
Virtual yaddaş mexanizminin
işinə dair nümunə 278
Fayl sistemi 281
Adlar 283
UNIX əməliyyat sistemində fayllar 284
Əgər onların sayı çox olarsa? 285
Giriş-çıxış əməliyyatlarının idarəedilməsi
altsistemi və qurğuların drayverləri 286
Tətbiqi proqram interfeysi 288
UNIX əməliyyat sistemi 290

MULTİMEDİA

İnformasiyanın rəqəmsal formada
təsvirinə keçid 294
Vektor və rastr qrafiki 297
Çözümlülük 300
3D-sürətləndiricisi nəyə qadirdir 304
Rəqəmsal fotoqrafiya 306
Rəqəmsal video 307
Audiomontaj 310

HESABLAMALAR

Kompüterdə hesablamalar 312
Yuvarlaqlaşdırmanın səhvləri 316
Hesablamalarda xətanın qiymətləndirilməsi 318
Hesablama üsulları 319
İnterval hesabı 321
Nyuton metodu 324
Nyuton metodunun dəqiqliyi 324
Nyuton nə yazır 325
Kompüter cəbri 326

İNTELLEKTUAL FƏALİYYƏTİN AVTOMATLAŞDIRILMASI

Kompüter modelləşdirilməsi 330
İnformasiya sistemlərinin qurulması 334
Ekspert sistemləri 336
Obrazların tanınması 339
"Düşüncə maşını" 340
Ev ofisi 343

OYUNLAR

Kompüter oyunları 346
Trenajorlar 349
"Tetris" fenomeni 349
Süni gerçəkliklər 351
Oyunlardan biliklərə doğru 356

KOMPÜTER

KOMPÜTERƏ QƏDƏRKİ DÖVRÜN TARİXİ

Hesablama maşınına gedən yol 360
Abak və sayğaclar 362
Kim idi birinci? 365
Blez Paskal 366
Qotfrid Vilhelm Leybnis 368
Çarlz Bebbic 371
Bebbicin maşınları 374
Herman Hollerit 379
Rusiyada əhalinin ilk siyahıya alınması 380
Avtomatlaşdırma və perfokartlar 382
"Elmi" kalkulyatorlar 384
Konrad Suze 386
Analoq maşınları 390

XX ƏSRDƏ KOMPÜTERİN TƏKAMÜLÜ

Birinci nəsil kompüterlər 394
Con Vinsent Atanasov 398
İkinci nəsil kompüterlər 400
Yadda saxlayan elektron-şüa boruları
və ultrasəsli yubatma xətləri 401
Ferrit nüvəcikləri və maqnit barabanları 402
Kəsilmə aparatı 405
Böyük sistemlər 406
Kompüterlərin üçüncü nəsli 407
Mini kompüterlər 410
Paralel hesablamalar 412
Aleksey Andreyeviç Lyapunov 416
Üçlük hesablama sistemi
və üçlük EHM 418
Çəkilər haqda Başe məsələsi 419
Sovetlər ölkəsində istehsal olunan EHM-lər 420
Mühəndis hesablamaları üçün maşın 422

MÜASİR KOMPÜTER

Kompüterlərin dördüncü nəsli 424
İnteqral sxemlərin növləri 428
Mikroprosessorların istehsalı 433
İnteqral sxemlərin layihələndirilməsi 434
EHM yaddaşının təkamülü 437
Maqnit yazılış 441
Keş bizə kömək edəcək 443
IBM firmasının "Şan-şöhrəti və acizliyi" 444
Stiv Cobs nə barədə bilmirdi 448

Macintosh əsrlər sərhədində 450
III Uilyam (Bill) Henri Geyts 452

XXI ƏSRİN KOMPÜTERİ

Kompüter və sağlamlıq 454
Əl-qolun, bədənin açılması üçün
edilən hərəkətlər 458
Neyroşəbəkələr 459
Süni neyron şəbəkələri 460
Nanotexnologiyalar 461
"Orada – dərinlikdə..." 464
Nanorobotlar 468
Hesablama maşınlarının gələcəyi 468
Mur qanunu 470
İnteqral texnologiyaların uğurları 471
Layihə normaları 472
Kvant kompüterləri 473
2010-cu ilin kompüteri 474

İNFORMASİYANIN ÖTÜRÜLMƏSİ

İNFORMASİYANIN ÖTÜRÜLMƏSİ VƏ KOMPÜTER ŞƏBƏKƏLƏRİ

Böyük məsafəyə informasiyanın ötürülməsi 480
Kompüterlər arasında informasiyanın ötürülməsi . . 482
Modem 484
ISDN və mobil telefon şəbəkələri 485
Modem necə işləyir 485
Hansı kompüter şəbəkələri vardır 488
Şəbəkə əməliyyat sistemi (ƏS) 491
Simsiz rabitə texnologiyası 491
Radioya gedən yolda 492
Nəyi nə ilə birləşdirməli 495

İNTERNET

İnternetin tarixi, quruluşu və inkişafı 498
İnternetə irişmə 503
Rabitə kanalları 506
Hansı kanalı seçməli? 507
Klassik İnternet xidmətləri 507
TELNET proqramının işi 508
FTP proqramının işi 509
Anonim FTP 510
"Elektron" məktubun üslubunun xüsusiyyətləri . . . 511
Elektron poçtun icad edilməsi 512
World Wide Web 513
İlk sayt 515
İnternet şəbəkəsinin miqyası 518
İnternetdə kommersiya 519
eBay.com saytı 520
İnternetdə axtarış 523
Necə axtarmalı 526
Harada axtarmalı 527
"Elektron hökumət" 528

Avropada virtual təhsil 530
"Avropada E-Austria" 531
Elektron kitabxanalar 531
İnternet və cəmiyyət 534
İnternetdə 2010-cu ilin səhəri 536
Proqramçı öz soyadını saytının
adı ilə əvəz etdi 538
İnternetdə söz azadlığı və senzura 539
İnternetin idarə edilməsi 541
İnternetə qadağa 541

İNFORMASİYANIN QORUNMASI

Rəqəmli əsrdə müəlliflik hüququ 542
Müəlliflik hüququ nədir 543
Müəlliflik hüququnun tarixindən 544
Müəlliflik hüququnun pozulmasına görə
məsuliyyət 545
Copyright və "Copyleft" 546
İnformasiya təhlükəsizliyi 548
Səhvlər və qəzalar 549
Viruslar 550
Bir virusun tarixçəsi 551
Ən məşhur "troyalı" 553
Sınaq müdaxiləsi 555
Hakerlər 559
"Açıq qapılar" 561
Kompüter cinayətkarlığı 561
Qəsdən ziyanvericilik 562
Sənaye casusluğu 563
Kompüter piratları 563
Pirat lüğəti 566
İnformasiyanın şifrlənməsi 568
Müasir kriptoqrafiya 573
"Enigma"nın quruluşu 574
Modula görə çıxıqlar halqası 579
Evklid alqoritmi 580
Fermanın kiçik teoremi və Eyler teoremi 581

MÜASİR İNFORMASİYA TEXNOLOGİYALARI

İnformasiya və postinformasiya cəmiyyəti 582
Kütləvi informasiya istehsalçıları 584
İnformasiya müharibələri 587
XXI əsrdə audio 588
Musiqinin proqramlaşdırılması 590
Yeni dövrdə kino və foto 593
Avtomobil və informasiya texnologiyaları 597
Elektron ev 600
Sehrli soba 601
Yaxın yüzillik üçün proqnozlar 602

Adlar göstəricisi 606
Terminlər göstəricisi 609

UŞAQLAR ÜÇÜN
ENSİKLOPEDİYA

İNFORMATİKA

Elmi tərcüməçilər və elmi redaktorlar: *Ramiz Alıquliyev*
fizika-riyaziyyat elmləri namizədi

Zərifə Cəbrayılova
texnika elmləri namizədi

Tofiq Kazımov
fizika-riyaziyyat elmləri namizədi

Rəsmiyyə Mahmudova

Əfruz Qurbanova

Buraxılışa məsul: *Əziz Güləliyev*

Kompüter səhifələyiciləri: *Ələkbər Kərimov*
Yeganə Əsgərova
Aslan Almasov

Kompüter işləri: *Fərhad Yusifov*
Anar Səmidov
Aybəniz Kərimova

Korrektor: *Zahid Məmmədov*

Çapa imzalanmışdır 10.07.2008. Format 84×108 $^{1}/_{16}$.
Ofset kağızı. Ofset çapı. Fiziki çap vərəqi 38,5.
Tirajı 25000 nüsxə. Sifariş №82.

"Şərq-Qərb" mətbəəsində çap olunmuşdur.
Bakı, Aşıq Ələsgər küçəsi, 17.