

“O‘zbekiston temir yo‘llari” DATK
Toshkent temir yo‘l muhandislari instituti

Shodimetov X.M., Abduqayumov B.N., Xolboyev O‘.N.

INFORMATIKA VA AXBOROT TEXNOLOGIYALARI

barcha ta‘lim yo‘nalishlari I-bosqich bakalavriat talabalari uchun
o‘quv qo‘llanma



Toshkent - 2012

Informatika va axborot texnologiyalari. O'quv qo'llanma.
X.M. Shodimetov, B.N.Abdugayumov, O'.N.Xolboyev ToshTYMI, T.:
2012, 162 b.

O'quv qo'llanma ToshTYMI barcha ta'lim yo'nalishlarida tahsil olayotgan 1-bosqich bakalavriat talabalari uchun mo'ljallangan bo'lib, "Informatika va axborot texnologiyalari" fanini o'zlashtirishlariga yordam beradi. Qo'llanmada "Informatika va axborot texnologiyalari" fani predmeti, kompyuterlar, Delphida dasturlash asoslari, differensial va integrallarni taqribiy hisoblash algoritmlari va dasturlari, ofis dasturlari, internet, HTML asoslari misollar yordamida batafsil tushuntirilgan.

O'quv qo'llanma institutning O'quv-uslubiy komissiyasi tomonidan nashrga tavsiya etilgan.

Taqrizchilar: J.S. Abdullayev – bosh mutaxassis;
N.A. Asqarov – i.q.f.n., dots.

Informatika – axborotlarning tuzilishi va xossalari haqidagi, shuningdek, axborotlarni yig'ish, saqlash va qayta ishlash usullari hamda vositalari haqidagi tabiiy fundamental fandır.

"**Informatika**" so'zi "**informatsiya**" (axborot) va "**avtomatika**" so'zlaridan hosil qilingan bo'lib, "axborotlar bilan avtomatik ishlash" ma'nosini beradi. Informatikaning asosiy obyekti axborot (informatsiya) hisoblanadi.

"**Informatsiya**" so'zi lotincha "**informatio**" so'zidan olingan bo'lib, tushuntirish, bayon qilish ma'nolarini anglatadi. Informatika fani ma'lumotlarni elektron hisoblash va qayta ishlash qurilmalarining rivojlanishi natijasida XX asming ikkinchi yarmida vujudga kelgan bo'lib (1960 yillarning boshlarida fransuz tilida "**informatique**" termini paydo bo'lgan), kompyuterlar bu fanning texnik vositasi hisoblanadi.

Kompyuter (EHM) larning qisqacha tarixi

XVII asrda Shikkard, Paskal, Leybnislar tomonidan dastlabki hisoblash mashinalari yaratildi: 1623 yilda Vilgelm Shikkard birinchi qo'shish amali bajaradigan hisoblash mashinasini yaratdi; 1642 yilda fransuz olimi Blez Paskal yaratgan **jamlash mashinasi** birinchi hisoblash mashinasi deb qabul qilingan va 50 nusxada ishlab chiqarilgan. Paskal mashinasi faqat qo'shish va ayirish amallarini bajara olgan; 1671-1674 yillarda Leybnis to'rtta arifmetik amallarni ham bajara oladigan **arifmometr** yaratdi.

1820-1830 yillarda ingliz olimi Ch. Bebbidj barcha arifmetik operatsiyalarni bajaradigan, turli masalalarni mustaqil yechadigan va tushunadigan **analitik mashina** loyihasini ishlab chiqdi. Uning g'oyasi bo'yicha mashina hisoblash jarayonida mutlaqo inson ishtirokisiz maxsus dasturlar asosida ishlashi ko'zda tutilgan edi. Bu mashinaning birinchi maketi 1834 yilda yaratildi. Ammo moddiy va texnik ta'minot yetarli bo'lmaganligi uchun Bebbidjga mashinani yaratish nasib etmadi. Shunga qaramay uning mashinasi dastur asosida ishlaydigan dunyodagi birinchi analitik mashina edi va bu murakkab masalalarni yechadigan matematik mashinalarning paydo bo'lishiga asos soldi.

XIX asr oxirlarida elektrik qurilmalarning paydo bo'lishi va elektrtexnika sohasining rivojlanishi natijasida hisoblash qurilmalarining elektrmexanika bosqichi boshlandi. 1887 yilda German Xollerit birinchi bo'lib elektrmexanik relelar asosida oddiy hisoblashlarni avtomatik bajaradigan mashina – **tabulyator** yaratdi. 1897 yilda Xollerit **Tabulating Machine Company** firmasiga asos soldi. Keyinchalik bu firma kompyuterlar ishlab chiqarish bo'yicha butun dunyoga mashur **IBM** (International Business

Machines) korporatsiyasiga aylantirildi.

1930 yillarda dastur yordamida boshqariladigan releli (elektromexanik) hisoblash mashinalari yaratila boshlandi. 1939-1941 yillarda Germaniyada Konrad Suz tomonidan *Z-3* deb nomlangan birinchi dastur yordamida boshqariladigan elektromexanik hisoblash mashinasi yaratildi. Bu mashina bir vaqtda 64 ta raqamni saqlab qolib, ikkita sonni 0,3 soniyada qo'shish, 5 soniyada ko'paytirish imkoniyatiga ega bo'lgan.

1940 yilda amerikalik muhandis G. Eytken Bebbidj loyhasi asosida ikkilik-o'nlik sistemadan foydalanib, *Mark-1* deb nomlangan birinchi universal dastur yordamida boshqariladigan hisoblash mashinasini yaratdi. *Mark-1* mashinasi mexanik ravishda ishlagan bo'lsa, 1947 yilda yaratilgan *Mark-2* to'raligicha elektromexanik mashina hisoblangan.

XX asr o'rtalarida hisoblash mashinalarining ishlash tezligi asosiy muammo sifatida qaraldi. Bu esa kompyuterlarning paydo bo'lishiga olib kela boshladi. Kompyuterning muhim xususiyati sifatida uning ishlash tezligi belgilandi.

1937-1942 yillarda amerikalik olimlar J. Atanasov va K. Berti rahbarligida birinchi EHMni yaratishga urinib ko'rishdi.

AQSh olimlari J. Mouchli va P. Ekkert rahbarligida 1945 yilda AQShning Pensilvaniya universiteti texnik xodimlari tomonidan birinchi EHM yaratildi va u *ENIAC* deb nomlandi. Bu kompyuter 18 mingta lampaga ega bo'lib, og'irligi 30 tonna, maydoni 120 m², bir soniyada 5 mingta amal bajargan.

Fon Neyman kompyuterlarda saqlanadigan va hisoblanadigan ma'lumotlar perfokartalar va perfolentalar yordamida emas, dastur yordamida kompyuterning o'zida saqlanish prinsipini ishlab chiqdi. Dastur yordamida saqlanadigan va hisoblanadigan birinchi kompyuter *EDSAC* nomli kompyuter bo'lib, 1949 yilda Buyuk Britaniyalik olim M. Uilks tomonidan yaratildi. Ushbu mashinadan boshlab kompyuterlarning birinchi avlodini hisoblash qabul qilingan.

Sobiq ittifoqda birinchi bo'lib akademik S.A. Lebedov rahbarligida 1951 yil Ukraina FA Elektronika institutida *MƏCM* deb nomlangan EHM yaratildi. 1954 yilda S.A. Lebedov rahbarligida *BƏCM* nomli EHM yaratilgan bo'lib, u 2048 ta xotira yacheykasiga ega bo'lib, 1 soniyada 9 ming amal bajarar edi. O'sha vaqda *BƏCM* jaxondagi eng tezkor EHM hisoblangan. Bundan tashqari shu davrda *Strela*, *Minsk*, *Dnepr*, *Ural*, *M-2*, *Mir* kabi EHMLar ham yaratilgan.

EHMLar avlodlari

Kompyuterlar, ya'ni EHMLar 5 ta avlodga bo'lib o'rganiladi.

1-avlod kompyuterlari 1950-yillarda yaratilgan kompyuterlardir. Ularning hajmi 60-80 m², og'irligi 30 tonna atrofida bo'lgan. Bu mashinalar minglab *elektron lampalar* asosida ishlashga mo'ljallangan bo'lib, xotira hajmi bir necha megabayt, ishlash tezligi 1 soniyada 1 mingtadan 10 mingtagacha amal bajargan. Bitta mashinani ishlatish uchun matematik, fizik, mexanik, elektrik kabi mutaxassislarning birgalikda ishlashi talab qilingan. Shuncha sarf-harajatlarga qaramay bu mashinalar oddiy hisob-kitoblarnigina amalga oshira olgan va asosan ilmiy va harbiy maqsadlarda qo'llanilgan. 1-avlod kompyuterlariga *ENIAC, EDVAC, EDSAC, UNIVAC, BECM, Strela, Minsk-1, M-20* va boshqalar kiradi.

2-avlod kompyuterlari 1960-yillarda yaratilgan. Ular 1-avlod mashinalariga qaraganda ancha ixcham bo'lib, lampalar o'rnida *tranzistorlar* ishlatilgan. Xotira hajmi bir necha o'n megabayt, ishlash tezligi 1 soniyada 100 mingtacha amalga yetgan. Bu mashinalarda qo'yilgan masalalarni yechish uchun dasturlash tillaridan foydalanila boshlangan. 2-avlod kompyuterlariga *Minsk2, Ryazan, ATLAS, BECM-4, BECM-6, Mir, Minsk-22, Minsk-32, M-220* va boshqalar kiradi.

3-avlod kompyuterlari 1970-yillarda yaratilgan. Bu mashinalarda tranzistorlar o'rnida minglab tranzistorlarni o'zida birlashtirgan *integral sxemalar* qo'llanilgan. Xotira hajmi 100 Megabaytdan ko'proq, bir soniyada bir milliontagacha amal bajara olgan. Shu davrdan boshlab monitor, klaviatura kabi qurilmalar alohida qurilma sifatida yaratila boshlangan. Shuningdek, ushbu davrdan boshlab *mikroprotsessorlar* paydo bo'lgan. Birinchi mikroprotsessor **Intel 4004** deb nomlanib, 1971 yilda **Intel** firmasi tomonidan yaratilgan. 3-avlod kompyuterlariga *Ural-11, Ural-12, ES* va *SM* seriyali EHMLar kiradi.

4-avlod kompyuterlari 1980-yillardan keyin yaratilgan mashinalar. Bu mashinalarda integral sxemalar o'rnida *katta integral sxemalar* ishlatiladi. Doimiy xotira hajmi 1 Gegabaytdan ortiq, ishlash tezligi 1 soniyada 1 milliondan ko'p amal bajaradi. Hozirgi kunda ishlab chiqarilayotgan kompyuterlar ham 4-avlod kompyuterlariga tegishli bo'lib, doimiy xotira hajmi Terabaytgacha, tezligi bir sekundda bir necha milliard amal bajarishgacha yetgan. 4-avlod mashinalariga *Elburs-2, M-10* EHMLaridan boshlab hozirgi kungacha chiqarilayotgan kompyuterlar kiradi.

5-avlod kompyuterlari loyihalashtirilgan va yaratilish ostonasida turgan kompyuterlardir. Bunday kompyuterlarda katta integral sxemalar o'rnida *neyron qatlamli optik tolalar* qo'llaniladi. 5-avlod kompyuterlarining hozirgi kompyuterlardan asosiy farqi shundan iborat bo'ladiki, ular sun'iy fikrlash qobiliyatiga ega bo'ladi va vaziyatga qarab ma'lum kerakli qarorlarni chiqaradi hamda texnologik jarayonlarni boshqarish vazifasi

yuklatiladi. Bundan tashqari ma'lumotlarni og'zaki kiritish va boshqarish ishlarini yo'lga qo'yish rejalashtirilgan.

Izoh

Yaponiyalik olimlar bunday imkoniyatlarga ega bo'lgan mashinalarni bir necha bor ishlab chiqarishga urinib ko'rinishi. Ammo ular 5-avlod kompyuterlari sifatida tan olinganicha yo'q. Yaponiyadan tashqari AQSh, Rossiya va bir necha G'arb mamlakatlari olimlari ham bu ish ustida jiddiy izlanishlar olib borishmoqda.

Kompyuterlar sinflari

Kompyuterlar tuzilishi va funksional imkoniyatlariga ko'ra shartli ravishda 3 ta sinfga bo'linadi:

- *superkompyuterlar*;
- *universal kompyuterlar*;
- *shaxsiy kompyuterlar*.

Superkompyuterlar juda murakkab va katta hajmdagi masalalarni yechishga mo'ljallangan maxsus ishlab chiqariladigan kompyuterlar bo'lib, avtomatlashtirilgan juda katta zavod va korxonalarda, ma'lum vazifalarni bajaradigan robotlarni boshqarishda, turli ilmiy tadqiqot muassasalarida, xususan fazoviy tadqiqot o'tkazadigan yoki ob-havo ma'lumotlarini aniqlaydigan tashkilotlarda, raketa va kosmik himoyalash masalalarida, atom va yadroviy tadqiqotlar o'tkazishda qo'llaniladi. Bu superkompyuterlar bir necha ming mikroprotsektorlarga ega bo'lib, bir sekundda trillionlab operatsiya bajara oladi. Bunday kompyuterlar asosan AQSh, Germaniya, Yaponiya, Buyuk Britaniya, Rossiya kabi fan-texnikasi rivojlangan mamlakatlarda mavjud.

Izoh

Hozirda dunyodagi eng katta superkompyuter AQShda joylashgan bo'lib, mingdan ortiq ilmiy tadqiqot muassasalari undan murakkab masalalarni yechishda tarmoq orqali foydalanishlari mumkin. Bu superkompyuter *Cray XI* deb nomlanib, 1 soniyada 52 trillion amal bajarish imkoniga ega.

Imkoniyat jihatidan 2-o'ringda Yaponiyadagi 1 soniyada 30 trillion amal bajaradigan *Year Simulator* superkompyuteri, 3-o'ringda 1 soniyada 12,4 trillion amal bajaradigan AQShdagi *ASCI White* superkompyuteri turadi.

Universal kompyuterlar (meynfreymlar) ham talabga qarab maxsus ishlab chiqariladigan kompyuterlar bo'lib, zavod, tashkilot, kompaniyalarda ma'lum murakkab va katta hajmli masalalarni yechishda, xususan juda katta o'lchamdagi massivlar bilan ishlashda, global va lokal tarmoqlarni boshqarishda qo'llaniladi.

Shaxsiy kompyuterlar (ShK) eng keng tarqalgan kompyuterlar bo'lib,

dunyodagi kompyuterlarning asosiy qismini tashkil etadi. Shaxsiy kompyuterlar ham o'z navbatida *stol kompyuteri (desktop)*, *noutbuk (notebook)*, *netbuk (netbook)*, *ishchi stansiya (work station)*, *cho'ntak kompyuterlari* kabi ko'plab turlarga bo'linadi. Shaxsiy kompyuterlar hozirgi kunda respublikamizdagi barcha ta'lim muassasalari, tashkilot, kompaniyalarda hamda boshqaruv, meditsina kabi boshqa muassasalarda, shuningdek har bir shaxsning kundalik hayotida keng foydalaniladi.

Kompyuter (EHM)lar umumiy ravishda *arifmetik-logik, boshqaruv, xotira va tashqi* qurilmalardan tashkil topadi.

Bundan buyon asosan shaxsiy kompyuterlarning *stol kompyuteri* va *noutbuk* turlari hamda ularning imkoniyatlari nazarda tutiladi.

Hozirgi kunda kompyuter ishlab chiqaruvchi ko'plab firmalar faoliyat yuritmoqda. Bularga *DEL, HP, IBM, Apple, Toshiba, Acer* kabi firmalarni misol keltirish mumkin.

Stol kompyuterining asosiy qurilmalari quyidagilar:

- *tizimli blok;*
- *monitor;*
- *klaviatura;*
- *sichqoncha.*

1. Tizimli blok kompyuterning eng asosiy qurilmasi hisoblanadi.

Uning asosiy qismlari quyidagilar:

- ❖ **tizimli plata** – bosh plata hisoblanib, mikroprotessor, operativ xotira kabi asosiy qurilmalar shu plataga mahkamlanadi. Shuningdek tizimli platada doimiy xotira, kesh xotira, audiokarta, videokarta kabi ko'plab qurilmalar joylashadi. Ba'zan bu qurilma *ona plata (motherboard)* deb ham ataladi. Chunki barcha qurilmalar mosligini tizimli plata ta'minlaydi;
- ❖ **mikroprotessor** – kompyuterning “miyasi” bo'lib, berilayotgan barcha buyruqlarni, amallarni va umuman barcha operatsiyalarni bajarishni ta'minlaydi. Bu qurilma CPU (Central Processing Unit – markaziy qayta ishlovchi qurilma) yoki oddiygina protessor deb ham ataladi. Hozirgi kunda asosan Intel firmasining *Pentium IV, Core 2 duo, Corei 7, i7* kabi mikroprotessorlari keng qo'llanilmoqda. Mikroprotessorlar uchun tezlik muhim ro'l o'ynaydi. Tezlik MHz (mega-gers), GHz (gegagers) larda o'lchanadi. Hozirda 2.5, 2.7 GHz va undan yuqori tezlikdagi mikroprotessorlar ishlab chiqarilmoqda;
- ❖ **tezkor (operativ) xotira** – kompyuter ishlab turganda foydalanilayotgan barcha tizimli va amaliy dasturlarni hamda kiritilayotgan va bajarilayotgan amallarni vaqtinchalik saqlab turadigan xotira qurilmasi. Bu qurilma OZU (оперативное запоминающее устройство), RAM (Random

access memory) nomlari bilan ham ataladi. Hozirgi vaqtda 2 Gb (gigabayt), 4 Gb li tezkor xotiralar ishlab chiqarilmoqda. Tezkor xotira hajmi qanchalik katta bo'lsa, kompyuterning ishlash tezligi va bir necha murakkab dasturlardan bir vaqtda foydalanish imkoniyatlari ortadi;

- ❖ **qattiq disk** – kompyuterning asosiy xotirasi bo'lib, barcha turdagi ma'lumotlarni va dasturlarni doimiy saqlab turishga mo'ljallangan xotira. Bu qurilma **vinchester** yoki **HDD** (hard disk drive) deb ham ataladi. Qattiq disk xotira hajmlari 80 Gb, 160 Gb, 250 Gb, 500 Gb, 1 Tb (terabayt) va boshqa ko'rinishlarda bo'ladi;
- ❖ **kesh xotira** – mikroprotsessor va operativ xotira orasiga o'rnatilib, kompyuter ishlash tezligini yanada oshirish uchun qo'llaniladi. Kesh xotira vazifasi va tuzilishi jihatidan deyarli operativ xotiraga o'xshash bo'lib, operativ xotiradan ancha tez ishlaydi. Kesh xotira hajmi hozirgi kunda odatda 1-3 Mb bo'ladi;
- ❖ **doimiy xotira** – kompyuterning ishlashini ta'minlab beruvchi kompleks dastur (BIOS) joylashgan bo'lib, **ПЗУ** (постоянное запоминающее устройство) yoki **ROM** (Read only memory) deb ham ataladi;

izoh

Doimiy xotiraga yozilgan dastur faqat bir marta ishlab chiqarilgan zavodda yoziladi va undan faqat o'qish uchun foydalaniladi. Bu dasturlar qurilmalar ishlashini tekshirish, klaviatura, monitor, xotira qurilmalari o'rtasida ma'lumot almashish bilan bog'liq doimiy amallarni bajarishni ta'minlaydi.

- ❖ **videokarta** (videoadapter) – monitorda video ma'lumotlarni aks ettirish uchun qo'llaniladi;
- ❖ **shinalar** – kompyuter qurilmalari aloqasini ta'minlash uchun qo'llaniladigan maxsus simlar;
- ❖ **portlar** – kompyuter qurilmalarini tizimli blokka ulash uchun mo'ljallangan maxsus joylar;
- ❖ **DVD-RW** (*Digital versatile disk rewriteable*) – kompakt disklardagi ma'lumotlarni o'qish va ularga yozish uchun mo'ljallangan qurilma.

Bulardan tashqari tizimli blokka *ichki modem, tarmoq adapteri, tovush adapteri, TV-tyuner* kabi kompyuter imkoniyatlarini oshiradigan ko'plab qurilmalar o'rnatilishi mumkin.

2. Monitor – ma'lumotlarni aks ettiruvchi qurilma. Hozirgi kunda asosan videografikli **SVGA** (Super video graphic array) va tezlik bilan rivojlanayotgan suyuq kristalli **LCD** (Liquid crystal display) turidagi monitorlar ishlatilmoqda. Monitor gorizontal va vertikal koordinatali mayda elektron nuqtalardan tashkil topgan bo'lib, bu nuqtalar *piksellar* deb ataladi. Piksellar soni qanchalik ko'p bo'lsa, tasvir shunchalik aniq ko'rinadi.

3. Klaviatura – kompyuterga ma'lumotlarni kiritish va boshqarish

uchun mo'ljallangan qurilma. U odatda standart 107 ta tugmadan iborat bo'ladi. Klaviatura tugmalari shartli ravishda *ma'lumotlarni kiritish tugmalari* (harflar, sonlar, belgilar), *boshqarish tugmalari* va *funksional tugmalarga* (F1,F2,...,F12) bo'linadi.

4. Sichqoncha – kompyuter ishlarini boshqarish uchun mo'ljallangan qurilma.

izoh

Sichqoncha qurilmasi ilgari qo'shimcha qurilma sifatida qaralgan bo'lsa, Windows OT paydo bo'lgandan so'ng kompyuterning asosiy qurilmalaridan biriga aylandi. Chunki Windows OT paydo bo'lgandan so'ng Fotoshop, Macromedia Flash, 3-Dmax, Corel Draw, AutoCad kabi axborot texnologiyalari imkoniyatlarini belgilaydigan, sichqoncha bilan ishlab bo'lmaydigan zamonaviy dasturlar paydo bo'ldi.

Shaxsiy kompyuterning qo'shimcha qurilmalariga quyidagilar kiradi:

- **printer** – ma'lumotlarni chop etuvchi qurilma. Hozirgi kunda lazerli printerlar ishlab chiqarilmoqda. Bunday printerlarda ma'lumotlar siyohni lazer nurlari yordamida yo'naltirish yordamida hosil qilinadi;
 - **skaner** – ma'lumotlarni kompyuterga kiritish qurilmasi;
 - **plotter** – ma'lumotlarni katta formatdagi qog'ozlarga yoki maxsus listlarga chiqarish qurilmasi;
 - **videoproektor** – kompyuterdagi ma'lumotlarni maxsus doskaga kattalashtirib beruvchi qurilma;
 - **elektron doska** – bir vaqtda oddiy doska va videoproektor sifatida foydalanish mumkin bo'lgan, bunda barcha amallarni bevosita elektron tarzda boshqarish imkonini beruvchi qurilma;
 - **modem** – kompyuterdagi raqamli ma'lumotlarni telefon tarmog'idagi tovushli signallarga va aksincha tovushli signallarni raqamli ma'lumotlarga aylantirib beruvchi qurilma bo'lib, internetga chiqishni ta'minlaydi;
 - **kompakt disk** – ma'lumotlarni tashqi xotirada saqlash qurilmasi. Kompakt diskning har xil imkoniyatlarga ega bo'lgan **CD-R**, **CD-RW**, **DVD-R**, **DVD-RW** kabi turlari mavjud;
 - **flesh-xotira** (fleshka) – ma'lumotlarni tashish uchun mo'ljallangan qurilma;
 - **videoglazok** – maxsus plyonkadagi yoki qog'ozdagi ma'lumotlarni kattalashtirib beruvchi qurilma;
 - **xab** – lokal tarmoqdagi kompyuterlarni birlashtiruvchi qurilma;
- Bulardan tashqari **Joystik** (sichqoncha o'rnida ishlatilib, maxsus o'yinlar o'ynash uchun mo'ljallangan qurilma), **Trekbot** (sharikli boshqaruv tizimiga ega bo'lib, odatda noutbuklarda sichqoncha o'rnida qo'llaniladi), **Digitayzer** (vazifasi skaner qurilmasiga o'xshash bo'lib, yuqori sifatdagi

hamda aniqlikdagi rasm va grafiklarni kompyuterga kiritish uchun mo'ljallangan qurilma), *ovoz chiqarish kolonkalari, mikrofon, raqamli fotoapparat, raqamli videokameralar* kabi ko'plab qurilmalar mavjud.

Kompyuterning dasturiy ta'minoti

Umuman olganda kompyuter ikki qismdan, ya'ni *fizik ta'minot* (hardware) va *dasturiy ta'minot* (software) dan iborat bo'ladi. Fizik qismga kompyuterning barcha qurilmalari kiradi.

Kompyuterning dasturiy ta'minoti ham o'z navbatida 3 qismga bo'linadi:

- o tizimli dasturlar;
- o amaliy dasturlar;
- o dasturlash tizimlari (instrumental dasturlar).

Tizimli dasturlarga kompyuter ishlashini ta'minlovchi, foydalanuvchilarga amaliy dasturlar bilan ishlashni qulaylashtiruvchi quyidagi turdagi dasturlar kiradi:

- operatsion tizimlar;
- qobiq operatsion tizimlar (NC va b.);
- utilitalar;
- drayverlar;
- arxivlash dasturlari;
- antivirus dasturlari va b.

Amaliy dasturlarga kompyuter foydalanuvchilari ishlatadigan barcha dasturlar, xususan, axborotlar bilan ishlaydigan, axborot texnologiyalari qo'llaniladigan, aniq masalalarni yechishga mo'ljallangan quyidagi turdagi dasturlar kiradi:

- matn muharrirlari (Bloknot, WordPad, Word, Tex va b.);
- nashriyot tizimlari (Adobe PageMaker, Ventura Publisher va b.);
- grafik va animatsiya redaktorlari (Paint, Corel Draw, Adobe Photoshop, Adobe Illustrator, Flash, 3DStudioMax va b.);
- avtomatik loyihalash tizimlari (AutoCAD, "Компас" va b.);
- elektron jadvallar (Excel, Lotus, SuperCalc va b.);
- ma'lumotlar bazasi hoshqarish tizimlari (Access, FoxPro, MySQL, Oracle va b.);
- taqdimot dasturlari (PowerPoint, Freelance Graphics va b.);
- matematik dasturlar (MathCAD, Maple, Mathlab va b.);
- dastur paketlari (Microsoft Office, FrameWork va b.);
- moliya va buxgalteriya dasturlari ("1С:Бухгалтерия", RS-BANK, ЭДИП, "АЛЬТ-Финансы" va b.);

- kompyuter o‘yinlari;
 - multimediya dasturlari (MediaPlayer, KMPlayer va b.);
 - videomontaj dasturlari (Movie Maker, Pinnacle Studio va b.);
 - elektron darsliklar, lug‘atlar, ma’lumotnomalar va h.k.
- Dasturlash tizimlariga barcha dasturlarni yaratuvchi, moslashtiruvchi va tuzatuvchi quyidagi turdagi dasturlar kiradi:
- dastur yaratish muhitlari (Delphi, Visual Basic, C++ va b.)
 - translyatorlar;
 - kompilyator va interpretatorlar;
 - tuzatuvchi dasturlar va b.

Windows operatsion tizimi

Windows eng keng tarqalgan, umumiy foydalanuvchilar uchun qulay bo‘lgan operatsion tizimdir.

Operatsion tizim deb foydalanuvchi bilan kompyuter o‘rtasida aloqani ta’minlab beruvchi tizimga aytiladi. Operatsion tizim juda ko‘p dasturlar to‘plamidan iborat bo‘lib, har bir dastur aniq bir vazifani bajaradi. Operatsion tizim (OT) larga Windows, Unix, Linux, MSDOS kabilar misol bo‘ladi.

Windows OT 1983 yildan Microsoft firmasi tomonidan ishlab chiqarila boshlangan bo‘lib, uning juda ko‘p versiyalari yaratilgan. Masalan, Windows 3.0 (1990 y.), Windows 95, Windows 98, Windows NT (2000), Windows XP (2001), Windows 7 (2009), Windows 8 (2012) va b.

Windows OT kompyuter yoqilishi bilan ishga tushadi va ekranda uning ishchi stoli hosil bo‘ladi. **Ishchi stol** asosiy soha va masalalar panelidan iborat.

Asosiy sohada quyidagilar joylashgan:

Мой компьютер – lokal disklar hamda hujjatlar saqlash uchun mo‘ljallangan papka joylashgan obyekt.

Мои документы – hujjatlar saqlash uchun mo‘ljallangan papka.

Корзина – o‘chirilgan fayllar saqlanadigan joy.

Сетевое окружение – lokal tarmoqqa ulangan kompyuterlarga murojaat qilish obyekt.

Internet Explorer – Internet sahifasini ko‘rish brauzeri va b.

Masalalar panelida **Пуск** tugmasi hamda soat, yozuv tili, antivirus kabi doimiy ravishda ishlatiladigan dasturlar yorliqlari joylashgan.

Пуск tugmasi bosilganda **Windowsning bosh menyusi** hosil bo‘ladi. Windowsning bosh menyusida **Мой компьютер**, **Мои документы**, foydalanilgan oxirgi dasturlar, shuningdek quyidagilar mavjud:

Все программы bo'limi orqali dasturlar ishga tushiriladi.

Недавние документы bo'limida foydalanilgan oxirgi hujjatlar ro'yxati joylashgan.

Панель управления bo'limi tizimli va amaliy dasturlarni sozlash, o'rnatish, o'chirish, kompyuter qurilmalarini dasturiy jihatdan bog'lash va sozlash ishlarini bajarishga mo'ljallangan.

Поиск menyusi yordamida ma'lumotlar va obyektlar qidiriladi.

Справка bo'limidan Windows va unda ishlash bo'yicha to'liq ma'lumot olish mumkin.

Выполнить menyusi orqali Windowsning ba'zi buyruqlarini sichqonchasiz va menyusiz tez bajarish mumkin.

Выход из системы buyrug'i boshqa foydalanish muhitiga o'tish va joriy foydalanish muhiti ishini tugallashni ta'minlaydi.

Выключение buyrug'i orqali kompyuterni o'chirish, Windowsni qayta yuklash kabi ishlar bajariladi.

Ma'lumki, "Windows" so'zi "oynalar" degan ma'noni bildiradi. Windowsda barcha dasturlar, obyektlar oynalar va darchalar ko'rinishida ochiladi.

Sichqonchanning chap tugmasi obyekt ustida bir marta bosilsa, obyekt belgilanadi, menyu ustida bosilsa, buyruq yoki dastur ishga tushiriladi. Sichqonchanning chap tugmasi ikki marta bosilganda obyekt ochiladi. Sichqonchanning o'ng tugmasi birov obyekt ustida bosilsa, o'sha obyektning kontekst menyusi ochiladi.

Контекст меню – obyekt ustida bajarish mumkin bo'lgan buyruqlar ro'yxatidir.

Папка – fayllar saqlanadigan maxsus obyekt.

Йорлиқ – obyektga murojaat qilishga imkon beruvchi nishon.

Total Commander qobiq operatsion tizimi

Odatda Windowsda ishlashni tezlashtirish va qulaylashtirish maqsadida qobiq operatsion tizimlardan foydalaniladi. Oldin MSDOS OT uchun ko'proq Norton Commander (NC) ishlatilgan bo'lsa, hozirgi kunda Windows OT uchun Total Commander (TC) qobiq operatsion tizimidan foydalaniladi.

Total Commander kompyuterga o'rnatilgan bo'lsa, uni ishga tushirish yorlig'i *Bosh menyuning Все программы* yoki *Ishchi stol, Masalalar paneli, Пуск tugmasi* kabi qulay menyularda ham hosil bo'ladi (1-rasm).



1-rasm Total Commander interfeysi

Dasturda menyular va uskunlar paneli orqali Windowsdagi deyarli barcha amallarni bajarish mumkin. Shuningdek quyidagi *funksional tugmalar* va tugmalar kombinatsiyalaridan keng foydalaniladi:

F1 – Total Commanderda ishlash bo'yicha to'liq ma'lumotnoma olish;

F2 – fayl yoki katalog nomini o'zgartirish;

F5 – fayl yoki katalogni nusxalash;

F6 – fayl yoki kataloglarni ko'chirish va nomini o'zgartirish;

F7 – yangi katalog yaratish;

F8 – fayl yoki katalogni o'chirish;

Tab – bir paneldan boshqasiga o'tish;

Alt+F1 – chap panelga disklar ro'yxatini chiqarish;

Alt+F2 – o'ng panelga disklar ro'yxatini chiqarish;

Alt+Enter – fayl yoki katalog xossalarini ko'rish;

Total Commander bilan birga Windowsda ham ko'p ishlatiladigan tugmalar kombinatsiyalari:

Ctrl+C – fayl yoki papkani almashinuv buferiga nusxalash;

Ctrl+X – fayl yoki papkani almashinuv buferiga ko'chirish;

Ctrl+V – almashinuv buferidagi fayl yoki papkani qo'yish;

Ctrl+Z – bajarilgan buyruqni bekor qilish;

Ctrl+Y – bekor qilingan buyruqni tiklash;

Ctrl+A – obyektidagi barcha fayl va papkalarni belgilash;

Alt+F4 – joriy fayl, papka yoki dasturni yopish;

Alt+Tab – ochilgan oynalarning keraklisiga o'tish;

Windows+D – ochilgan barcha oynalarni masalalar paneliga yig'ish.

Sanoq sistemalari

Kompyuterda barcha turdagi axborotlar faqat sonli ma'lumotlar ko'rinishida yaratiladi va qayta ishlanadi. Foydalanuvchi uchun u kerakli ko'rinishda, ya'ni matn, son, rasm, video ma'lumotlar shaklida ekrandagina namoyon bo'ladi. Ixtiyoriy bitta belgi kompyuter xotira qurilmasidagi ketma-ket joylashgan sakkizta elektron yacheykani egallaydi. Elektron yacheykada signal bor bo'lsa, 1, signal yo'q bo'lsa, 0 deb belgilash kiritiladi. Demak, barcha ma'lumotlar 0 va 1 raqamlaridan iborat bo'ladi. 0 va 1 raqamlari ketma-ketliklari va ular ustidagi amallar qoidalari *ikkilik sanoq sistemasini* tashkil etadi. Har qanday belgi bir *bayt*, bitta signal bir *bit* hisoblanadi. Demak, bitta belgi sakkizta 0 va 1 laming biror ketma-ketligidan iborat bo'ladi. Barcha harflar, sonlar, standart buyruqlar, ranglar, belgilar (jami 256 ta) ASCII (American standart code information interchange) yoki ANSI (American national standart institute) kodlash tizimi asosida 0 va 1 lar ko'rinishida ifodalanadi.

Axborotlarning o'lchov birliklari orasidagi bog'lanish quyidagicha:

$$1 \text{ bayt} = 8 \text{ bit},$$

$$1 \text{ Kb (kilobayt)} = 2^{10} \text{ bayt},$$

$$1 \text{ Mb (megabayt)} = 2^{10} \text{ Kb} = 2^{20} \text{ bayt},$$

$$1 \text{ Gb (gegabayt)} = 2^{10} \text{ Mb} = 2^{20} \text{ Kb} = 2^{30} \text{ bayt},$$

$$1 \text{ Tb (terabayt)} = 2^{10} \text{ Gb} = 2^{20} \text{ Mb} = 2^{30} \text{ Kb} = 2^{40} \text{ bayt}.$$

Izoh

Yuqoridagilardan tashqari Pb (petabayt; 1 Pb = 2^{50} bayt), Eb (eksabayt; 1 Eb = 2^{60} bayt), Zb (zetabayt; 1 Zb = 2^{70} bayt), Yb (yottabayt; 1 Yb = 2^{80} bayt) kabi axborot o'lchov birliklari ham mavjud.

Ikkilik va kundalik turmushda qo'llaniladigan o'nlik sanoq sistemalaridan tashqari qadimda beshlik, yettilik, sakkizlik sanoq sistemalari ham ishlatilgan. Hozirgi kunda ikkilik sanoq sistemasida hosil bo'ladigan juda katta sonlarni ifodalashni soddalashtirish maqsadida o'n oltilik sanoq sistemasidan ham foydalaniladi. Biz bu sanoq sistemalaridan ikkilik, sakkizlik, o'nlik va o'n oltilik sanoq sistemalarini ko'rib o'tamiz. Bu sistemalardagi natural sonlar ketma-ketligi quyidagi moslikda bo'ladi:

I-jadval. Sanoq sistemalari va ular orasidagi moslik

2 lik	o	1	10	11	100	101	110	111	1000	1001	1010	1011	1100	1101	1110	1111	10000
8 lik	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	20
10 lik	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
16 lik	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10

Ixtiyoriy p sanoq sistemasidagi $a_1 a_2 \dots a_{n(p)}$ butun sonni o nlik sanoq

sistemasiga o'tkazish uchun quyidagi formuladan foydalaniladi:

$$a_1 a_2 \dots a_{n[p]} = a_1 p^{n-1} + a_2 p^{n-2} + \dots + a_n p^0 \quad (1)$$

Agar berilgan son $a_1 a_2 \dots a_n a_{n+1} a_{n+2} \dots a_{n+m} [p]$ ko'rinishidagi kasr son bo'lsa, o'nlik sanoq sistemasiga o'tkazish formulasi quyidagicha bo'ladi:

$$a_1 a_2 \dots a_n a_{n+1} \dots a_{n+m}[p] = a_1 p^{n-1} + a_2 p^{n-2} + \dots + a_n p^0 + a_{n+1} p^{-1} + \dots + a_{n+m} p^{-m} \quad (2)$$

bu formulalarda $a_1, a_2, \dots, a_n, \dots, a_{n+m}$ lar sonni tashkil etuvchi raqamlar, p – berilgan sonning asosi, $1, 2, \dots, n$ lar berilgan sonning butun qismi raqamlari nomerlari, $n+1, n+2, \dots, n+m$ lar berilgan sonning kasr qismining raqamlari nomerlari.

O'nlik sanoq sistemasida berilgan butun sonni ixtiyoriy sanoq sistemasiga o'tkazish uchun berilgan son o'tilayotgan sistema asosiga bo'linma bo'luvchidan kichik bo'lguncha bo'lib boriladi va bo'linma qoldiqlar bilan teskari tartibda olinadi.

O'nlik sanoq sistemasida berilgan o'nli kasr ko'rinishidagi sonni ixtiyoriy sanoq sistemasiga o'tkazish uchun berilgan sonning butun qismi alohida, kasr qismi alohida o'tkaziladi va chiqqan natijalar qo'shiladi. Bunda, berilgan sonning butun qismi o'tilayotgan sistema asosiga bo'linma bo'luvchidan kichik bo'lguncha bo'lib boriladi va bo'linma qoldiqlar bilan teskari tartibda olinadi, berilgan sonning kasr qismi esa o'tilayotgan sistema asosiga toki kasr qism yo'qolguncha ketma-ket ko'paytirib boriladi va natijalarning butun qismlari ketma-ket olinadi, bunda kasr qismiga ko'paytirilayotganda butun qismi doimo nol deb qaraladi.

Ikkilik sanoq sistemasidan sakkizlikka, sakkizlikdan o'n oltilikka o'tkazish kabi hollarda, berilgan son avvalo o'nlikka o'tkazilib, hosil bo'lgan o'nlikdagi son kerakli sistemaga o'tkaziladi.

Misolalar

Sonlarni bir sanoq sistemasidan boshqasiga o'tkazish

1) $1011011_{[2]}$, $3507_{[8]}$, $D3A_{[16]}$ sonlarini o'nlik sanoq sistemasiga o'tkazing.

$$1011011_{[2]} = 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 64 + 16 + 8 + 2 + 1 = 91_{[10]}$$

$$3507_{[8]} = 3 \cdot 8^3 + 5 \cdot 8^2 + 0 \cdot 8^1 + 7 \cdot 8^0 = 3 \cdot 512 + 5 \cdot 64 + 7 = 1536 + 320 + 7 = 1863_{[10]}$$

$$D3A_{[16]} = D \cdot 16^2 + 3 \cdot 16^1 + A \cdot 16^0 = 13 \cdot 256 + 48 + 10 = 3386_{[10]}$$

2) $91_{[10]}$, $1863_{[10]}$, $3386_{[10]}$ sonlarini mos ravishda ikkilik, sakkizlik va o'n oltilik sanoq sistemalariga o'tkazing.

$$\begin{array}{r}
 91 \overline{) 2} \\
 90 \overline{) 45} \overline{) 2} \\
 \underline{1} \overline{) 44} \overline{) 22} \overline{) 2} \\
 \underline{1} \overline{) 22} \overline{) 11} \overline{) 2} \\
 \underline{0} \overline{) 10} \overline{) 5} \overline{) 2} \\
 \underline{1} \overline{) 4} \overline{) 2} \overline{) 2} \\
 \underline{1} \overline{) 2} \overline{) 1} \\
 \underline{0}
 \end{array}$$

$$91_{(10)} = 1011011_{(2)}$$

$$\begin{array}{r}
 1863 \overline{) 8} \\
 1856 \overline{) 232} \overline{) 8} \\
 \underline{7} \overline{) 232} \overline{) 29} \overline{) 8} \\
 \underline{0} \overline{) 24} \overline{) 3} \\
 \underline{5}
 \end{array}$$

$$1863_{(10)} = 3507_{(2)}$$

$$\begin{array}{r}
 3386 \overline{) 16} \\
 3376 \overline{) 211} \overline{) 16} \\
 \underline{10} \overline{) 208} \overline{) 13} \\
 \underline{3}
 \end{array}$$

$$3386_{(16)} = D3A_{(2)}$$

3) $11,001_{(2)}$, $5,27_{(8)}$ va $S, A_{(16)}$ sonlarini o'nlik sanoq sistemalariga o'tkazing.

$$11,001_{(2)} = 1 \cdot 2^1 + 1 \cdot 2^0 + 0 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} = 3 + 0,125 = 3,125_{(10)}$$

$$5,27_{(8)} = 5 \cdot 8^0 + 2 \cdot 8^{-1} + 7 \cdot 8^{-2} = 5 + 0,25 + 0,109375 = 5,359375_{(10)}$$

$$S, A_{(16)} = C \cdot 16^0 + A \cdot 16^{-1} = 12 + 0,625 = 12,625_{(10)}$$

4) $3,125_{(10)}$, $5,359375_{(10)}$ va $12,625_{(10)}$ sonlarini mos ravishda sakkizlik va o'n oltilik sanoq sistemalariga o'tkazing.

$$\begin{array}{r}
 3 \overline{) 2} \\
 \underline{2} \overline{) 1} \\
 \underline{1}
 \end{array}$$

$$3_{(10)} = 11_{(2)}$$

$$\begin{array}{r}
 0,125 \times 2 \\
 0,25 \times 2 \\
 0,5 \times 2 \\
 1
 \end{array}$$

$$0,125_{(10)} = 0,001_{(2)}$$

$$\text{Demak, } 3,125_{(10)} = 11,001_{(2)}$$

$$\begin{array}{r}
 5 \overline{) 8} \\
 \underline{0} \overline{) 0} \\
 \underline{5}
 \end{array}$$

$$5_{(10)} = 5_{(8)}$$

$$\begin{array}{r}
 0,359375 \times 8 \\
 2,875 \times 8 \\
 7
 \end{array}$$

$$0,359375_{(10)} = 0,27_{(8)}$$

$$\text{Demak, } 5,359375_{(10)} = 5,27_{(8)}$$

$$\begin{array}{r}
 12 \overline{) 16} \\
 \underline{0} \overline{) 0} \\
 \underline{12}
 \end{array}$$

$$12_{(10)} = C_{(16)}$$

$$\begin{array}{r}
 0,625 \times 1 \\
 10
 \end{array}$$

$$0,625_{(10)} = 0, A_{(16)}$$

$$\text{Demak, } 12,625_{(10)} = C, A_{(16)}$$

Sanoq sistemalaridagi sonlar ustida amallar

1) Ikki sanoq sistemasidagi quyidagi amallarni bajaring:

a) $1011101_{(2)} + 111011_{(2)} = x_{(2)}$, $x_{(2)} - 111011_{(2)} = y_{(2)}$, x, y larni toping.

$$\begin{array}{r}
 1011101_{(2)} \\
 + 111011_{(2)} \\
 \hline
 10011000_{(2)}
 \end{array}$$

Demak,

$$b) 101_{(2)} \cdot 111_{(2)} = x_{(2)},$$

$$x_{(2)} : 111_{(2)} = y_{(2)}, \quad x, y \text{ larni toping.}$$

$$\begin{array}{r} 101_{[2]} \\ \times 111_{[2]} \\ \hline 101 \\ 101 \\ 101 \\ \hline 100011_{[2]} \end{array}$$

$$\begin{array}{r} 100011 \mid 111 \\ \underline{111} \mid 101 \\ 111 \\ \underline{111} \\ 0 \end{array}$$

Demak, $x=100011_{[2]}$,

$y=101_{[2]}$

2) Sakkizlik sanoq sistemasidagi quyidagi amallarni bajaring:

a) $624_{[8]} + 765_{[8]} = x_{[8]}$, $x_{[8]} - 765_{[8]} = y_{[8]}$, x, y larni toping.

$$\begin{array}{r} 624_{[8]} \\ + 765_{[8]} \\ \hline 1611_{[8]} \end{array}$$

$$\begin{array}{r} 1611_{[8]} \\ - 765_{[8]} \\ \hline 624_{[8]} \end{array}$$

Demak, $x=1611_{[8]}$,

$y=624_{[8]}$

b) $45_{[8]} : 34_{[8]} = x_{[8]}$, $x_{[8]} \cdot 34_{[8]} = y_{[8]}$, x, y larni toping.

$$\begin{array}{r} 45_{[8]} \\ \times 34_{[8]} \\ \hline 224 \\ 157 \\ \hline 2014_{[8]} \end{array}$$

$$\begin{array}{r} 2014 \mid 34 \\ \underline{160} \mid 45 \\ 214 \\ \underline{214} \\ 0 \end{array}$$

Demak, $x=2014_{[8]}$,

$y=45_{[8]}$

3) O'n oltilik sanoq sistemasidagi quyidagi amallarni bajaring:

a) $9DF_{[16]} + A57_{[16]} = x_{[16]}$, $x_{[16]} - A57_{[16]} = y_{[16]}$, x, y larni toping.

$$\begin{array}{r} 9DF_{[16]} \\ + A57_{[16]} \\ \hline 1436_{[16]} \end{array}$$

$$\begin{array}{r} 1436_{[16]} \\ - A57_{[16]} \\ \hline 9DF_{[16]} \end{array}$$

Demak, $x=1436_{[16]}$,

$y=9DF_{[16]}$

b) $6B_{[16]} : 12_{[16]} = x_{[16]}$, $x_{[16]} \cdot 6B_{[16]} = y_{[16]}$, x, y larni toping.

$$\begin{array}{r} 6B_{[16]} \\ \times 12_{[16]} \\ \hline D6 \\ 6B \\ \hline 786_{[16]} \end{array}$$

$$\begin{array}{r} 786 \mid 6B \\ \underline{6B} \mid 12 \\ D6 \\ \underline{D6} \\ 0 \end{array}$$

Demak, $x=786_{[16]}$

$y=12_{[16]}$

Algoritmlar

“Algoritm” so‘zi buyuk o‘zbek matematigi Al-Xorazmiy nomidan olingan bo‘lib, u birinchi bo‘lib arab raqamlaridan foydalangan holda arifmetik amallarni bajarish qoidasini bayon etgan.

Algoritm – qo‘yilgan masalani yechimga olib keluvchi amallarning mantiqiy ketma-ketligidir.

Har qanday masalani yechish uchun, xususan dastur tuzish uchun avvalo algoritm tuzish maqsadga muvofiqdir. Algoritm tuzish masalani yechishning asosiy jarayoni, ya‘ni murakkab qismi hisoblanadi. Tayyor algoritimga dastur tuzish esa har qanday malakali dasturchi uchun qiyinchilik tug‘dirmaydi. Bitta masalani yechish uchun turli xil algoritmlar tuzish mumkin va bu algoritmlarning barchasi to‘g‘ri bo‘lishi mumkin. Ammo barcha yuz berishi mumkin bo‘lgan holatlarni hisobga olgan holda yechimga eng qisqa yo‘l bilan, murakkab va ortiqcha amallarsiz olib keluchi optimal algoritm tuzish muhim ahamiyat kasb etadi.

Algoritm quyidagi xossalarga ega: *aniqlik, tushunarlilik, ommaviylik, uzluksizlik, natijaviylik.*

Aniqlik xossasi. Algoritm shunday tuzilgan bo‘lishi kerakki, masaladagi barcha o‘zgaruvchilar va boshqa elementlar aniq ravishda berilgan bo‘lishi lozim.

Tushunarlilik xossasi. Algoritmning barcha qadamlari tushunarli, qisqartirishlarsiz berilishi lozim.

Uzluksizlik xossasi. Algoritm bo‘laklarga ajratilgan va bu bo‘laklar uzluksiz bog‘langan bo‘lishi lozim.

Ommaviylik xossasi. Tuzilgan algoritm bir turdagi masalalarning barchasi uchun o‘rinli bo‘lishi lozim.

Natijaviylik xossasi Algoritmning chekli qadamlaridan so‘ng albatta natijaga erishilishi lozim.

Algoritmlar uchta turga bo‘linadi: *chiziqli, tarmoqlanuvchi va takrorlanuvchi* (siklik) algoritmlar.

Chiziqli algoritmlar deb undagi har bir amallarning hech qanday shart va takrorlanishlarsiz bajariladigan turiga aytiladi.

Tarmoqlanuvchi algoritmlar deb berilgan masalada ba‘zi amallarning ma‘lum bir shartlar asosida bajarilish jarayoni qatnashgan algoritmlarga aytiladi.

Takrorlanuvchi algoritmlar biror-bir shart bajarilguncha yoki biron parametring har xil qiymatlari asosida chekli ravishda takrorlanishlar yuz beradigan jarayonlar qatnashgan algoritmlardir.

Algoritmlar asosan so‘zlar, jadvallar, blok-sxemalar, formulalar,

dasturlar ko‘rinishida yoziladi.

Algoritmnlarni blok-sxema shaklida tasvirlash qulayroq va tushunarliroq bo‘lgani uchun masalaning algoritmlari ko‘proq blok-sxema ko‘rinishida tuziladi.

Blok-sxemalarning asosiy elementlari quyidagilar:



– algoritmning boshlanishi va tugallanishini bildiradi;



– o‘zgaruvchilar qiymatlarini kiritish va natijani chiqarish uchun ishlatiladi;



– shartni tekshirish jarayonida qo‘llaniladi;



– ifodalarni hisoblash va o‘zgarmaslarni e‘lon qilishni bildiradi;



– takrorlanuvchi jarayonlarni ifodalashda ishlatiladi.

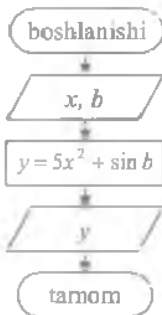
Bloklar gorizontaal va vertikal to‘g‘ri chiziqlar, strelkalar orqali bog‘lanadi.

Chiziqli algoritmlarga doir misollar

1. $y = 5x^2 + \sin b$, bunda $x, b \in \mathbb{R}$ ifodaning qiymatini hisoblash algoritmini tuzing.

Blok sxema ko‘rinishida:

So‘zlar orqali:



1. Algoritm tuzishni boshlash.

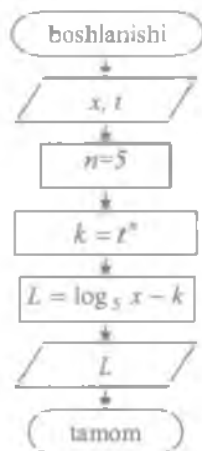
2. x va b larning qiymatlarini kiritish.

3. $y = 5x^2 + \sin b$ ifodani hisoblash.

4. y ni ekranga chiqarish.

5. Algoritm tuzishni tugallash.

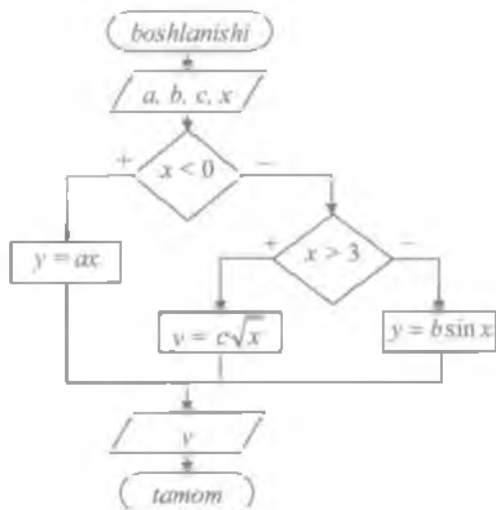
2. $L = \log_5 x - k$, bunda $k = t^n$, $n = 5$, $x, t \in \mathbb{R}$, $x > 0$ ifodaning qiymatini hisoblash algoritmini tuzing.



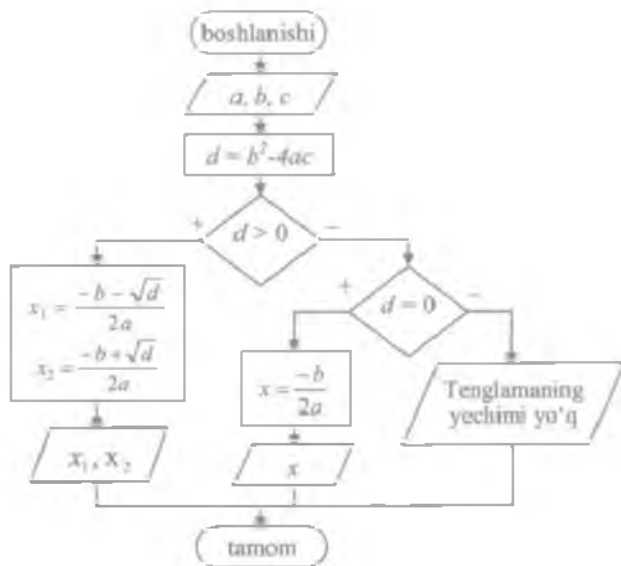
Tarmoqlanuvchi algoritmlarga doir misollar

3. Ifodaning qiymatini hisoblash algoritmini tuzing.

$$y = \begin{cases} ax, & \text{agar } x < 0, \\ b \sin x, & \text{agar } 0 \leq x \leq 3, \text{ bunda } a, b, c, x \text{ lar ixtiyoriy haqiqiy sonlar.} \\ c\sqrt{x}, & \text{agar } x > 3. \end{cases}$$

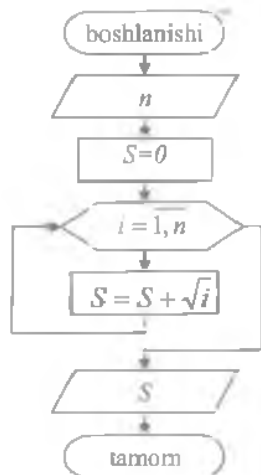


4. $ax^2 + bx + c = 0$ ko'rinishdagi kvadrat tenglamaning ildizlarini topish algoritmini tuzing. Bunda a, b, c lar ixtiyoriy nolga teng bo'lmagan haqiqiy sonlar.

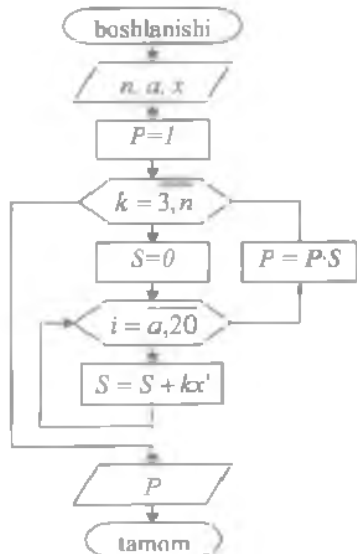


Takrorlanuvchi algoritmlarga doir misollar

5. $S = \sum_{i=1}^n \sqrt{i}$ ifodasini hisoblash algoritmini tuzing.



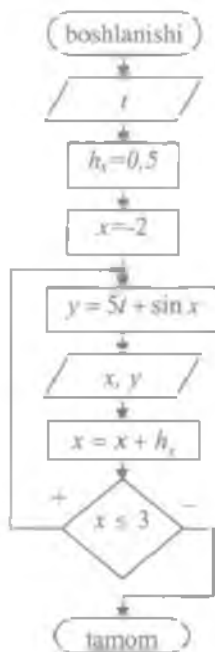
6. $P = \prod_{k=1}^n \sum_{i=1}^{20} kx^i$ ifodasini hisoblash algoritmini tuzing.



7. $y = 5t + \sin x$, bunda

$$-2 \leq x \leq 3, h_x = 0,5$$

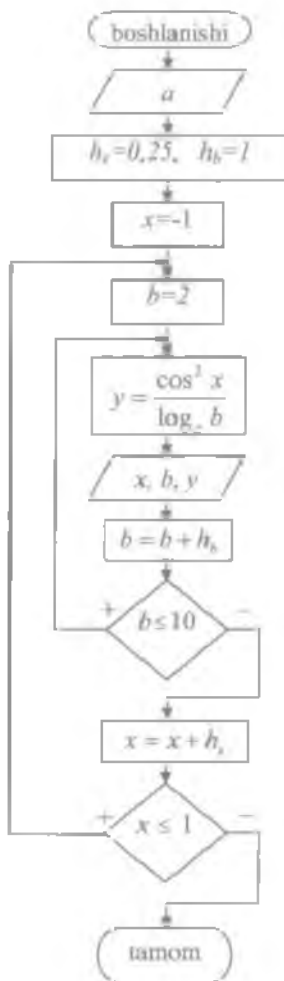
ifodaning qiymatlarini ekranga chiqarish algoritmini tuzing.



3. $y = \frac{\cos^2 x}{\log_a b}$, bunda

$$-1 \leq x \leq 1, h_x = 0,25, 2 \leq b \leq 10$$

ifodaning qiymatlarini ekranga chiqarish algoritmini tuzing.



Delphi dasturlash tili va muhiti, Delphida konsol ilova, strukturasi, elementlari, alifbosi va nomlar

Masalani yechish algoritmi tuzib chiqilgandan so'ng dastur tuziladi.

Dastur – algoritmning kompyuter tushunadigan tilda yozilishidir.

Dasturlash esa dastur tuzish jarayonidir.

Hozirgi kungacha 500 dan ortiq dasturlash tillari yaratilgan. Dasturlash tillari ko'pincha biror sohadagi masalalarni yechuvchi dasturlar yaratishga mo'ljallangan bo'ladi. Masalan, 1954 yilda ishlab chiqilgan **FORTRAN** tili asosan muhandislik va texnik masalalarni, 1959 yilda yaratilgan **KOBOL** tili iqtisodiy masalalarni, 1960 yilda yaratilgan **ALGOL** tili ilmiy-texnik masalalarni, 1965 yilda yaratilgan **Basic** tili, shuningdek **Pascal, C, Java** tillari universal tillar qatoriga kiradi va ular yordamida ilmiy, texnikaviy, muhandislik, iqtisodiy va boshqa sohalardagi masalalarni yechishga mo'ljallangan dasturlar tuzish mumkin.

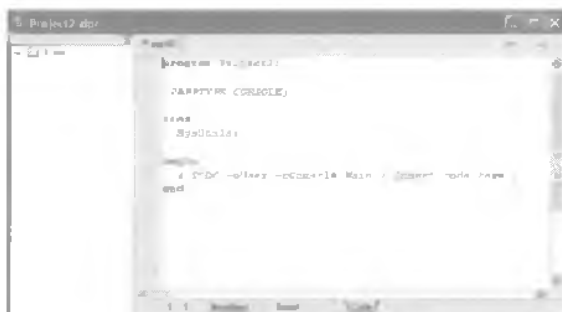
Pascal tili 1971 yilda Shveysariyaning Syurix shahri Oliy texnika maktabi professori Niklaus Virt tomonidan yaratilgan bo'lib, 1979 yildan boshlab birinchi hisoblash mashinasi muallifi fransuz olimi Blez Paskal sharafiga **Pascal tili** deb atala boshlangan. **MS DOS** operatsion tizimi o'rnatilgan kompyuterlarda Pascal tilidagi dasturlar Turbo Pascal, Borland Pascal dasturlash muhitlarida tuzilar edi. Windows operatsion tizimi paydo bo'lgandan so'ng Borland firmasining **Delphi** vizual dasturlash muhiti paydo bo'ldi. Delphi muhitida dasturlar **Object Pascal** tilida tuziladi. Hozirgi kunda Pascal tiliga zamonaviy juda ko'p yangi texnologiyalar tadbiq etilganligi uchun ba'zan bu til Delphi tili deb ham yuritilmoqda.

Delphi orqali turli tipdagi dasturlar – oddiy dasturlardan tortib to ma'lumotlar bazasi hamda internet bilan ishlovchi dasturlar yaratish mumkin.

Delphi dasturlash muhiti *Пуск* → *Программы* → *Borland Delphi* → *Delphi* menyular ketma-ketligi orqali ishga tushiriladi. Natijada Delphining obyektga yo'naltirilgan vizual dasturlash muhiti hosil bo'ladi (9-rasm).

Bu muhitda komponentlar yordamida murakkab vizual dasturlar tuzishdan oldin avvalo dasturlash tilini o'rganish zaruriyati tug'iladi. Delphi tarkibida ana shunday komponentlarsiz dasturlar tuzishga imkon beruvchi, ya'ni Turbo Pascal va Borland Pascal muhitlariga o'xshash muhit mavjud. Bu yerda tuzilgan dasturlar **konsol ilovalar** yoki **konsol dasturlar** deyiladi. Konsol dastur tuzish uchun *File* menyusining *New - Other* buyruqlar ketma-ketligi bosiladi. Hosil bo'lgan oynadan *Console Application* bandi tanlanib, OK tugmasi bosiladi.

Natijada Konsol dasturlash muhiti hosil bo'ladi (2-rasm).



2-rasm. Konsol dasturlash muhiti

Hosil bo'lgan oynaga tuzilgan dastur kiritiladi va **Run** menyusining **Run** buyrug'i yoki F9 tugmasi bosilib natija olinadi.

Delphi tilida **dastur tuzilishi** umumiy holda quyidagicha bo'ladi:

- 1) dastur sarlavhasi;
- 2) nishonlar bo'limi;
- 3) o'zgarmaslar bo'limi;
- 4) turlar bo'limi;
- 5) o'zgaruvchilarni e'lon qilish bo'limi;
- 6) protsedura va funksiyalar;
- 7) dastur tanasi (hisoblash bo'limi).

- Dastur sarlavhasi *Program* xizmatchi so'zi bilan boshlanib, lotin harfi bilan boshlangan nom qo'yiladi.

Masalan, *Program Dastur;*, *Program d12;*, *Program kv_teng;* sarlavha satrlari to'g'ri,

Program 12;, *Program 2dastur;*, *Program dacmyp;*, *Program d 12;*, *Program kv-teng;* sarlavha satrlari noto'g'ri yozilgan.

- *Nishonlar bo'limi* *Label* xizmatchi so'zi bilan boshlanib, nishonlar ro'yxati ketma-ket vergullar orqali yoziladi. Masalan, *Label 1;* *Label 1, 2, nishon1, m;*
- *O'zgarmaslar bo'limi* *Const* xizmatchi so'zi bilan boshlanib, o'zgarmaslar ketma-ket nuqtali vergul bilan yoziladi. Masalan, *Const n=6;*, *m=-5.7;*, *mk='Informatika';*, *b=4*n-trunc(m);*
- *Turlar bo'limida* mavjud turlardan tashqari yoki ular asosida yangi turlar kiritish mumkin.
- *O'zgaruvchilarni e'lon qilish bo'limi* *Var* xizmatchi so'zi bilan boshlanib, o'zgaruvchilar turlari bilan yoziladi. Masalan, *var n:integer;* *b:real;* *a:string;* *t:boolean;*
- *Protsedura va funksiyalar bo'limlarida* qism dasturlar yoziladi. Bu bo'lim alohida mavzuda keltiriladi.
- *Dastur tanasi* dasturning asosiy qismi bo'lib, *begin* va *end* xizmatchi so'zlari orasiga yoziladi va *end* so'zidan keyin nuqta qo'yiladi.

Bu bo'limlardan dastur sarlavhasi va dastur tanasi dastur tuzishda

albatta ishtirok etadi. Qolgan bo'limlarning ishtiroki masala berilishiga va dastur tuzuvchiga bog'liq.

Delphi tili o'z **alfavitiga** ega. Bunga *harflar, raqamlar, maxsus belgilar va ishlatilmaydigan belgilar* kiradi.

– *Harflarga A dan Z gacha va a dan z gacha* lotin harflarining bosh va kichik harflari kiradi.

– *Raqamlarga 0 dan 9 gacha* arab raqamlari va o'n oltilik sanoq sistemasi raqamlari kiradi.

– *Maxsus belgilarni* shartli ravishda *ajratuvchilar, tinish belgilari, amal belgilari* va *xizmatchi so'zlarga* ajratish mumkin.

• *Ajratuvchi* bo'sh joy (probel) tashlanganda, Enter tugmasi bosilganda yoki izoh yozilganda hosil bo'ladi. Izohlar dasturning ixtiyoriy joyida katta qavs ichida berilishi mumkin va u dastur ishlashiga hech qanday ta'sir ko'rsatmaydi. Masalan, `readln(a,b);` {O'zgaruvchilar qiymati kiritilmoqda}. Izohlarni (* *) orasida yoki // belgisidan keyin kiritish ham mumkin.

• *Tinish belgilariga* quyidagilar kiradi:

(,), (*, *), [,], (., .), {, }, «, », ,, ,, ,, //, :=, .., ^, @, \$, #.

• *Amal belgilariga* quyidagilar kiradi: +, -, *, /, =, <, >, <=, >=.

• *Xizmatchi so'zlarga* *and, array, as, asm, begin, case, class, const, constructor, destructor, dispinterface, div, do, downto, else, end, except, exports, file, finalization, finally, for, function, goto, if, implementation, in, inherited, initialization, inline, interface, is, label, library, mod, nil, not, object, of, or, out, packed, procedure, program, property, raise, record, repeat, resource string, set, shl, shr, string, then, threadvar, to, try, type, unit, until, uses, var, while, with, xor* so'zlari kiradi. Bu so'zlarni birini ikkinchisining o'mida ishlatish, o'zgaruvchilarni belgilash mumkin emas.

– *Ishlatilmaydigan belgilarga* masalan rus alfaviti harflarini kiritish mumkin. Ular izohlarda, belgili va qatorli o'zgaruvchilarning qiymatlari sifatida ishlatilishi mumkin.

Nomlar (Identifikatorlar) o'zgaruvchilarni, o'zgartmaslarni, turlarni, protseduralarni, funksiyalarni, sinflarni, modullarni, dastur sarlavhalarini belgilashda ishlatiladi. Nomlar lotin harflari va arab raqamlaridan tashkil topadi hamda lotin harfidan boshlanadi. Nomlarda «_» belgisi ham ishtirok etishi mumkin. Maxsus belgilar va ishlatilmaydigan belgilar nomlar tarkibiga kiritilmaydi. Shuningdek, xizmatchi so'zlar ham nom sifatida qo'llanilmaydi. Mumkin bo'lgan nomlarga misollar: **x, y, dastur_1, max.**

Mumkin bo'lmagan nomlarga misollar: **Idastur** – raqamdan boshlangan; **With** – xizmatchi so'z; **Натужа** – kirill harflaridan iborat; **Dastur.1** – maxsus belgi qatnashgan; **Nishon 1** – bo'sh joy qatnashgan.

Turlar, o'zgarmaslar, o'zgaruvchilar, standart funksiyalar

Turlar

Delphida ma'lumotlar turlari standart va dasturchi tomonidan tuziladigan turlarga bo'linadi.

Standart turlar butun, haqiqiy, belgili, qatorli, mantiqiy turlarga bo'linadi. Bundan tashqari ko'rsatkichli, variant, sanoqli, sana va vaqt kabi standart turlar ham mavjud.

➤ *Butun turlar* butun sonlarni tasvirlashda ishlatiladi. Butun turlar va ularning qiymatlar sohasi quyidagicha:

2-jadval. Butun turlar va ularning qiymatlar sohasi

Tur	Qiymatlar sohasi
<i>Integer</i>	-2147483648 .. 2147483647
<i>Cardinal</i>	0 .. 4294967295
<i>Shortint</i>	-128 .. 127
<i>Smallint</i>	-32768 .. 32767
<i>Logint</i>	-2147483648 .. 2147483647
<i>Int64</i>	$-2^{63} .. 2^{63}-1$
<i>Byte</i>	0 .. 255
<i>Word</i>	0 .. 65535
<i>Longword</i>	0 .. 4294967295

Bulardan *Integer* va *Int64* turlari ko'proq ishlatiladi. Butun o'zgaruvchilarga qiymat kiritilayotganda e'lon qilingan tur qiymatlari sohasidan chiqib ketmasligi lozim. Agar amallar natijasida o'zgaruvchi qiymati ko'rsatilgan diapazondan chiqib ketse, diapazon boshidan boshlanadi. Masalan, o'zgaruvchi *Smallint* turiga tegishli va qiymati 32767 ga teng bo'lsa, unga 1 soni qo'shilganda o'zgaruvchi qiymati -32768 ga teng bo'ladi.

➤ *Haqiqiy turlar* haqiqiy sonlarni ifodalashda ishlatiladi. Haqiqiy turlar va ularning qiymatlar sohasi quyidagicha:

3-jadval. Haqiqiy turlar va ularning qiymatlar sohasi

Tur	Qiymatlar sohasi
<i>Real</i>	$5.0 \cdot 10^{-324} .. 1.7 \cdot 10^{108}$
<i>Real48</i>	$2.9 \cdot 10^{-39} .. 1.7 \cdot 10^{39}$
<i>Single</i>	$1.5 \cdot 10^{-43} .. 3.4 \cdot 10^{38}$
<i>Double</i>	$5.0 \cdot 10^{-324} .. 1.7 \cdot 10^{38}$
<i>Extended</i>	$3.6 \cdot 10^{-4951} .. 1.1 \cdot 10^{4932}$
<i>Comp</i>	$-2^{63}+1 .. 2^{63}+1$
<i>Currency</i>	-922337203685477.5808 .. 922337203685477.5807

Haqiqiy sonlar kiritilganda va hosil qilinganda berilgan diapazondan chiqib ketmasligi lozim, aks holda xatolik chiqaradi yoki ishlamaydi. Haqiqiy sonlar uchun tur tanlashda kerakli aniqlikka e'tibor beriladi.

- *Belgili turlar* Delphi alfavitidagi ixtiyoriy bitta simvolni saqlashga mo'ljallangan va ularning qiymatlari bittalik qo'shtimoq (apostrof) ichiga olib yoziladi. Belgili turlarning ANSIChar, WideChar, Char turlari mavjud. Belgilar ASCII yoki ANSI kodlash tizimiga ko'ra o'z tartib raqamlariga ega.
- *Qatorli turlar* ham belgili turlar kabi ixtiyoriy simvollarini saqlaydi va ularning qiymatlari *String* turiga tegishli bo'ladi. Ammo bunda simvollar soni bittadan ortiq bo'lishi mumkin. Ularning qiymatlari ham apostrof ichiga olib yoziladi.
- *Mantiqiy turlar* faqat ikkita qiymatni qabul qilishi mumkin: *true* (rost) va *false* (yolg'on). Mantiqiy turlarning Boolean, ByteBool, WordBool, LongBool turlaridan asosan Boolean turi qo'llaniladi.

Yuqoridagi turlardan tashqari dasturchi mustaqil ravishda dastur strukturasi turlar bo'limida yangi tur kiritishi mumkin. Masalan, Type Rang = (Qizil, yashil, ko'k, sariq, oq); fasl = (bahor, yoz, kuz, qish); raqam = 0..9; Shundan so'ng o'zgaruvchilarni e'lon qilish bo'limida masalan, quyidagicha yozish mumkin: var a:fasl; b:raqam; n:raqam;

O'zgaruvchilar va o'zgarmaslar

Dasturlash jarayonida qiymati o'zgaraydigan kattaliklar ***o'zgarmaslar*** deyiladi. O'zgarmaslar ixtiyoriy turdagi ma'lumotlarni qabul qilishi mumkin. Ular Const xizmatchi so'zi bilan boshlanib, o'zgarmaslar qiymatlari bilan beriladi. Masalan, const n=6; m=-5.7; mk='Informatika'; b=4*n-trunc(m);

Dastur ishlashi mobaynida qiymatlari o'zgarishi mumkin bo'lgan kattaliklar ***o'zgaruvchilar*** deyiladi va ular dasturda Var so'zi bilan e'lon qilinadi. Masalan, var n:integer; b:real; a1:string; t:boolean;. Agar bir nechta o'zgaruvchilar bir turga mansub bo'lsa, ularni birgalikda e'lon qilish ham mumkin. Masalan, var m,n:integer;

Delphida bazasida oldindan kiritilgan standart funksiya va protseduralar mavjud. Ulardan dastur tuzishda to'g'ridan-to'g'ri foydalanish mumkin. Standart funksiyalarning matematik, turlarni almashtiruvchi, qatorlar bilan ishlovchi, massiv sonlarini qayta ishlovchi, fayllarni boshqaruvchi, protsedura va funksiyalarga doir turlari mavjud. Masalan ba'zi matematik standart funktsiyalar 4-jadvalda keltirilgan.

Matematikadagi boshqa funksiyalar jadvalda keltirilgan standart funksiyalardan foydalanib yoziladi.

Standart funksiya	Nomi va matematik ko'rinishi	Argument turi	Natija turi
Abs(x)	Absolyut qiymat $- x $	Butun yoki haqiqiy	Argument turiga mos
Arctan(x)	Arktangens – $Arctgx$	Butun yoki haqiqiy	Haqiqiy
Cos(x)	Kosinus – $cosx$	Butun yoki haqiqiy	Haqiqiy
Exp(x)	Eksponenta – e^x	Butun yoki haqiqiy	Haqiqiy
Frac(x)	Argumentning kasr qismi	Butun yoki haqiqiy	Haqiqiy
Int(x)	Argumentning butun qismi	Butun yoki haqiqiy	Butun
Ln(x)	Natural logarifm – $\ln x$	Butun yoki haqiqiy	Haqiqiy
Pi	π soni: 3,14159265358...	Argumenti yo'q	Haqiqiy
Round(x)	Argumentga yaqin butun son	Butun yoki haqiqiy	Butun
Sin(x)	Sinus – $\sin x$	Butun yoki haqiqiy	Haqiqiy
Sqr(x)	Argument kvadrati – x^2	Butun yoki haqiqiy	Argument turiga mos
Sqrt(x)	Kvadrat ildiz – \sqrt{x}	Butun yoki haqiqiy	Haqiqiy
Trunc(x)	Argumentning butun qismi	Butun yoki haqiqiy	Butun

Ko'p ishlatiladigan ba'zi boshqa standart funksiyalar va protseduralar:

- *Ord(a)* – a simvolning kodlash tizimidagi tartib raqamini hosil qiladi;
- *Chr(i)* – i tartib raqamli simvolni hosil qiladi;
- *IntToStr(k)* – k butun sonni qatorli turga aylantiradi;
- *StrToInt(t)* – qatorli turdagi butun sonni butun turga aylantiradi;
- *FloatToStr(x)* – x haqiqiy sonni qatorli turga aylantiradi;
- *StrToFloat(t)* – qatorli turdagi haqiqiy sonni haqiqiy turga aylantiradi;
- *Succ(t)* – sanoqli turdagi t dan oldingi pozitsiyada turgan qiymatni aniqlaydi;
- *Pred(t)* – sanoqli turdagi t dan keyingi pozitsiyada turgan qiymatni aniqlaydi.

Delphi da ifodalarni hisoblashlarda quyidagi amal belgilaridan foydalaniladi.

5-jadval. Amal belgilari va ularning ustunlik darajalari

Amallar	Ustunlik darajasi
@, not	1
*, /, div, mod, and, shl, shr, as	2
+, -, or, xor	3
=, <, <=, >, >=, in, is	4

Oddiy qavslar yordamida amallarning bajarilish ketma-ketligini o'zgartirish mumkin.

Operatorlar, o'zlashtirish operatori, tarkibiy va bo'sh operator, ma'lumotlarni kiritish va chiqarish protseduralari

Dastur tuzishda asosiy buyruqlar va amallar operatorlar yordamida amalga oshiriladi. Operatorlarni shartli ravishda oddiy va murakkab operatorlarga bo'lish mumkin. *Oddiy operatorlar* deganda tarkibida qisman operatorlar bo'lmagan operatorlar tushuniladi. Bularga o'zlashtirish operatori, o'tish operatori, bo'sh operator, protsedura operatorlari kiradi. Murakkab operatorlarga tarkibiy operator, shartli operator (if), tanlash operatori (case), parametrli takrorlanish operatori (for), sharti oldin va sharti keyin beriladigan takrorlanish operatorlari (while, repeat) va bog'lash operatori (with) kiradi.

O'zlashtirish operatori. O'zlashtirish operatori dasturlash jarayonida o'zgaruvchiga yangi qiymat berishda, ifodalarni o'zlashtirishda ishlatiladi.

Umumiy ko'rinishi quyidagicha: $A:=B$;

bunda: A – o'zgaruvchi;

B – ifoda yoki son.

O'zgaruvchi va o'zlashtirilayotgan ifodaning (sonning) turi bir xil yoki bir-biriga mos bo'lishi lozim. Masalan, A butun turdagi o'zgaruvchi bo'lsa, B ham butun son yoki natijasi doimo butun son chiqadigan ifoda bo'lishi lozim. Agar A haqiqiy turdagi o'zgaruvchi bo'lsa, u holda B butun yoki haqiqiy turdagi son yoki natijasi doimo butun yoki haqiqiy son chiqadigan ifoda bo'lishi lozim.

O'zlashtirish operatorini qo'llashga doir misollar:

$z:=b+c$; (b va c butun bo'lsa, z ham butun, b va c dan birortasi yoki ikkisi ham haqiqiy bo'lsa, z haqiqiy turda bo'ladi);

$y:=\cos(\exp(3*\ln(x)))$; (x butun yoki haqiqiy, y haqiqiy tur);

$t:=a<5 \text{ and } c>0$; (a va c butun yoki haqiqiy, t mantiqiy tur);

$s1:=\text{'Axborot'}$; ($s1$ qatorli tur).

Tarkibiy operator. Tarkibiy operator deganda *begin* va *end* xizmatchi so'zlari orasiga olib yoziladigan operatorlar guruhi tushuniladi. Bunda *begin* va *end* xizmatchi so'zlari *operatorlar qavisi* deb ham yuritiladi. Tarkibiy operator bir nechta operatorlarni bitta operator sifatida tasvirlash uchun ishlatiladi.

Dastur tanasida har qanday ifodadan so'ng nuqtali vergul qo'yiladi, hech qanday operatorsiz nuqtali vergul qo'yilsa, **bo'sh operator** yozilgan deyiladi. Ammo ifodadan so'ng *end*, *until*, *else* xizmatchi so'zlari kelsa, bu xizmatchi so'zlardan oldin nuqtali vergul qo'yilmaydi.

Ma'lumotlarni kiritish va chiqarish. Konsol ilovada ma'lumotlarni klaviatura yordamida kiritish *read* va *readln*, ekranga chiqarish *write* va

writeln standart protseduralari orqali amalga oshiriladi.

Ma'lumotlarni kiritish protsedurasining umumiy ko'rinishlari quyidagicha:

Read(c1,c2,...,cn);

Readln(c1,c2,...,cn);

Readln;

Bu yerda s_1, s_2, \dots, s_n – o'zgaruvchilar nomlari.

Protseduradagi n ta qiymat ketma-ket probellar yordamida yoki *Enter* tugmasi orqali kiritilgandan so'ng *Enter* tugmasini bosish lozim va bu qiymatlar avtomatik s_1, s_2, \dots, s_n o'zgaruvchilarga o'zlashtiriladi.

Read(c1,c2,...,cn); protsedurasi yozilganda kiritilgan n ta qiymat mos ravishda c_1, c_2, \dots, c_n larga o'zlashtiriladi va qator davomida ma'lumotlar kiritilishi va chiqarilishi mumkin. Bunda cn o'zgaruvchi qiymatidan keyin shu qatorda kiritilgan ortiqcha qiymatlar keyingi kiritish protsedurasi o'zgaruvchilariga o'zlashtiriladi.

Readln(c1,c2,...,cn); protsedurasi yozilganda kiritilgan n ta qiymat mos ravishda c_1, c_2, \dots, c_n larga o'zlashtiriladi va kursor pozitsiyasi avtomatik ravishda keyingi qatorga o'tadi. Bunda cn o'zgaruvchi qiymatidan keyin shu qatorda kiritilgan ortiqcha qiymatlar hisobga olinmaydi hamda hech qanday o'zgaruvchiga o'zlashtirilmaydi.

Readln; protsedurasi yozilganda kompyuter klaviaturadagi ixtiyoriy tugma bosilishini kutadi va kiritilgan simvolni hech qanday o'zgaruvchida saqlamaydi.

Ma'lumotlarni va o'zgaruvchilar qiymatlarini ekranga chiqarish protsedurasining umumiy ko'rinishlari quyidagicha:

Write(t1,t2,...,tn);

Writeln(t1,t2,...,tn);

Writeln;

Bu yerda t_1, t_2, \dots, t_n – o'zgaruvchilar nomi yoki Delphi tili alfavitidagi ixtiyoriy turdagi ma'lumotlar. Protseduradagi n ta o'zgaruvchilar qiymatlari yoki ma'lumotlar ekranga chiqadi.

Write(t1,t2,...,tn); protsedurasi yozilganda t_1, t_2, \dots, t_n o'zgaruvchilar qiymatlari yoki ma'lumotlar yonma-yon ekranga chiqadi va shu qatorda boshqa ma'lumotlar ham chiqarilishi hamda o'zgaruvchilar qiymatlari kiritilishi mumkin.

Writeln(t1,t2,...,tn); protsedurasi yozilganda t_1, t_2, \dots, t_n o'zgaruvchilar qiymatlari yoki ma'lumotlar yonma-yon ekranga chiqadi va kursor pozitsiyasi avtomatik ravishda keyingi qatorga o'tadi.

Writeln; protsedurasi kursor pozitsiyasini keyingi qatorga o'tkazishni ta'minlaydi.

Masalan, $a=3$, $b=5$, firma='Intel' bo'lsin. Konsol ilovaga quyidagi protseduralar kiritilsa, ekranda natijalar ushbu ko'rinishda bo'ladi.

Protseduralar	Natijalar
writeln(a,b);	35
Writeln(a, b);	35
Writeln(a, ' ', b);	3 5
Writeln('a=',a, ' b=',b);	A=3, b=5
Writeln(firma, ' firmasi');	Intel firmasi
Writeln('a>b ', a>b);	a>b FALSE

Ekranga chiqadigan ma'lumotlarning ko'rinishini yaxshilash uchun o'zgaruvchilarga yoki ma'lumotlarga ma'lum bir format qo'llash mumkin. Bu holda ekranga chiqarish protsedurasining ko'rinishi quyidagicha bo'ladi:

Writeln(y1:k1,y2:k2,...,yn:kn);

bunda $y1, y2, \dots, yn$ – o'zgaruvchilar yoki ma'lumotlar;

$k1, k2, \dots, kn$ – qiymatlar egallaydigan pozitsiyalar miqdorlari.

Masalan, $a:=5$; $b:=27$; $c:=362$; bo'lsin. Konsol ilovaga quyidagi protseduralar kiritilsa, ekranda natijalar ushbu ko'rinishda bo'ladi.

Protseduralar	Natijalar
writeln(a,b,c);	5
writeln(a:2,b:2,c:2);	
writeln(a:3,b:3,c:3);	
writeln(a:4,b:4,c:4);	
writeln(7:2,'a',3,3>2:5);	7 a T R U E

Olingan natijalardan ko'rinadiki, agar ajratilgan pozitsiya miqdori ma'lumot kengligidan ortiq bo'lsa, ma'lumot o'ng tomonidan tekislanadi, kam bo'lsa pozitsiya miqdori avtomatik ma'lumot kengligiga o'zgaradi.

Haqiqiy sonlar uchun bu protsedura quyidagicha bo'ladi.

Writeln(y1:k1:l1,y2:k2:l2,...,yn:kn:ln);

bunda $k1, k2, \dots, kn$ – haqiqiy sonlarning umumiy pozitsiyalari miqdorlari;

$l1, l2, \dots, ln$ – kasr qismlarining pozitsiyalari miqdorlari.

Masalan, $a:=5,21$; $b:=26,37$; bo'lsin. Konsol ilovaga quyidagi protseduralar kiritilsa, natijalar ushbu ko'rinishda bo'ladi.

Protseduralar	Natijalar
Writeln(a);	5
Writeln(b);	2
Writeln(a:4:2,b:6:2);	5
Writeln(a:7:3,b:5:1);	
Writeln(a:2:3, ' ', b:3:3);	5
Writeln('a=',a:5:1, ' b=',b:3:0);	a = 5.21, b = 26

Bunda sonning kasr qismi pozitsiya miqdoriga mos ravishda chiqadi.

Shuning uchun ham sonning umumiy pozitsiyasi miqdori kasr qismi miqdoridan kamida ikkita ortiq yozilishi lozim. Aks hollarda kompyuter xato chiqarmasligi uchun sonning butun qismini ham to'liq ekranga chiqaradi.

Misollar

1. $y = 5x^2 + \sin b$, $x, b \in \mathbf{R}$ ifodani hisoblash dasturi tuzilsin.

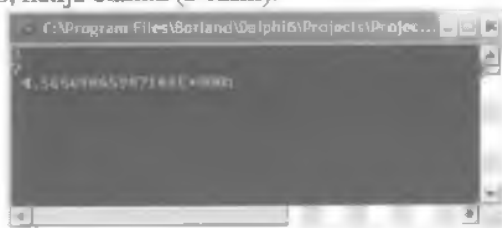
Bu ifodaga algoritmlar mavzusida blok-sxema tuzilgan edi. Dastur tuzish uchun Delphi'ni ishga tushirib, *File/New/Other/Console Application* menyular orqali konsol ilova yaratamiz. Bunda quyidagi ilova loyihasi hosil bo'ladi:

```
program Project2;  
{$APPTYPE CONSOLE}  
uses SysUtils;  
begin  
  { TODO -oUser -cConsole Main : Insert code here }  
end.
```

Bu loyihada dasturning umumiy ko'rinishi tayyor. Tuzilgan algoritimga ko'ra dasturga kerakli bo'limlar va operatorlarni kiritamiz.

```
program Project2;  
{$APPTYPE CONSOLE}  
uses SysUtils;  
var x,b,y:real;  
begin  
  readln(x,b);  
  y:=5*sqr(x)+sin(b);  
  writeln(y);  
  readln  
end.
```

F9 tugmasini bosib, hosil bo'lgan oynada masalan, x ga 3, b ga 7 qiymatni kiritib, natija olamiz (3-rasm).



3-rasm. Dastur natijasi

Demak, natija taxminan 45,657 ga teng ekan.

Dasturdagi `{$APPTYPE CONSOLE}` (ya'ni Application type Console – konsol turidagi loyiha) so'zi konsol ilovaligini bildirish uchun Delphi tomonidan avtomatik qo'shiladi. Uni o'chirish mumkin emas. `uses SysUtils;` so'zi ham kerakli modulardan foydalanish uchun ochilgan. Bu so'zni ham o'chirmagan ma'qul.

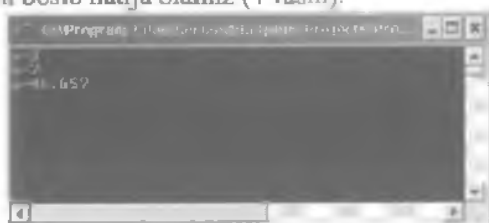
Tuzilgan dastur eng qisqa ko'rinishdir. Foydalanuvchilar uchun tushunarliroq bo'lishi uchun dasturga ba'zi izohlar qo'shamiz:

```

program Project1;
{$APPTYPE CONSOLE}
uses SysUtils;
var x,b,y:real; {o'zgaruvchilar e'lon qilinmoqda}
begin
  write('x='); readln(x); {o'zgaruvchilar qiymati kiritilmoqda}
  write('b='); readln(b);
  y:=5*sqr(x)+sin(b);
  writeln('y=',y:6:3); readln
end.

```

F9 tugmasini bosib natija olamiz (4-rasm).



4-rasm. Dastur natijasi

Dasturni File menyusining Save buyrug'i orqali saqlab qo'yish mumkin.

2. $L = \log_5 x - k$, bunda $k = t^n$, $n = 5$, $x, t \in \mathbb{R}, x > 0$ ifodaning qiymatini hisoblash dasturini tuzing.

```

program Project1;
{$APPTYPE CONSOLE}
Uses SysUtils;
const n=5;
var x,t,k,L:real;
begin
  write('x=');readln(x);
  write('t=');readln(t);
  k:=exp(n*ln(t));
  L:=ln(x)/ln(5) - k;
  writeln('L=',L:6:2); readln
end.

```

Shartli operator, o'tish operatori

Shartli operator. Hayotdagi masalalar doimo to'g'ridan-to'g'ri osongina hisoblanavermaydi. Ularni yechish jarayonida ko'plab shartlarga duch kelamiz. Shartlarning bajarilish yoki bajarilmasligiga qarab u yoki bu jarayonlar hisoblanadi. Delphi tilida shartlarni tekshirish uchun *if* shartli operatori mavjud bo'lib, uning umumiy ko'rinishda quyidagicha:

If A then B1 else B2;

bunda: *A* – mantiqiy ifoda;

B1 va *B2* – operatorlar;

if – agar, *then* – bo'lsa, *else* – aks holda ma'nolarini bildiradi. Agar *A* mantiqiy ifoda *true* qiymat qabul qilsa *B1* operator, aks holda, ya'ni *false* qiymat qabul qilsa, *B2* operator bajariladi.

Shartni tekshirishda quyidagi mantiqiy belgilar ishlatilishi mumkin:

> (katta), < (kichik), <= (kichik yoki teng), >= (katta yoki teng), <> (teng emas), = (teng).

Bundan tashqari quyidagi mantiqiy amal va bog'lanishlar ham ishlatiladi:

Not (inkor), *And* (mantiqiy va), *Or* (mantiqiy yoki).

Masalan, *M1* va *M2* mantiqiy ifodalar quyidagi ko'rinishda berilganda, ularning natijalari va mantiqiy amallar bajarilish natijalarini yozaylik:

M1	M2	M1 natijasi	M2 natijasi	M1 and M2	M1 or M2	Not M1
5>3	Abs(x)>=x	True	True	True	True	False
'd'>'c'	Sqr(2)<3.5	True	False	False	True	False
5<3	'b'>'a'	False	True	False	True	True
'd'<'c'	'n'<'a'	False	False	False	False	True

Mantiqiy ifodalarni biror mantiqiy o'zgaruvchiga o'zlashtirish ham mumkin. Masalan, $F := (2 * pi > sqrt(pi)) \text{ or } (5 < exp(2))$; bunda *F* o'zgaruvchining natijasi *True* bo'ladi.

Yuqoridagi umumiy ko'rinishdagi shartli operator *to'liq ko'rinishdagi shartli operator* deyiladi. Agar faqat *A* shart bajarilganda biror operator berilgan bo'lsa, quyidagi qisqa ko'rinishdagi shartli operatoridan foydalaniladi:

If A then B;

Bunda *A* shart bajarilmasa hech qanday operator bajarilmaydi.

Agar shart bajarilmaganda operator berilgan bo'lsa, qisqa ko'rinishdagi shartli operatorni quyidagicha yozish mumkin:

If not(A) then B;

Bajariladigan operatorlar bir nechta shartlarga bog'liq bo'lsa,

shartlarning har bir qavs ichiga olinib, *and* va *or* mantiqiy bog'lanishlar yordamida yoziladi.

Masalan, x ning qiymati $(-2; 2)$ oraliqda bo'lsa, $y=2x$ ifoda bajarilsin. Bu shartli ifoda dasturda quyidagicha ifodalanadi:

if $(x > -2)$ *and* $(x < 2)$ *then* $y := 2 * x$;

Shartli operatoridagi bajariladigan operator bitta emas bir nechta bo'lsa, ular tarkibiy operator sifatida yoziladi, ya'ni *begin* va *end* xizmatchi so'zlari orasiga olib yoziladi.

Masalan, *if* $x > 5$ *then begin* $y := 2 * x$; $z := \ln(x)$ *end else begin* $y := \text{sqr}(x)$; $z := \text{exp}(x)$ *end*;

Shartli operator tarkibida yana shartli operator kelishi mumkin.

Masalan, *if* $A1$ *then* $B1$ *else if* $A2$ *then* $B2$ *else* $B3$;

O'tish operatori. Dasturlash jarayonida ba'zi hollarda boshqaruvni dasturning biror qismiga to'g'ridan-to'g'ri o'tkazishga to'g'ri keladi. Bunday holatlarda shartsiz o'tish operatoridan foydalaniladi.

Shartsiz o'tish operatorining umumiy ko'rinishi quyidagicha:

Goto n ;

bu yerda: n – nishon nomi;

goto – o'tish ma'nosini bildiradi.

Nishon sifatida identifikatorlar yoki natural sonlar ishlatilishi mumkin.

Masalan, *goto* N ; yoki *goto* 7 ;

Nishon dasturning yuqori qismida *Label* xizmatchi so'zi yordamida e'lon qilinishi lozim.

Masalan, *label* $n, 7$;

Boshqaruv o'tiladigan joyga nishon ikki nuqta yordamida qo'yiladi.

Masalan, n : yoki 7 : kabi.

Misollar

1. $ax^2 + bx + c = 0$ kvadrat tenglamaning ildizlarini topish dasturi tuzilsin.

```
program Project1;
```

```
var a,b,c,d,x,x1,x2:real;
```

```
begin
```

```
  writeln('kvadrat tenglamaning a,b,s koeffitsiyentlarini kiriting');
```

```
  readln(a,b,c);
```

```
  if a=0 then
```

```
    begin x:=-c/b; writeln('x=',x:4:2); end
```

```
  else
```

```
    begin
```

```
      d:=sqr(b)-4*a*c;
```

```

if d>0 then
  begin
    x1:=(-b-sqrt(b))/(2*a);
    x2:=(-b+sqrt(b))/(2*a);
    writeln('x1=',x1:4:2,' x2=',x2:4:2)
  end
else
  if d=0 then
    begin x:=-b/(2*a); writeln('x=',x:4:2); end
  else writeln('tenglama yechimga ega emas')
end;
readln
end.

```

Bu dasturda o'tish operatoridan foydalanish ham mumkin:

```

program Project1;
label 1;
var a,b,c,d,x,x1,x2:real;
begin
  writeln('kvadrat tglamaning a,b,s koeffitsiyentlarini kiriting');
  readln(a,b,c);
  if a=0 then
    begin
      x:=-c/b;
      writeln('x=',x:4:2); goto 1
    end;
  d:=sqrt(b)-4*a*c;
  if d>0 then
    begin
      x1:=(-b-sqrt(b))/(2*a);
      x2:=(-b+sqrt(b))/(2*a);
      writeln('x1-',x1:4:2,' x2=',x2:4:2); goto 1
    end
  if d=0 then
    begin
      x:=-b/(2*a);
      writeln('x=',x:4:2); goto 1;
    end
  writeln('tenglama yechimga ega emas');
1: readln
end.

```

Tanlash operatori

Ko'p hollarda biror-bir parametrning qiymatiga qarab operatorlar guruhidan keraklilarini bajarishga to'g'ri keladi. Bunday hollarda tanlash operatorini ishlatish maqsadga muvofiq. Tanlash operatorining umumiy ko'rinishi quyidagicha:

```
Case k of
  A1: S1;
  A2: S2;
  ...
  An: Sn
else P
End;
```

bu yerda: case – tanlash;
of – dan ma'nolarini bildiradi;
k – operator selektori;
A1, A2, ..., An – nishonlar;
S1, S2, ..., Sn, P – operatorlar.

k ning qiymati A1, A2, ..., An nishonlardan biriga teng bo'lsa, mos ravishda S1, S2, ..., Sn operatorlardan biri bajariladi, aks holda P operator bajariladi.

Masalan, shartli operator yordamida yozilgan If B then S1 Else S2; ifodasini tanlash operatori yordamida quyidagicha yozish mumkin:

```
Case B of
  True: S1;
  False: S2
End;
```

Operator selektori – k ifoda yoki o'zgaruvchi bo'lib, qiymati tartiblangan turlardan biriga (butun, simvulli, mantiqiy, sanoqli) tegishli bo'lishi lozim. Nishonlar o'rnida tartiblangan turlar qiymatlari bo'ladi, haqiqiy va qatorli turlar ma'lumotlaridan foydalanish mumkin emas. Bitta qatoridagi nishonlar bir yoki bir nechta qiymatlardan iborat bo'lishi mumkin. Agar qiymatlar diapazondan iborat bo'lsa, ularning birinchi va oxirgi qiymatlarini ikki nuqta bilan ko'rsatish ham mumkin. Masalan, bitta nishon o'rnida 3,4,5,6 nishonlar berilgan bo'lsa, bulami 3..6 kabi yozish mumkin. Shuningdek, bitta operator o'rnida tarkibiy operator kelishi mumkin.

Misollar

1. Oy tartib raqami kiritilganda qaysi faslga tegishli ekanligini aniqlab beruvchi dastur tuzilsin.

```
Program fasl;  
var n:integer;  
    f:string;  
begin  
    writeln('Oy tartib raqamini kiriting');  
    readln(n);  
    case n of  
        3,4,5: f:='Bahor';  
        6,7,8: f:='Yoz';  
        9,10,11: f:='Kuz';  
        12,1,2: f:='Qish'  
    end;  
    writeln(n,'-oy ',f,' faslga tegishli');  
    readln  
end.
```

2. Fandan olingan umumiy ball kiritilganda bahoni ekranga chiqarib beruvchi dastur tuzilsin.

```
Program baholash;  
var n:integer;  
    b:string;  
begin  
    writeln('ballni kiriting');  
    readln(n);  
    case n of  
        0..55: b:='Ikki';  
        56..70: b:='Uch';  
        71..85: b:='To'rt';  
        86..100: b:='Besh';  
    end;  
    writeln('Bahongiz ',b);  
    readln  
end.
```

Parametrli takrorlanish operatori

Ayrim masalalarda ma'lum hisoblashlarni ko'p marta takror bajarishga to'g'ri keladi. Bunday takrorlanishlarni ifodalash uchun *for*, *while*, *repeat* takrorlanish operatorlari qo'llaniladi.

For operatori takrorlanishlar soni oldindan aniq bo'lgan hollarda ishlatiladi. Uning umumiy ko'rinishi quyidagicha:

For i:=n1 to n2 do S;

bu yerda: *i* – sikl o'zgaruvchisi bo'lib faqat tartiblangan turlar qiymatlarini qabul qiladi;

n1, *n2* – sikl o'zgaruvchisining birinchi va oxirgi qiymatlari;

S – sikl tanasi, ya'ni bajariladigan operator. Agar sikl tanasi ikki va undan ortiq operatorlardan iborat bo'lsa, ular tarkibiy operatorga ichida yoziladi.

For operatori quyidagicha ishlaydi: dastlab sikl o'zgaruvchisining birinchi va oxirgi qiymati aniqlab olinadi, so'ngra sikl o'zgaruvchisi aniqlangan birinchi qiymatni qabul qiladi va schyotchik kabi avtomatik keyingi qiymatlarni qabul qilaveradi. Bunda agar qabul qilingan qiymatlar sikl o'zgaruvchisining oxirgi qiymatidan kichik yoki teng bo'lsa, sikl tanasi bajariladi, aks holda sikl tugatiladi.

For operatorining quyidagi ko'rinishi ham mavjud:

For i:=n2 downto n1 do S;

Bu operatorning dastlabki ko'rinishdagi operatorlardan farqi shundaki, bunda sikl o'zgaruvchisining qiymati oxirgi qiymatdan birinchi qiymatgacha kamayib boradi.

Misollar

1. $\prod_{i=3}^n \frac{1}{x}$ ifodani hisoblash dasturi tuzilsin.

Program dasturl;

var i,n:integer; P,x:real;

begin

writeln('x va n ni kiriting');

readln(x,n);

P:=1;

For i:=3 to n do

P:=P*1/exp(i*ln(x));

writeln('P=',P:6:4);

readln

end.

2. $\sum_{k=a}^{10} \prod_{i=5}^m kx^i$ ifodani hisoblash dasturi tuzilsin.

```
Program dastur2;  
var i,n:integer;  
    P,x:real;  
begin  
  writeln('x va n ni kiriting');  
  readln(x,n);  
  S:=0;  
  for k:=a to 10 do  
    begin  
      P:=1;  
      For i:=5 to m do  
        P:=P*k*exp(i*ln(x));  
      S:=S+P  
    end;  
  writeln('S=',S:6:2);  
  readln  
end.
```

Sharti oldin va sharti keyin beriladigan takrorlanish operatorlari

While operatori

While sikl operatori takrorlanishlar soni oldindan aniq bo'lmagan hollarda qo'llaniladi va sharti oldin beriladigan sikl operatori hisoblanadi.

Operatorming umumiy ko'rinishi quyidagicha:

While B Do S;

bu yerda *B* – mantiqiy ifoda;

S – sikl tanasi bo'lib, bir yoki bir nechta operatorlardan iborat bo'lishi mumkin.

Operator quyidagicha ishlaydi: dastlab mantiqiy ifoda hisoblanadi, agar bu ifoda qiymati *true* bo'lsa, *S* operator bajariladi va boshqaruv yana operator boshiga o'tkaziladi hamda mantiqiy ifoda takroran hisoblanadi. Bu jarayon mantiqiy ifoda *false* qiymat qabul qilguncha davom etadi.

Misollar

1. $y = k^2 \ln x - \pi$, $2 \leq x < 5$, $h_x = 0,4$ ifodaning qiymatlarini hisoblash dasturi tuzilsin.

Program dastur1;


```

const hx=0.4;
var y,k,x:real;
begin
  readln(k);
  x:=2;
  while x<=5 do
    begin
      y:=sqr(k)*ln(x)-pi;
      writeln('x=',x:3:1,' y=',y:6:2);
      x:=x+hx
    end;
  readln
end.

```

2. $\prod_{i=3}^n \frac{1}{x^i}$ ifodani hisoblash dasturi tuzilsin.

Bunday ifodalarni hisoblash dasturlari odatda *for* operatori yordamida tuziladi. Ammo bu ko'rinishdagi ifodalarni *while* operatori yordamida tuzish ham mumkin:

```

Program dastur2;
var i,n:integer;
    P,x:real;
begin
  readln(n,x);
  P:=1; i:=3;
  while i<=n do
    begin
      P:=P*i/exp(i*ln(x));
      i:=i+1
    end;
  writeln('P=',P:6:4);
  readln
end.

```

Dasturda i va n o'zgaruvchilarni butun tur sifatida e'lon qilish shart emas, ya'ni haqiqiy tur deb e'lon qilish ham mumkin.

While operatori yordamida ixtiyoriy murakkalikdagi sikllarni tashkil qilish imkoniyati mavjud.

$$3. \quad y = \frac{\cos^2 x}{\log_a b}, \quad \text{bunda } -1 \leq x \leq 1, \quad h_x = 0,25, \quad 2 \leq b \leq 10, \quad h_b = 0,5$$

ifodaning qiymatlarini ekranga chiqarish dasturini tuzing.

```

Program dastur3;
const hx=0.25; hb=0.5;
var x,y,a,b:real;
begin
  readln(a);
  x:=1;
  while x<=-1 do
    begin
      b:=2;
      while b<=10 do
        begin
          y:=sqr(cos(x))/(ln(b)/ln(a));
          writeln('x=',x:3:1,' b=',b:3:1,' y=',y:6:2);
          b:=b+hb
        end;
        x:=x+hx
      end;
      readln
    end.

```

Repeat operatori

Repeat sikl operatori ham takrorlanishlar soni oldindan aniq bo'lmagan hollarda ishlatiladi hamda sharti keyin beriladigan sikl operatori deb ham ataladi.

Operatorning umumiy ko'rinishi quyidagicha:

Repeat S_1, S_2, \dots, S_n until B ;

bu yerda: B – mantiqiy ifoda;

S_1, S_2, \dots, S_n – bajariladigan operatorlar.

Operator quyidagicha ishlaydi: dastlab S_1, S_2, \dots, S_n – operatorlar bajariladi. So'ng mantiqiy ifoda hisoblanadi, agar mantiqiy ifoda qiymati *false* bo'lsa, boshqaruv yana operator boshiga o'tkaziladi hamda operatorlar takroran bajariladi. Takrorlanishlar mantiqiy ifoda *true* qiymat qabul qilguncha davom etadi. Demak, *repeat* operatorida *for* va *while* operatorlaridan farqli ravishda sikl tanasi kamida bir marta bajariladi.

Misollar

While operatori yordamida tuzilgan dasturlarni *repeat* operatori orqali

yozaylik.

1. $y = k^2 \ln x - \pi$, $2 \leq x \leq 5$ $h_x = 0,4$ ifodaning qiymatlarini hisoblash dasturi tuzilsin.

```
Program dastur1;  
const hx=0.4;  
var y,k,x:real;  
begin  
  readln(k);  
  x:=2;  
  repeat  
    y:=sqr(k)*ln(x)-pi;  
    writeln('x=',x:3:1,' y=',y:6:2);  
    x:=x+hx  
  until x>5;  
  readln  
end.
```

2. $\prod_{i=3}^n \frac{1}{x^i}$ ifodani hisoblash dasturi tuzilsin.

```
Program dastur2;  
var i,n:integer; P,x:real;  
begin  
  readln(n,x);  
  P:=1; i:=3;  
  repeat  
    P:=P*1/exp(i*ln(x));  
    i:=i+1  
  until i>n;  
  writeln('P=',P:6:4);  
  readln  
end.
```

3. $y = \frac{\cos^2 x}{\log_b a}$, bunda $-1 \leq x \leq 1$, $h_x = 0,25$ $2 \leq b \leq 10$ $h_b = 0,5$ ifodaning

qiymatlarini ekranga chiqarish dasturini tuzing.

```
Program dastur3;  
const hx=0.25; hb=0.5;  
var x,y,a,b:real;  
begin
```

```

readln(a);
x:=-1;
repeat
  b:=2;
  repeat
    y:=sqr(cos(x))/(ln(b)/ln(a));
    writeln('x=',x:3:1,' b=',b:3:1,' y=',y:6:2);
    b:=b+hb
  until b>10;
  x:=x+hx
until x>1;
readln
end.

```

Murakkab sikllarda takrorlanish operatorlarini aralashtirib qo'llash ham mumkin. Masalan, tashqi siklni *while*, ichki siklni *repeat* operatori yordamida, yoki *repeat-while*, *for-repeat*, *while-for* kabi juftliklardan foydalanish ham mumkin.

Yuqorida tuzilgan dasturni *while-repeat* juftligidan foydalanib yozamiz.

```

Program dastur4;
const hx=0.25; hb=0.5;
var x,y,a,b:real;
begin
  readln(a);
  x:=-1;
  while x<=1 do
    begin
      b:=2;
      repeat
        y:=sqr(cos(x))/(ln(b)/ln(a));
        writeln('x=',x:3:1,' b=',b:3:1,' y=',y:6:2);
        b:=b+hb
      until b>10;
      x:=x+hx
    end;
  readln
end.

```

Massivlar, bir o'lchovli massivlar

Ko'p hollarda jadval yoki matritsalar ko'rinishidagi ma'lumotlar bilan ish yuritish kerak bo'ladi. Jadvalda ma'lumotlar juda ko'p bo'lganligi sababli, ularning har bir yacheykasidagi elementni mos ravishda bittadan o'zgaruvchiga qiymat qilib berilsa, ular ustida ish bajarish ancha murakkabliklar va noqulayliklarga olib keladi. Shu sababli, dasturlashda bunday muammolar massivlarni ishlatish yordamida hal qilinadi.

Massiv – bu bir nom bilan belgilangan qiymatlar guruhi yoki jadvaldir. Massivlar bir o'lchovli yoki ko'p o'lchovli bo'lishi mumkin. Vektorni bir o'lchovli massiv, matritsani ikki o'lchovli massiv deb qarash mumkin.

Bir o'lchovli A_i , $i = \overline{k1..kn}$ massiv elementlari dasturda $A[i]$ kabi ishlatiladi va umumiy ko'rinishda quyidagicha e'lon qilinadi:

$\text{var } A: \text{array}[k1..kn] \text{ of } T;$

bu yerda: A – massiv nomi;

$k1, kn$ – massiv elementlari sonining quyi va yuqori chegaralari bo'lib, ular aniq butun sonlar yoki o'zgarmaslar bo'lishi lozim;

T – massiv elementlari turi.

Masalan,

$\text{Const } n=20;$

$\text{Var } A: \text{array}[1..30] \text{ of integer};$

$B: \text{array}[3..n] \text{ of real};$

Bunda A – 1 dan 30 gacha tartiblangan 30 ta butun sonlardan iborat bir o'lchovli jadvaldir. V – 3 dan n gacha tartiblangan (n ga o'zgarmaslar bo'limida aniq qiymat berilgan) $n-2$ ta haqiqiy sonlardan tashkil topgan bir o'lchovli jadvaldir.

Massivlarni e'lon qilishdan maqsad, massiv elementlari uchun operativ xotiradan joy ajratishdir.

Massiv elementlari qiymatlarini kiritish odatda *for* operatori orqali amalga oshiriladi.

Masalan, *for* $i:=1$ to 10 *do read* ($A[i]$);

Bu misolda A massivning 10 ta elementi qiymati klaviaturadan ketma-ket kiritiladi.

Xuddi shuningdek, massiv elementlarini ekranga ketma-ket chiqarish ham mumkin.

Masalan, *For* $i:=1$ to 10 *do Write*($A[i]$);

Misollar

1. N ta haqiqiy sonlardan tashkil topgan bir o'lchovli massiv elementlari yig'indisini topish dasturini tuzing.

```

Program summassiv;
var i,n: integer;
    S:real;
    a: array[1..100] of real;
begin
    write ('Massiv o'lchamini kiriting n=');
    readln(n);
    writeln('Massiv elementlarini kiriting');
    for i:=1 to n do readln(a[i]);
        S:=0;
    for i:=1 to n do S:=S+a[i];
    writeln('S=',S:6:2);
    readln
end.

```

2. 10 ta butun sonlardan iborat bir o'lchovli massivning eng kichik elementini va uning joylashgan o'mini topish dasturini tuzing.

Bajarish:

Bu masalada dastlab birinchi element eng kichik deb olinib, keyin boshqa elementlar bilan solishtirib chiqiladi va kerakli o'rinda almashtirishlar bajariladi.

Program eng kichik;

```

var i,min: integer;
    a: array[1..10] of integer;
begin
    writeln ('Massivning 10 ta elementini kiriting');
    for i:=1 to 10 do read(a[i]);
    min:=1;
    for i:=2 to 10 do if a[i]<a[min] then min:=i;
    writeln('Izlanayotgan eng kichik element: ',a[min], ' Element
    tartib tartib raqami: ',min );
    readln
end.

```

3. Bir o'lchovli massiv elementlarini o'sish tartibida joylashtirish dasturini tuzing.

Bajarish:

Massivlarni tartiblashning bir necha usullari mavjud. Ulardan tanlash va almashtirish usullarini qarab chiqamiz.

Tanlash usuli yordamida massivni o'sish bo'yicha tartiblashtirish algoritmi quyidagicha:

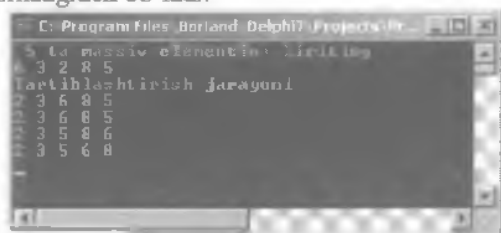
1. Massivning birinchi elementidan boshlab qarab chiqilib, eng kichik element topiladi.
 2. Birinchi element bilan eng kichik element joylari almashtiriladi.
 3. Ikkinchi elementidan boshlab qarab chiqilib, eng kichik element topiladi.
 4. Ikkinchi element bilan eng kichik element joylari almashtiriladi.
 5. Bu jarayon bitta oxirgi elementgacha takrorlanadi.
- Algoritm dasturi masalan 5 ta element uchun quyidagicha bo'ladi:

```

Program tartiblash1;
const s=5;
var i,j,min,k,buf: integer;
    a: array[1..s] of integer;
begin
  writeln(s, ' ta massiv elementini kiriting');
  for k:=1 to s do read(a[k]);
  writeln ('Tartiblashtirish jarayoni');
  for i:=1 to s-1 do
    begin
      min:=i;
      for j:=i+1 to S do
        if a[j]<a[min] then min:=j;
      buf:=a[i]; a[i]:=a[min]; a[min]:=buf;
      for k:=1 to s do write(a[k], ' ');
      writeln;
    end;
  readln;readln
end.

```

Natija 5-rasmdagidek bo'ladi:



5-rasm. Dastur natijasi

Almashtirish usuli yordamida massivni o'sish bo'yicha tartiblashtirish algoritmi quyidagicha:

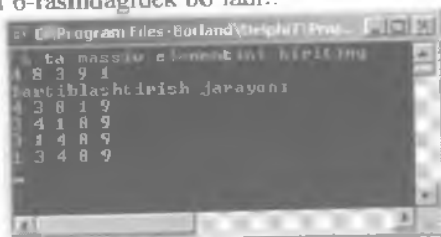
1. Massivning birinchi elementidan boshlab ketma-ket hamma qo'shni elementlar bir-biri bilan solishtirilib, agar birinchisi ikkinchisidan kichik bo'lsa, ularning joyi almashtirilib boriladi.
 2. Bu jarayon davomida kichik qiymatli elementlar massiv boshiga, katta elementlar esa oxiriga siljitib boriladi.
 3. Bu jarayon massiv elementlar sonidan bitta kam takrorlanadi.
- Algoritm dasturi 5 ta element uchun quyidagicha bo'ladi:

```

Program tartiblash2;
const s=5;
var i,k,buf:integer;
    a:array[1..s] of integer;
begin
  writeln(S, ' ta massiv elementini kiriting');
  for k:=1 to s do read(a[k]);
  writeln('Tartiblashtirish jarayoni');
  for i:=1 to s-1 do
    begin
      for k:=1 to s-1 do
        if a[k]>a[k+1] then
          begin
            buf:=a[k]; a[k]:=a[k+1]; a[k+1]:=buf;
          end;
      for k:=1 to s do write(a[k], ' ');
      writeln;
    end;
  readln; readln
end.

```

Dastur natijasi 6-rasmdagidek bo'ladi:



6-rasm. Dastur natijasi

Ikki o'lovli massivlar

Ikki o'lovli $B_{i,j}$, $i = \overline{k1, kn}$, $j = \overline{c1, cm}$ massiv elementlari dasturda $B[i,j]$ kabi ishlatiladi va umumiy ko'rinishda quyidagicha e'lon qilinadi:

var B:array[k1..kn,c1..cm] of T;

bu yerda: B – massiv nomi;

$k1, kn$ – massiv elementlari qatorlari sonining quyi va yuqori chegaralari;

$c1, cm$ – massiv elementlari ustunlari sonining quyi va yuqori chegaralari;

T – massiv elementlari turi.

Masalan,

Const n=20; m=30;

Var B:array[2..n,n..m] of integer;

Bunda B – 2 dan n gacha tartiblangan $n-1$ ta qatordan, n dan m gacha tartiblangan $m-n+1$ ta ustundan iborat $(n-1)*(m-n+1)$ ta butun sonlardan tashkil topgan ikki o'lovli jadvaldir.

Misollar

1. $\sum_{i=1}^{10} \prod_{j=2}^n \sqrt{a_{ij}}$ ni hisoblash dasturini tuzing.

Program massiv21;

var i,j,n:integer;

S,P:real;

a: array[1..10,1..10] of real;

begin

write('n ni kiriting: '); readln(n);

writeln('Massiv elementlarini kiriting');

for i:=1 to 10 do

for j:=2 to n do readln(a[i,j]);

S:=0;

for i:=1 to 10 do

begin

P:=1;

for j:=2 to n do

P:=P*sqrt(a[i,j]);

S:=S+P

end;

writeln('S=',S:6:2);

readln

end.

2. $m \times n$ o'lchamli ikki o'lchovli massivning 3-ustunini k soniga ko'paytirishdan hosil bo'lgan massivni ekranga chiqarish dasturini tuzing.

```
Program massiv22;  
var i,j,m,n,k: integer;  
    a:array[1..10,1..10] of integer;  
begin  
    write('qatorlar soni m=');  
    readln(m);  
    write('ustunlar soni n=');  
    readln(n);  
    writeln ('Massiv elementlarini kiriting');  
    for i:=1 to m do  
        for j:=1 to n do  
            readln(a[i,j]);  
    write('ko'paytiriladigan son k=');  
    readln(k);  
    for i:=1 to m do  
        a[i,3]:= a[i,3]*k;  
    for i:=1 to m do  
        begin  
            for j:=1 to n do  
                write(' a['i','j']=',a[i,j]);  
            writeln  
        end;  
    readln  
end.
```

Dinamik massivlar

Dinamik massivlar oldindan elementlari soni aniq bo'lmagan massivlarni qo'llashda ishlatiladi.

Bir o'lchovli dinamik massiv umumiy ko'rinishda quyidagicha e'lon qilinadi:

var A.array of T;

bu yerda: A – jadval nomi;

T – massiv elementlari turi.

Masalan, *var vek:array of real;*

Bunda haqiqiy sonlardan tashkil topgan *vek* nomli bir o'lchovli massiv e'lon qilinmoqda.

Massiv uzunligi hisoblash bo'limida *SetLength* protsedurasi yordamida

aniqlanishi lozim.

Masalan, $SetLength(vek, 10)$;

Bunda vek nomli massiv uchun operativ xotiradan 0 dan 9 gacha bo'lgan 10 ta joy ajratilmoqda. Demak, bu massiv elementlarini $vek[0]$, $vek[1]$, ..., $vek[9]$ lar tashkil etadi.

Misol

1. $y_i = 2inx_i + b$, $i = \overline{0, n}$ ni hisoblash dasturini tuzing.

Program dinamikmasi;

var x,y:array of real;

i,n: integer;

b:real;

begin

write('n=');

readln(n);

write('b=');

readln(b);

setLength(x,n);

setLength(y,n);

writeln('x massiv elementlarini kiriting');

for i:=0 to n do

read(x[i]);

readln;

for i:=0 to n do

begin

y[i]:=2*x[i]+b;

writeln('x[' ,i, ']=',x[i]:3:1, ' y[' ,i, ']=',y[i]:5:2)

end;

readln

end.

Ikki o'lchovli dinamik massiv umumiy ko'rinishda quyidagicha e'lon qilinadi:

var B.array of array of T;

bu yerda B – jadval nomi;

T – massiv elementlari turi.

Masalan, var matr:array of array of real;

Bunda haqiqiy sonlardan tashkil topgan matr nomli ikki o'lchovli massiv e'lon qilinmoqda.

Ikki o'lchovli dinamik massiv uzunligi ham SetLength protsedurasi

yordamida aniqlanadi.

Masalan, `SetLength(matr,4,3)`;

Bunda 4x3 o'lchamli matr nomli massiv uchun operativ xotiradan 12 ta joy ajratiladi. Demak, bu massiv elementlarini `matr[0,0]`, `matr[0,1]`, `matr[0,2]`, `matr[1,0]`, `matr[1,1]`, `matr[1,2]`, `matr[2,0]`, `matr[2,1]`, `matr[2,2]`, `matr[3,0]`, `matr[3,1]`, `matr[3,2]` lar tashkil etadi.

Misol

2. Ikki o'lchovli *a* va *v* massivlar yig'indisini topish dasturi tuzilsin.

Program dmas2;

var a,b,c:array of array of integer;

 i,j,m,n: integer;

begin

 write('m=');

 readln(m);

 write('n=');

 readln(n);

 setLength(a,m,n);

 setLength(b,m,n);

 setLength(c,m,n);

 writeln(m*n,' ta a massiv elementlarini kiriting');

 for i:=0 to m-1 do

 for j:=0 to n-1 do

 read(a[i,j]);

 writeln(m*n,' ta b massiv elementlarini kiriting');

 for i:=0 to m-1 do

 for j:=0 to n-1 do

 read(b[i,j]);

 for i:=0 to m-1 do

 for j:=0 to n-1 do

 c[i,j]:=a[i,j]+b[i,j];

 for i:=0 to m-1 do

 for j:=0 to n-1 do

 writeln('c['i','j']='',c[i,j]);

 readln;

 readln;

end.

Xuddi shuningdek uch o'lchovli dinamik massivlarni ham ishlatish mumkin. Ular umumiy ko'rinishda quyidagicha e'lon qilinadi:

var R:array of array of array of T;

bu yerda: R – jadval nomi;
 T – massiv elementlari turi.

Masalan, var *kub.array of array of array of integer*;

Bunda butun sonlardan tashkil topgan *kub* nomli uch o'lchovli massiv e'lon qilinmoqda.

Uch o'lchovli dinamik massiv uzunligi ham *SetLength* protsedurasi yordamida aniqlanadi.

Masalan, *SetLength(kub,1,1,2)*;

Bunda $2 \times 2 \times 3$ o'lchamli *kub* nomli massiv uchun operativ xotiradan 12 ta joy ajratiladi. Demak, bu massiv elementlarini *kub*[0,0,0], *kub*[0,0,1], *kub*[0,0,2], *kub*[0,1,0], *kub*[0,1,1], *kub*[0,1,2], *kub*[1,0,0], *kub*[1,0,1], *kub*[1,0,2], *kub*[1,1,0], *kub*[1,1,1], *kub*[1,1,2] lar tashkil etadi.

Misol

3. $y_{i,j,k} = 2x_{i,j,k}$, $i = \overline{0,1}$, $j = \overline{0,2}$, $k = \overline{0,3}$ ni hisoblash dasturini tuzing.

Program dmas3;

var x,y:array of array of array of integer;

 i,j,k: integer;

begin

 setLength(x,2,3,4);

 setLength(y,2,3,4);

 writeln('x massiv elementlarini kriting');

 for i:=0 to 1 do

 for j:=0 to 2 do

 for k:=0 to 3 do

 read(x[i,j,k]);

 readln;

 for i:=0 to 1 do

 for j:=0 to 2 do

 for k:=0 to 3 do

 begin

 y[i,j,k]:=2*x[i,j,k];

 writeln('x['i','j','k,']=',x[i,j,k], ' y['i','j','k,']=',y[i,j,k])

 end;

 readln;

 end.

Protsedura va funksiyalar

Ayrim masalalarni yechishda ma'lum parametrlarning har xil qiymatlarida bir xil hisoblashlarni (jarayonlarni) bajarishga to'g'ri keladi. Bunday hollarda dastur hajmini kichraytirish maqsadida *qism dasturlardan* foydalaniladi. Bunda takrorlanadigan operatorlar ketma-ketligi uchun umumiy ko'rinishda bitta qism dastur tuziladi va kerak bo'lgan joylarda unga murojaat qilinadi.

Delphi tilida *qism dasturni protsedura yoki funksiya qism dasturi* ko'rinishida yozish mumkin. Protsedura yoki funksiyaga dasturning qayeridan murojaat qilinsa, boshqaruv yana shu joyga qaytib keladi. Dasturda e'lon qilingan o'zgaruvchilar shu dasturdagi protsedura va funksiyalarga nisbatan *global o'zgaruvchilar* deyiladi. Protsedura va funksiyalar ichida e'lon qilingan o'zgaruvchilar esa *lokal o'zgaruvchilar* deyiladi. Lokal o'zgaruvchilarning ta'sir doirasi shu qism dastur ichiga tegishli bo'ladi, xolos.

Har qanday qism dasturni kichik bir dastur deb qarash mumkin. Protsedura va funksiyalar ham dasturga o'xshab dastur strukturasi barcha bo'limlariga ega bo'lishi mumkin. Ya'ni qism dasturda sarlavha satri, nishonlar, o'zgaruvchilar, turlar, o'zgaruvchilar, protsedura va funksiyalar, hisoblash bo'limlari ishtirok etishi mumkin. Bulardan dastur strukturasi degidek sarlavha satri va hisoblash bo'limlari albatta ishtirok etadi. Demak, protsedura va funksiyalar ichma-ich joylashishlari mumkin.

Protsedura qism dasturi

Protsedura sarlavha satri umumiy ko'rinishda quyidagicha bo'ladi:

Procedure <protsedura nomi>(<formal parametrlar>);

Masalan, $\sum_{i=1}^n \frac{1}{\sqrt{x}}$ ko'rinishidagi ifodalar uchun umumiy ko'rinishda

quyidagicha protsedura tuzish mumkin:

```
Procedure sum(t1,t2: integer);  
var i:integer;  
begin  
  S:=0;  
  for i:=t1 to t2 do S:=S+1/sqrt(x)  
end;
```

Bu protseduraga dasturdan quyidagicha murojaat qilinadi:

Sum(n,m);

Protseduraga murojaat qilinganda faktik parametrlar qiymatlari mos ravishda formal parametrlarga uzatiladi. Shuning uchun formal va faktik parametrlar soni teng va ularning mos ravishdagi turlari bir xil bo'lishi

shart. Bunda S, n, m lar global, $t1, t2, i$ lar lokal o'zgaruvchilar hisoblanadi.

Misollar

1. $y = \frac{\min(\sqrt{a}, 2*b)}{3 * \min(a^2, 7)}$ ifodani hisoblash dasturini tuzing.

```
program Project1;
var y,a,b,min,c1,c2:real;
procedure kichigi(t1,t2:real);
begin
  if t1<t2 then min:=t1 else min:=t2
end;
begin
  readln(a,b);
  kichigi(sqrt(a),2*b);
  c1:=min;
  kichigi(sqr(a),7);
  c2:=min;
  y:=c1/(3*c2);
  writeln('y=' ,y:5:2);
  readln
end.
```

Bu dasturda protseduraga birinchi murojaat qilishda $\text{sqrt}(a)$ va $2*b$ lar mos ravishda x va y larga uzatiladi. Ularning kichigi min o'zgaruvchida hosil bo'ladi. Bu qiymatni biror o'zgaruvchiga o'zlashtirish lozim, aks holda ikkinchi marta protseduraga murojaat qilinganda dastlabki natija o'chib ketadi. Dasturda qiymat $c1$ ga o'zlashtiriladi. Ikkinchi marta murojaat qilinganda hosil bo'ladigan natija $c2$ o'zgaruvchiga o'zlashtiriladi.

Yuqoridagi dasturni quyidagicha yozish ham mumkin:

```
Program Project2;
var y,a,b,c1 ,c2:real;
procedure kichigi(t1,t2:real; var min:real);
begin
  if t1<t2 then min:=t1 else min:=t2
end;
begin
  readln(a,b);
  kichigi(sqrt(a),2*b,c1);
  kichigi(sqr(a),7,c2);
```

```

y:=c1/(3*c2);
writeln('y=' ,y:5:2);
readln
end.

```

Bunda c_1 va c_2 o'zgaruvchilar ham protseduraga murojaat qilishda qatnashmoqda va mos ravishda protsedura parametrlari ro'yxatidagi min o'zgaruvchiga uzatiladi. Topilgan minimum qiymatlar c_1 va c_2 larga o'zlashtiriladi.

Funksiya qism dasturi

Funksiya qism dasturi va ularni tashkil etish ham protseduraga o'xshash bo'lib, sarlavha satri umumiy ko'rinishda quyidagicha bo'ladi:

Function <funksiya nomi>(<formal parametrlar>):<funksiya natijasi turi>;

Masalan, *Function Min* ($x,y:real$): *real*;

Funksiya qism dasturining protseduradan asosiy farqi shundaki, unga murojaat qilinganda natija faqat bitta bo'lib, u shu funksiya nomiga yoki *result* nomli standart o'zgaruvchiga o'zlashtiriladi.

Misol. Yuqoridagi $y = \frac{\min(\sqrt{a}, 2*b)}{3*\min(a^2, 7)}$ ifodani funksiya qism dasturi yordamida hisoblash dasturini tuzing.

```

Program Project1;
var y,a,b,min,c1,c2:real;
function kichigi(t1,t2:real):real;
begin
  if t1<t2 then kichigi:=t1 else kichigi:=t2
end;
begin
  readln(a,b);
  y:=kichigi(sqrt(a),2*b)/(3*kichigi(sqrt(a),7));
  writeln('y=' ,y:5:2);
  readln
end.

```

Fayllar bilan ishlash

Dasturda kiritiladigan va chiqariladigan ma'lumotlar soni ko'p miqdorda bo'lganda ularni faylda saqlash ma'qulroq. Bu ma'lumotlar

oddiy matnli fayllarda saqlanadi. Fayl o'zgaruvchisi quyidagicha e'lon qilinadi:

var f: textfile;

Dasturda fayl bilan ishlash uchun avvalo fayl o'zgaruvchisi (faylning mantiqiy nomi) bilan diskdagi fayl (faylning fizik nomi) *Assign* funksiyasi yordamida bog'lanishi kerak. Shundan so'ng faylni *Rewrite*, *Reset* yoki *Append* funksiyalarining keraklisi orqali ochib, zarur bo'lgan ishlar bajariladi. Fayl bilan biror ish bajarilgandan so'ng uni *Close* funksiyasi yordamida yopish lozim. Agar fayl bilan bir necha marta amal bajarish lozim bo'lsa, har bir amaldan so'ng uni *Close* funksiyasi yordamida yopib, *Reset* yoki *Append* funksiyalardan biri orqali qaytadan ochish lozim.

Fayllar bilan ishlovchi maxsus funksiyalar:

- *Assign* – fayl o'zgaruvchisi (faylning mantiqiy nomi) bilan diskdagi faylni (faylning fizik nomini) bog'laydi. Masalan: *Assign(f,'D:\fayl\fl.txt');*
- *Rewrite* – mantiqiy fayl yaratadi va uni ochadi. Masalan: *Rewrite(f);*
- *Reset* – faylni o'qish uchun ochadi. Masalan: *Reset(f);*
- *Append* – yozish uchun faylni ochadi. Masalan: *Append(f);*
- *Close* – faylni yopadi. Masalan: *Close(f);*
- *Read* – fayldan o'qish funksiyasi. Masalan: *Read(f, t);* buyrug'i faylda kursor turgan joydagi elementni *t* o'zgaruvchiga o'zlashtiradi.
- *Write* – faylga yozish funksiyasi. Masalan: *Write(f, k);* buyrug'i faylda kursor turgan joyga *k* elementni yozadi.
- *Eof* – fayl oxirini aniqlaydi. Bu funksiya fayl oxiriga yetganda *true* qiymat qabul qiladi.
- *Eoln* – qator oxirini aniqlaydi. Bu funksiya qator oxiriga yetganda *true* qiymat qabul qiladi.
- *SeekEoln* – *Eoln* funksiyasining vazifasini bajarib, qator oxiridagi probel va tabulyatsiyalarni hisobga olmaydi.

Misollar

1. 3x4 o'lchamli 2 o'lchovli massiv elementlari kvadratlarini faylga yozish dasturini tuzing.

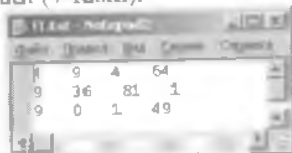
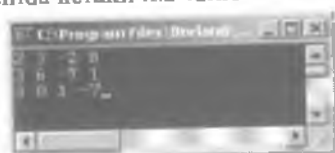
```
program Project1;  
var f:textfile; i,j:integer;  
    a:array[1..3,1..4] of integer;  
begin  
    for i:=1 to 3 do  
        for j:=1 to 4 do read(a[i,j]);  
    assign(f,'d:\fayl\fl.txt');
```

```

rewrite(f);
for i:=1 to 3 do
begin
for j:=1 to 4 do write(f,sqr(a[i,j]),' ');
writeln(f);
end;
close(f)
end.

```

Dasturni ishga tushirib, *A* massiv elementlari kiritilsa, d:\fayl\f1.txt fayli ichida kerakli ma'lumotlar hosil bo'ladi (7-rasm).



7-rasm. Massiv elementlarini kiritish va f1.txt faylida hosil bo'lgan ma'lumot

2. d:\fayl\f1.txt faylidagi 3x4 o'lchamli 2 o'lchovli massivni ekranga chiqarish dasturini tuzing.

```

program Project2;
var f:textfile; i,j:integer;
a:array[1..3,1..4] of integer;
begin
assign(f,'d:\fayl\f1.txt');
Reset(f);
for i:=1 to 3 do
begin
for j:=1 to 4 do read(f,a[i,j]);
readln(f);
end;
close(f);
for i:=1 to 3 do
begin
for j:=1 to 4 do write(a[i,j],' ');
writeln
end;
readln
end.

```

Dasturni kiritib, natija olamiz (8-rasm).



8-rasm. Dastur natijasi

Tuzilgan dasturda fayldagi massiv elementlari sonini bilganligimiz uchun *for* operatoridan foydalandik. Odatda fayldagi ma'lumotlar o'qishdan oldin sanalmaydi. U holda fayldagi massiv elementlarini o'qish va ekranga chiqarish uchun quyidagicha dastur tuziladi:

```
program Project1;  
var f:textfile; k,t,i,j:integer;  
    a:array[1..3,1..4] of integer;  
begin  
    assign(f,'d:\fay\fl.txt');  
    Reset(f);  
    i:=0;  
    while not eof(f) do  
        begin  
            i:=i+1;  
            j:=0;  
            while not seekcoln(f) do  
                begin  
                    j:=j+1;  
                    read(f,a[i,j]);  
                end;  
            readln(f);  
        end;  
    close(f);  
    for k:=1 to i do  
        begin  
            for t:=1 to j do  
                write(a[k,t], ' ');  
            writeln;  
        end;  
    readln  
end.
```

Delphi ning vizual imkoniyatlari, Delphi dasturlash muhiti

Delphi konsol ilovasi yordamida dasturlash tili, strukturasi, dastur tuzish qoidalari o'rganib chiqilgandan so'ng dasturlar Delphi ning haqiqiy imkoniyatlarini ochib beruvchi vizual dasturlash muhitida tuziladi.

Delphi ning vizual dasturlash muhiti Windows operatsion tizimi uchun mo'ljallangan dasturlash vositasi bo'lib, kompyuterda dastur tuzish ishlarini avtomatlashtiradi. Delphi tizimi dastur tuzishni sezilarli darajada soddalashtiradi va dastur tuzuvchi ishining asosiy qismini tizimga yuklaydi. Delphi tizimi dasturni loyihalash va yaratish vaqtini kamaytiradi hamda Windows muhitida ishlovchi dastur ilovalarini tuzish jarayonini osonlashtiradi.

Delphi ishga tushirilganda Delphining vizual dasturlashga imkon beruvchi dasturlash muhiti hosil bo'ladi (9-rasm).



9-rasm. Delphi dasturlash muhiti

Dasturlash muhiti elementlari quyidagilar:

- **sarlavha satri.** Bu yerda sistemali menyu tugmasi, dastur va fayl nomi hamda oynani boshqarish tugmalari o'rin olgan;
- **menyu qatori.** Dastur *File, Edit, Search, View, Project, Run, Component, Database, Tools, Window, Help* menyularidan iborat bo'lib, barcha buyruqlar menyularda qulay tarzda joylashtirilgan;

- uchinchi qatorning chap tomonida odatda *uskunalar paneli* joylashgan bo'ladi. Bu yerda Delphi muhitida eng ko'p ishlatiladigan buyruqlar vizual ko'rinishdagi tugmachalar shaklida tasvirlangan bo'lib, foydalanuvchi ishini tezlashtirishga mo'ljallangan;
- uchinchi qatorning ikkinchi yarmida odatda *komponentlar palitrası* joylashgan bo'ladi. Bu yerda ko'p miqdordagi obyektlar (componentlar) jamlangan bo'lib, ular Delphi muhitining dastur yaratish vositasi hisoblanadi;
- ekranning o'rta qismida Delphi muhitining ishchi obyektı – *forma* (Form) joylashgan. Komponentlar (obyektlar) shu formaga joylashtiriladi. Ushbu forma eng asosiy obyekt hisoblanib, dasturlashning yakuniy natijasi shu formada aks etadi.
- formaning ostki qavatida *kod redaktori* (Unit) joylashgan. Barcha dasturlash kodlari shu oynada yoziladi va tahrirlanadi.
- formaning chap tomonida *obyektlar shajarasi* (Object TreeView) va *obyekt inspektori* (Object Inspector) oynalari joylashgan. Obyektlar shajarasi oynasi orqali obyektning joylashish strukturasi osongina ko'rish va bu obyektlar bilan bog'liq jarayonlarga o'tish imkonini beradi. Obyekt inspektori oynasida belgilangan obyektning xossalarini ko'rish va o'zgartirish imkoniyati mavjud.

Delphi dasturlash muhiti asosiy tushunchalari:

Obyekt. Vizual dasturlashda bu atama muloqot oynasi va muloqot oynasida joylashadigan komponentlar – *forma*, *buyruq tugmasi* (button), *matni axborot kiritiluvchi maydon* (edit), *menyu* va boshqa komponentlar sifatida tushuniladi.

Xossa. Bunda obyekt xususiyatlari tushuniladi. Masalan uning o'lchami, ekran yoki formada joylashishi, sarlavhasi yoki obyektidagi matn.

Hodisa. Yaratilgan dasturning ishlashi jarayonida bajariladigan harakatlar nomi hodisa deyiladi. Masalan, obyekt ustida sichqonchani chap tugmasini bosish – *OnClick*. Bu hodisalarning ishlatilishi natijasida ma'lum amallar bajariladi.

Delphi da dastur yaratish jarayoni ikki bosqichga bo'linadi:

- 1) *formalarni loyihalashtirish;*
- 2) *dasturlashtirish.*

Formani loyihalashtirishdagi asosiy amallar:

a) avvalo *Komponentlar palitrasidan* kerakli komponentlarni tanlab formaga joylashtirish lozim. Buning uchun *komponentlar palitrasida* kerakli obyekt belgilanadi va bu komponent formaning qayeriga

joylashtirilishi kerak bo'lsa, o'sha yerda sichqonchanning chap tugmasi bir marta bosiladi;

b) formada joylashgan komponentni sichqoncha yordamida ixtiyoriy joyga ko'chirish va o'lchamlarini o'zgartirish mumkin;

c) odatda obyektning xossalari *obyekt inspektori oynasining xossalari* (Properties) qismidan o'zgartiriladi. Buning uchun formada obyekt yoki obyektlar belgilanadi va xossalari qismidan kerakli xossa qiymatlari kiritiladi yoki o'zgartiriladi.

Yuqoridagi amallardan so'ng tuzayotgan dasturimizning tashqi ko'rinishi hosil bo'ladi. Bu dastur muayyan foydali ishni bajarishi uchun ikkinchi bosqich, ya'ni *dasturlashtirish bosqichiga* o'tamiz:

1. formada har bir obyekt ma'lum voqealarga bog'langan bo'lishi mumkin. Bu bog'likliklarni hosil qilish va ko'rish uchun formada kerakli komponentni, masalan, Label1 komponentini belgilaymiz;
2. *obyekt inspektori oynasining hodisalar* (Events) bo'limiga o'tamiz. Bu bo'lim ham ikki qismdan iborat. Chap tomonda hodisa nomlari, o'ng tomonda belgilangan obyektga mos hodisalar keltirilgan;
3. chap tomondan kerakli hodisani topib, o'ng tomondan bu hodisaga mos bo'lgan katakcha ustida sichqonchanning chap tugmasini ikki marta bosamiz.

Izoh

Bir komponent uchun bir nechta hodisalar yaratish yoki umuman yaratmaslik ham mumkin.

Natijada ekranda *kod redaktori* oynasi hosil bo'ladi va unga masalan Onclick hodisasi uchun quyidagi ifoda avtomatik qo'shiladi:

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  
end;
```

Delphi muhiti shu qadar qulay avtomatlashirilganki, dastur tuzuvchi faqat kursor turgan *begin* va *end* xizmatchi so'zlar orasiga o'z amallarini yozsa hamda foydalangan o'zgaruvchilarini e'lon qilsa kifoya. Qolgan barcha bo'limlarni Delphi ishga tushirilgandan boshlab o'zi avtomatik ravishda yozib boradi.

Forma tashkil etuvchilari bilan *kod redaktori* orasida uzviy bog'liqlik mavjud. Masalan, formaga biror komponent joylashtirilsa, dastur kodi ham avtomatik o'zgaradi.

Ma'lumki, Delphining konsol ilovasida ma'lumotlar to'g'ridan-to'g'ri *read* va *readln* protseduralari yordamida kiritilib, *write* va *writeln* protse-

duralari yordamida chiqarilardi. Delphi vizual dasturlash muhitida esa ma'lumotlarni kiritish va chiqarish ma'lum bir komponentlar, masalan, *Edit*, *Memo*, *Label*, *StringGrid* kabi komponentlar yordamida amalga oshiriladi.

Bu komponentlarning barchasi qatorli turlarni qabul qiladi. Shu sababli kiritilayotgan ma'lumot boshqa turda bo'lsa, komponentga yozilgan bu ma'lumot *kod redaktorida* kerakli turga aylantirib olinishi va aksincha, ekranga chiqarilayotgan ma'lumot qatorli turga aylantirib olinishi lozim. Buning uchun maxsus protseduralar mavjud. Ularning asosiyarlari quyidagilar:

- ***StrToFloat*** (*const S: string*): *Real*;
S qatomi haqiqiy turga aylantiradi.
- ***StrToInt*** (*const S: string*): *Integer*;
S qatomi butun turga aylantiradi.
- ***StrToBool*** (*const S: string*): *Boolean*;
S qatomi mantiqiy turga aylantiradi.
- ***FloatToStr*** (*a: real*): *string*;
a haqiqiy sonni qatorli turga aylantiradi.
- ***IntToStr*** (*k: Integer*): *string*;
k butun sonni qatorli turga aylantiradi.
- ***BoolToStr*** (*k: Boolean*): *string*;
k mantiqiy qiymatni qatorli turga aylantiradi.

Masalan, *x* haqiqiy va *n* butun sonlar kiritilishi kerak bo'lsa, formaga loyihalash vaqtida *x* soni uchun *Edit1*, *n* soni uchun *Edit2* komponentlarini joylashtiramiz. Kod redaktorida kerakli joyda quyidagini yozamiz:

```
x:=strtofloat(edit1.text);  
n:=strtoint(edit2.text);
```

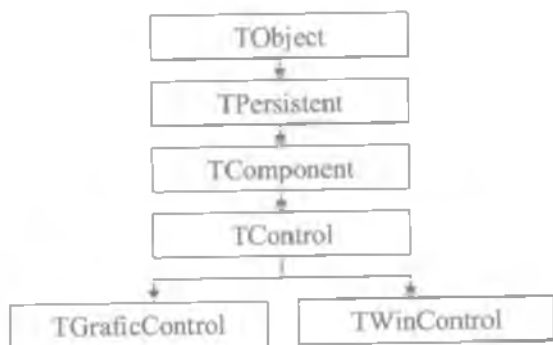
Masalan, *x* haqiqiy va *n* butun sonlar ekranga chiqishi kerak bo'lsa, formaga loyihalash vaqtida *x* soni uchun *Edit3*, *n* soni uchun *Edit4* komponentlarini joylashtiramiz. Kod redaktorida kerakli joyda quyidagini yozamiz:

```
Edit3.Text:=floattostr(x);  
Edit4.Text:=inttostr(n);
```

Sinflar va komponentlar

Komponentlar obyekt tushunchasining eng asosiy qismi hisoblanadi. Delphida komponentlar *Vizual komponentlar kutubxonasi* (Visual Component Library-VCL) deb nomlangan murakkab shajaraviy struktura

ko'rinishida joylashtirilgan. Quyida ushbu struktura keltirilgan:



VCL tarkibiga kiruvchi **TObject** sinfi boshqa hamma sinflarning eng yuqorisi bo'lib, obyektlarni tuzish va boshqarish imkonini beradi.

TPersistent sinfi **TObject** sinfidan kelib chiqadi va u obyektlarni tashkil qilish uchun o'zida bir necha usullarni saqlaydi.

TComponent sinfi barcha komponentlarni o'zida birlashtiradi.

TControl sinfi xossalar, usullar va vizual komponentlar xodisalarini ta'minlab beradi. Bu lar yordamida klaviaturadan va sichqonchadan foydalangan holda ma'lumotlarni ekranga chiqarish va dasturga kiritish mumkin.

TWinControl sinfi **TControl** sinfining davomchisi bo'lib, oyna elementlarini boshqarish va yaratish uchun ishlatiladi.

TGraphicControl sinfi ham **TControl** sinfining davomchisi bo'lib, grafik elementlarni boshqarish uchun ishlatiladi.

Keltirilgan strukturada har bir sinf o'zidan oldingi sinflarga tegishli hisoblanadi. Bu sinflarga tegishli bo'lgan komponentlarning ba'zi xossa va hodisalari quyidagilar:

Xossalar

Left: Integer – ekranning chap chegarasidan komponent yuqori chap chegarasigacha bo'lgan masofani piksellar hisobida o'zgartiradi.

Top: Integer – ekranning yuqori chegarasidan komponent yuqori chap chegarasigacha bo'lgan masofani piksellar hisobida o'zgartiradi.

Width: Integer – komponent kengligini piksellar hisobida o'zgartiradi.

Height: Integer – komponent balandligini piksellar hisobida o'zgartiradi.

Align: TAlign – komponentning o'zi joylashgan obyektida qanday holatda joylashishini bildiradi. Talign tipiga tegishli qiymatlar:

AlNone – joylashish o'zgarmaydi.

AlTop – komponent balandligi o'zgarmagan, kengligi obyekt kengligiga teng bo'lgan holda obyekt yuqorisidan joylashadi.

AlBottom – komponent balandligi o'zgarmagan, kengligi obyekt kengligiga teng bo'lgan holda obyekt quyi qismidan joylashadi.

AlLeft – komponent kengligi o'zgarmagan, balandligi obyekt balandligiga teng bo'lgan holda obyekt chap tomonidan joylashadi.

AlRight – komponent kengligi o'zgarmagan, balandligi obyekt balandligiga teng bo'lgan holda obyekt o'ng tomonidan joylashadi.

AlClient – komponent kengligi va va balandligi butun obyekt bo'yicha kengaygan holatda joylashadi.

AutoSize: Boolean – agar bu xossa *true* qiymat qabul qilsa, u holda komponent o'lchamlari uning tashkil etuvchisiga qarab avtomat ravishda o'zgaradi. Masalan, *Edit* komponenti balandligi unga yozilgan matn shriftiga qarab avtomatik o'zgaradi.

Caption: TCaption – komponent sarlavhasini qo'yish. Eslatma sifatida shuni aytib o'tish kerakki, Delphi muhitida butun, haqiqiy, simvolli, qatorli turlar bilan birga boshqa ko'pgina turlar ham joriy qilingan. Masalan, *Caption* xossasi *TCaption* turiga tegishli bo'ladi.

Color: TColor – komponent fonini aniqlaydi. Quyida *TColor* turiga tegishli bo'lgan ba'zi qiymatlar keltirilgan:

<i>clBlack</i>	qora
<i>clGreen</i>	yashil
<i>clSilver</i>	kumush rang
<i>clRed</i>	qizil
<i>clBlue</i>	ko'k
<i>clYellow</i>	sariq
<i>clWhite</i>	oq
<i>clWindow</i>	oyna rangida
<i>clAqua</i>	havorang
<i>clGray</i>	kulrang
<i>clNavy</i>	to'q ko'k
<i>clFunchia</i>	binafsha rang

Enabled: Boolean – xossa qiymati *true* bo'lsa, komponent aktiv holatga, *false* bo'lsa, passiv holatga o'tadi.

Hint: String – komponent ustiga sichqoncha ko'rsatkichi olib borilganda paydo bo'ladigan matnni aniqlaydi.

ShowHint: Boolean – xossa *true* qiymat qabul qilsa, *Hint* xossasi aktiv, *false* qiymat qabul qilsa, passiv holatga o'tadi.

Text: TCaption – komponentga yozilgan matnni aniqlaydi.

Visible: Boolean – xossa *true* qiymat qabul qilsa, komponent ekranda ko'rinib turadi, *false* qiymat qabul qilsa, novizual holatga o'tadi.

Protседuralar

BringToFront – komponentni barcha komponentlar old qismiga joylashtiradi.

SendToBack – komponentni barcha komponentlar ostki qavatiga joylashtiradi.

Hide – joriy komponentni unda joylashgan barcha ichki komponentlar bilan birgalikda ko'rinmas holatga o'tkazadi. Bu buyruq bajarilganda **Visible** xossasi avtomatik ravishda **False** qiymatga ega bo'ladi.

Show – komponentni unda joylashgan barcha komponentlar bilan birga vizual holatga o'tkazadi. Bu buyruq bajarilganda **Visible** xossasi avtomatik ravishda **True** qiymatga ega bo'ladi.

Refresh – komponentni tozalaydi.

Hodisalar

OnClick – bu hodisa sichqoncha chap tugmasi obyekt ustida bir marta bosib qo'yib yuborilganda yuzaga keladi.

OnMouseDown – sichqoncha chap tugmasi obyekt ustida bir marta bosilganda yuzaga keladi.

OnMouseMove – komponent ustiga sichqoncha ko'rsatkichi olib borilganda yuzaga keladi.

Delphi komponentlari

Delphi komponentlar palitrasida 400 dan ortiq turli komponentlar mavjud bo'lib, ular qo'llanilish sohasiga qarab bo'limlarga ajratilgan. Masalan, eng ko'p ishlatiladigan *Standard* komponentlar palitrasida quyidagilar o'rin olgan:

- **MainMenu** – dastur bosh menyusi. Komponent murakkab ierarxiya turdagi menyu yaratish imkonini beradi.
- **PopupMenu** – kontekst menyu yaratishga mo'ljallangan bo'lib, sichqonchani o'ng tomonini bosish bilan hosil bo'ladi.
- **Label** – bu komponent forma oynasida uncha uzun bo'lmagan bir qatorli yozuvni chiqarishda ishlatiladi.
- **Edit** – forma oynasida qatorli ma'lumotlarni kiritish, tahrirlash va chiqarishda ishlatiladi.
- **Memo** – forma oynasida ko'p qatorli ma'lumotlarni kiritish, tahrirlash va chiqarishda ishlatiladi.
- **Button** – buyruqlarni bajarishni ta'minlash uchun ishlatiladi.

- **CheckBox** – tasdiqlash, tanlash yoki aksincha holatlarni ko'rsatishda qo'llaniladi.
- **RadioButton** – komponent variantni tanlash imkoniyatini hosil qilishda ishlatiladi.
- **ListBox** – variantlarni ro'yxat bo'yicha taqdim etish va tanlash imkoniyatini yaratish uchun ishlatiladi.
- **ComboBox** – variantlarni tugma yordamida ochilgan ro'yxat bo'yicha taqdim etish va tanlash imkoniyatini yaratish uchun ishlatiladi.
- **ScrollBar** – o'tkazish tugmalarini bosil qilish uchun ishlatiladi.
- **GroupBox** – bir necha bog'liq komponentlarni guruhlashda ishlatiladi.
- **RadioGroup** – bu komponent variantlardan birini tanlash imkoniyatini hosil qilishda ishlatiladi. Yangi tanlash tugmasi bosilganda oldin tanlangan variant avtomatik bekor qilinadi.
- **Panel** – komponentlarni birlashtirishda ishlatiladi.
- **ActionList** — foydalanuvchi dasturga markazlashgan holda ta'sir qilishi uchun ishlatiladi.

Label, Edit, Memo matnli komponentlari, Button tugmasi

Label komponenti

Joylashishi: komponentlar palitrasining Standard bo'limi.

Komponent ekranda matnli axborotlarni, turli nom va sarlavhalarni joylashtirishga mo'ljallangan.

Label uchun bundan oldingi mavzuda sanab o'tilgan xossalar birga yana quyidagi xossalar ham joriy etilgan:

Alignment: *TAlignment;*

Bu xossa matnning komponentda joylashish holatini aniqlaydi.

TAlignment turi quyidagi qiymatlarni o'z ichiga oladi:

taLeftjustify – matnni chap tomonga tekislaydi;

taRightjustify – matnni o'ng tomonga tekislaydi;

TaCenter – matnni markazga tekislaydi.

AutoSize: *Boolean;*

Xossa *true* qiymat qabul qilsa, komponentning kengligi va balandligi unga yozilgan matnga mos ravishda avtomatik o'zgaradi.

Layout: *TtextLayout;*

Matnning vertikal o'q bo'yicha tartiblanishini aniqlaydi. *TtextLayout* turi quyidagi qiymatlarga ega:

tlTop – yuqoridan tekislash;

tlBottom – quyidan tekislash;

tlCenter – markazga tekislash.

Label komponenti uchun eng asosiy xossa *Caption* xossasidir. Bu xossa aynan ekranga chiqishi kerak bo'lgan matnни aniqlaydi.

***TCustomEdit* sinfi**

Delphida ma'lumotlarni kiritish, chiqarish va tahrirlashga mo'ljallangan komponentlar mavjud. Ulardan eng asosiylari *TCustomEdit* sinfining *Edit* va *Memo* komponentlaridir. Quyida ushbu ikki komponent uchun eng asosiy xossa va amallar keltirilgan:

Xossalar:

AutoSize: *Boolean*;

Bu xossa True qiymatni qabul qilsa, komponent o'lchami unga yozilayotgan matnga qarab avtomatik o'zgaradi.

MaxLength: *Integer*;

Komponentga nechtagacha simvol joylashtirish mumkinligini aniqlaydi.

PasswordChar: *char*;

Bu xossa komponentga yozilgan matnning barcha harflari bir xil, ko'rsatilgan simvol ko'rinishida tasvirlanishini bildiradi. Bu imkoniyat dasturda parol tashkil etishda qo'llaniladi.

ReadOnly: *Boolean*;

Xossa *true* qiymatni qabul qilsa, komponentdagi yozuvni faqat o'qish mumkin bo'lib, nusxa olish, o'chirish, o'zgartirish mumkin emas.

Protseduralar:

Clear;

Komponentdagi barcha matnни o'chiradi.

ClearSelection;

Matnning belgilangan qismini o'chiradi.

CopytoClipboard;

Belgilangan matnни almashinuv buferiga nusxalaydi.

CuttoClipboard;

Belgilangan matnни almashinuv buferiga ko'chiradi.

PasteFromClipboard;

Almashinuv buferidagi matnни komponentdagi kursor turgan joyga qo'yadi.

SelectAll;

Komponentdagi barcha matnни belgilaydi.

Edit komponenti

Joylashishi: komponentlar palitrasining Standard ho'limi.

Bu komponent ma'lumotlarni kiritish, chiqarish va tahrirlashga mo'ljallangan. Uning asosiy xossasi *Text* hisoblanib, aynan uning qiymati kiritilayotgan, chiqarilayotgan yoki tahrirlanayotgan ma'lumot hisoblanadi. *TEdit* sinfi uchun yuqoridagilar bilan birga quyidagi hodisa ham joriy etilgan:

OnChange;

Dastur ishlab turganda komponentdagi matn o'zgartirilganda yuzaga keladi.

Memo komponenti

Joylashishi: komponentlar palitrasining Standard ho'limi.

Memo komponenti ham xuddi *Edit* kabi ma'lumotlarni kiritish, tahrirlash va ekranda aks ettirish uchun mo'ljallangan. Uning *Edit* dan eng asosiy farqi ma'lumot faqat bitta satrdan emas, balki ixtiyoriy sondagi satriardan tashkil topishi mumkin.

TMemo sinfi uchun quyidagi xossalar ham joriy etilgan:

ScrollBars: TScrollStyle;

Memo ga vertikal va gorizontal chizg'ichni qo'yadi yoki uni olib tashlaydi. *TScrollStyle* turiga quyidagi qiymatlar tegishli:

ssNone – chizg'ich yo'q;

ssHorizontal – faqat gorizontal chizg'ich bor;

ssVertical – faqat vertikal chizg'ich bor;

ssBoth – ikkala chizg'ich ham bor.

Lines: TStrings;

Komponentdagi matn qatorlarini aniqlaydi.

TStrings sinfi

TStrings sinfiga tegishli ba'zi xossalar:

Strings[Index: Integer]: String;

Index qiymatli qatorni aniqlaydi.

Text: String;

Memo ga yozilgan matnни aniqlaydi.

TStrings sinfiga tegishli ba'zi amallar:

Add(const S: string);

Memo dagi mavjud matn oxiriga *S* matnни qo'shadi.

Clear; virtual; abstract;

Memo ni tozalaydi.

Delete(Index: Integer);

Index qiymatli qatorni o'chiradi.

Exchange(Index1, Index2: Integer);

Index1 va *Index2* qatorlar o'zini almashtiradi.

Insert(Index: Integer; const S:string);

Index tartib raqamli qatorga *S* satrni qo'shadi.

LoadFromFile(const FileName:string);

FileName nomli fayldagi ma'lumotni *Memo* ga chiqaradi.

Move(CurIndex, NewIndex: Integer);

CurIndex nomerli satrdagi matnni *NewIndex* nomerli satrga ko'chiradi.

SaveToFile(const FileName:string);

Memo dagi matnni *FileName* nomli faylga saqlaydi.

Button tugmasi

Joylashishi: komponentlar palitrasining Standard bo'limi.

Bu komponent bosilishi natijasida o'ziga birlashtirilgan amallarni bajarishni ta'minlaydi.

Label, Edit, Memo va Button komponentlari yordamida dasturlar tuzish

Misol. $ax^2+bx+c=0$ ko'rinishidagi kvadrat tenglamaning ildizlarini topib beruvchi dastur tuzing.

Bajarish:

1. Delphi muhitini ishga tushiramiz. Natijada avtomatik ravishda yangi loyiha (fayl) hosil bo'ladi. *File/New Application* buyrug'i orqali qaytadan yangi loyiha yaratish ham mumkin.
2. *File/Save All* buyrug'i orqali loyihamizni bitta alohida papkaga saqlab qo'yamiz.
3. Formaga 4 ta *Label*, 3 ta *Edit*, 1 ta *Memo* va 1 ta *Button* tugmalarini joylashtiramiz (10-rasm).

Ular avtomatik mos ravishda *Label1*, *Label2*, *Label3*, *Label4*, *Edit1*, *Edit2*, *Edit3*, *Memo1*, *Button1* nomlarini oladi. Komponentlarning joylashishi va o'lchamini sichqoncha yordamida o'zgartirishimiz mumkin.

4. *Label1* komponentning *Caption* xossasiga " $a*x^2+bx+c=0$ tenglamaning koeffitsientlarini kiriting" so'zlarini yozamiz. Bu matn ikki qatorga va o'rtaga yozilishi uchun komponent uchun quyidagi xossalarni o'rnatamiz.

AutoSize - False

WordWrap - True

Alignment - taCenter



10-rasm. Forma loyihasi

Komponentning *Font* xossasi qiymat darchasida turgan uch nuqta ustida sichqoncha chap tugmasini bosamiz. Hosil bo'lgan shrift oynasidan shrift o'lchamini kattalashtirib, rang beramiz.

5. *Label2*, *Label3* va *Label4* larning *Caption* xossasiga mos ravishda "a=", "b=" va "c=" yozuvlarini kiritamiz. Har uch komponentning *Label1* kabi rang va shrift o'lchamlarini o'zgartiramiz.
6. Dastur ishga tushgan vaqtda qiymat kiritiladigan maydonlar bo'sh turishi uchun *Edit1*, *Edit2* va *Edit3* larning *Text* xossasi qiymatida turgan yozuvni o'chiramiz. *Label* lar kabi *Edit* larning rang va shrift o'lchamlarini o'zgarimiz.
7. *Memo* komponentining *Lines* xossasi qiymat darchasidagi uch nuqta ustida sichqoncha chap tugmasini bosamiz. Hosil bo'lgan oynada "Memo1" so'zini o'chirib, *OK* tugmasini bosamiz.
8. *Button1* tugmasining *Caption* xossasi qiymatiga "Natija" so'zini kiritamiz.
9. "Natija" tugmasi ustida sichqoncha chap tugmasini ikki marta bosamiz. Hosil bo'lgan *kod redaktoridagi* protseduraga quyidagilarni kiritamiz:

```

procedure TForm1.Button1Click(Sender: TObject);
var a,b,c,d,x,x1,x2:real;
begin
  Memo1.Clear;
  a:=strtofloat(Edit1.text);
  b:=strtofloat(Edit2.text);
  c:=strtofloat(Edit3.text);
  if a=0 then

```

```

begin
  x:=-c/b;
  Memo1.Lines.Add('x='+FloatToStr(x));
end
else
begin
  d:=sqrt(b)-4*a*c;
  if d>0 then
    begin
      x1:=(-b-sqrt(d))/(2*a);
      x2:=(-b+sqrt(d))/(2*a);
      Memo1.Lines.Add('x1= '+FloatToStr(x1));
      Memo1.Lines.Add('x2= '+FloatToStr(x2));
    end
  else
    if d=0 then
      begin
        x:=-b/(2*a);
        Memo1.Lines.Add('x='+FloatToStr(x));
      end
    else Memo1.Lines.Add('tenglama yechimga ega emas')
  end;
end;
end.

```

10. F9 funksional tugmasi yoki *Run* menyusining *Run* buyrug'i orqali dasturni ishga tushiramiz. Natijada tuzgan dasturimizning yakuniy formasi hosil bo'ladi (11-rasm). Formada *a*, *b*, *c* larga kerakli qiymatlarni kiritib natijalar olish mumkin.
11. Bajirilgan ishlarni saqlab qo'yish uchun yana bir marta *File/Save* buyrug'ini bosamiz.

Grafik-tasvirlar yaratishga mo'ljallangan sinf va buyruqlar

Delphi da grafik va tasvirlarni yaratishga mo'ljallangan quyidagi sinflar mavjud:

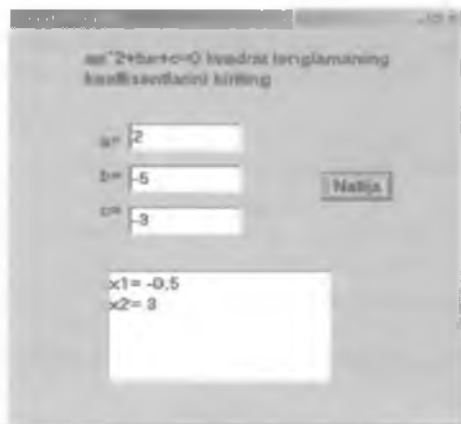
TFont (shrift), **TPen** (qalam), **TBrush** (cho'tka), **TCanvas** (Kanva).

Bu sinflar uskunalari va xossalari yordamida grafik va tasvirlar hosil qilishda quyidagi komponentlar obyekt sifatida ishlatiladi:

Image – asosan tayyor rasm va grafiklarni ekranga chiqarish uchun ishlatiladi.

Shape – obyektida tayyor oddiy geometrik shakllar: kvadrat, aylana, ellips kabilarni aks ettirishda ishlatiladi.

PaintBox – qo‘l mehnati yordamida ixtiyoriy grafik va tasvirlar hosil qilish uchun mo‘ljallangan.



11-rasm. Kvadrat tenglamani yechish dasturi

TFont sinfi

TFont sinfi shrift xossa va xususiyatlarini aniqlaydi.

TFont sinfidagi quyidagi xossalari joriy etilgan:

Color: *TColor*;

Bu xossa yozuv rangini aniqlaydi.

Height: *Integer*;

Shrift balandligini piksellar hisobida aniqlaydi.

Style: *TFontStyle*;

Yozuv ko‘rinishini aniqlaydi.

TFontStyle turining quyidagi qiymatlari mavjud:

fsBold – qalin yozuv;

fsItalic – qiya yozuv;

fsUnderline – tagiga chizilgan yozuv.

TPen sinfi

Bu sinf ixtiyoriy chiziq chizishga mo‘ljallangan qalamning xususiyatlarini aniqlaydi.

TPen sinfining asosiy xossalari quyidagilar:

Color: *TColor*;

Qalam rangini aniqlaydi.

Mode: *TPenMode*;

Chiziq holatini aniqlaydi. *TPenMode* turining ba'zi qiymatlari quyidagilar:

PmBlack – qalam har doim qora rangda chizadi;

PmWhite – har doim oq rangda chizadi;

PmCopy – har doim *Color* xossasi rangi bo'yicha chizadi;

PmMerge – ekran va *Color* xossasi ranglari aralashmasi bo'yicha chizadi.

Width: Integer;

Chiziq qalinligini aniqlaydi.

***TBrush* sinfi**

Bu sinf sohani bo'yashda qo'llaniladigan cho'tka xususiyatlarini aniqlaydi.

Cho'tkaning asosiy xossalari quyidagilar:

Color: TColor;

Cho'tka rangini aniqlaydi. Odatda u oq rangda bo'ladi.

Style: TBrushStyle;

Cho'tka naqshini aniqlaydi. *TBrushStyle* turiga tegishli ba'zi qiymatlar:

bsSolid – yaxlit naqsh;

bsClear – naqshsiz;

bsHorizontal – gorizontali chizikli naqsh;

bsVertical – vertikal chizikli naqsh;

bsCross – katak ko'rinishidagi naqsh.

***TCanvas* sinfi**

Bu sinf shrift, qalam va cho'tkalar yordamida hosil qilinadigan grafik va tasvirlar obyektini aniqlaydi. *TCanvas* sinfining obyekt mavjud emas, u *Image*, *PainBox*, *Shape*, *Form* kabi komponentlarni obyekt sifatida qabul qiladi.

TCanvas sinfining asosiy xossalari quyidagilar:

Brush: Tbrush;

Obyektdagi cho'tka xossalarini aniqlaydi.

Font: Tfont;

Obyektdagi shrift xossalarini aniqlaydi.

Pen: Tpen;

Obyektdagi qalam xossalarini aniqlaydi.

PenPos: Tpoint;

Qalamning obyektida joriy vaqtdagi pozitsiyasini (koordinatasini) aniqlaydi.

Pixels[X,Y: Integer]: Tcolor;

(X,Y) koordinatali piksel rangini aniqlaydi.

Canvas obyektlarida oddiy grafik va shakllarni yaratishga mo'ljallangan ba'zi buyruqlar:

LineTo(X,Y : Integer);

Qalamning joriy vaqtdagi pozitsiyasidan (X,Y) koordinatali nuqtagacha to'g'ri chiziq chizadi (dastlabki holda joriy pozitsiya (0,0) koordinatada bo'ladi). Bu holda qalamning *PenPos* – joriy pozitsiyasi (X,Y) koordinataga o'zgaradi.

MoveTo(X,Y: Integer);

Qalamning joriy vaqtdagi pozitsiyasi (X,Y) koordinataga ko'chadi.

Rectangle(X1,Y1,X2,Y2 : Integer);

Yuqori chap burchak koordinatasi (X1,Y1), quyi o'ng burchak koordinatasi (X2,Y2) bo'lgan to'g'ri to'rtburchak chizadi.

Ellipse(X1,Y1,X2,Y2 : Integer);

Yuqori chap burchak koordinatasi (X1,Y1), quyi o'ng burchak koordinatasi (X2,Y2) bo'lgan to'g'ri to'rtburchakli shablon ichiga ellips chizadi.

TextOut(X,Y : Integer; const Text: string);

Yuqori chap burchak koordinatasi (X,Y) bo'lgan to'g'ri to'rtburchakli shablon ichiga buyruqda ko'rsatilgan *Text* yozuvini chiqaradi.

TextHeight(const Text : String): Integer;

Text yozuvining balandligini aniqlaydi.

TextWidth(const Text : String): Integer;

Text yozuvining uzunligini aniqlaydi.

Image, Shape, Paintbox, Timer komponentlari

Image komponenti

Joylashishi: komponentlar Palitrasining Additional bo'limi.

TImage sinfiga tegishli bo'lgan *Image* komponenti formada ba'zi turdagi rasm va kartinalarni joylashtirishga mo'ljallangan. Rasm va kartinalar .ico, .bmp, .wmf kengaytmali fayllar bo'lishi lozim.

TImage sinfiga *TCanvas* sinfining xossalari bilan birga yana quyidagi xossalar ham joriy qilingan:

Canvas: TCanvas;

Komponentda tasvir shakllanishini bildiradi.

Center: Boolean;

Xossa *true* qiymatni qabul qilsa, tasvir komponent markaziga, *false* qiymat qabul qilsa, tasvir komponent yuqori chap burchagiga joylashadi.

Picture: TPicture;

Komponentda joylashadigan rasmni aniqlaydi.

Stretch: *Boolean*;

Rasmning o'lchami komponent o'lchamiga moslashishini aniqlaydi.

Shape komponenti

Joylashishi: komponentlar Palitrasining Additional bo'limi.

Bu komponent tayyor standart shakllarni aks ettirish uchun mo'ljallangan. *TShape* sinfida quyidagi xossalalar ham joriy qilingan:

Brush: *TBrush*;

Shakl ichini bo'yovchi cho'tkani aniqlaydi.

Pen: *TPen*;

Shakl chegarasini chizuvchi qalamni aniqlaydi.

Shape: *TShapeType*;

ekranga chiquvchi shakl turini aniqlaydi.

TShapeType turi quyidagi qiymatlarni qabul qiladi:

StRectangle – to'g'ri to'rtburchak;

StSquare – kvadrat;

StRoundRect – qirrasiz aylana shaklidagi to'g'ri to'rtburchak;

StRoundSquare – qirrasiz aylana shaklidagi kvadrat;

StEllipse – ellips;

StCircle – aylana.

PaintBox komponenti

Joylashishi: komponentlar palitrasining System bo'limi.

TPaintBox sinfining *PaintBox* komponenti chegaralangan sohada ixtiyoriy chizishga mo'ljallangan. *TPaintBox* sinfi *TCanvas* sinfiga tegishli bo'lgan barcha xossalarni o'z ichiga oladi.

Timer komponenti

Joylashishi: komponentlar palitrasining System bo'limi.

Timer sinfining *Timer* komponenti novizual komponent bo'lib, yordamchi komponent hisoblanadi. Bu komponent kerakli amallarning biror vaqtda yoki vaqt oraliqlarida bajarilish imkonini beradi.

Timer sinfida joriy etilgan xossa va hodisalar:

Enabled: *Boolean*;

Xossa *True* qiymat qabul qilsa, *Timer* ishga tushadi, *false* qiymat qabul qilsa *Timer* ishlashdan to'xtaydi.

Interval: *Cardinal*;

OnTimer hodisasiga yuklangan amallarning yuz berish oraliqlarini millisoniyalarda aniqlaydi.

OnTimer;

Bu hodisa intervalda berilgan vaqt tugagan vaqtda sodir bo'ladi. Vaqt bilan bog'liq amallar aynan shu hodisaga birlashtiriladi.

Image va Timer komponentlari yordamida dasturlar tuzish

Misol. Dastur uchun "Заставка" yarating.

"Заставка" dastur ishga tushishidan oldin yoki dastur ishlashi jarayonida dasturchi, dastur yaratilish vaqti, umuman ixtiyoriy ma'lumotlarni namoyish etishda ishlatiladi. Masalan, Windows OT, ofis dasturlari ishga tushayotganda bir necha soniya vaqt davomida dastur haqida qisqacha ma'lumotnoma ekranda paydo bo'ladi.

"Заставка"ni ixtiyoriy vaqtda ekranga chiqarib, ixtiyoriy vaqtda yo'qoladigan qilib, yoki ma'lum vaqt oraliqlarida ekranga chiqadigan qilib yaratish mumkin. "Заставка"ga chiqadigan rasm *.bmp* kengaytmali bo'lishi lozim. Buni ixtiyoriy grafik redaktoridan foydalanib, hosil qilish mumkin.

Bajarish.

1. *D:\MyProject\Zastavka* papkasini yaratamiz.
2. Delphi ni ishga tushiramiz. *File/New/Application* buyrug'i yordamida yangi loyiha yaratamiz yoki oldin yaratgan ixtiyoriy dasturimizni ochamiz. Loyihani *D:\MyProject\Zastavka* papkasiga saqlab qo'yamiz.
3. Delphi menyusining *Tools/Image Editor* buyrug'ini bosamiz. Natijada ekranda Delphi dasturining grafik redaktori hosil bo'ladi (12-rasm).



12-rasm. Image Editor oynasi

4. Grafik fayl hosil qilish uchun *File/Open* buyrug'i orqali oldin yaratgan ixtiyoriy *.bmp* kengaytmali faylni ochishimiz yoki *File/New/Bitmap File* buyrug'i yordamida yangi fayl hosil qilishimiz mumkin. Biz keyingi buyruqni tanlaymiz. Ochilgan muloqot oynasidagi *Width*

qiymatiga 300, *Height* qiymatiga 200, *Color* qismidan *SuperVGA* ni tanlab, OK tugmasini bosamiz. Hosil bo'lgan ramkaga ixtiyoriy ko'rinish chizamiz. Masalan, ramkaga biror rangdagi fon berib, uskunalar paneli yordamida o'rta quyidagicha yozamiz:

DASTUR 2012 YILDA ... TOMONIDAN YARATILGAN

5. Faylimizni *File/Save* buyrug'i orqali *D:\MyProject\Zastavka* papkasiga *Rasm1.bmp* nomi bilan saqlaymiz. *File/Exit* buyrug'i orqali grafik redaktoridan chiqamiz.
6. Komponentlar palitrasining *System* bo'limidan formaga 1 ta *Timer* komponentini joylashtiramiz. U *Timer1* deb nom oladi. *Timer1* ning *Interval* xossasiga millisekund hisobida 3000 sonini kiritamiz.
7. Komponentlar palitrasining *Additional* bo'limidan formaga 1 ta *Image* komponentini joylashtiramiz. *Image1* deb nom oladigan componentning *Picture* xossasida turgan uch nuqta ustida sichqoncha chap tugmasini bosamiz. Hosil bo'lgan tasvir yuklash oynasining *Load* buyrug'i orqali *D:\MyProject\Zastavka\Rasm1.bmp* faylini ochamiz va OK tugmasini bosamiz. Rasm to'liq ko'rinishi uchun *Image* komponentining *Autosize* xossasiga *True* qiymatni kiritamiz.
8. *Timer1* komponenti ustida sichqoncha chap tugmasini ikki marta bosamiz. Hosil bo'lgan kod redaktoridagi protseduraga quyidagi buyruqlarni yozamiz:

```
Procedure TForm1.Timer1Timer(Sender: TObject);  
Begin  
    Image1.Free;  
    Timer1.Free;  
End.
```

Bu protsedura dastur ishga tushirilgandan so'ng 3000 millisekund o'tgach, zastavka butunlay yo'qolishini anglatadi.

9. Dasturni ishga tushirib, uning qanday ishlashini ko'rishimiz mumkin. Uning ko'rinishi quyidagicha bo'ladi (13-rasm).
10. Bajargan ishlarimizni saqlash uchun yana bir marta saqlash tugmasini bosamiz.



13-rasm. Zastavka ko'rinishi

Agar zastavka 3000 millisoniyadan so'ng yo'qoliishi sharti o'f miga har 3000 millisoniyadan so'ng yo'qolib, har 3000 millisoniyadan so'ng paydo bo'lsin degan masala qo'yilgan bo'lsa, yuqoridagi protseduraning begin va end so'zlari orasidagi yozuvni quyidagicha o'zgartiramiz:

```

Procedure TForm1.Timer1Timer(Sender: TObject);
Begin
  If Image1.Visible=true then Image1.Hide
  Else Image1.Show
End;

```

Masala sharti bajarildi.

Chiziqli algebraik tenglamalar sistemasini yechishning Gauss usuli

Ma'lumki, ko'pgina muhandislik masalalari chiziqli tenglamalar sistemasini yechishga olib kelinadi.

Bizga quyidagi tenglamalar sistemasi berilgan bo'lsin:

$$\begin{cases}
 a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = a_{1n+1}, \\
 a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = a_{2n+1}, \\
 \dots \\
 a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = a_{nn+1}.
 \end{cases} \quad (1)$$

Yoki matritsa ko'rinishida $AX = B$, bu yerda

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}, \quad B = \begin{pmatrix} a_{1n+1} \\ a_{2n+1} \\ \dots \\ a_{nn+1} \end{pmatrix}$$

Gauss usulida tenglamalar noma'lumlari ketma-ket yo'qotib boriladi. Ya'ni 2-tenglamadan x_1 ni, 3-tenglamadan x_1 va x_2 larni va hokazo. Natijada sistema uchburchak matritsali sistemaga keltiriladi va undan x_n, x_{n-1}, \dots, x_1 noma'lumlar topiladi. Gauss usulining algoritmi quyidagicha:

Har bir k -qatordagi ($k = j+1, \dots, n$) elementlarga mos ravishda j -qatordagi elementlarni $-a_{kj}/a_{jj}$ koeffitsiyentlarga ko'paytirib qo'shiladi. Natijada j -ustundagi elementlarning bosh diagonaldan pastdagilari nolga aylanadi. Bu almashtirishlar B vektor ustida ham bajariladi. Algoritmning bu jarayoniga Gauss usulining to'g'ri yo'li deyiladi.

Hosil bo'lgan uchburchakli tenglamalar sistemasidan noma'lumlarni topishga Gauss usulining teskari yo'li deyiladi. Bu sistemadan $x_n = a_{n,n+1}/a_{nn}$ topiladi. Bu yerda $a_{nn}, a_{n,n+1}$ lar almashtirishlardan so'ng hosil bo'lgan uchburchak matritsali tenglamalar sistemasi oxirgi tenglamasining ozod hadi va x_n oldidagi koeffitsiyent.

So'ngra x_{n-1} topiladi: $x_{n-1} = (a_{n-1,n+1} - a_{n-1,n}x_n)/a_{n-1,n-1}$.

Noma'lumlar umumiy holda quyidagicha topiladi:

$$x_i = \frac{a_{i,n+1} - \sum_{j=i+1}^n a_{ij}x_j}{a_{ii}}, \quad i = n, n-1, \dots, 1. \quad (2)$$

(1) ko'rinishdagi tenglamalar sistemani yechish algoritmi blok-sxema ko'rinishida quyidagicha bo'ladi:

Dasturi:

Delphi dasturini ishga tushiramiz va yangi loyiha yaratib, formaga 4 ta Label, 1 ta Edit, 3 ta StringGrid, 2 ta Button komponentlarini joylashtiramiz. Label1, Label2, Label3, Label4 komponentlarining *Caption* xossalari-ga mos ravishda “n ni kiriting va OK ni bosing ($n \leq 15$)”, “Noma”lular koeffitsiyentlari”, “Ozod hadlar”, “Natija” sozlarini kiritib chiqamiz, Edit1 ning “Text” xossasi qiymatini o’chiramiz, Button tugmalariga mos ravish-da “Ozod hadlar”, “Natija” so’zlarini yozamiz. StringGrid lar hajmlarini kerakli hajmda o’zgartirib chiqamiz, StringGrid1 ning *ColCount* (qatorlar soni) va *RowCount* (ustunlar soni) xossalari-ga 1 sonini kiritamiz, *FixedCols* (fiksirlangan qatorlar soni), *FixedRows* (fiksirlangan ustunlar soni) xossalari-ga 0 sonini kiritamiz, StringGrid larning *DefaultColWidth* xossasi orqali jadval ustunlari kengligini o’zgartiramiz, ma’lumotlarni kiritishga ruxsat berilishi uchun StringGrid1 va StringGrid2 larning *Options* xossalari ichidagi *goEditing* xossasi qiymatini “True”ga o’zgarti-ramiz. Natijada forma 14-rasmdagi ko’rinishga keladi:



14-rasm. Chiziqli tenglamalar sistemasini yechish dasturi loyihasi

Button1 tugmasini sichqoncha yordamida ikki marta bosib, hosil bo’lgan protseduraga quyidagi dasturni kiritamiz:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  n:=StrToInt(Edit1.Text);
  StringGrid1.ColCount:=n; StringGrid1.RowCount:=n;
```

```

StringGrid2.RowCount:=n; StringGrid3.RowCount:=n;
for i:=0 to n-1 do
for j:=0 to n-1 do StringGrid1.Cells[i,j]:= "";
for j:=0 to n-1 do StringGrid2.Cells[0,j]:= "";
for j:=0 to n-1 do StringGrid3.Cells[0,j]:= "";
end;

```

F12 tugmasi yordamida formaga qaytib, Button2 tugmasi ustida sichqoncha chap tugmasini ikki marta bosamiz. Hosil bo'lgan navbatdagi protseduraga quyidagi dasturni kiritamiz.

```

procedure TForm1.Button2Click(Sender: TObject);
var a:array[1..15, 1..16] of real;
    x:array[1..15] of real; c,s,d:real; i,j,k,l,p:integer;
begin

```

```

    for i:=1 to n do
        begin
            for j:=1 to n do
                a[i,j]:=StrToFloat(StringGrid1.Cells[j-1,j-1]);
                a[i,n+1]:=StrToFloat(StringGrid2.Cells[0,i-1])
            end;

```

```

        for k:=1 to n do
            begin
                if a[k,k]=0 then
                    begin
                        l:=k;
                        while a[l+1,k]=0 do l:=l+1;
                        for p:=k to n+1 do
                            begin
                                d:=a[k,p];
                                a[k,p]:=a[l+1,p];
                                a[l+1,p]:=d;
                            end;
                        end;
                        for i:=k to n-1 do
                            begin
                                c:=a[i+1,k];
                                for j:=k to n+1 do
                                    a[i+1,j]:=(a[k,j]/a[k,k])*c-a[i+1,j];
                                end;
                            end;
                        x[n]:=a[n,n+1]/a[n,n];

```

```

for k:=n-1 downto 1 do
begin
  S:=0;
  for i:=k+1 to n do
    S:=S+a[k,i]*x[i];
  x[k]:=(a[k,n+1]-S)/a[k,k];
end;
for i:=1 to n do
  StringGrid3.Cells[0,i-1]:=floattostr(x[i])
end;

```

F9 tugmasini bosib, yakuniy ishchi formani hosil qilamiz. Dastur yordamida masalan, quyidagi tenglamalar sistemasini yechamiz (15-rasm).

$$\begin{cases}
 3x_1 + 2x_2 - 5,8x_3 + 2x_4 + 4,6x_5 + 4x_6 + 3,5x_7 = 3,5 \\
 -5x_1 + 7x_2 + 3x_3 - 1,5x_4 + 3x_5 + 2x_6 + x_7 = 1 \\
 2,6x_1 - 6,7x_2 + 5x_3 + 2x_4 + 6x_5 + 6x_6 + 6x_7 = -5,6 \\
 3,8x_1 + 4x_2 + 3x_3 + 7x_4 + 10x_5 + 4x_6 = 0 \\
 -4x_1 - 2x_2 + 2,5x_3 + 9x_4 + 5,2x_5 + 2x_6 + 4x_7 = 3 \\
 8x_1 + 5x_2 - 1,6x_3 + 12x_4 + 7x_5 + 7,5x_7 = 7 \\
 4x_1 - 6x_2 + 3x_3 + 8x_4 + 3x_5 + 11x_6 + 4x_7 = 10
 \end{cases}$$



15-rasm. Chiziqli tenglamalar sistemasini yechish dasturi

Eng sodda taqribiy integrallash formulalari

Bizdan $[a, b]$ kesmada aniqlangan uzluksiz $f(x)$ funksiyani integrallash talab qilingan bo'lsin.

Agar $f(x)$ funksiyaning boshlang'ich funksiyasi ma'lum va u $F(x)$ ga teng bo'lsa, u holda berilgan funksiyaning aniq integrali quyidagi Nyuton-Leybnis formulasi yordamida hisoblanar edi:

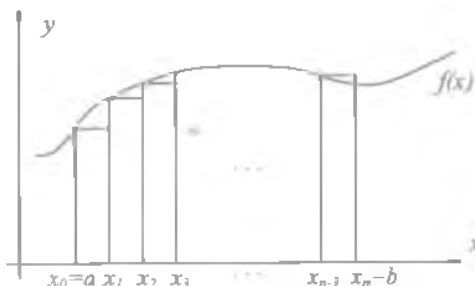
$$\int_a^b f(x) dx = F(b) - F(a), \quad \text{bunda } F'(x) = f(x) \quad (1)$$

Ma'lumki, amaliyotda berilgan funksiyaning boshlang'ich funksiyasini har doim ham topib bo'lmaydi. Topilsa ham juda murakkab ifodalanganligi uchun bunday funksiyalarning aniq integrallarini taqribiy hisoblash uchun kvadratur yoki *kuhatur* formulalardan foydalaniladi.

Eng sodda kvadratur formulalarga to'g'ri to'rtburchaklar, trapetsiya va Simpson kvadratur formulalari misol bo'la oladi.

To'g'ri to'rtburchaklar formulasi

Faraz qilaylik, bizga $f(x)$ funksiyaning grafigi 16-rasmda keltirilgan ko'rinishda berilsin.



16-rasm. Sohani to'g'ri to'rtburchaklar ko'rinishida tasvirlash

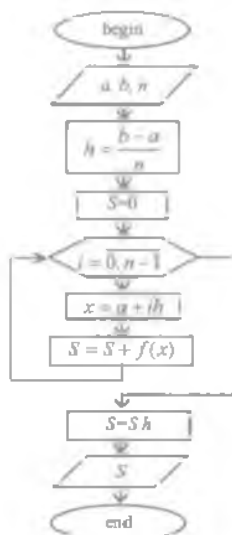
$[a, b]$ kesmani teng qadamlar bilan n ta teng bo'lakka bo'lamiz va bo'limish nuqtalarini $x_k = a + kh$ bilan belgilaymiz. Bu yerda $h = (b - a) / n$, $k = 0, 1, \dots, n$. y_k bilan $f(x)$ funksiyaning x_k nuqtadagi qiymatini belgilaymiz, ya'ni $y_k = f(x_k)$. U holda $f(x)$ funksiyaning $[a, b]$ kesmadagi aniq integrali hosil qilingan to'g'ri to'rtburchaklarning yig'indisiga taqriban teng bo'ladi:

$$\int_a^b f(x) dx \approx h \sum_{k=0}^{n-1} f(x_k) \quad (2)$$

Hosil qilingan kvadratur formulaga *chap to'g'ri to'rtburchaklar formulasi* deyiladi. Xuddi shu usul bilan *o'ng to'g'ri to'rtburchaklar formulasi* ham hosil qilinadi:

$$\int_a^b f(x) dx \cong h \sum_{k=1}^n f(x_k) \quad (2)$$

Algoritmi:



Misol. $\int_{0,25}^{0,75} \frac{dx}{\sqrt{\cos x - x^2}}$ aniq integralni *to'g'ri to'rtburchaklar formulasi*

yordamida taqribiy hisoblash dasturi tuzilsin.

Dasturi. Delphi da yangi loyiha yaratib, formaga misol berilishini kiritish uchun 1 ta Label, kiritiladigan va chiqariladigan ma'lumotlar uchun 4 ta Label va 4 ta Edit hamda 1 ta Button komponentlarini joylashtiramiz. *Form1* ning *Caption* xossasi qiymat darchasiga "T To'rtburchaklar usuli" so'zini, *Label1* komponenti *Caption* xossasiga misol berilishini, *Label2*, *Label3*, *Label4*, *Label5* komponentlariga mos ravishda "a=", "b=", "n=", "S=" so'zlarini kiritamiz. Edit larning *Text* xossalaridagi qiymatlarni o'chiramiz. *Button1*ning *Caption* xossasiga "OK" so'zini kiritamiz. Natijada forma 17-rasmdagi holatga keladi.

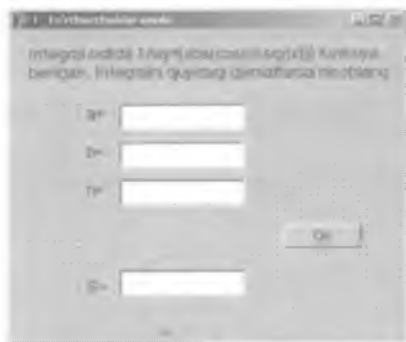
Button1 tugmasi ustida sichqoncha chap tugmasini ikki marta bosib, hosil bo'lgan protseduraga quyidagi dasturni kiritamiz:

```

procedure TForm1.Button1Click(Sender: TObject);
var S,a,b,h,x:real; n,i:integer;
  
```

Begin

```
a:=StrToFloat(Edit1.Text);  
b:=StrToFloat(Edit2.Text);  
n:=StrToInt(Edit3.Text);  
h:=(b-a)/n;  
S:=0;  
for i:=0 to n-1 do  
  begin  
    x:=a+i*h;  
    S:=S+1/sqrt(abs(cos(x)-sqr(x)))  
  end;  
S:=S*h;  
Edit4.Text:=FloatToStr(S)  
end;
```



17-rasm. To'g'ri to'rtburchaklar dasturi loyihasi

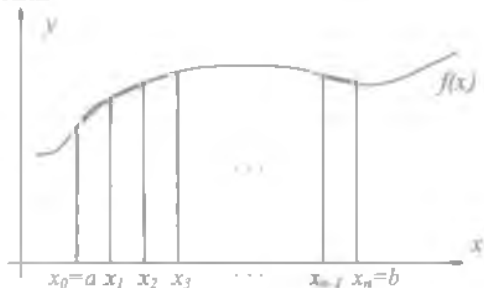
F9 tugmasini bosib, kerakli qiymatlarni kiritamiz. Natija 18-rasmdagidek bo'ladi:



18-rasm. Integralni to'g'ri to'rtburchaklar formulasi yordamida hisoblash dasturi

Trapetsiya formulasi

Bu yerda ham $f(x)$ funksiyaning grafigi 19-rasmda keltirilgan ko'rinishda berilsin.



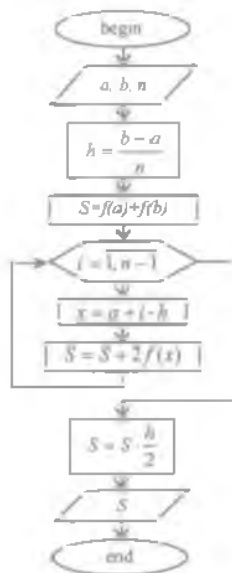
19-rasm. Sohani trapetsiyalar ko'rinishida tasvirlash

U holda $f(x)$ funksiyaning $[a, b]$ kesmadagi aniq integrali hosil qilingan trapetsiyalarning yig'indisiga taqriban teng bo'ladi:

$$\int_a^b f(x) dx \approx \frac{h}{2} (f(a) + f(b) + 2 \sum_{k=1}^{n-1} f(x_k)) \quad (3)$$

Keltirilgan kvadratur formulaga *trapetsiya formulasi* deyiladi.

Algoritmi:



Misol. Yuqoridagi $\int_{0.25}^{1.75} \frac{dx}{\sqrt{|\cos x - x^2|}}$ aniq integralni *trapetsiya formulasi*

yordamida taqribiy hisoblash dasturi tuzilsin.

Dasturi. Dastur tuzishda yuqoridagi formadan foydalanamiz. Dastur protsedurasi quyidagicha bo'ladi:

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var S,a,b,h,x:real;
```

```
    n,i:integer;
```

```
function f(t:real):real;
```

```
begin
```

```
    f:=1/sqrt(abs(cos(t)-sqr(t)))
```

```
end;
```

```
begin
```

```
    a:=StrToFloat(Edit1.Text);
```

```
    b:=StrToFloat(Edit2.Text);
```

```
    n:=StrToInt(Edit3.Text);
```

```
    h:=(b-a)/n;
```

```
    S:=f(a)+f(b);
```

```
    for i:=1 to n-1 do
```

```
        begin
```

```
            x:=a+i*h;
```

```
            S:=S+2*f(x)
```

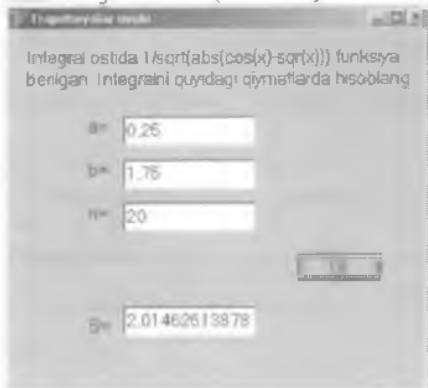
```
        end;
```

```
    S:=S*h/2;
```

```
    Edit4.Text:=FloatToStr(S)
```

```
end;
```

F9 tugmasini bosib natija olamiz (20-rasm).

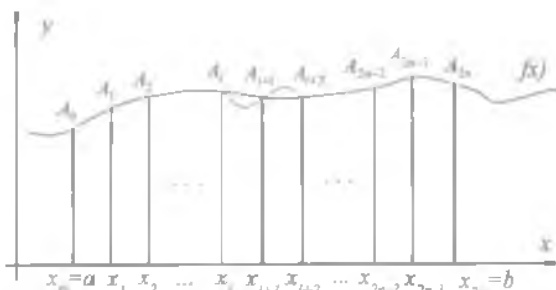


20-rasm. Integralni *trapetsiya* formulasi yordamida hisoblash dasturi

Simpson formulasi

Bu formulani hosil qilish uchun $[a, b]$ kesmani $2n$ ta teng bo'laklarga ajratamiz, ya'ni $x_i = a + ih$, $h = \frac{b-a}{2n}$, $i = 0, 2n$.

Shu x_i nuqtalardagi $f(x)$ funksiyaning qiymatlarini $y_i = f(x_i)$ bilan belgilaymiz. $f(x)$ funksiyaning grafigi 21-rasmda keltirilgan ko'rinishda bo'lsin.



21-rasm. Egri chiziq bo'laklarini parabolalar bilan almashtirish

Ma'lumki, A_i nuqtaning koordinatalari x_i, y_i bo'ladi. Shuningdek, A_i, A_{i+1}, A_{i+2} nuqtalardan yagona $y = ax^2 + bx + c$ parabola o'tadi. Bu yerdagi a, b, c noma'lumlarni x_i va y_i lar orqali topamiz. Parabolani $[x_i, x_{i+2}]$ kesmada integrallaymiz. Natijani berilgan $f(x)$ ning shu kesmadagi integrali sifatida qabul qilamiz. Ya'ni

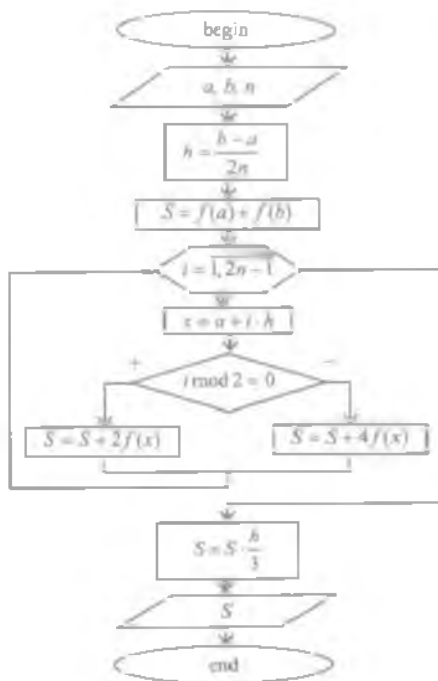
$$\int_{x_i}^{x_{i+2}} f(x) dx \cong \frac{h}{3} [f(x_i) + 4f(x_{i+1}) + f(x_{i+2})] \quad (4)$$

Topilgan formulada i ga $0, 2, 4, \dots, 2n-2$ qiymatlar berib, ularni qo'shib chiqsak, quyidagi formula hosil bo'ladi:

$$\int_a^b f(x) dx \cong \frac{h}{3} [f(a) + f(b) + 4(f(x_1) + f(x_3) + \dots + f(x_{2n-1})) + 2(f(x_2) + f(x_4) + \dots + f(x_{2n-2}))] \quad (5)$$

Bu formula *Simpson kvadratur formulasi* deyiladi.

Aniq integralni *Simpson formulasi* yordamida taqribiy hisoblash algoritmi va yuqoridagi $\int_{0.75}^{0.75} \frac{dx}{\sqrt{\cos x - 1}}$ integralni hisoblash dasturi quyida keltirilgan.



Dasturi. Dastur tuzishda yuqoridagi formadan foydalanamiz. Dastur protsedurasi quyidagicha bo'ladi:

```

procedure TForm1.Button1Click(Sender: TObject);
var S,a,b,h,x:real;
    n,i:integer;
function f(t:real):real;
begin
    f:=1/sqrt(abs(cos(t)-sqr(t)))
end;
begin
    a:=StrToFloat(Edit1.Text);
    b:=StrToFloat(Edit2.Text);
    n:=StrToInt(Edit3.Text);
    h:=(b-a)/(2*n);
    S:=f(a)+f(b);
    for i:=1 to 2*n-1 do
        begin
  
```

```

x:=a+i*h;
if i mod 2=0 then S:=S+2*f(x) else S:=S+4*f(x)
end;
S:=S*h/3;
Edit4.Text:=FloatToStr(S)
end;

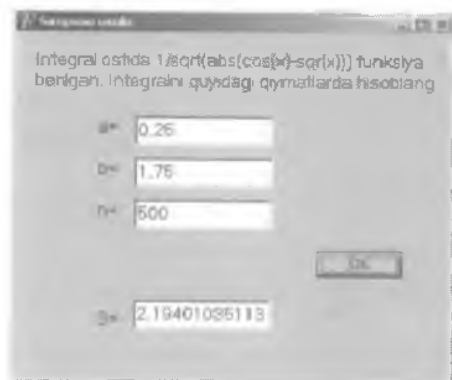
```

F9 tugmasini bosib natija olamiz (22-rasm).



22-rasm. Integralni *Simpson* formulasi yordamida hisoblash dasturi

Bu dasturlarda natijaning aniqligini oshirish uchun bo'laklashlar sonini yetarli darajada oshirish lozim. Masalan $n=500$ da natija 23-rasmdagidek bo'ladi:



23-rasm. Integralni *Simpson* formulasi yordamida hisoblash dasturi

Differensial tenglamalarni taqribiy yechish usullari

Fan-texnika sohasining turli tarmoqlaridagi ko'pgina masalalar differensial tenglamalarni yechishga keltiriladi. Differensial tenglamalarni taqribiy yechishning juda ko'p usullari mavjud. Ulardan eng sodda va hisoblashga qulay usullardan Eylar va Runge-Kutta usullarini ko'rib chiqamiz.

Eylar usuli

Bizga $y' = f(x, y)$ (1) tenglamaning $y(x_0) = y_0$ (2) shartni qanoatlantiradigan yechimini topish talab qilingan bo'lsin. $f(x, y)$ funksiya qaralayotgan oraliqda x bo'yicha uzluksiz bo'lib, y bo'yicha Lipshts shartini qanoatlantirsin:

$$|f(x, y_2) - f(x, y_1)| \leq K|y_2 - y_1|, K - \text{const.}$$

Bizdan (1), (2) masalaning $y(x_i)$, $x_i = x_0 + ih$ ($i = 1, 2, \dots$) taqribiy qiymatini qadami h bo'lgan bir o'lchovli to'rdada aniqlash talab qilingan bo'lsin. Buning uchun quyidagi ma'lum tenglikdan foydalaniladi:

$$y(x_{i+1}) - y(x_i) = \int_{x_i}^{x_{i+1}} f(x, y(x)) dx \quad (3)$$

Endi (3)-tenglikdagi integralni taqribiy ravishda chap to'g'ri to'rtburchaklar kvadratur formulasi bilan almashtiramiz va $y(x_i)$ ning qiymatini y_i bilan belgilab, quyidagi **Eylar formulasi**ni hosil qilamiz:

$$y_{i+1} = y_i + h \cdot f(x_i, y_i), \quad i = 0, 1, 2, \dots \quad (4)$$

Agar (1) va (2) masalaning x_i nuqtadagi aniq yechimini $y(x_i)$ bilan, uning taqribiy yechimini y_i bilan belgilasak, Eylar usulining xatoligi quyidagi tengsizlik orqali baholanadi:

$$|y(x_i) - y_i| \leq \frac{hN}{2L} [e^{L(x_i - x_0)} - 1] \quad (5)$$

bu yerda N, L lar doimiy sonlar.

Runge-Kutta usuli

Berilgan (1) va (2) masalani Runge-Kutta usuli bilan yechishda $y(x)$ funksiya Teylor qatoriga yoyiladi va qatordan hosila qatnashmaydigan qilib o'zgartirilgan qismi olinadi. Runge-Kutta usulida ixtiyoriy i -qadamdagi taqribiy hisoblash formulasi quyidagicha olinadi:

$$y_{i+1} = y_i + Z_i \quad (6)$$

Z_i larning tanlanishiga qarab Runge-Kutta usulining turli tartibdagi formulalari hosil qilinadi.

Runge-Kutta usulining 3-tartibli formulasi uchun

$$Z_i = \frac{1}{6}(k_1^{(i)} + 4k_2^{(i)} + k_3^{(i)}), \quad (7)$$

bu yerda

$$k_1^{(i)} = hf(x_i, y_i), \quad k_2^{(i)} = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1^{(i)}}{2}\right), \quad k_3^{(i)} = hf\left(x_i + h, y_i + k_2^{(i)} - k_1^{(i)}\right).$$

To'rtinchi tartibli Runge-Kutta formulasi quyidagicha bo'ladi:

$$Z_i = \frac{1}{6}(k_1^{(i)} + 2k_2^{(i)} + 2k_3^{(i)} + k_4^{(i)}), \quad (8)$$

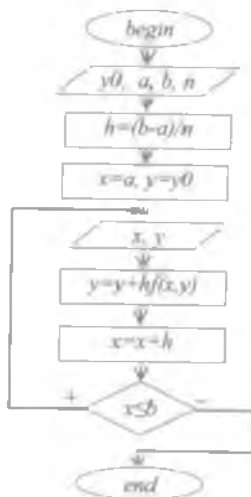
bu yerda

$$k_1^{(i)} = hf(x_i, y_i), \quad k_2^{(i)} = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_1^{(i)}}{2}\right), \\ k_3^{(i)} = hf\left(x_i + \frac{h}{2}, y_i + \frac{k_2^{(i)}}{2}\right), \quad k_4^{(i)} = hf(x_i + h, y_i + k_3^{(i)}).$$

Eyler va Runge-Kutta formulalari yordamida algoritm va dasturlar tuzish

Misol. $y' = 0,5y - x^2$, $y_0 = 1,5$, $x \in [2,3]$, $n = 10$ differensial tenglamani Eyler formulasi yordamida taqribiy yechish algoritmi va dasturi tuzilsin.

Algoritmi:



Dasturi:

Delphi dasturini ishga tushiramiz va yangi loyiha yaratib, formaga misol berilishini kiritish uchun 1 ta Label, kiritiladigan y_0, a, b, n lar uchun 4 ta Label va 4 ta Edit, chiqariladigan ma'lumotlar uchun 1 ta Label va 1 ta Memo hamda 1 ta Button komponentlarini joylashtiramiz. *Form1* ning *Caption* xossasi qiymat darchasiga "Eylar usuli" so'zini, *Label1* komponenti *Caption* xossasiga misol berilishini, *Label2*, *Label3*, *Label4*, *Label5*, *Label6* komponentlariga mos ravishda " $y_0=$ ", " $a=$ ", " $b=$ ", " $n=$ ", "Javob" so'zlarini kiritamiz. Edit larning *Text* xossalaridagi qiymatlarni o'chiramiz. *Memol*ning *Lines* xossasiga kirib, "Memol" so'zini o'chiramiz. *Button1*ning *Caption* xossasiga "OK" so'zini kiritamiz. Natijada forma 24-rasmdagi holatga keladi:



24-rasm. Differensial tenglamani Eylar usuli yordamida yechish dasturi loyihasi

Button1 tugmasi ustida sichqoncha chap tugmasini ikki marta bosib, hosil bo'lgan protseduraga quyidagi dasturni kiritamiz:

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
Var x,y,y0,a,b,h:real; n:integer;
```

```
Begin
```

```
  Memol.Clear;
```

```
  y0:=StrToFloat(Edit1.Text);
```

```
  a:=StrToFloat(Edit2.Text);
```

```
  b:=StrToFloat(Edit3.Text);
```

```
  n:=StrToInt(Edit4.Text);
```

```
  h:=(b-a)/n;
```

```
  x:=a; y:=y0;
```

```

repeat
  Memo1.Lines.Add('x='+FloatToStr(x)+' y='+FloatToStr(y));
  y:=y+b*(0.5*y-sqr(x));
  x:=x+h;
until x>b;
end;

```

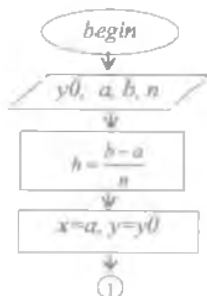
Dasturni kiritib bo'lgandan so'ng F9 tugmasini bosamiz. Hosil bo'lgan ishchi formada y_0, a, b, n larga qiymatlar kiritib natija olamiz (25-rasm).

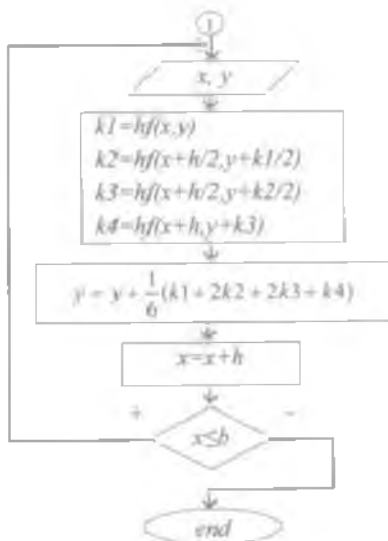


25-rasm. Differensial tenglamani *Euler* usuli yordamida yechish dasturi

Мисол. Yuqoridagi $y' = 0,5y - x^2$, $y_0 = 1,5$, $x \in [2,3]$, $n = 10$ differensial tenglama Runge-Kutta usuli yordamida taqribiy yechilsin.

Algoritmi:





Dasturi. Dastur tuzishda yuqoridagi formadan foydalanamiz. Dastur protsedurasi quyidagicha bo'ladi:

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
Var x,y,y0,a,b,h,k1,k2,k3,k4:real; n:integer;
```

```
function f(t1,t2:real):real;
```

```
begin
```

```
  f:=0.5*t2-sqr(t1)
```

```
end;
```

```
Begin
```

```
  Memo1.Clear;
```

```
  y0:=StrToFloat(Edit1.Text);
```

```
  a:=StrToFloat(Edit2.Text);
```

```
  b:=StrToFloat(Edit3.Text);
```

```
  n:=StrToInt(Edit4.Text);
```

```
  h:=(b-a)/n;
```

```
  x:=a; y:=y0;
```

```
  repeat
```

```
    Memo1.Lines.Add('x='+FloatToStr(x)+' y='+FloatToStr(y));
```

```
    k1:=h*f(x,y); k2:=h*f(x+h/2,y+k1/2);
```

```
    k3:=h*f(x+h/2,y+k2/2); k4:=h*f(x+h,y+k3);
```

```
    y:=y+(k1+2*k2+2*k3+k4)/6;
```

```
    x:=x+h
```

until x>b

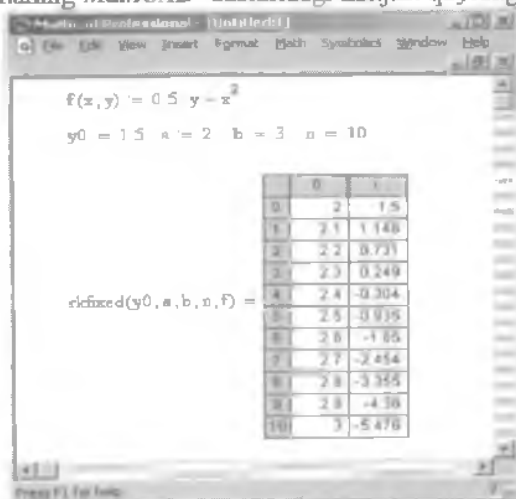
end;

F9 tugmasini bosamiz va dasturda kerakli qiymatlarni kiritamiz. Natija 26-rasmdagidek bo'ladi.



26-rasm. Differensial tenglamani Runge-Kutta usuli yordamida yechish dasturi

Bu tenglamaning MathCAD dasturidagi natijasi quyidagicha:



27-rasm. Differensial tenglamani MathCAD dasturida yechish

Axborot texnologiyalari

Hozirgi XXI asr – axborot asrida axborotlarning haddan tashqari ortib borishi “Axborot texnologiyalari” fanining uzluksiz va shiddat bilan rivojlanishini taqozo etadi. Bugungi kunda axborotlar shu qadar ko‘pki, ularni muntazam ravishda saralash, qayta ishlash, saqlash usullarini, vositalarini yaratish zamon talabi bo‘lib bormoqda.

“Axborot texnologiyalari” fani barcha turdagi axborotlarni foydalanuvchiga tushunarli tarzda yetkazish uchun qayta ishlash, formatlash, saralash, uzatish usullarini, texnologiyalarini o‘rganadi. Axborotlar bilan ishlash jaryonlariga axborot texnologiyalari maksimal darajada tadbiq etilganda ular yuqori sifat bosqichiga ko‘tariladi.

Hozirgi kunda axborot va kompyuter texnologiyalari kundalik turmushda eng ko‘p qo‘llaniladigan texnologiyaga aylanib bormoqda. Chunki hayotning qaysi sohasini olmaylik, qanday amallarni bajarmaylik, albatta axborotlar bilan ish ko‘ramiz. Ya‘ni ularni qabul qilamiz, qayta ishlaymiz va uzatamiz.

“Axborot texnologiyalari” terminidagi “texnologiya” so‘zi grekchadan olingan bo‘lib, “techné” – san‘at, mahorat; “logos” – fan, ta‘limot ma‘nolarini anglatadi. Texnologiya – usul va vositalar majmuasidir. Texnologiya qo‘llanilishi natijasida obyektning boshlang‘ich holati o‘zgarib, oldindan belgilangan, talabga javob beradigan holatga keladi.

Axborot texnologiyalari – axborotlarni yig‘ish, saqlash, qayta ishlash, qidirish, qabul qilish, uzatish usul va vositalaridir.

“Axborot texnologiyalari” vositalariga kompyuterning asosiy va qo‘shimcha qurilmalari, ularda mavjud bo‘lgan axborotlarni yaratuvchi, qayta ishlovchi hamda uzatuvchi **Microsoft Word, PowerPoint, Excel, Access, Internet Explorer** kabi barcha dasturlar misol bo‘la oladi.

Microsoft Word matn muharriri, dastur interfeysi

Word dasturi matn muharriri bo‘lib, matnli ma‘lumotlarni kiritish, formatlash, tahrirlash amallarini bajarishga mo‘ljallangan. Bundan tashqari jadval ko‘rinishidagi ma‘lumotlarni hosil qilish va tahrirlash, rasmlar va kartinalar joylashtirish, har xil matematik, statistik, iqtisodiy simvollar va formulalarni kiritish, turli ko‘rinishdagi figuralar, chiziqqlar yordamida blok-sxemalar chizish, sonli ma‘lumotlar asosida diagrammalar hosil qilish va boshqa shu kabi ko‘plab turdagi ishlarni bajarish mumkin.

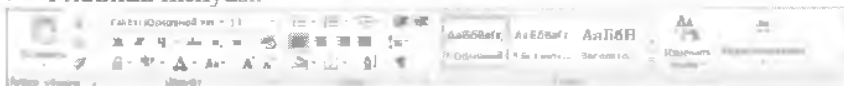
Word dasturi *Microsoft office* paketi tarkibidagi dastur bo‘lib, dasturlash texnologiyalarining rivojlanib borishi natijasida uning yangidan-yangi

- **menuyu satri.** Menyularda dasturda ishlatish mumkin bo'lgan buyruqlar keltirilgan. Dastur menyulari: **Файл, Главная, Вставка, Разметка страницы, Ссылки, Рассылки, Рецензирование, Вид;**
- **uskunalar paneli.** Menuyu buyruqlari panelda *piktogrammali buyruqlar* ko'rinishida joylashtirilgan bo'lib, birortasining tanlanishi mos buyruq bajarilishiga olib keladi. Bunda piktogramma ustiga sichqoncha ko'rsatkichi olib borilganda mos buyruq nomi paydo bo'ladi;
- **gorizontal va vertikal chizg'ichlar.** Bu chizg'ichlar sahifa parametrlarini o'zgartirish va ko'rib turish uchun mo'ljallangan;
- **ishchi soha.** Ish obyekti bo'lib, ma'lumotlar shu yerda yoziladi va tahrirlanadi;
- **o'tkazish paneli.** Matnning ekranga sig'magan qismlarini ko'rish va ularga o'tishni ta'minlaydi;
- **holatlar satri.** Bu yerda foydalanuvchiga kursor holati, hujjat haqida umumiy ma'lumotlar, ba'zi tugmalar holatlari haqida ma'lumotlar berib boriladi. Shuningdek, dasturning boshqa rejimiga o'tish, sahifa masshtabini o'zgartirish ishlarini bajarish mumkin.

Word dasturi menyulari

Menyularning ko'p ishlatiladigan buyruqlari quyidagilar:

- **Файл** menyusi buyruqlari:
 - *Сохранить* (Ctrl+S) – faylni xotirada saqlash;
 - *Сохранить как* (F12) – faylni boshqa nom va kengaytma bilan saqlash;
 - *Открыть* (Ctrl+O) – saqlab qo'yilgan faylni ochish;
 - *Заккрыть* (Ctrl+W) – hujjat oynasini yopish;
 - *Сведения* – joriy hujjat haqida to'liq ma'lumot olish;
 - *Последние* – Word dasturi bo'yicha ochilgan yoki foydalangan oxirgi hujjatlar hamda papkalar ro'yxatini ko'rish va ularni ochish;
 - *Создать* (Ctrl+N) – yangi hujjat yaratish;
 - *Печать* (Ctrl+P) – fayl sahifalarini chop qilish;
 - *Справка* (F1) – dastur va unda ishlash bo'yicha ma'lumotlar olish;
 - *Параметры* – dasturda ishlash parametrlarini o'zgartirish va uskunalar panelini sozlash;
 - *Выход* (Alt+F4) – dasturdan chiqish.
- **Главная** menyusi.



29-расм. Главная menyusi

- Boshqa menyular buyruqlari bo'limlarga ajratilgan.
- Menyuning **Буфер обмена** bo'limi buyruqlari:
 - *Вырезать* (Ctrl+X) – belgilangan ma'lumotlarni almashinuv buferiga ko'chirish;
 - *Копировать* (Ctrl+C) – belgilangan ma'lumotlarni almashinuv buferiga nusxalashi;
 - *Вставить* (Ctrl+V) – almashinuv buferidagi ma'lumotlarni kursor turgan joyga qo'yish;
 - *Формат по образцу* – kursor turgan joydagi ma'lumot formatini boshqa ma'lumot sohasida qo'llash;
- **Шрифт** bo'limi:
 - Bo'limning quyi o'ng burchagidagi tugma bosilganda **Шрифт** oynasi hosil bo'ladi (30-rasm).



30-rasm. **Шрифт** oynasi

Oynadan shrift turi, o'lchami kabi parametrlar tanlanib, OK tugmasi bosilganda belgilangan matn kerakli shriftga o'tadi. Ammo quyidagi buyruqlarni bevosita bo'limning o'zi orqali bajarish qulay:

- *Шрифт* – shrift turini o'zgartirish;
- *Размер шрифта* – shrift o'lchamini o'zgartirish;
- *Полужирный* (Ctrl+B) – belgilangan matnni qalin shriftga aylantirish;
- *Курсив* (Ctrl+I) – belgilangan matnni qiya ko'rinishga o'tkazish;
- *Подчеркнутый* (Ctrl+U) – tagiga chizilgan matnni aniqlaydi;

- *Подстрочный знак* (Ctrl+ “=”) – indeksga yozishni ta’minlaydi. Bu holda yana shu buyruq bosilganda odatiy yozish holatiga o’tadi;
- *Надстрочный знак* (Ctrl+Shift+“=”) – darajaga yozishni ta’minlaydi. Bunda ham odatiy yozish rejimiga o’tish uchun piktogramma qayta bosiladi;
- **Абзац bo’limi buyruqlari:**



31-rasm. Главная menyusining Абзац bo’limi

- *Маркеры* – abzatsga marker qo’yish;
- *Нумерация* – abzatslarni raqamlash;
- *Сортировка* – abzatslarni saralash;
- *Выравнивание ...* – joriy abzatsni kerakli tomonlar bo’yicha tekislash;
- *Интервал* – abzats qatorlari orasidagi masofani o’zgartirish;
- *Границы* – abzatsga chegara qo’yish.
- **Редактирование bo’limi buyruqlari:**
- *Найти* (Ctrl+F) – hujjatdagi ma’lumotlarni qidirish;
- *Заменить* (Ctrl+H) – hujjatdagi bir xil ma’lumotlarni avtomatik tarzda boshqasiga almashtirish;
- *Перейти* (Ctrl+G) – hujjatning kerakli sahifalariga yoki bo’limlariga avtomatik o’tish.

➤ **Вставка menyusi:**



32-rasm. Вставка menyusi

- **Страницы bo’limi buyruqlari:**
- *Титульная страница* – faylga titul varag’i o’rnatish;
- *Разрыв страницы* – kursordan keyingi matnning doimo yangi qatordan yozilishini ta’minlash;
- **Таблицы bo’limi:**
- Bo’limdagi **Таблицы** buyrug’i bosilganda jadval tuzishning quyidagi usullari taklif qilinadi:
- *Вставка таблицы* – katakchalarni belgilash usuli bilan jadval yaratish;
- *Вставить таблицу* – ustun va satrlar sonini ko’rsatish usuli bilan jadval yaratish;

- *Нарисовать таблицу* – jadvalni mustaqil qo'lda chizish;
 - *Таблица Excel* – sahifada Excel jadvalini hosil qilish;
 - *Экспресс таблицы* – tayyor dizaynli jadvallarni o'rnatish.
- Hosil qilingan jadval tanlanganda uni tahrirlash uchun **Работа с таблицами** sarlavhali **Конструктор** va **Макет** nomli maxsus menyular paydo bo'ladi.

- **Конструктор** menyusi buyruqlari:
 - *Стили таблиц* – jadvalni turli dizayndagi ko'rinishlarda tasvirlash;
 - *Заливка* – joriy katak yoki belgilangan jadvalga rang berish;
 - *Границы* – joriy katak yoki belgilangan jadvalning chegaralarini passiv holatga o'tkazish;
 - *Ластик* – katak chegarasini passiv holatga o'tkazish;
- **Макет** menyusi:

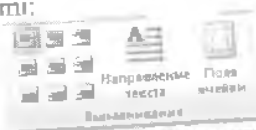


33-rasm. Макет menyusi

- **Строки и столбцы** bo'limi:

Bo'limning **Удалить** buyrug'i bosilganda quyidagi menyular hosil bo'ladi:

 - *Удалить ячейки* – joriy katak yoki belgilangan kataklarni o'chirish;
 - *Удалить столбцы* – joriy ustun yoki belgilangan ustunlarni o'chirish;
 - *Удалить строки* – joriy qator yoki belgilangan qatorlarni o'chirish;
 - *Удалить таблицу* – joriy jadvalni o'chirish;
 - *Вставить сверху* – joriy qatordan oldin yangi qator hosil qilish;
 - *Вставить снизу* – joriy qatordan keyin yangi qator qo'shish;
 - *Вставить слева* – joriy ustundan oldin yangi ustun hosil qilish;
 - *Вставить справа* – joriy ustundan keyin yangi ustun qo'shish.
- **Объединение** bo'limi:
 - *Объединить ячейки* – belgilangan kataklarni birlashtirish;
 - *Разделить ячейки* – katakni satr va ustunlarga bo'lish;
 - *Разделить таблицу* – jadvalni bo'lish.
- **Выравнивание** bo'limi:



34-rasm. Макет menyusining Выравнивание bo'limi

Bo'limning chap tomonida matnni katak bo'ylab yuqori-chapga, yuqori-o'rtaga kabi turli tekislash buyruqlari keltirilgan:

- *Направление текста* – matni vertikal yoki gorizontal ko‘rinishga o‘tkazish;
- *Поля ячейки* – katak parametrlarini o‘rnatish;
- **Данные** bo‘limi buyruqlari:
- *Сортировка* – belgilangan qatorlarni matn yoki sonli ma‘lumotga ko‘ra saralash;
- *Преобразовать в текст* – jadvalni matn ko‘rinishiga o‘tkazish;
- *Формула* – jadvalda avtomat tarzda hisoblashlarni qo‘llash.
- **Вставка** menyusining **Иллюстрации** bo‘limi buyruqlari:
- *Рисунок* – sahifaga fayldan rasm qo‘yish;
- *Картинка* – sahifaga Office bazasidan kartina qo‘yish;
- *Фигуры* – blok-sxemalar, turli geometrik figuralar, strelkalar qo‘yish;
- *SmartArt* – turli uslubdagi grafik va shakllardan tashkil topgan obyektlar bo‘lib, ma‘lumotlarni tushunish va tasavvur qilishda katta yordam beradi;
- *Диаграмма* – sonli ma‘lumotlar asosida diagrammalar tashkil qilish;
- *Снимок* – ochiq holdagi oynalar yoki ekranning ixtiyoriy qismini rasimga tushirish va uni sahifaga joylashtirish.

Yuqorida qo‘yilgan obyektlar tanlanganda ularni turli ko‘rinishlarda o‘ta qulay tarzda tahrirlashga imkon beruvchi maxsus menyular paydo bo‘ladi.

- **Ссылки** bo‘limining **Гиперссылка** buyrug‘i belgilangan obyektga gipermurojaat o‘rnatadi.
- **Колонтитул** bo‘limi buyruqlari:
 - *Верхний колонтитул* – sahifaning yuqori qismiga kolontitul qo‘yish;
 - *Нижний колонтитул* – sahifaning quyi qismiga kolontitul qo‘yish;
 - *Номер страницы* – sahifalarni raqamlash.
- **Текст** bo‘limi buyruqlari:
 - *Подпись* – matnli ma‘lumotlarni obyekt sifatida qo‘yish;
 - *WordArt* – belgilangan matni WordArt stiliga o‘tkazish;
 - *Объект* – turli dasturlar obyektlarini joylashtirish.
- **Символы** bo‘limi buyruqlari:
 - *Формула* – formulalar yozish va ularni tahrirlash;
 - *Символ* – kerakli simvollarni qo‘yish.
- **Разметка страницы** menyusini:
 - **Параметры страницы** bo‘limi buyruqlari:
 - *Поля* – ma‘lumotlar chegaralarini o‘zgartirish;
 - *Ориентация* – sahifalarni kitob yoki albom shakliga o‘tkazish;
 - *Размер* – qog‘oz o‘lchamini o‘zgartirish;
 - *Колонки* – ma‘lumotlarni ustunlarga ajratish.

- **Фон страницы** bo'limi buyruqlari:
 - *Цвет страницы* – sahifaga fon berish;
 - *Границы страницы* – sahifaga chegara qo'yish.
- **Абзац** bo'limi buyruqlari:
 - *Отступ* – abzatsga chap va o'ng tomondan joy tashlash;
 - *Интервал* – abzatsdan oldingi va keyingi oraliqni aniqlash.
- **Упорядочить** bo'limi buyruqlari:
 - *Положение* – belgilangan obyektning salifada joylashishini aniqlaydi;
 - *Обтекание текста* – ma'lumotlarning belgilangan obyektga nisbatan joylashishini aniqlaydi;
 - *Группировать* – belgilangan obyektlarni birlashtirish;
 - *Повернуть* – obyektни burish.
- **Вид менюси** buyruqlari:
 - *Разметка страницы* – sahifaning pechatga moslangan ko'rinishi bo'lib, odatda shu rejimda ishlanadi;
 - *Линейка* – ekranga chizg'ichni qo'yish yoki olib tashlash;
 - *Масштаб* – sahifa masshtabini o'zgartirish;
 - *Новое окно* – joriy hujjat aks etgan yangi oyna ochish;
 - *Разделить* – oynani bo'lish;
 - *Снять разделение* – bo'lishni bekor qilish.

Shuningdek, quyidagi tugmalar yoki tugmalar kombinatsiyalaridan doimiy ravishda foydalaniladi:

- Home – kursorni qator boshiga o'tkazish;
- End – kursorni qator oxiriga o'tkazish;
- Page Up – sahifa bo'ylab oldingi ekranga o'tish;
- Page Down – sahifa bo'ylab keyingi ekranga o'tish;
- Ctrl+Home – kursorni hujjatning birinchi qatoriga o'tkazish;
- Ctrl+End – kursorni hujjatning oxirgi qatoriga o'tkazish;
- Shift+F5 – kursorni oldingi pozitsiyasiga o'tkazish;
- Delete – belgilangan qismni o'chirish;
- Shift+F3 – belgilangan matn registrini o'zgartirish;
- Ctrl+A – hujjatdagi barcha ma'lumotni belgilash;
- Esc – bajarilayotgan jarayonni bekor qilish;
- Ctrl+Z – buyruqni bekor qilish;
- Ctrl+Y – bekor qilingan buyruqni tiklash;

PowerPoint dasturi, slaydlar va taqdimotlar

Diplom ishlari, dissertatsiya himoyalari, korxonalar, tashkilot, ta'lim muassasalari taqdimotlari va boshqa ko'rsatmalar prezentatsiya dasturlari yordamida namoyish etiladi. Bundan tashqari televideniya namoyish qilinadigan turli reklama roliklari, ko'rsatuvlarda ham taqdimot dasturlari animatsiya effektlaridan foydalaniladi.

PowerPoint taqdimot yaratishga mo'ljallangan dastur bo'lib, Microsoft Office paketi tarkibidagi dastur hisoblanadi.

PowerPointda yaratilgan fayl *pptx* kengaytmali fayl hisoblanadi.

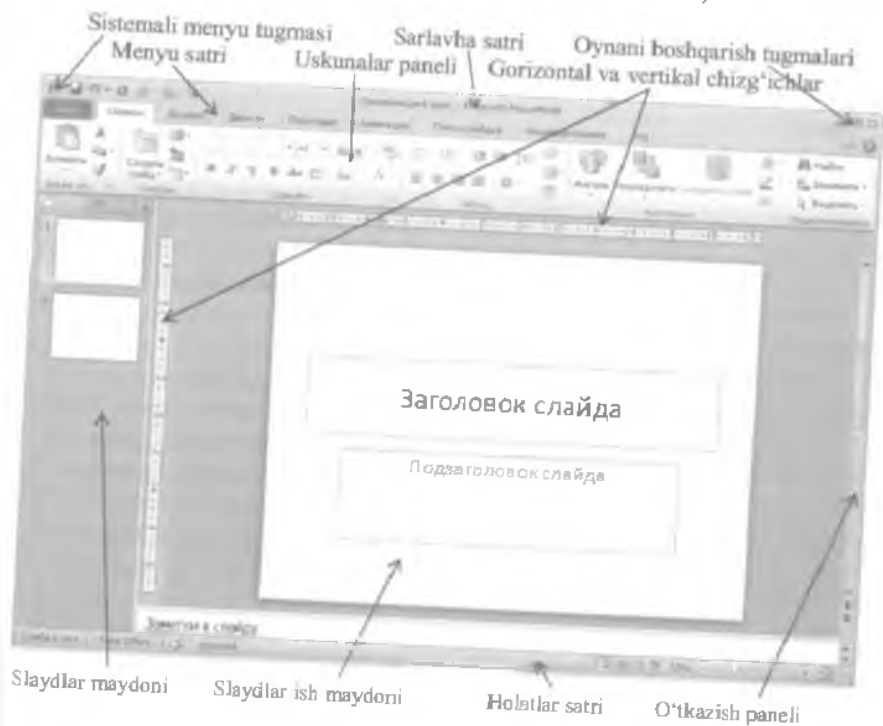
Taqdimot (Prezentatsiya) – bu slaydlar ketma-ketligidir.

Slayd – bu o'zida matn, grafik, rasm, jadval, ovoz va boshqa animatsiya samaralarini mujassam etgan taqdimot varag'idir.

PowerPoint dasturi odatda **Пуск-Бee программы Microsoft Office**

Microsoft PowerPoint menyular ketma-ketligi orqali ishga tushiriladi.

Natijada **PowerPoint** dasturi oynasi hosil bo'ladi (35-rasın).



35-rasm. PowerPoint dasturi oynasi

- Dastur oynasi elementlari quyidagilar:
- **sarlavha satri.** Bu satrda sistemali menyu tugmasi, fayl va dastur nomi hamda oynani boshqarish tugmalari o'rin olgan.
 - **menyu satri.** Sarlavha satridan keyin menyu qatori joylashib, ularda dasturda foydalanish mumkin bo'lgan buyruqlar keltirilgan. PowerPoint dasturi menyulari quyidagilar: *Файл, Главная, Вставка, Дизайн, Переходы, Анимация, Показ слайдов, Рецензирование, Вид.*
 - **uskunalar paneli.** Menyu buyruqlari panelda piktogrammalı buyruqlar ko'rinishida joylashtirilgan bo'lib, birortasining tanlanishi mos buyruq bajarilishiga olib keladi;
 - **gorizontal va vertikal chizg'ichlar.** Bu chizg'ichlar sahifa parametrlarini o'zgartirish va ko'rib turish uchun mo'ljallangan;
 - **slaydlar maydoni.** Slaydlar ro'yxati joylashgan bo'lib, slaydlar ustidagi umumiy amallar, boshqa slaydlarga o'tish ishlari bajariladi;
 - **slayd ish maydoni.** Ish obykti bo'lib, slayd ma'lumotlari, elementlari shu yerda hosil qilinadi va tahrirlanadi;
 - **o'tkazish paneli.** Matnning ekranga sig'magan qismlarini ko'rish va ularga o'tishni ta'minlaydi;
 - **Holatlar satri.** Bu yerda foydalanuvchiga kursor holati, taqdimot va slaydlar haqida umumiy ma'lumotlar, ba'zi tugmalar holatlari haqida ma'lumotlar berib boriladi. Shuningdek, dasturning boshqa rejimiga o'tish, sahifa masshtabini o'zgartirish ishlarini bajarish mumkin.

PowerPoint dasturida taqdimot yaratish

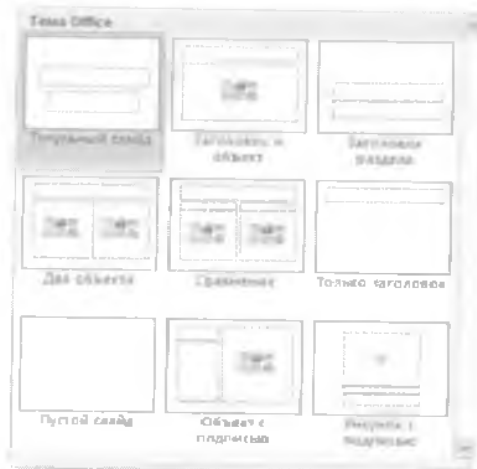
PowerPoint dasturida taqdimot yaratish jarayoni shartli ravishda ikki bosqichga bo'linadi:

1. Taqdimotni loyihalashtirish va ma'lumotlarni kiritish;
2. Taqdimot va slaydlar animatsiya samaralarini hosil qilish.

Taqdimot kamida bitta slayddan iborat bo'ladi. Shuning uchun dastur ishga tushirilganda bitta bo'sh slayddan iborat **Презентация1** nomli taqdimot hosil bo'ladi. Hosil bo'lgan slayd odatda titul slaydi maketi ko'rinishida bo'ladi. Slayd maketini o'zgartirish uchun **Главная** menyusining **Слайды** bo'limidagi **Макет** buyrug'i bosiladi. Natijada bir qancha maket ko'rinishlari hosil bo'ladi (36-rasm).

Ular quyidagilar:

- **Титульный слайд.** *Titul slaydi* maketi slayd sarlavhasi va kichik sarlavha elementidan tashkil topadi.



36-расм. Слайдлар макетлари namunalari

- **Заголовок и объект.** *Sarlavha va obyekt* maketi slayd sarlavhasi hamda jadval, diagramma, SmartArt, rasm, kartina, multimediyani ma'lumotlaridan birini joylashtirishga imkon beradi.
- **Заголовок раздела.** *Bo'lim sarlavhasi* maketi orqali kichik va katta sarlavha qo'yish mumkin.
- **Два объекта.** *2 ta obyekt* maketi slayd sarlavhasi hamda 2 ta matn yoki obyektlar joylashtirishga mo'ljallangan.
- **Сравнение.** *Solishtirish* maketida slayd sarlavhasi hamda 2 ta matn yoki obyekt va ularning har biriga sarlavha qo'yish imkoniyati mavjud.
- **Только заголовок.** *Faqat sarlavha* maketida slayd sarlavhasinigina yozish mumkin.
- **Пустой слайд.** *Bo'sh slayd* maketida slaydga bevosita ma'lumot kiritish imkoniyati mavjud emas.
- **Объект с подписью.** *Izohli obyekt* maketi matn yoki obyekt va unga sarlavhali izoh yozishga mo'ljallangan.
- **Рисунок с подписью.** *Izohli rasm* maketida rasm va unga sarlavhali izoh yozish imkoniyati mavjud.

Yuqoridagi maketlarning har birida maketlarning ixtiyoriy elementini o'zgartirish, o'chirib tashlash, boshqa obyektlar joylashtirish imkoniyatlari mavjud.

Ixtiyoriy maketli slaydga ma'lumot kirish uchun kerakli shablon ustida sichqonchani chap tugmasi bir marta bosiladi. Natijada matn kiritish uchun kursor yoki obyekt joylashtirish uchun mos ravishdagi oyna paydo bo'ladi.

PowerPoint dasturida bajariladigan asosiy buyruqlar va amallar

➤ **Файл** menyusi buyruqlari:

- *Сохранить* (Ctrl+S) – faylni xotirada saqlash;
- *Сохранить как* (F12) – faylni boshqa nom va kengaytma bilan saqlash;
- *Открыть* (Ctrl+O) – saqlab qo'yilgan faylni ochish;
- *Закрыть* (Ctrl+W) – taqdimot oynasini yopish;
- *Сведения* – joriy taqdimot haqida to'liq ma'lumot olish;
- *Последние* – PowerPoint dasturi bo'yicha ochilgan yoki foydalanilgan oxirgi taqdimotlar hamda papkalar ro'yxatini ko'rish va ularni ochish;
- *Создать* (Ctrl+N) – yangi taqdimot yaratish;
- *Печать* (Ctrl+P) – slaydlarni chop qilish;
- *Справка* (F1) – dastur va unda ishlash bo'yicha ma'lumotlar olish;
- *Параметры* – dasturda ishlash parametrlarini o'zgartirish va uskunalar panelini sozlash;
- *Выход* (Alt+F4) – dasturdan chiqish.

➤ **Главная** menyusi.

■ Menyuning **Буфер обмена** bo'limi buyruqlari:

- *Вырезать* (Ctrl+X) – belgilangan slaydni yoki ma'lumotlarni almashinuv buferiga ko'chirish;
- *Копировать* (Ctrl+C) – belgilangan slaydni yoki ma'lumotlarni almashinuv buferiga nusxalash;
- *Вставить* (Ctrl+V) – almashinuv buferidagi slaydni yoki ma'lumotlarni kursor turgan joyga qo'yish;
- *Формат по образцу* – belgilangan slayd yoki kursor turgan joydagi ma'lumot formatini mos ravishda boshqa slayd yoki ma'lumot sohasiga qo'llash;

■ **Слайды** bo'limi buyruqlari:

- *Макет* – joriy slayd maketini o'zgartirish;
- *Создать слайд* – slayd yaratish. Bu buyruq yuqori va pastki qismlardan iborat (37-rasm).



37-расм. Главная menyusining Слайды bo'limi

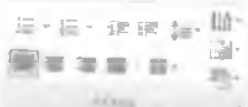
Buyruqning yuqori qismi bosilganda joriy slayddan keyin joriy slayd dizaynidagi yangi slayd hosil bo'ladi.

Buyruqning pastki qismi bosilganda hosil bo'lgan maketlar

namunalardan birini tanlash yordamida kerakli maketdagi slaydni yaratish mumkin.

Buyruqning pastki qismi bosilganda shuningdek, quyidagi buyruqlar ham hosil bo'ladi:

- *Дублировать выделенные слайды* – belgilangan slaydlarni nushtalash;
- *Слайды из структуры* – faylda saqlangan boshqa struktura asosida slayd yaratish;
- *Повторное использование слайдов* – boshqa taqdimotlar slaydlaridan foydalanish.
- **Шрифт** bo'limi. Bo'lim orqali shrift turi, o'lchami, rangini o'zgartirish, shriftni qalin yoki kursiv ko'rinishga o'tkazish kabi ishlarni bajarish mumkin.
- **Абзац** bo'limi buyruqlari:



38-rasm. Главная menyusining Абзац bo'limi

- *Маркеры* – abzatsga marker qo'yish;
- *Нумерация* – abzatslarni raqamlash;
- *Интервал* – abzats qatorlari orasidagi masofani o'zgartirish;
- *Выравнивание ...* – joriy abzatsni kerakli tomonlar bo'yicha tekislash;
- *Направление текста* – matn yo'nalishini o'zgartirish;
- *Выровнять текст* – matn shablon bo'ylab vertikal tekislash;
- *SmartArt* – matnli ma'lumotlarni vizual ko'rinishida tasvirlash.
- **Рисование** bo'limi buyruqlari:
 - *Упорядочить* – obyektlarni tartiblash amallarini bajaradi;
 - *Экспресс стили* – matnli yoki ixtiyoriy obyektning "ekspress" stiliga o'tkazish;
 - *Заливка фигуры* – obyekt fonini o'zgartirish;
 - *Контур фигуры* – obyekt chegaralarini o'zgartiradi;
 - *Эффекты фигур* – obyektga effekt qo'llash.
- **Редактирование** bo'limi buyruqlari:
 - *Найти (Ctrl+F)* – hujjatdagi ma'lumotlarni qidirish;
 - *Заменить (Ctrl+H)* – hujjatdagi bir xil ma'lumotlarni avtomatik tarzda boshqasiga almashtirish;

• **Вставка** menyusini:



39-rasm. Вставка menyusini

- **Таблицы bo'limi:**
Bo'limdagi **Таблицы** buyrug'i bosilganda jadval tuzishning quyidagi usullari taklif qilinadi:
 - *Вставка таблицы* – katakchalarni belgilash usuli bilan jadval yaratish;
 - *Вставить таблицу* – ustun va satrlar sonini ko'rsatish usuli bilan jadval yaratish;
 - *Нарисовать таблицу* – jadvalni mustaqil qo'lda chizish;
 - *Таблица Excel* – slaydda *Excel* jadvalini hosil qilish;
 Hosil qilingan jadval tanlanganda avtomatik paydo bo'ladigan **Работа с таблицами** sarlavhali **Конструктор** va **Макет** nomli maxsus menyular yordamida jadvalni ixtiyoriy ko'rinishda tahrirlash mumkin.
- **Изображение bo'limi buyruqlari:**
 - *Рисунок* – slaydga fayldan rasm qo'yish;
 - *Картинка* – slaydga *Office* bazasidan kartina qo'yish;
 - *Снимок* – ochiq holdagi oynalar yoki ekranning ixtiyoriy qismini rasimga tushirish va uni slaydga joylashtirish;
 - *Фотоальбом* – fayldan o'qilgan rasmlar asosida fotoalbom tayyorlash.
- **Иллюстрации bo'limi buyruqlari:**
 - *Фигуры* – blok-sxemalar, turli geometrik figuralar, strelkalar qo'yish;
 - *SmartArt* – slaydga *SmartArt* obyektini joylashtirish;
 - *Диаграмма* – sonli ma'lumotlar asosida diagrammalar tashkil qilish;
- **Ссылки bo'limining Гиперссылка va Действие buyruqlari** belgilangan obyektga gipermurojaat o'rnatadi.
- **Текст bo'limi buyruqlari:**
 - *Надпись* – matnli ma'lumotlarni obyekt sifatida qo'yish;
 - *Колонтитулы* – slaydga kolontitul qo'yish;
 - *WordArt* – belgilangan matnni *WordArt* stiliga o'tkazish;
 - *Номер слайда* – slaydlarni raqamlash;
 - *Объект* – turli dasturlar obyektlarini joylashtirish.
- **Символы bo'limi buyruqlari:**
 - *Формула* – formulalar yozish va ularni tahrirlash;
 - *Символ* – kerakli simvollarni qo'yish.
- **Мультимедиа bo'limi buyruqlari:**
 - *Видео* – slaydga videoklip o'rnatish;
 - *Звук* – slaydga audioklip o'rnatish.
 Yuqorida qo'yilgan obyektlar tanlanganda ularni turli ko'rinishlarda qulay tarzda tahrirlashga imkon beruvchi maxsus menyular paydo bo'ladi.
- **Дизайн menyusi:**
 - **Параметры страницы bo'limi buyruqlari:**

- *Параметры страницы* – sahifa parametrlarini o'zgartirish;
- *Ориентация слайда* – sahifalarni kitob yoki albom shakliga o'tkazish.
- **Темы** bo'limi. Bu yerda turli uslubdagi tayyor dizaynli slayd namunalari mavjud. Slaydlar maydonida kerakli slaydlar belgilanib, bo'limdagi hiror namuna sichqoncha yordamida tanlanganda belgilangan slaydlar tanlangan dizayn turiga o'tadi. Bo'limdagi **Цвета, Шрифты** va **Эффекты** buyruqlari yordamida dizayn foni va shrift ranglari, effekti kabi parametrlarni o'zgartirish mumkin.
- **Переходы** menyusi:
 - **Просмотр** bo'limining **Просмотр** buyrug'i yordamida tanlangan o'tish parametrlari belgilangan slaydlarga qanday qo'llanilayotganligini ko'rish mumkin.
 - **Переход к этому слайду** bo'limida slayd hosil bo'lishining turli usullari keltirilgan. Ulardan birortasini joriy slaydga qo'llash uchun kerakli usulni tanlash kifoya. Bo'limdagi **Параметры эффектов** buyrug'i yordamida tanlangan usulning yo'nalish uslubini o'zgartirish mumkin.
 - **Время показа слайдов** bo'limi buyruqlari:
 - *Звук* – taqdimotga ovoz o'rnatish. Buning uchun **Звук** darchasini ochib, *Access* bazasidagi standart ovozlardan biri yoki *Другой звук* buyrug'i yordamida ochilgan oynadan ixtiyoriy wav kengaytmali fayl tanlanadi. Darchadagi **Непрерывно** buyrug'i bosilsa, tanlangan ovoz taqdimot namoyishi tugaguncha qaytadan yangrayveradi;
 - *Длительность* – joriy slayd hosil bo'lish vaqtini bildiradi;
 - *Смена слайда* – keyingi slaydga o'tishni aniqlaydi. Buyruqning **По щелчку** bandi tanlansa, keyingi slayd kerakli tugma bosilgandagina hosil bo'ladi. Buyruqning **После** bandi avtomatik o'tishni bildiradi. Bunda joriy slayd namoyish vaqtini ko'rsatish mumkin;
- Bo'limdagi **Применить ко всем** buyrug'i bosilsa, menyudagi tanlangan parametrlar barcha slaydlar uchun qo'llaniladi.
- **Анимация** menyusi:
 - **Просмотр** bo'limining **Просмотр** buyrug'i yordamida tanlangan animatsiya effektlari joriy slaydga qanday qo'llanilayotganligini ko'rish mumkin;
 - **Анимация** bo'limi oynasidan tanlangan slayd elementi uchun kirish, belgilanish va chiqish effektlaridan biri o'rnatiladi. **Параметры эффектов** buyrug'i orqali tanlangan animatsiya uslubini o'zgartirish mumkin. Bo'limning quyi o'ng burchagidagi tugmani bosish yordamida hosil qilingan oynadan joriy animatsiya parametrlari o'zgartiriladi.
 - **Расширенная анимация** bo'limi buyruqlari:
 - *Добавить анимацию* – elementga qo'shimcha animatsiya o'rnatish;

- *Область анимации* – joriy slayd uchun qo'llanilgan animatsiyalar ro'xhati oynasini hosil qilish yoki olib tashlash;
- *Анимация по образцу* – tanlangan element animatsiyalarini boshqa elementga qo'llash.
- **Время показа слайдов** bo'limi buyruqlari:
 - *Начало* – animatsiya boshlanishini bildiradi. Buyruq parametrlari:
 - ♦ **По шелчку** – animatsiya tugma bosilganda ishga tushadi;
 - ♦ **С предыдущим** – oldingi animatsiya bilan birgalikda ishga tushadi;
 - ♦ **После предыдущего** – oldingi animatsiyadan keyin ishga tushadi;
 - *Длительность* – animatsiya vaqti uzunligini aniqlaydi;
 - *Задержка* – animatsiyadan keyingi pauza vaqtini aniqlaydi;
 - *Изменить порядок анимации* – animatsiyalar tartibini o'zgartirish;
- **Показ слайдов менюси:**
 - **Начать показ слайдов** bo'limi buyruqlari:
 - *С начала* (F5) – taqdimotni namoyish etish;
 - *С текущего слайда* – taqdimotni joriy slayddan boshlab namoyish etish;
 - *Произвольный показ* – qismaniy taqdimot yaratish;
 - **Настройка** bo'limi buyruqlari:
 - *Настройка демонстрации* – taqdimot parametrlarini sozlash;
 - *Скрыть слайд* – slaydni vaqtinchalik namoyishdan olib tashlash;
 - *Настройка времени* – taqdimot namoyishini vaqtga moslashtirish;
 - *Запись показа слайдов* – taqdimotga vaqt bo'yicha ovoz yozish.
- **Вид менюси buyruqlari:**
 - *Обычный* – taqdimot yaratish va uni tahrirlashga mo'ljallangan rejim bo'lib, odatda shu rejimda ishlanadi;
 - *Сортировщик слайдов* – slaydlarni saralash rejimini o'rnatish;
 - *Линейка* – ekranga lineykani qo'yish yoki olib tashlash;
 - *Масштаб* – slayd masshtabini o'zgartirish;
 - *Новое окно* – joriy taqdimot aks etgan yangi oyna ochish.

Excel elektron jadvali, Excel interfeysi

Jadval ko'rinishdagi ma'lumotlarni tahlil qilish, hisob-kitob ishlarini olib borish uchun maxsus amaliy dasturlar yaratilgan. Bunday dasturlar **elektron jadvallar** deb ataladi. **Excel** dasturi ham elektron jadvallar qatoriga kiradi.

Elektron jadvallar, xususan **Excel** dasturi bank, soliq sohalarida juda keng qo'llaniladi. Shu bilan birga barcha tashkilot, korxonalar va o'quv muassasalarining buxgalteriya bo'limlarida eng ko'p ishlatiladigan dastur

hisoblanadi. Bundan tashqari har bir kishi ish jarayonida Excel dasturiga murojaat qilishiga to'g'ri keladi.

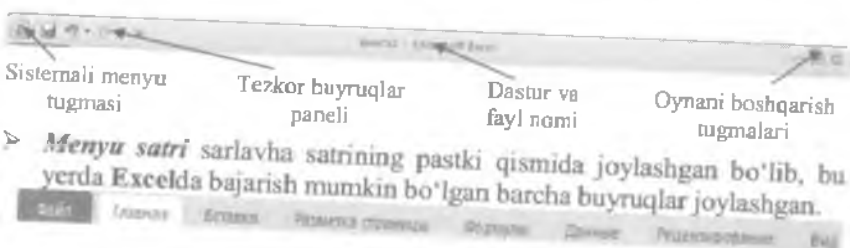
Excel dasturi Microsoft Office paketi tarkibidagi dastur bo'lib, **Русская-Программа-Microsoft Office Excel** menyular ketma-ketligi orqali ishga tushiriladi. Natijada Excel oynasi hosil bo'ladi (40-rasm):



40-rasm. Excel elektron jadvali interfeysi

Excel dasturi oynasi elementlari quyidagilar:

➤ **Sarlavha satri.** Bu yerda sistemali menyu tugmasi, tezkor buyruqlar paneli, dastur va fayl nomi, hamda oynani boshqarish tugmalari o'rin olgan.



➤ **Menyu satri** sarlavha satrining pastki qismida joylashgan bo'lib, bu yerda Excelda bajarish mumkin bo'lgan barcha buyruqlar joylashgan.

- **Uskunalar paneli.** Menyú satridan keyin uskunalar paneli joylashgan. Excel menyú buyruqlari piktogrammalar (tugmalar) ko'rinishida tasvirlangan.



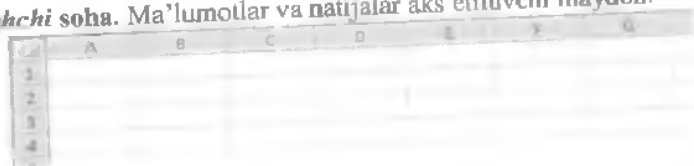
- **Nomlar maydoni.** Nomlar maydoni uskunalar panelidan keyingi qatorning chap tomonida joylashgan. Bu maydonda faol yacheyka nomi (manzili) ko'rinib turadi.



- **Formulalar satri** faol katakdagi qiymatning formulasini ko'rish yoki joriy katakka ma'lumot va formulalar kiritish uchun mo'ljallangan.



- **Ishchi soha.** Ma'lumotlar va natijalar aks etiluvchi maydon.



- **Ishchi varaqlar paneli.** Panel orqali boshqa ishchi varaqqa o'tish va yangi ishchi varaq hosil qilish mumkin.



- **Holatlar satri.** Bu satrda bajarilayotgan buyruq va ba'zi tugmalar holati to'g'risida ma'lumotlar berib boriladi. Shuningdek, dasturning boshqa rejimiga o'tish, sahifa mashtabini o'zgartirish ishlarini bajarish mumkin.



Excelda tayyorlangan fayl **xlsx** kengaytmali bo'lib, bu fayl **ishchi kitob (Книга)** deb yuritiladi. Ishchi kitob bir nechta **ishchi varaq (Лист)** lardan tashkil topadi. Ishchi varaq 16384 ta ustun 1048576 ta qatorlar (Excel 2010 versiyasiga ko'ra) kesishmasidan iborat bo'lgan mayda **kataklardan** (yacheykalardan) iborat. Ustunlar lotin alifbosining bosh harflari, qatorlar esa natural sonlar bilan nomlangan. Har bir katakning ham o'z nomi (**adres**) mavjud. Masalan, **B** ustun va **6**-qator kesishgan yacheyka **B6** deb

nomlanadi. Faol katakning nomini nomlar maydoniga qarab ham aniqlash mumkin. Har bir katak son, matn, formula va boshqa ma'lumotlar aks etishi uchun mo'ljallangan.

Katakdan so'ng maydon tushunchasi kiritiladi. Maydon **Exceld**a kataklar guruhidagi ma'lumotlarga funksiyalarni qo'llashda ishlatiladi.

Maydon deb, yonma-yon yoki ketma-ket joylashgan kataklar guruhiga aytiladi. Maydonning adresini ko'rsatish uchun uning yuqori chap va pastki o'ng kataklarining nomi ikki nuqta bilan ajratib yoziladi.

Maydonga misollar:

A1:B1 — **A1** va **B1** kataklardan iborat maydon;

B1:B100 — **B** ustundagi birinchi 100 ta katak;

B2:E6 — **B** dan **E** gacha bo'lgan 4 ta ustun va 2 dan 6 gacha bo'lgan 5 ta satrlarda joylashgan 20 ta katak.

Ishchi kitob va *ishchi varaqlar* ustida bajariladigan ba'zi asosiy amallar:

❖ *Ishchi kitob (fayl) yaratish.* **Файл - Создать - Новая книга - Создать** buyruqlar ketma-ketligi bosiladi.

❖ *Xotiradagi faylni ochish.* **Файл - Открыть** buyrug'i yordamida ochilgan muloqot oynadan kerakli fayl tanlanib, **Открыть** buyrug'i bosiladi.

❖ *Faylni xotirada saqlash.* **Файл - Сохранить** tugmasi bosiladi.

❖ *Faylni chop qilish.* **Файл - Печать** buyrug'i bosiladi va ochilgan oynadan printer turi, chop etiladigan varaqlar raqamlari va boshqa parametrlar ko'rsatilib, **Печать** tugmasi bosiladi.

❖ *Faylni yopish.* **Файл - Закрыть** buyrug'i bosiladi.

❖ *Ishchi varaq yaratish va uni o'chirish.* **Ишчи varaq yaratish** - **Вставить лист (Shift+F11)** buyrug'i bosilganda yangi *ishchi varaq* hosil bo'ladi. **Ишchi varaqni yo'qotish** uchun esa panelda *ishchi varaq* yorlig'ining kontekst menyusi ochilib, **Удалить** buyrug'i tanlanadi.

❖ *Ishchi varaq nomini o'zgartirish.* **Excel** *ishchi varaqlari* odatda **Лист1**, **Лист2** va h.k. so'zlar bilan nomlanib boradi. Lekin *ishchi varaq* nomini xohishga qarab o'zgartirish ham mumkin. Buning uchun *ishchi varaq* yorlig'i ustida sichqoncha o'ng tugmasi bosilib, hosil bo'lgan kontekst menyudan **Переименовать** buyrug'i bosiladi. Bu holda *ishchi varaq* yorlig'ida kursor paydo bo'ladi. Yangi nom kiritilib, **Enter** tugmasi bosiladi.

❖ *Maydonlarni mahkamlash va uni hekor qilish.* *Ishchi varaqning* barcha kataklari ekranga sig'maganligi sababli ishlash jarayonida ba'zan mo'ljal yo'qotib qo'yiladi, ya'ni ishlayotgan soha ko'rinmay qoladi. Bu hol bo'lmasligi uchun maydon mahkamlab qo'yiladi.

- Mahkamlangan maydon doimo ekranda ko'rinib turadi. Maydonni mahkamlash uchun **Вид** menyusining **Закрепить области** buyrug'i bosiladi. Hosil bo'lgan navbatdagi menyudan yana **Закрепить области** buyrug'i bosilganda faol katakning chap va yuqori qismi mahkamlanadi. **Закрепить верхнюю строку** yoki **Закрепить первый столбец** buyrug'i bosilganda mos ravishda birinchi qator yoki birinchi ustun mahkamlanadi. Mahkamlashni bekor qilish uchun **Снять закрепление областей** buyrug'i bosiladi.
- ❖ **Satr va ustun qo'shish.** Bir yoki bir nechta satrlar yoki ustunlarni ularning nomlari orqali belgilab, kontekst menyusining **Вставить** buyrug'ini bosish mos ravishda yangi satrlar yoki ustunlar qo'shilishiga olib keladi.
 - ❖ **Satr va ustunlarni yo'qotish.** Satr yoki ustunlarni ularning nomlari orqali belgilab, kontekst menyusining **Удалить** buyrug'i bosiladi.
 - ❖ **Ustun kengligini o'zgartirish.** Главная menyusining **Формат** buyrug'i yordamida ochilgan menyudan **Ширина столбца** buyrug'i bosiladi. Hosil bo'lgan muloqot oynasida kerakli qiymat kiritilib, **OK** tugmasi bosiladi.
 - ❖ **Satr balandligini o'zgartirish.** Excelda odatda satr kengligi "normal" stilidagi shrift o'lchamida bo'ladi. Bu holda katakka kiritilgan eng katta harf yoki son ham ko'rinib turadi. Shrift o'lchami qanchalik kattalashtirilsa, satr balandligi shunga mos ravishda o'zgaradi. Satr balandligini ixtiyoriy o'zgartirish xuddi ustun kengligini o'zgartirish kabi amalga oshiriladi.
 - ❖ **Dasturdan chiqish.** Dasturni butunlay yopish uchun **Файл** menyusining **Выход** buyrug'i, yoki **Alt+F4** tugmalar kombinatsiyasi, yoki oynaning yuqori o'ng burchagidagi **X** tugmasi bosiladi.

Ma'lumotlarni kiritish va tahrir qilish

Excel kataklaridagi ma'lumotlar odatda quyidagi 3 turda bo'lishi mumkin: sonli qiymat, matn, formula. Bundan tashqari ishchi varaqlarda rasmlar, diagrammalar, tugmalar va boshqa obyektlar joylashishi mumkin.

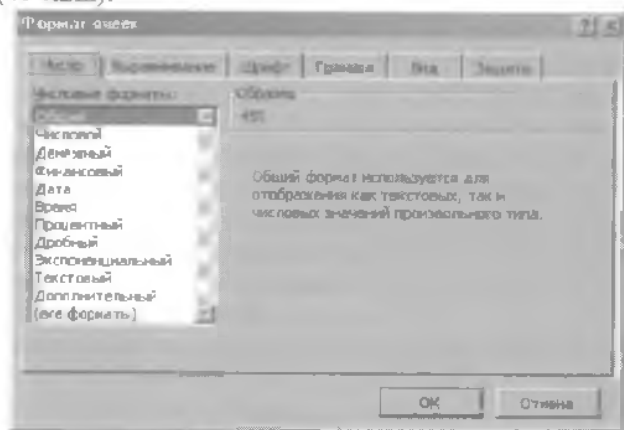
Sonli qiymat, matn, formula va funksiyalarni kiritish uchun avvalo sichqonchani chap tugmasi ma'lumot kiritilishi lozim bo'lgan katakda bir marta bosiladi. Ma'lumot kiritilib, **Enter** tugmasi bosiladi.

◆ Sonli qiymatlar

Excelda kiritilayotgan sonlar quyidagi raqam va belgilardan tashkil topishi mumkin: **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, (), /, ,, \$, %, ., E, e.**

Kiritilgan sonlar odatda katakning o'ng tomoniga tekislanadi.

Sonli formatlarni qo'llash. Excelda turli tuman masalalarni yechish imkoniyati mavjud. Bu masalalarda sonlar har xil ko'rinishda ishlatilishi mumkin. Masalan, foiz, pullik, vaqt, sana, matn, moliyaviy, eksponensial, kasr, umumiy va boshqa formatlar ko'rinishida. Bunday formatlarni qo'llash uchun katak yoki kataklar guruhi belgilanib, Главная menyusining **Формат – Формат ячеек** buyruqlar ketma-keligi yoki kontekst menyuning **Формат ячеек** buyrug'i tanlanadi. Natijada quyidagi muloqot oynasi ochiladi (41-rasm):



41-rasm. Формат ячеек oynasi

Bu oynaning Число panelidagi **Числовые форматы** bo'limidan kerakli format tanlanganda o'ng tomondagi **Образец** sohasida son formatining umumiy ko'rinishi hosil bo'lib, uning pastki qismida tanlangan format to'g'risida qisqacha tushuntirma paydo bo'ladi. Biror turdagi format tanlanib, **OK** tugmasi bosiladi.

◆ **Matnlar va ularni kiritish**

Ko'p hollarda ishchi varaqlarga matn yozishga ham to'g'ri keladi. Misol uchun, satr va ustun sarlavhalari, sonli ma'lumotlarga tushuntirma, jadval mavzusi matnlar yordamida ifodalanadi. Matnlar sonlar, harflar, belgilar ketma-ketligidan tashkil topishi mumkin. Masalan, biror katakka **128-qator** deb kiritilsa, Excel bu ma'lumotni matn sifatida tushunadi. Matnlar ham xuddi sonlar kabi kiritiladi va odatda chap tomonga tekislanadi.

◆ **Ma'lumotlarni tahrirlash**





Excel kataklaridagi ma'lumotlar ustida har qanday amalni bajarish uchun kerakli katak yoki maydon belgilab olinishi lozim.

Katakni belgilash. Har qanday katak ustida sichqonchanning chap

tugmasini bir marta bosish natijasida katak belgilanadi. Belgilangan katak *faol katak* (faol yacheyka) yoki *joriy katak* deyiladi.

Maydon tanlash. Maydon ustida har qanday amalni bajarish uchun avvalo maydon belgilanishi (tanlanishi) lozim. Buning uchun sichqonchanning chap tugmasi kerakli maydonning yuqori chap katagi ustida bosilib, qo'yib yuborilmasdan quyi o'ng katagigacha surib boriladi. Bu ishni **Shift** tugmasi va kursorni boshqarish tugmalaridan foydalanib ham bajarish mumkin. Bu amallar yonma-yon joylashgan kataklar uchun o'rinlidir. Yonma-yon joylashmagan kataklar uchun esa, **Ctrl** tugmasini bosib turgan holda sichqoncha yordamida kerakli maydonlar tanlanadi.



Excelda kiritilgan ma'lumotlarni tahrirlash amallari mavjud. Ularning ayrimlari quyida keltirilgan:

- ❖ **kiritilgan ma'lumotni o'chirish.** Buning uchun ma'lumot turgan yacheyka faollashtiriladi va klaviaturadagi **Delete** tugmasi bosiladi;
 - ❖ **ma'lumotlarni nusxalash.** Katakda ma'lumotning nusxasini olish uchun katak faollashtiriladi va **Главная** menyusining **Копировать** yoki kontekst menyusining **Копировать** (yoki **Ctrl+C**) buyrug'i bosiladi. Qo'yilishi kerak bo'lgan katak faollashtirilib, **Главная** menyusining **Вставить** yoki kontekst menyusining **Вставить** (yoki **Ctrl+V**) buyrug'i bosiladi;
 - ❖ **ma'lumotlarni ko'chirish.** Katakda ma'lumotni ko'chirish uchun katak faollashtiriladi va **Главная** menyusining **Вырезать** yoki kontekst menyusining **Вырезать** (yoki **Ctrl+X**) buyrug'i bosiladi. Qo'yilishi kerak bo'lgan katak faollashtirilib, **Главная** menyusining **Вставить** yoki kontekst menyusining **Вставить** (yoki **Ctrl+V**) buyrug'i bosiladi;
 - ❖ **katakda ma'lumotni o'zgartirish.** Ba'zi hollarda katakka noto'g'ri kiritilgan ma'lumotlarni to'g'rilash uchun uni o'chirib, qaytadan yozmasdan unga o'zgina o'zgartirish kiritish yetarli bo'ladi. Bu ishni bir necha usullar bilan bajarish mumkin:
 - a) katakcha ustida sichqonchanning chap tugmasi 2 marta bosiladi va bu holda katak ichida kursor paydo bo'ladi. Kerakli qism to'g'rilanib, **Enter** tugmasi bosiladi;
 - b) katakcha faollashtiriladi va **F2** funksional tugma bosiladi. Kerakli o'zgartirishlar kiritilib, **Enter** tugmasi bosiladi;
 - c) katakcha faollashtiriladi. Bu vaqtda katakdagi ma'lumotlar formulalar satrida ko'rinadi. Sichqonchanning chap tugmasi formulalar satrida bir marta bosiladi. Kerakli o'zgartirishlar kiritilib, **Enter** tugmasi bosiladi.
- Bu uch usulni qo'llaganda ham formulalar satrining chap tomonida qo'shimcha 2 ta tugma hosil bo'ladi:  . Bunda  tugmasi tahrirlashni bekor qiladi,  tugmasi esa tahrirlashni tugatadi. Bu tugmalarning o'rtiga

mos ravishda **Esc** va **Enter** tugmalarini bosish ham mumkin.

◆ **Ma'lumotlarni formatlash**

Ma'lumotlarni formatlash bo'yicha bir qancha amallar mavjud. Ularning barchasini yuqoridagi **Формат ячеек** oynasidan bajarish mumkin. Barcha formatlash amallarini qo'llash uchun albatta formatlanish kerak bo'lgan soha belgilab olinishi lozim. Ba'zi formatlash amallari:

- ❖ **tekislash.** Bu kabi ishlarni bajarish uchun **Формат ячеек** oynasining **Выравнивание** paneli ochiladi. Panelning **Выравнивание** sohasidagi **по горизонтали** oynachasidan gorizontal tekislash, **по вертикали** oynachasidan vertikal tekislash buyruqlari tanlanadi. Panelning **Оформление** sohasidan katakka sig'magan ma'lumotni qanday joylashtirilish kerakligi ko'rsatiladi. Masalan, **переносить по словам** bandi tanlansa, katakning qatoriga sig'magan ma'lumot katakning navbatdagi qatoridan yozilishini ko'rsatadi. **Автоподбор ширины** bandi tanlansa, katakdagi uzun ma'lumot katakka moslashishini bildiradi. Agar **объединение ячеек** bandi tanlansa, ma'lumot uzunligiga qarab katak keyingi kataklar bilan birlashadi. Kerakli bandlar tanlab bo'lingandan so'ng **OK** tugmasi bosiladi.
- ❖ **ma'lumot shrifti va o'lchamini o'zgartirish.** Ma'lumotlar shrifti va o'lchamini o'zgartirish uchun **Формат ячеек** oynasining **Шрифт** panelidan mos ravishda **Шрифт** va **Размер** oynachalaridan kerakli qiymatlar tanlanib, **OK** tugmasi bosiladi. Bu amalni uskunalar panelining oynachalaridan foydalanib bajarish ham mumkin;
- ❖ **rang berish.** Ma'lumotlar va ma'lumotlar egallab turgan soha ajralib, chiroyli ko'rinib turishi uchun ularga biror rang berish maqsadga muvofiqdir. Buning uchun kerakli katak yoki soha belgilanib, katak yoki soha foni uchun  tugmasi, katak yoki sohada joylashgan ma'lumotlar rangi uchun esa  tugmasini bosish kerak;
- ❖ **katakka yangi satr qo'shish.** Ba'zi vaziyatlarda bir katakka katak enini o'zgartirmagan holda uzun matnni joylashtirishga to'g'ri keladi. Bunday hollarda **Alt+Enter** tugmalar kombinatsiyalaridan foydalanib, matning qolgan qismini navbatdagi qatordan yozish mumkin. Bu holda **Excel** joriy katak uchun **matnni ko'chirish** holatini o'rnatadi.

Excelda formula va funksiyalar

Formula va funksiyalar Elektron jadvalda ishlashda eng asosiy vosita hisoblanadi. **Excel** oddiy matn muharriridan shunisi bilan farq qiladi. Formula va funksiyalar kataklardagi boshlang'ich qiymatlar orqali ma'lum amallar yordamida yangi zarur qiymat hosil qilish uchun ishlatiladi.

Boshlang'ich qiymatlar o'zgartirilganda formula yoki funksiya yordamida hosil qilingan qiymat ham avtomatik ravishda o'zgaradi. Bu esa formula yoki funksiyalarni bir marta yozib, boshlang'ich qiymatlarni xohlagancha o'zgartirib, yangi-yangi natijalar olish imkonini beradi.

Agar biror katakka qandaydir formula kiritilib, **Enter** tugmasi bosilsa, formula o'rnida shu formula bo'yicha hisoblangan qiymat hosil bo'ladi. Biror katakdagi sonli qiymat qanday formula yordamida hisoblanganligini bilish uchun katak faollashtiriladi. Natijada formulalar satrida yozilgan formula hosil bo'ladi.

Katakka kiritiladigan formulada quyidagi elementlar bo'lishi mumkin:

- arifmetik amallar;
- katak yoki maydon adresi, yoki ularning nomi;
- sonlar;
- ishchi jadval funksiyalari.

Formulada quyidagi amallardan foydalanish mumkin:

6-jadval. Excelda amal belgilari

Amal belgisi	Nomi	Ustunlik darajasi
^	Darajaga ko'tarish	1
*	Ko'paytirish	2
/	Bo'lish	2
+	Qo'shish	3
-	Ayirish	3
&	Konkatenatsiya (ulash)	4
=	Tenglikni mantiqiy solishtirish	5
>	Kataga mantiqiy solishtirish	5
<	Kichikka mantiqiy solishtirish	5

Jadvaldan ko'rinib turibdiki, misol uchun darajaga ko'tarish amali birinchi bo'lib bajariladi. Oddiy qavslar yordamida amallarning tabiiy bajarilish ketma-ketligini o'zgartirish mumkin.

Formulalarga misollar:

=15*0,5 - 15 soni 0,5 soniga ko'paytiriladi;

=A1+C1 - A1 va C1 katakdagi sonlar qo'shiladi;

=MAKC(B1:D5) - B1:D5 maydonida joylashgan eng katta qiymat topiladi;

=B2^(2/3) - B2 katakda turgan sonning $\frac{2}{3}$ - darajasi topilmoqda;

=(2-ABS(0,2+A2))/(1+A2^3) - $\frac{2-|0,2+A2|}{1+A2^3}$ formula hisoblanmoqda.

Formulalarni kiritish

Formulani kiritish uchun dastlab formula kiritilishi kerak bo'lgan katak faollashtiriladi. Shundan so'ng katakning o'zida yoki formulalar satrida ma'lum formula kiritiladi. Formula har doim "=" (tenglik) belgisidan boshlanishi shart.

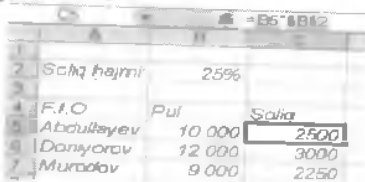
Formula to'liq kiritib bo'lingandan so'ng albatta **Enter** tugmasi bosilishi lozim.

Adreslar

Formulalarda adreslar muhim o'rin tutadi. Adreslar *nisbiy*, *mutlaq* va *aralash* adreslarga bo'linadi. Ularning bir-biridan farqi quyida misollar yordamida tushuntirilgan.

Nisbiy adreslar. Faraz qilaylik, **D4** katakda $=B4+C4$ formula yozilgan bo'lsin. Agar bu formuladan **D5** va **D6** kataklarga nusxa olinsa, bu kataklarda mos ravishda $=B5+C5$ va $=B6+C6$ formulalar paydo bo'ladi. Bu yerda **Excel** formulalarda qatnashgan adreslarni ularning tutgan o'rniga qarab o'zgartirmoqda. E'tibor beriladigan bo'lsa, birinchi formula ikki katak chapda joylashgan son bilan bir katak chapda joylashgan son yig'indisidan iborat. Ikkinchi va uchinchi formulalarda ham aynan birinchi qoidadan nusxa olingan, ya'ni bu formulalar ham ikki katak chapda joylashgan son bilan bir katak chapda joylashgan son yig'indisidan iborat. Bu misolda birinchi formuladagi **B4** va **C4** adreslar nisbiy adreslar hisoblanadi. Xullas, formuladan nusxa olinganda unda qatnashgan adreslar o'zgarsa, ular **nisbiy adreslardir**.

Mutlaq adreslar. Ba'zi holatlar bo'ladiki, formuladan nusxa olinganda unda qatnashgan ba'zi adreslar o'zgarmay qolishi lozim 42-rasmdagi **Excel** jadvaliga qaraylik.



	A	B	C
1			
2	Soliq hajmi	25%	
3			
4	F.I.O	Pul	Soliq
5	Abdullayev	10 000	2500
6	Daniyorov	12 000	3000
7	Muradov	9 000	2250

42-rasm. **Excel** elektron jadvalida masala

Bu misolda **B2** katakda soliq foizi, **C5** katakda **B5** katakdan olingan soliq miqdori keltirilgan. **C5** katakda $=B5*\$B\2 formula yozilgan. Bu formuladan **C6** va **C7** kataklarga nusxa olinsa, mos ravishda $=B6*\$B\2 va $=B7*\$B\2 formulalar hosil bo'ladi. Bu misoldan ko'rinib turibdiki, nisbiy adres o'zgardi, mutlaq adres o'zgarmadi, bu yerda **\$B\$2** adres mutlaq


adresdir. Xullas, har bir mutlaq adresda ikkitadan \$ belgi ishtirok etadi, birinchisi ustun, ikkinchisi satr raqami oldida.

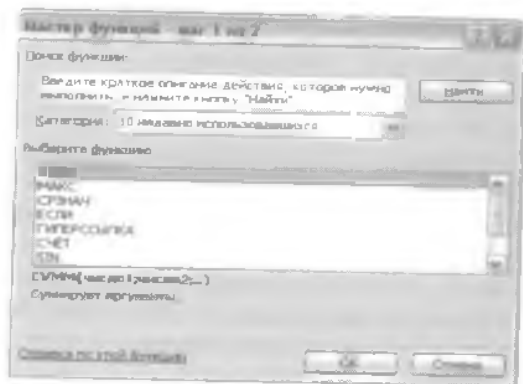
Aralash adreslar. Yana shunday holatlar bo'ladiki, formulada qatnashgan adresning faqat bir qismi, ya'ni yoki ustun nomi yoki satr raqami o'zgaradi. Bunday adreslar **aralash adreslar** deyiladi.

Funksiyalar

Funksiyalar Excel bazasiga oldindan kiritilgan bo'lib, ular formulalar yozishda qo'llaniladi. Funksiyalarning qo'llanilishi:

- *ba'zi to'g'ridan-to'g'ri hisoblash mumkin bo'lmagan formulalarni yozishda;* masalan, B2 dan B20 gacha bo'lgan kataklarda joylashgan sonlarning eng kattasini B22 katakda hosil qilish kerak bo'lsin. Bu masalani oddiy formulalar yordamida hosil qilib bo'lmaydi. Buning uchun maxsus **MAKC** funksiyasi mavjud bo'lib, B22 katakda **=MAKC(B2:B20)** formulani yozish zarur va yetarli;
- *katta hajmli ma'lumotlar ustida bajariladigan formulalarni sodda ko'rinishda yozishda;* masalan, B15 katakda B4 dan B14 gacha bo'lgan kataklarda joylashgan sonlarning o'rta arifmetigini hosil qilish kerak bo'lsin. Buni formula yordamida hal qiladigan bo'lsak, B15 katakka **=(B4+B5+B6+B7+B8+B9+B10+B11+B12+B13+B14)/11** formula kiritilishi lozim. Ushbu oddiy formulani kiritish uchun ancha vaqt va mehnat talab etiladi. Agar bu son 11 ta emas, juda ko'p bo'lsachi? Bunday kattaroq formulalarni yozishda funksiyalardan foydalaniladi. Yuqoridagi formula **CP3HA4** funksiyasi yordamida yoziladigan bo'lsa, B15 katakka **=CP3HA4(B4:B14)** formula kiritilishi kifoya;
- *matematikadagi ko'pgina maxsus funksiyalarni hisoblashlarda;* masalan, D3 katakda joylashgan sonning sinusini D4 katakka chiqarish lozim bo'lsa, D4 katakka **=SIN(D3)** formula yoziladi.

Funksiyalarni katakka to'g'ridan-to'g'ri katakning o'zida yoki formulalar satrida kiritish mumkin. Lekin **Exceld**a funksiyalarni kiritish bundanda osonlashtirilgan bo'lib, bu ish funksiya ustasi yordamida amalga oshiriladi. Funksiya ustasini ishga tushirish uchun **Формулы** menyusining **Вставить функцию** yoki formulalar satridagi  tugmasi bosiladi. Natijada funksiya ustasi oynasi hosil bo'ladi (43-rasm).



43-рasm. Функция устаси oynаси

Bu oynaning **Категория** darchasida 13 turkumdagi funksiyalar menyusi joylashgan. Biror turkumdagi menyu tanlansa, oynaning **Выберите функцию** darchasida shu turkumga mos funksiyalar ro'yxati paydo bo'ladi. Bu ro'yxatdan kerakli funkciya tanlanadi, shunda darchaning pastki tomonida tanlangan funkciyaning qisqacha tavsifi va yo'riqnomasi paydo bo'ladi. Bu yerdan funkciyaning nima uchun va qanday ishlatilishini bilib olish mumkin. Funkciya tanlanib, **OK** tugmasi bosiladi. Natijada shu funkciyaning paneli paydo bo'ladi. Odatda bu panelda ko'pincha funkciya ustasining o'zi hisoblanayotgan maydon va usulni taklif etadi. Tekshirib, to'g'riligiga ishonch hosil qilinsa, **OK** tugmasini bosish mumkin. Bunday bo'lmasa, o'zimiz panel oynasiga to'g'ri qiymatlarni kiritib, **OK** tugmasini bosishimiz lozim.

Bu funksiyalarning ha'zilari quyidagilar:

1. **СУММ**. Excelda eng ko'p ishlatiladigan funksiyalardan biri **СУММ** hisoblanadi. Bu funkciya sonlar va ko'rsatilgan maydondagi qiymatlar yig'indisini topishga imkon beradi.

Umumiy ko'rinishi: **СУММ(1-qiymat;2-qiymat; ... ; n-qiymat)**

bu yerda 1-qiymat; 2-qiymat; ... ; n-qiymatlar argumentlar ro'yxati. Ular o'rniida aniq sonlar, maydon yoki katak adreslari kelishi mumkin.

Masalan, **B2:D50** maydonda joylashgan sonlar yig'indisini topish uchun **=СУММ(B2:D50)** formulani yozish, 30 soni bilan **A2:E2** maydonda joylashgan sonlar yig'indisini topish uchun **=СУММ(A2:E2;30)** formulani yozish, **C2:E10** va **F11:H15** maydonlarda joylashgan sonlar yig'indisini topish uchun esa **=СУММ(C2:E10;F11:H15)** formulani yozish kifoyadir.

Joylashishi: **Мастер функций** oynasi **Категория** darchasining **Математические** bo'limi.

2. **CP3HACH.** Bu funksiya sonlar va berilgan maydonda joylashgan qiymatlar o'rtta arifmetigini topishda ishlatiladi.
Umumiy ko'rinishi: CP3HACH(1-qiymat;2-qiymat; ... ;n-qiymat)
 Masalan, C3:C20 maydonda joylashgan sonlarning o'rtta arifmetigini topish uchun =CP3HACH(C3:C20) formuladan foydalanish mumkin.
Joylashishi: Статистические bo'limi.
3. **MAKCS.** Bu funksiya sonlar va berilgan maydonda joylashgan qiymatlarning eng kattasini topish uchun qo'llaniladi.
Umumiy ko'rinishi: MAKCS(1-qiymat;2-qiymat; ... ;n-qiymat)
 Masalan, B2:D10 maydonda joylashgan sonlarning eng kattasini topish uchun =MAKCS(B2:D10) formula yozilishi lozim.
Joylashishi: Статистические bo'limi.
4. **MIN.** Bu funksiya sonlar va berilgan maydonda joylashgan qiymatlarning eng kichigini topish uchun qo'llaniladi.
Umumiy ko'rinishi: MIN(1-qiymat;2-qiymat; ... ;n-qiymat)
 Masalan, 30 soni bilan C2:C10 maydonda joylashgan sonlarning eng kichigini C11 katakda hosil qilish kerak bo'lsa, C11 katakka =MIN(30;C2:C10) formula kiritilishi lozim.
Joylashishi: Статистические bo'limi.
5. **ПРОИЗВЕД.** Sonlar va berilgan maydonda joylashgan qiymatlar ko'paytmasini topish uchun ishlatiladi.
Umumiy ko'rinishi: ПРОИЗВЕД(1-qiymat;2-qiymat; ... ;n-qiymat)
 Masalan, B17 katakda B2:B15 maydonda joylashgan sonlar ko'paytmasini hosil qilish uchun B17 katakka =ПРОИЗВЕД(B2:B15) formulani kiritib, Enter tugmasini bosish yetarli.
Joylashishi: Математические bo'limi.
6. **ФАКТР.** Faktorialni hisoblash uchun ishlatiladi.
Umumiy ko'rinishi: ФАКТР(qiymat)
 Masalan, C5 katakda turgan son faktorialini hisoblash uchun =ФАКТР(C5) formula yoziladi.
Joylashishi: Математические bo'limi.
7. **СТЕПЕНЬ.** Darajani hisoblash uchun ishlatiladi.
Umumiy ko'rinishi: СТЕПЕНЬ(qiymat;daraja)
 Masalan, 5¹⁰ ni hisoblash uchun =СТЕПЕНЬ(5;10) formula yozilishi lozim.
Joylashishi: Математические bo'limi.
8. **НАИБОЛЬШИЙ.** Massivning k-katta elementini topish uchun ishlatiladi.
Umumiy ko'rinishi: НАИБОЛЬШИЙ(massiv;k)
 Masalan, A2:B6 maydonda joylashgan sonlarning 3-kattasini topish

uchun =**НАИБОЛЬШИЙ**(A2:B6;3) formula ishlatiladi.

Joylashishi: Статистические bo'limi.

9. **НАИМЕНЬШИЙ**. Massivning k-kichik elementini topish uchun ishlatiladi.

Umumiy ko'rinishi: **НАИМЕНЬШИЙ**(massiv;k)

Masalan, A2:D3 maydonda joylashgan sonlarning 2-kichigini topish uchun =**НАИМЕНЬШИЙ**(A2:D3;2) formula ishlatiladi.

Joylashishi: Статистические bo'limi.

10. **МУМНОЖ**. Massivlar ko'paytmasini topadi.

Umumiy ko'rinishi: **МУМНОЖ**(1-массив, 2-массив)

Masalan, A1:D3 maydonda joylashgan massivga B5:C8 maydondagi massivni ko'paytirish uchun 3x2 o'lchamli maydon belgilab olinadi va =**МУМНОЖ**(A1:D3;B5:C8) formula kiritilib, Ctrl+Shift+Enter tugmalari birgalikda bosiladi.

Joylashishi: Математические bo'limi.

11. **ЕСЛИ**. Qiymatlarni mantiqiy ifodaga ko'ra chiqarishni ta'minlaydi.

Umumiy ko'rinishi: **ЕСЛИ**(mantiqiy ifoda;1-qiymat;2-qiymat)

Bunda, mantiqiy ifoda bajarilsa, 1-qiymat, bajarilmasa, 2-qiymat ekranga chiqadi.

Masalan, B2 katakka kiritilgan son 55 dan katta bo'lsa, o'tdi, 55 va undan kichik bo'lsa, o'tmadi so'zini B4 katakka chiqarish lozim bo'lsa, B4 katakka =**ЕСЛИ**(B2>55; "o'tdi";"o'tmadi") formulasi yoziladi.

Joylashishi: Логические bo'limi.

12. **СЧЁТЕСЛИ**. Massivning berilgan shartni qanoatlantiradigan qiymatlari sonini topishga imkon beradi.

Umumiy ko'rinishi: **СЧЁТЕСЛИ**(massiv;shart)

Masalan, B2:B5 maydonda joylashgan sonlarning 55 dan kattalari sonini topish uchun =**СЧЁТЕСЛИ**(B2:B5;">55") formula kiritilishi lozim.

Joylashishi: Статистические bo'limi.

13. **СУММЕСЛИ**. Massivning berilgan shartni qanoatlantiradigan qiymatlari yig'indisini topishga imkon beradi.

Umumiy ko'rinishi: **СУММЕСЛИ**(1-maydon;shart;2-maydon)

Masalan, A2:A6 maydonda joylashgan sonlarning 55 dan kattalari yig'indisini topish uchun =**СУММЕСЛИ**(A2:A6;">55") formula kiritilishi lozim.

Masalan, biror katakka =**СУММЕСЛИ**(A2:A56;"Aprel";B2:B56) formula kiritilgan bo'lsa, A2:A56 maydondagi "Aprel" so'zlariga mos keluvchi B2:B56 maydon kataklaridagi sonlar yig'indisi hosil bo'ladi.

Joylashishi: Математические bo'limi.

14. **СУММПРОИЗВ.** Massivlarning mos ravishdagi qiymatlari ko'paytmalarining yig'indisini topishda ishlatiladi.

Umumiy ko'rinishi: СУММПРОИЗВ(1-массив, 2-массив,...) Masalan, A2:B6 va D3:E7 maydonlarda joylashgan massivlarning mos ravishdagi qiymatlarining ko'paymalari yig'indisini topish uchun =СУММПРОИЗВ(A2:B6;D3:E7) formula kiritilishi lozim.

Joylashishi: Математические bo'limi.

Shunday standart funksiyalar ham borki, ularni funksiya ustasi yordamida yozgandan ko'ra, ularni klaviatura orqali tezroq kiritish mumkin. Quyida shunday ba'zi matematik standart funksiyalar keltirilgan:

7-jadval. Excelda matematik standart funksiyalar

Funksiya	Nomi
ABS(argument)	Absolyut qiymat
ACOS(argument)	Arkkosinus
ASIN(argument)	Arksinus
ATAN(argument)	Arktangens
COS(argument)	Kosinus
EXP(argument)	Eksponenta
LN(argument)	Natural logarifm
LOG10(argument)	O'nli logarifm
LOG(argument:asos)	Berilgan asosga ko'ra argument logarifmi
ПИ()	π
SIN(argument)	Sinus
TAN(argument)	Tangens

Yuqoridagi funksiyalarda argument o'rniga katak adresi yoki aniq son yoziladi.

Excelda diagrammalar qurish va ularni tahrirlash

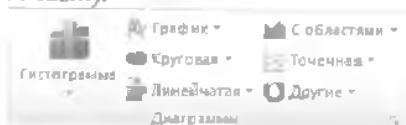
Excelning asosiy imkoniyatlaridan yana biri diagrammalar qurishdir.

Diagrammalar jadval ko'rinishdagi ma'lumotlarning grafik ko'rinishdagi tasviridir. Inson ko'zi sonli ma'lumotlardan ko'ra, grafik va tasvirlarni yaxshiroq ko'ra oladi. Aslida diagramma bizga hech narsa bermaydi. Lekin sonli ma'lumotlarni tushunish va tasavvur qilishda diagrammalarning roli beqiyosdir. Shuning uchun ham odatda Excelda barcha hisob-kitob ishlari hajarib bo'lingandan so'ng, boshlang'ich qiymat va natijalar asosida diagrammalar quriladi.

Har xil masalalarni yechish jarayonida turli ma'lumotlar hosil bo'ladi. Tabiiyki, bu ma'lumotlarning grafiklari ham turlicha bo'ladi. Excelda juda ko'p turdagi diagrammalarni yasash imkoni mavjud. Masalan, **chiziqli**

doira, gistogramma, yuza, halqa ko'rinishdagi diagrammalar. Bir masala uchun bu diagrammalarning barchasidan foydalanish o'rinli emas. Misol uchun, guruh talabalarining fanlar bo'yicha olgan ballarini yoki stipendiya miqdorlarini tasvirlash uchun diagrammalarning *chiziqli (линейчатая), Gistogramma* turlaridan, jahon mashina bozorida davlatlarning ulushlari yoki uyali aloqa kompaniyalarining abonentlar bo'yicha ulushlarini tasvirlash uchun *doira, xalqa* turlaridan foydalanish maqsadga muvofiq bo'ladi.

Diagramma hosil qilish uchun odatda avvalo kerakli maydon yoki maydonlar belgilab olinadi. So'ngra **Вставка** menyusining diagrammalar bo'limiga o'tiladi (44-rasm).



44-rasm. Diagrammalar bo'limi

Bu yerda diagrammalarning bir qancha turlari mavjud. Ularning birortasi tanlanganda shu turga mos diagramma ko'rinishlari paydo bo'ladi. **Другие** bo'limi tanlanganda boshqa turdagi diagrammalar va ularning ko'rinishlari ham paydo bo'ladi. Bu ko'rinishlardan keraklisi bosilganda ishchi sohada diagramma hosil bo'ladi.

Hosil qilingan diagramma odatda ba'zi tahrirlashlarga muhtoj bo'ladi. Diagramma tanlanganda uni tahrirlovchi **Конструктор, Макет, Формат** menyulari paydo bo'ladi.

Конструктор menyusini: –

- o Menyuning **Тип** bo'limidagi **Изменить тип диаграммы** buyrug'i tanlanganda barcha turdagi diagramma turlari paydo bo'ladi. Bu turlarning biror ko'rinishi tanlanib, **ОК** buyrug'i bosiladi.
- o **Тип** bo'limidagi **Сохранить как шаблон** buyrug'i yordamida hosil qilingan diagramma yangi tur sifatida saqlab qo'yilishi va keyinchalik boshqa ma'lumotlar sohasiga qo'llanilishi mumkin.
- o Menyuning **Данные** bo'limidagi **Строка/столбец** buyrug'i yordamida ma'lumotlarning aniqlanish va qiymatlar sohasini almashtirish mumkin.
- o **Данные** bo'limidagi **Выбрать данные** buyrug'i orqali ma'lumotlar sohasi o'zgartiriladi.
- o Menyuning **Макеты диаграмм** bo'limi yordamida joriy diagramma maketini o'zgartirish imkoniyati mavjud.
- o Menyuning **Стили диаграмм** bo'limi orqali diagramma kerakli dizaynga o'tkaziladi.

- Menyuning **Переместить диаграмму** buyrug'i yordamida diagrammani alohida varaqqa yoki mavjud ishchi varaqlarning birortasiga o'tkazish mumkin.

Макет menyusi:

- Menyuning **Подписи** bo'limidagi **Название диаграммы** buyrug'i yordamida diagrammaga sarlavha qo'yish yoki olib tashlash mumkin.
 - **Подписи** bo'limidagi **Названия осей** buyrug'i orqali diagrammaning koordinata o'qlarini nomlash yoki ularni olib tashlash imkoniyati mavjud.
 - **Подписи** bo'limidagi **Легенда** buyrug'i diagrammani tasvirlash uchun izoh joylashtirishga imkon beradi.
 - **Подписи** bo'limidagi **Подписи данных** buyrug'i yordamida bevosita diagrammada qiymatlarni aks ettirish mumkin.
 - **Подписи** bo'limidagi **Таблица данных** buyrug'i diagrammaga jadvalning o'zini ham joylashtiradi.
 - Menyuning **Оси** bo'limidagi **Оси** buyrug'i diagrammaga koordinata o'qlarini joylashtiradi yoki olib tashlaydi.
 - **Оси** bo'limidagi **Сетка** buyrug'i diagrammani tushurishni osonlashtirish uchun gorizontal va vertikal ko'rinishlarda to'rlar joylashtiradi.
- Формат** menyusi yordamida diagrammadagi yozuvlar va diagramma elementlarini turli ko'rinishlarda tasvirlash ishlari bajariladi.

МББТ, Access dasturi, Access interfeysi

Ma'lumki, hozirgi kunda fan-texnika sohasining rivojlanishi, axborot texnologiyalarining rivojlanishi, umuman insoniyat faoliyatining kengayib, rivojlanib va takomillashib borishi natijasida har bir sohada katta hajmdagi axborotlar oqimi hosil bo'lmoqda. Bu esa kompyuter va axborot texnologiyalari yordamida axborotlarni yig'ish, saralash, qayta ishlash, saqlash, qulay tarzda ifodalash, uzatish ishlarini tez va aniq tarzda bajarishga mo'ljallangan ma'lumotlar bazalarini yaratishni taqozo etadi. Bunday masalalar ma'lumotlar bazasini boshqarish tizimlari yordamida hal etiladi.

Ma'lumotlar bazasini boshqarish tizimi (МББТ) – bu ma'lumotlar bazasini yaratishga mo'ljallangan dasturlar tizimidir.

Ma'lumotlar bazasi (МБ) – bu o'zaro bog'langan va tartiblangan ma'lumotlar majmuasini kiritish, saqlab turish, kerakli parametrlar bo'yicha saralash, qidirish, foydalanuvchiga qulay tarzda chiqarib berish va uzatish ishlarini bajarishga mo'ljallangan yaxlit tizimdir.

МББТ vazifasiga ma'lumotlar bazasini boshqarishning quyidagi xususiyatlari kiradi:

- *ma'lumotlar olish*: foydalanuvchining talabiga binoan axborot va so'rov turidan qat'iy nazar unga qulay tarzda javob berish;
- *ma'lumotlarni modifikatsiyalash*: berilgan axborotni foydalanuvchi talabiga mos holda o'zgartirish;
- *ishonchlilik darajasi*: qurilmalar ishi tasodifan to'xtatilganda ma'lumotlar bazasini qayta tiklash imkoniyati;
- *ma'lumotlarni himoyalash*: MBdan ruxsatsiz foydalanishning cheklanganligi;
- *MBdan tarmoqda foydalanish*: ma'lumotlardan bir vaqtda bir necha kishining bir-biriga halaqit bermasdan foydalanishi. MB asosan quyidagi modellar asosida tuziladi:
- daraxtsimon (iyerarxik) modellar;
- tarmoqli (to'rtli) modellar;
- relyatsion modellar.

Ierarxik modelda obyektlar shajaraviy struktura ko'rinishida joylashtiriladi. To'rtli modelda esa quyi sinfdagi obyektlar ham bog'langan bo'lishi mumkin. Relyatsion modelda obyektlar ikki o'lchovli jadval ko'rinishida beriladi va bu jadvallar unikal maydon orqali bog'langan bo'ladi.

MBBT samaradorligi quyidagilar bilan baholanadi:

- so'rovlarni vaqtida bajarish;
- ma'lumotlarni qidirish vaqti;
- ma'lumotlar bazasidagi obyektlarni import va eksport qilish qobiliyati;
- yangilash, qo'shish, yo'qotish kabi ko'p ishlatiladigan amallarni bajarish tezligi;
- ko'p foydalanuvchili rejimda parallel murojaat qilishning ko'pligi;
- hisobotni hosil qilish tezligi.

Umuman olganda ma'lumotlar bazasi bilan ishlovchi juda ko'p dasturlar mavjud. Hozirgi kunda *Oracle, SQL Server, MySQL, dBASE, FoxPro, Informix, Paradox, Microsoft Access* kabi MBBTlar keng qo'llanilmoqda. Bundan tashqari *Visual Basic, Delphi, C#, Java* kabi universal dasturlash muhitlarida ham ma'lumotlar bazasini yaratish va uni boshqarish uchun qulay imkoniyatlar yaratilgan.

Microsoft Access dasturi MB yaratishga va uni boshqarishga mo'ljallangan, relyatsion modellar asosida qurilgan MBBTdir.

Access dasturi **Руск - Все программы - Microsoft office - Microsoft Access** menyular ketma-ketligi orqali ishga tushiriladi. Natijada dastur oynasi hosil bo'lib, dastlab xotiradagi MBni ochish yoki yangi MB yaratish kerak bo'ladi (45-rasm).

Закрывать базу данных – MBni yopish;

Сведения – joriy MB haqida to'liq ma'lumot olish hamda MBni shifrlash;

Последние – ochilgan yoki foydalanilgan oxirgi MBlar ro'yxatini ko'rish va ularni ochish;

Создать – yangi MB yaratish;

Печать (Ctrl+P) – joriy obyektни chop etish;

Справка (F1) – dastur va unda ishlash bo'yicha ma'lumotlar olish;

Параметры – dasturda ishlash parametrlarini o'zgartirish va uskunalar panelini sozlash;

Выход (Alt+F4) – MBBTdan chiqish.

Dasturning boshqa menyulari buyruqlari bevosita MB obyektlari bilan ishlashga mo'lgallanligi uchun ularni o'rni kelganda ko'rib o'tamiz.

Accessda jadval tuzish

Ma'lumotlar bazasidagi asosiy obyekt jadval hisoblanadi. Chunki barcha ma'lumotlar jadvallarda saqlanadi. Har qanday MB kamida bitta jadvaldan tashkil topadi. Shuning uchun ham yangi MB yaratish buyrug'i bosilganda avtomatik ravishda **Таблица1** nomli bo'sh jadval hosil bo'ladi. Bunda jadvaldagi ustunlar maydon, satrlar esa yozuv deb ataladi. Har bir maydon bir xil turga mansub turli ma'lumotlardan iborat bo'ladi. Har bir yozuv esa bitta obyektga doir turli ma'lumotlardan tashkil topadi.

Maydon turlariga misollar:

- ◆ **Текстовый.** Bu turdagi maydonga 255 tagacha simvollar kiritilishi mumkin.
- ◆ **Поле MEMO.** Maydon odatda xarakteristik ma'lumotlarni kiritishga mo'ljallangan bo'lib, 65535 tagacha bo'lgan simvollarni qabul qila oladi.
- ◆ **Числовой.** Sonli ma'lumotlarni kiritish;
- ◆ **Дата/время.** Vaqt va sanani ifodalash;
- ◆ **Денежный.** Pul formatidagi ma'lumotlarni kiritish;
- ◆ **Счетчик.** Qatorlarni avtomatik sanash.
- ◆ **Логический.** Mantiqiy tanlash amalini o'rnatish;
- ◆ **Поле объекта OLE.** OLE turidagi ma'lumotlarni kiritish;
- ◆ **Вложение.** Rasm, tasvir va ixtiyoriy fayllarni saqlash;
- ◆ **Вычисляемый.** Ma'lumotni hisoblash natijasida hosil qilish;
- ◆ **Гиперссылка.** Obyektga gipermurojaat o'rnatish.

Accessda MB obyektlari bilan ishlash uchun turli rejimlar mavjud.

Xususan, jadvallar bilan quyidagi rejimlarda ishlash mumkin:

- **Таблица** – ma'lumotlarni ko'rish va bevosita kiritishga mo'ljallangan.
- **Сводная таблица** – jadvalning kerakli qismidan hisobot jadvali tuzishga mo'ljallangan;
- **Сводная диаграмма** – jadvalning kerakli qismini diagramma ko'rinishida ifodalashga xizmat qiladi;
- **Конструктор** – jadval maydonlari xossalari o'zgartirish va yangi maydonlar qo'shishga mo'ljallangan.

Ochilgan jadvalni boshqa rejimga o'tkazish uchun Главная menyusining Режим buyrug'i bosiladi.

Yangi jadval Создание menyusining Таблица bo'limidagi quyidagi usullardan biri yordamida yaratiladi:

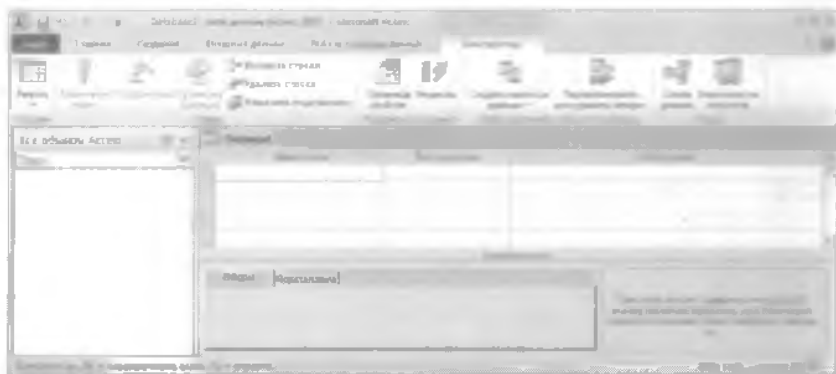
- **Таблица** – yangi bo'sh jadval yaratish. Bunda jadval maydonlari nomlari va turlari bevosita ma'lumot kiritish yo'li bilan aniqlanadi. Buyruq bosilganda Таблица rejimida Таблица1 nomli bo'sh jadval hosil bo'ladi.



46-rasm. Accessda Таблица rejimida jadval yaratish muhiti

Jadvalning 1-ustunida Счетчик turidagi Код maydoni hosil bo'lib, avtomatik tarzda natural sonlar bilan raqamlab boriladi. Код maydoni unikal maydon hisoblanadi va **критерий** deyiladi hamda jadval shu maydon orqali boshqa jadvallar bilan bog'lanadi. Yangi maydon qo'shish uchun uskunalar panelidan yoki jadvaldagi Щелкните для добавления buyrug'i bosilganda hosil bo'lgan ro'yxatdan maydon turi tanlanadi va ustun nomi kiritilib, Enter tugmasi bosiladi. Shu tariqa boshqa maydonlar ham yaratiladi. Shundan so'ng yozuvlar hosil qilinadi, ya'ni ma'lumotlar kiritiladi. Hosil bo'lgan jadval yopilayotganda jadvalga nom beriladi.

- **Конструктор таблиц** – конструктор rejimida jadval yaratish. Bunda maydonlar nomlari va turlari oldindan ko'rsatiladi. Buyruq bosilganda Конструктор rejimida Таблица1 nomli jadval hosil bo'ladi.

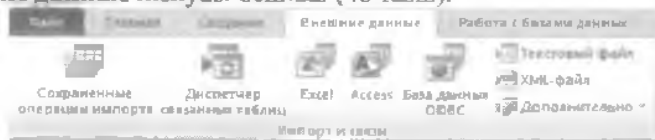


47-рasm. Accessda **Конструктор** rejimida jadval yaratish muhiti

Jadvalning **Имя поля** ustuniga maydonlar nomlari kiritiladi. **Тип данных** ustunidan mos ravishda maydonlar turlari tanlab boriladi. Jadvalning pastki qismidagi **Свойства поля** bo'limidan maydon xossalari o'zgartirish mumkin. Jadvalni keyinchalik boshqa jadvallar bilan bog'lash uchun biror unikal maydonga uskunalar panelidagi **Ключевое поле** buyrug'ini bosish orqali kalit qo'yiladi. Shundan so'ng jadval yopiladi va yangi nom kiritib saqlab qo'yiladi. Agar biror maydonga kalit qo'yilmagan bo'lsa, jadval yopilayotganda dasturning o'zi avtomatik kalit maydon qo'yishiga rozilik berish lozim. Ma'lumot kiritish uchun jadval **Таблица** rejimida qaytadan ochiladi.

➤ **Списки SharePoint** – internet tarmog'i orqali SharePoint bazasi yordamida jadval hosil qilish.

Bu usullardan tashqari ma'lumotlarni import qilish yo'li bilan jadval yaratish yoki jadval yaratmasdan MBga boshqa MBdagi yoki Excel ishchi varag'idagi jadvallarni bog'lab qo'yish ham mumkin. Buning uchun **Внешние данные** menyusi ochiladi (48-rasm).



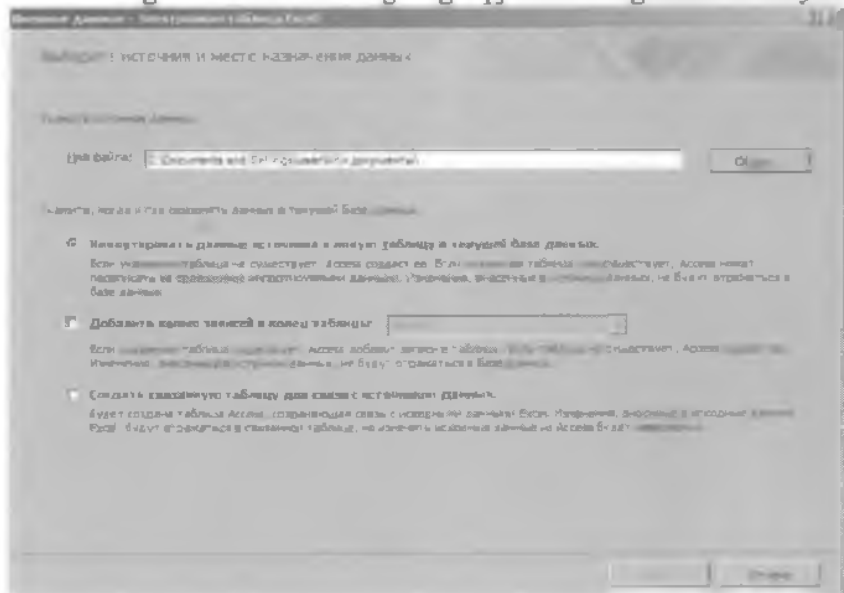
48-рasm. **Внешние данные** menyusining **Импорт и связи** bo'limi

Bu holda **Импорт и связи** bo'limida Excel, Access va boshqa import qilish va bog'lash mumkin bo'lgan dasturlar ro'yxati hosil bo'ladi.

Masalan, Excel ishchi varag'iga murojaat qilish uchun menyu bo'limidagi **Excel** buyrug'i bosiladi. Hosil bo'lgan **Внешние данные** – **Excel** oynasining (49-rasm) **Имя файла** darchasida **Обзор** buyrug'i

o'qali Excelda tayyorlangan fayl manzili va nomi ko'rsatiladi. Shuningdek, oynadagi quyidagi usullardan biri tanlanib, OK buyrug'i bosiladi:

- **Импортировать данные источника в новую таблицу в текущей базе данных** – ma'lumotlarni joriy MB yangi jadvaliga import qilish. Hosil bo'lgan navbatdagi oynalardan maydonlar nomi va turini o'zgartirish, kalit maydon aniqlash, jadvalni nomlash ishlari bajarilib, **Готово** tugmasi bosiladi. Agar import jarayonida xatoliklar aniqlansa, bu haqda xabar paydo bo'ladi va ular alohida jadvalda ko'rsatiladi;
- **Добавить копию записей в конец таблицы** – ma'lumotlarni ko'rsatilgan jadval oxiriga qo'shish. Bunda OK tugmasini bosishdan oldin buyruq yonidagi darchadan joriy MB jadvalini tanlab olish lozim. Shuningdek, import qilinayotgan va joriy jadval maydonlari soni, nomi va turi mos kelishi zarur.
- **Создать связанную таблицу для связи с источником данных** – boshqa manbadagi jadvalga bog'liq bo'lgan jadval yaratish. Bu holda tashqi jadval o'zgartirilganda avtomatik ravishda joriy MBdagi jadval ham o'zgaradi. Ammo MBdagi bog'liq jadvalni o'zgartirib bo'lmaydi.



49-tasm. Внешние данные – Excel oynasi

Import qilingan jadvallarning ustun turlari va parametrlarini ixtiyoriy vaqtda o'zgartirish va ma'lumotlarini tahrirlash mumkin.

MBdagi qator bo'yicha bog'liq jadvallarni ularning kalit maydonlari

yordamida bir-biriga bog'lab qo'yish lozim. Buning uchun **Работа с базами данных** menyusining **Схема данных** buyrug'i bosiladi. Hosil bo'lgan **Конструктор** menyusining **Отобразить таблицу** buyrug'i orqali ochilgan oynadan kerakli jadvallar **Добавить** buyrug'i yordamida tashlab olinadi. Jadvallarning kalit maydonlari sichqonchani surish yo'li bilan bog'lanadi (50-rasm).



50-rasm. Схема данных oynasi

Access da forma yaratish

MBga ma'lumotlarni kiritish, o'zgartirish, ularni ko'rish, so'rovlar tashkil qilish va umuman MB interfeysini yaratish maqsadida foydalanuvchilarga qulay bo'lgan elektron shablonlar tuziladi. Bunday shablonlar **formalar** deb ataladi. Formalar ma'lumotlar manbalari, masalan jadvallar yoki so'rovlar bilan bevosita bog'langan bo'ladi.

Accessda formalar bilan quyidagi rejimlarda ishlash mumkin:

- **Форма.** Formaning foydalanish uchun mo'ljallangan ishchi ko'rinishi;
- **Макет.** Formani sozlashning umumiy ko'rinishi bo'lib, bunda ma'lumotlar real aks etib turadi. Bu esa forma elementlarini ma'lumotlarga moslashtirgan holda sozlash imkonini beradi;
- **Конструктор.** Formani sozlashning batafsil ko'rinishi bo'lib, forma va undagi elementlar ko'rinishi va xossalarini ixtiyoriy tarzda o'zgartirish, yangi elementlar joylashtirish va boshqa **Макет** rejimida mumkin bo'lmagan o'zgartirishlarni amalga oshirishga imkon beradi.

Ochilgan formani boshqa rejimga o'tkazish uchun **Главная** menyusining **Режим** buyrug'i bosiladi va yuqoridagilardan biri tanlanadi.

Odatda formalar turli maqsadlarda tuziladi. Kerakli formani yaratish uchun **Создание** menyusining **Формы** bo'limidagi quyidagi buyruqlardan biri tanlanadi:

- **Форма.** Jadval yoki so'rov ma'lumotlari bilan ishlovchi formalar yaratishga mo'ljallangan bo'lib, kerakli manba tanlangandagina buyruq aktivlashadi. Buyruq bosilganda **Макет** rejimida tanlangan manbaning barcha ma'lumotlari aks etgan forma hosil bo'ladi. Yaratilgan formadan **Форма** rejimida to'g'ridan-to'g'ri foydalanish yoki zaruriyat

bo'lsa, uni **Макет** yoki **Конструктор** rejimida o'zgartirish mumkin.

- **Конструктор форм.** Formani to'liq mustaqil yaratishga mo'ljallangan bo'lib, buyruq bosilganda **Конструктор** rejimida boshqa obyektlar bilan bog'lanmagan bo'sh forma hosil bo'ladi. Unga kerakli komponentlarni mustaqil ravishda joylashtirish, formani manba bilan bog'lash va boshqa forma bilan bog'liq ixtiyoriy amallarni bajarish mumkin.
- **Пустая форма.** Bunda ham yuqoridagi kabi formani mustaqil ravishda yaratish mumkin. Buyruq bosilganda **Макет** rejimida bo'sh forma hosil bo'ladi. Shuningdek, o'ng tomonda **Список полей** paneli paydo bo'lib, undagi jadvallardan kerakli maydonlarni formaga joylashtirish imkoniyati mavjud.
- **Мастер форм.** Master yordamida jadval va so'rovlar asosida forma yaratish. Bu usul tanlanganda oyna hosil bo'lib, undagi **Таблицы и запросы** darchasidan jadval yoki so'rov tanlab olinadi (51-rasm).



51-rasm. **Мастер форм** oynasi






Darchaning pastki qismidagi **Доступные поля** sohasidan kerakli maydonlar strelkalar yordamida **Выбранные поля** maydoniga o'tkaziladi. Bunda bog'langan bir nechta jadval yoki so'rovlar maydonlaridan foydalanish mumkin. **Далее** tugmasi bosilganda hosil bo'lgan oynada quyidagi strukturalar taklif etiladi:

- **в один столбец** – jadvalning bitta qatoridagi ma'lumotlar ustun bo'ylab ketma-ket joylashgan maydonlarda hosil bo'ladi;
- **ленточный** – barcha ma'lumotlar ustun ko'rinishda ketma-ket joylashadi;
- **табличный** – barcha ma'lumotlar odatdagi ikki o'lchovli jadval ko'rinishida joylashadi;
- **выровненный** – maydonlarning ko'rinishi va turiga qarab tartiblashtirilgan forma tuzish.

Kerakli struktura tanlanib, Далее tugmasi bosiladi. Navbatdagi oynada formaga nom kiritilib, Готово tugmasi bosiladi.

Yaratilgan forma Конструктор yoki Макет rejimiga o'tkazilganda formani sozlashga mo'ljallangan Конструктор, Упорядочить, Формат nomli qo'shimcha menyular hosil bo'ladi.

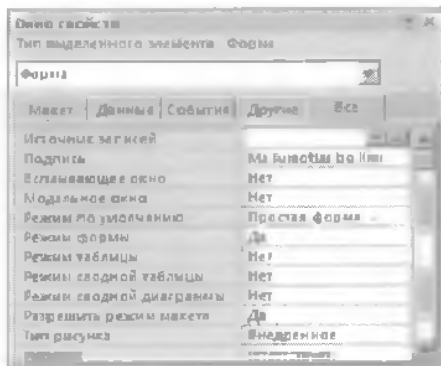
Конструктор menyusidagi ba'zi komponentlar va ularning vazifalari:

- **Поле** (*maydon* - ) – jadval qiymatlarini aks ettirish;
 - **Надпись** (*yozuv* - ) – ma'lumot yozish;
 - **Кнопка** (*tugma* - ) – buyruqqa giper murojaat o'rnatish;
 - **Поле с списком** (*ro'yxatli maydon* - ) – ma'lumotni ro'yxatdan tanlash;
 - **Рисунок** (*rasm* - ) – fayldagi rasmni joylashtirish;
- Barcha komponentlar quyidagi turlardan biriga tegishli bo'ladi:
- **bog'langan komponentlar** – MB manbasidagi ma'lumotlarni aks ettirishga yoki manbaga ma'lumotlar kiritishga xizmat qiladi;
 - **erkin komponentlar** – MB interfeysini hosil qilish, kerakli ma'lumotlarni ekranda aks ettirishda ishlatiladi;
 - **hisoblanuvchi komponentlar** – ma'lumotlari hisoblash natijasida hosil bo'ladigan komponentlar tushuniladi.

Конструктор menyusini buyruqlari:

- **Эмблема** – formaga emblema qo'yish;
- **Заголовок** – formaga sarlavha qo'yish;
- **Дата и время** – joriy sana va vaqtni ifodalash;
- **Добавить поля** – jadvaldagi mavjud maydonni joylashtirish;
- **Формат** menyusini buyruqlari;
- **Фоновый рисунок** – fon sifatida rasm joylashtirish;
- **Заливка фигуры** – komponent fon uchun rang tanlash;

Forma va uning barcha elementlari xossalari maxsus *Xossalar oynasida* (Окно свойств) aniqlab boriladi. Formadagi ixtiyoriy obyekt Макет yoki Конструктор rejimida tanlanib, Конструктор menyusining Страница свойств buyrug'i (F4) bosilganda obyektning to'liq xossalari, masalan, nomi, qiymati, joylashish holati, bajaradigan vazifasi kabilar aks etgan *Xossalar oynasi* hosil bo'ladi (52-rasm). Oynada joriy komponentning ixtiyoriy xossasini o'zgartirish mumkin.



52-rasm. Xossalar oynasi

Jadval kabi formalarni ham boshqa MBlardan import qilib olish yoki eksport qilish mumkin.

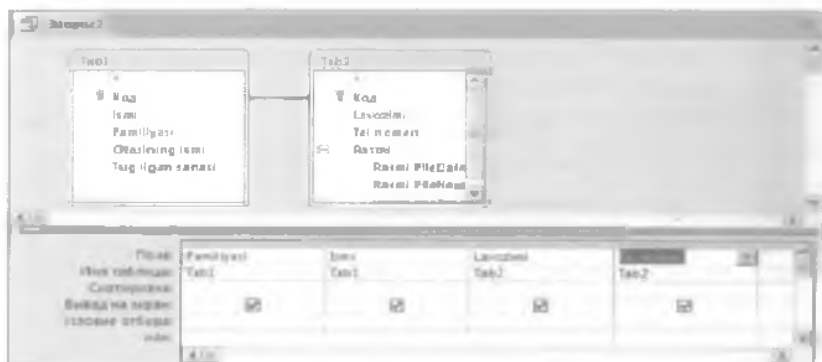
Accessda so'rov tuzish

So'rovlar ma'lumotlar bazasidagi ma'lumotlarni ko'rish, ma'lumotlar qo'shish, o'zgartirish va o'chirishda katta qulayliklar yaratadi. So'rovlar shuningdek, ma'lumotlarni filtrlashda, hisoblash jarayonlarini amalga oshirishda, hisobotlar tashkil qilishda hamda ma'lumotlarni boshqarish ishlarini avtomatlashtirishda keng foydalaniladi.

So'rovlarning ba'zi turlarini ko'rib chiqamiz.

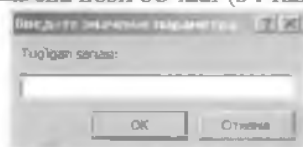
➤ **Tanlangan maydonlarni chiqarish.** Ba'zan jadval yoki bog'langan jadvallarning kerakli maydonlarinigina ko'rishga to'g'ri keladi. Bunday so'rovni yaratish uchun **Создание** menyusining **Запросы** bo'limidagi **Конструктор запросов** buyrug'i bosiladi. Hosil bo'lgan oynadan kerakli jadvallar **Добавить** buyrug'i yordamida so'rov oynasiga tashlanadi. Jadvallardagi kerakli maydonlar sichqoncha chap tugmasini ikki marta bosish orqali **Поле** maydoniga joylashtiriladi (53-rasm).

Сортировка bo'limidan maydonlarni o'sish yoki kamayish tartibida chiqarish, **Вывод на экран** bo'limidan maydonlarning ekranga chiqish yoki chiqmaslik sharti tanlanadi. **Конструктор** yordamchi menyusining **Выполнить** buyrug'i yordamida so'rovni ishlatib ko'rish mumkin. Tayyorlangan so'rov oynasini yopib unga nom beriladi.



53-rasm. So'rov oynasi

- **Tantlangan maydonlarni shartlarga ko'ra chiqarish.** Ba'zi hollarda jadvallarning kerakli maydonlaridagi barcha qator ma'lumotlari emas, balki shartni qanoatlantiradiganlari kerak bo'ladi. Bu kabi so'rov ham yuqoridagidek yaratiladi. Ammo bunda maydonlarning **Условие отбора** darchasiga kerakli shartlar yoziladi. Masalan, *tug'ilgan sanasi* maydoni shartiga *Like [Tug'ilgan sanasi:]* protsedurasi kiritilsa, so'rov ishga tushirilganda darcha hosil bo'ladi (54-rasm).



54-rasm. So'rov darchasi

Bunda kerakli sana kiritilib, OK tugmasi bosiladi. Natijada kiritilgan sanada tug'ilganlar ma'lumotlari hosil bo'ladi. Darchada masalan, * begisi kiritilsa, barcha qatorlar, 02* kiritilsa, 02 bilan boshlangan sana, *7 kiritilsa, 7 raqami bilan tugagan sana, *5* kiritilsa, 5 raqami qatnashgan sanada tug'ilganlar ma'lumotlari hosil bo'ladi.

- So'rovlarning so'rovnoma orqali taqdim etilishi foydalanuvchilarga juda qulaydir. So'rovnoma yaratishni bir misol yordamida ko'rib chiqamiz. So'rovlar tashkil qilish uchun avvalo so'rovnoma tuziladi. Buning uchun konstruktor rejimida forma yaratamiz va unga izlanadigan ma'lumotlar turlari uchun maydonlar va yozuvlar joylashtirib chiqamiz. Maydonlarning yozuvlariga maydon nomlarini, yozuvlarga yulduzcha (*) belgisini kiritib chiqamiz (55-rasm).

Маълумотларни қўйиш	
Ишти	Свободный
Фамилияси	Свободный
Ташаббус	Свободный
Тарихи	Свободный
Дароми	Свободный
Телефон номери	Свободный

55-расм. Излaш формаси лoйиhasи

Yozuvlar maydonlardagi qiymatlarni qabul qiladi va so'rovda maydonlar qiymati o'rinda ishlatiladi. Yozuvlardagi yulduzchalar forma rejimida ko'rinishligi uchun yozuvlarning xossalar oynasidan **Вывод на экран** darchasidagi qiymatlar **Нет** qiymatga aylantirib chiqiladi.

Biror maydon ustida sichqoncha o'ng tugmasini bosib, hosil bo'lgan kontekst menyudan **Обработка событий** buyrug'i bosiladi. Hosil bo'lgan oynadan **Программы** bandi tanlanib, **OK** tugmasi bosiladi. Natijada Access ning **Visual Basic** dasturlash muhiti ochiladi.

Bu yerda **Поле** maydoni uchun avtomatik yozilgan bo'sh protsedurani o'zgartiramiz. Avvalo **Поле** uchun hodisa darchasidan **change** hodisasini tanlaymiz.

Protseduraga quyidagi dasturni kiritamiz:

```
Private Sub Поле0_Change()
```

```
Надпись13.Caption = Поле0.Text
```

```
If Надпись13.Caption = "" Then Надпись13.Caption = ""
```

```
End Sub
```

Bunda protseduraning dastlabki qatorida izlash formasida **Поле0** yoniga joylashtirilgan **Надпись13** yozuviga **Поле0** maydonidagi ma'lumot o'zlashtirilmoqda. Keyingi qatorda agar **Надпись13** bo'sh qiymat qabul qilgan bo'lsa, fil'trlash olib tashlanadi. Bu protsedurani boshqa maydon va yozuvlar uchun ham yozib chiqamiz va **Visual Basic** oynasini hamda izlash formasini yoramiz.

Создание menyusining **Запросы** bo'limidagi **Конструктор запросов** buyrug'ini bosamiz. Hosil bo'lgan oynadan kerakli jadvallarni **Добавить** buyrug'i yordamida so'rov oynasiga tashlaymiz.

Jadvallardagi kerakli maydonlarni sichqoncha chap tugmasini ikki marta bosish orqali **Поле** maydoniga joylashtiramiz va so'rov tuzilayotgan maydonlarning **Условие отбора** darchasiga kerakli shartlarni yozib chiqamiz. Masalan, **Иsmi** maydoni uchun: `Like(forms)![Излaш]![Надпись13]`.

[Caption]+'*' Or Like[forms]![Izlash]![Надпись 13].[Caption]+'*'.
So'rov oynasini yopib, izlash formasini ochamiz. Konstruktor rejimiga o'tib, formaga bitta tugma joylashtiramiz. Avtomatik ochilgan oynadan **Разное – Выполнить запрос** bandini tanlab, **Далее** tugmasini bosamiz. Ochilgan navbatdagi oynalardan yaratilgan so'rov, tugmaga yozuv, nom tanlanib, **Готово** tugmasi bosiladi.

Izlash formasini boshqa formalar bilan bog'lash uchun tugmalar tashlanib gipermurojaatlar qo'yiladi. Shu bilan so'rovnoma tayyor bo'ladi va ma'lumotlarni kerakli parametrlar bo'yicha qidirish mumkin (56-rasm).



56-rasm. Izlash formasi

Accessda ma'lumotlarni izlash, saralash, filtrlash

MBda ma'lumotlar odatda juda ko'p bo'lganligi uchun kerakli ma'lumotlarni qidirish, saralash, biror parametr bo'yicha filtrlash ishlarini bajarishga to'g'ri keladi.

➤ **Ma'lumotlarni izlash.** Buning uchun Главная menyusining **Найти** buyrug'i bosiladi. Natijada Поиск и замена oynasining Поиск paneli ochiladi (57-rasm).



57-rasm. Ma'lumotlarni qidirish oynasi

Bunda panelning **Образец** darchasiga izlanayotgan ma'lumot yoziladi. **Поиск в дarchasidan** qidiruv obyekt uchun joriy maydon (**Текущее поле**) yoki to'liq obyekt (**Текущий документ**) tanlanadi. **Совпадение** darchasidan qidirilayotgan ma'lumot uchun *ixtiyoriy qismiga mos* (**С любой части поля**), *to'liq mos* (**Поля целиком**), *boshlang'ich qismi mos* (**С начала поля**) turlaridan biri tanlanadi. **Просмотр** darchasidan barcha qatorlar bo'yicha (**Все**), oldingi qatorlardan (**Вверх**), keyingi qatorlardan (**Вниз**) izlash kerakligi ko'rsatilib, **Найти Далее** tugmasi bosiladi.

- **Ma'lumotlarni saralash.** Главная menyusining **Сортировка и фильтр** bo'limidagi **По возрастанию** buyrug'i bosilganda joriy maydon ma'lumotlari o'sish tartibida (alifbo asosida) joylashadi. Boshqa maydonlar ma'lumotlari ham mos ravishda o'zgaradi. **По убыванию** buyrug'i ma'lumotlarni kamayish tartibida joylashtiradi. **Удалить сортировку** buyrug'i saralashni bekor qiladi.
- **Ma'lumotlarni filtrlash.** Главная menyusining **Сортировка и фильтр** bo'limidagi **Фильтр** buyrug'i yoki maydon sarlavhasi o'ng qismidagi tugma bosilganda joriy maydon turiga mos buyruq hosil bo'ladi. Bunda sonlar (**Числовые фильтры**), matnlar (**Текстовые фильтры**), sana (**Фильтры дат**) yoki boshqa turlar bo'yicha filtrlash buyruqlari foydalanuvchilarga ma'lumotlarni filtrlashning barcha imkoniyatlarini yaratadi.

Accessda hisobot tayyorlash

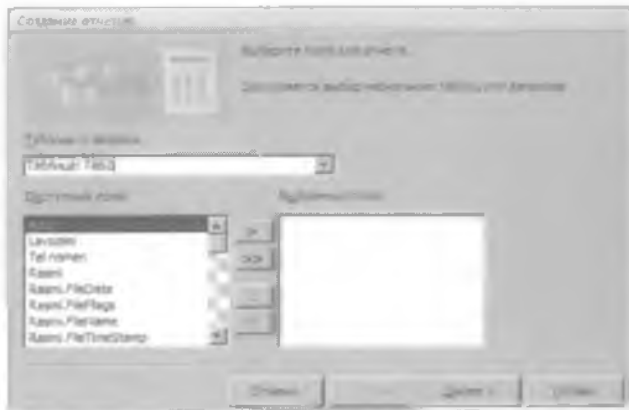
Ma'lumotlar bazasidagi jadval yoki so'rov ma'lumotlarini foydalanuvchilarga tushunarli tarzda yetkazib berish yoki chop etish uchun maxsus hisobotlar yaratiladi.

Hisobot – bu natijalar aks etgan qog'ozli hujjatdir.

Hisobot yaratish uchun **Создание** menyusining **Отчеты** bo'limidagi quyidagi buyruqlardan biri tanlanadi:

Отчет. Obyektlar panelidan biror jadval yoki so'rov tanlanib, **Отчет** buyrug'i tanlanganda mos ravishda jadvaldagi barcha ma'lumotlar yoki so'rov bo'yicha ma'lumotlar avtomatik ravishda hisobot varaqlari ko'rinishida joylashadi. Hisobot yopilganda unga nom beriladi.

Мастер отчетов. Bu usulda hisobotning ko'pgina parametrlarini tanlashga imkoniyat bo'ladi. Buyruq bosilganda bir nech bosqichli hisobot yaratish paneli paydo bo'ladi (58-rasm).



58-rasm. Hisobot yaratish oynasi

Panelning **Таблицы и запросы** bo'limidan kerakli jadval yoki so'rov tanlanib, **Доступные поля** darchasidan kerakli maydonlar strelka yordamida **Выбранные поля** darchasiga o'tkaziladi (bunda bog'langan bir nechta jadvallardan kerakli maydonlarni tanlab olish ham mumkin). **Далее** tugmasi bosilganda hosil bo'lgan navbatdagi oynalardan yozuvlarni saralash parametri, hisobot maketi, varaq shakli (kitob yoki albom shaklida) tanlanib, hisobotga nom kiritiladi va **Готово** tugmasi bosiladi.

Bundan tashqari **Конструктор отчетов** yoki **Пустой отчет** buyruqlari orqali hisobotlarni mustaqil yaratish ham mumkin.

Yaratilgan hisobotlarga konstruktor yoki maket rejimida turli o'zgartirishlar kiritish mumkin.

Компьютер тarmoqlari, Internet

Компьютерлар o'rtasida ma'lumot almashish imkonini beruvchi tizim **kompyuter tarmoqlari** deyiladi. Bunda tizim deganda tarmoqni boshqaruvchi tashkilotlar, kompyuterlar, ulanish simlari, portlari, boshqa kerakli qurilmalar, dasturiy tizimlar majmuasi tushuniladi.

Компьютер tarmoqlarining yaratilishi juda katta imkoniyatlar eshigini ochdi: axborotlar oqimini keskin tezlashtirdi, dunyoning istalgan nuqtasi bilan o'ta tezkor, arzon, qulay ravishda ma'lumotlar almashish imkonini berdi. Doimiy ravishda kerakli xabarlar istalgan joyga yetib bormoqda. Компьютерlarning qimmat resurslaridan, masalan printer, plotter, modem kabi qurilmalardan birgalikda foydalanish, kataloglar, fayllar va dasturiy tizimlarni bir vaqtda yoki istalgan vaqtda birgalikda ishlatish, boshqarish imkoniyatlari paydo bo'ldi.

Tarmoqlar mablag'larning, vaqtning, ishchi kuchlarning minimal tejalishiga olib kelmoqda.

Hozirgi kunda tarmoqlar orqali juda ko'p servis xizmatlari yo'lga qo'yilmoqda. Tarmoqlar shu qadar keng qo'llanilmoqdaki, ularning yo'qligini hattoki tasavvur etish mumkin bo'lmay qoldi.

Kompyuter tarmoqlari lokal va global tarmoqlarga bo'linadi.

Biror institut, korxonalar, tashkilot, kompaniya ichida mavjud kompyuterlar o'rtasida ma'lumot almashish imkonini beruvchi tizim *lokal tarmoq* deyiladi. Lokal tarmoq yaratishdan asosiy maqsad ofis ishlarini avtomatlashtirish, korxonalar boshqaruv sistemalari, loyihalar, texnologik jarayonlar, bank va axborot sistemalarini boshqarish, kompyuter resurslaridan birgalikda foydalanish, ma'lumot almashish va boshqalar hisoblanadi.

Global kompyuter tarmog'i deb, ma'lum bir qit'a yoki dunyo miqyosidagi kompyuterlar o'rtasida ma'lumot almashish imkonini beruvchi tizimga aytiladi.

Hozirgi kungacha dunyo bo'yicha ko'plab global kompyuter tarmoqlari yaratilgan. Ularning ba'zilari quyidagilar:

ARPANET (Advanced research projects agency network) (1969 yilda yaratilgan) – AQShning mudofaa vazirligi tomonidan tashkil qilingan eng mashhur eski tarmoqlardan biri.

BITNET (Because it's time network) (1981 yil) – Shimoliy Amerika va Yevropa qit'asi mamlakatlarini birlashtiruvchi tarmoq.

CSNET (Computer and science network) (1981 yil) – kompyuter va fan tarmog'i bo'lib, butun dunyo bo'yicha amal qiladi.

EARN (European academic research network) – Yevropa ilmiy tadqiqot muassasalari o'rtasidagi aloqani ta'minlovchi tarmoq.

EUNET (Europe union network) – Yevropa kompyuterlar tarmog'i uyushmasi.

USENET (United states education network) (1979 yil) – AQShning barcha oliy ta'lim muassasalarini birlashtiruvchi tarmoq.

NSFNET – (National science foundation network) – AQShning milliy ilmiy fondi tarmog'i bo'lib, AQShdagi mingdan ortiq ilmiy-tadqiqot institutlarini, korporatsiya va hukumat idoralarini birlashtiradi. Dunyodagi eng yirik superkompyuterga ulangan bo'lib, murakkab masalalarni yechishda undan foydalanish imkonini beradi.

INTERNET (International network) – xalqaro kompyuter tarmog'i bo'lib, butun dunyo kompyuterlari o'rtasida ma'lumot almashish imkonini beradi.

INTERNET tarmog'i ARPANET tarmog'i rivojlantirilib, unga CSNET, NSFNET kabi ko'plab tarmoqlar birlashtirilishi natijasida

vujudga kelgan.

Internet dunyodagi eng katta va keng qamrovli global kompyuter tarmog'i bo'lib, 200 dan ortiq mamlakatlarga kirib borgan. 150 dan ko'proq mamlakatlarda keng miqyosda ishlatiladi va 150 milliondan ortiq abonentlarga ega.

Internet tarmog'ining asosiy yacheykalari bu kompyuterlar va ularni bog'lovchi lokal tarmoqlardir.

Internet *texnik, dasturiy ta'minot va informatsion* qismlardan tashkil topgan.

Internetning texnik qismi kompyuterlar, ulanish simlari, telefon tarmog'i, modem, Switch, Hub kabi qurilmalar majmuasidan tashkil topgan bo'lib, ulardan ixtiyoriysining vaqtinchalik ishdan chiqishi internet tarmog'ining umumiy faoliyatiga aslo ta'sir etmaydi.

Internetning dasturiy ta'minoti tarmoqqa ulangan turli kompyuterlar va tarmoq vositalarini bir-biriga mantiqiy bog'lash, ma'lumotlarni ma'lum bir standart ko'rinishda foydalanuvchilarga yetkazib berish, axborotlarni qidirish, ularni saqlab turish, tarmoqda axborot xavfsizligini ta'minlash vazifalarini bajaradi.

Internetning informatsion qismi tarmoqqa joylashtirilgan turli matnli, sonli, grafik, rasm, audio, video va boshqa ko'rinishidagi ma'lumotlardir.

Kompyuterni internetga ulash uchun telefon liniyasi, ichki yoki tashqi modem, qaydnomalar dastur ta'minoti bo'lishi lozim va yaqinroq bo'lgan provayderlarga murojaat qilinadi.

Provayderlar bu kompyuterlarning internet tarmog'i bilan aloqasini ta'minlovchi va ularga turli internet xizmatlarini ko'rsatuvchi tashkilotlardir. Hozirda **Uzpak, Sarkor, Uznet, Buston** provayderlari keng turdagi internet xizmatlarini ko'rsatmoqda.

Internet tarmog'iga tashkilot yoki muassasalar kompyuterlarini ulash uchun qo'shimcha ravishda koaksial yoki optik totali simlar, xab (Hub), modem va Svich (Switch) qurilmalari, qo'shimcha dasturiy tizimlar kerak bo'ladi.

Internet xizmatlari, Internet protokollari

Hozirgi kunda Internet tarmog'ida inson faoliyatining ixtiyoriy sohasiga oid deyarli barcha ma'lumotlarni topish mumkin. Borgan sari internetning axborot va xizmat ko'lamini ortib bormoqda. Bu yerda har bir sohaga oid oddiy ma'lumotlardan tortib, fan, texnika, madaniyat, sport, san'at, adabiyot va boshqa juda ko'p sohalarga oid qimmatli ma'lumotlarni olish mumkin. Internet tarmog'i dastlab ma'lumotlarni uzatish, qabul qilish va

lardagi kitoblar elektron shaklda ham mavjud va bu kutubxonalar internet tarmog'iga ulangan. Elektron kutubxonalar haqida www.vlibrary.freenet.uz sahifasidan ma'lumot olish mumkin. Hozirgi kunda elektron kutubxonalar son-sanoqsiz. Oliy ta'lim muassasalari, litsey, kollej va maktablar, ilmiy-tadqiqot muassasalari, ba'zi tashkilotlar ham o'zlarining elektron kutubxonalariga ega.

Ba'zi elektron kutubxonalar:

www.lcweb.loc.gov – Kongress kutubxonasining elektron ko'rinishi bo'lib, dunyoning eng yirik virtual kutubxonalaridan biridir. Kongress kutubxonasida turli sohaga doir 115 milliondan ortiq kitob va hujjatlar saqlanadi.

www.vlib.org – Virtual kutubxonasi iqtisod, biznes, fan-texnika, ilmiy va boshqa sohalardagi kitoblar bilan to'ldirilgan eng katta kutubxonalardan biri.

www.zivonet.uz – O'zbekistondagi oliy o'quv yurtlari, akademik litsey, kollej talabalari va maktab o'quvchilari uchun xizmat qiluvchi markazlashtirilgan eng katta elektron kutubxonalardan biri.

www.uch2.tashjit.uz – ToshTYMI elektron kutubxona tizimi manzili.

Internetning ba'zi tushunchalari:

Router – internetda ma'lumotlar oqimini qulay va yaqin yo'l bilan manzilga yetkazishni ta'minlovchi dasturlar majmuasidir.

Trafik – internetdagi ma'lumotlar oqimining hajmi.

Proxy serverlar – internetda ko'pchilik murojaat qiladigan serverlar ma'lumotlari nusxalari saqlanadigan qo'shimcha serverlar.

Brauzerlar (browsers) – HTML tilida tayyorlangan Web sahifalarni foydalanuvchiga tushunarli ko'rinishda tasvirlab berish uchun yaratilgan maxsus dasturlar. Bunday dasturlar qatoriga Internet Explorer, Netscape Navigator, Opera, Safari, Firefox brauzerlarini kiritish mumkin.

URL (Uniform resource locator) – internet sahifalari elektron manzili va unga kirish usulining to'liq nomi. Masalan, TTYMI web-sayti *URL* adresi quyidagicha: <http://www.tashjit.uz>. Bunda: <http://www> – *URL* texnologiyasi, [tashjit](http://www.tashjit.uz) – tashkilot nomi, [uz](http://www.tashjit.uz) – domen nomi.

Elektron manzillarning domenlariga misollar:

edu (education) – ta'lim muassasalari;

com (commercial) – tijorat muassasalari;

org (organization) – tashkilotlar;

gov (government) – hukumat muassasalari;

net (network) – telekommunikatsion muassasalar;

int (international) – xalqaro muassasalar;

Geografik joylashuvlar bo'yicha domenlar:

uz – O'zbekiston;
us – AQSh;
ru – Rossiya;
uk – Buyuk Britaniya;
ca – Kanada;
de – Germaniya.

Internet qaydnomalari (protokollari).

TCP/IP (Transmission control protocol/Internet protocol) – internet tarmog'ida uzatish qaydnomalari majmuasining nomi. TCP/IP tarmoqning dasturlar qismi bo'lib, undagi har bir qism aniq bir vazifani bajaradi.

TCP/IP tarkibiga kiruvchi ba'zi bir qaydnomalar:

TCP – ma'lumotlar uzatilishini qo'llab-quvvatlovchi qaydnoma.

IP – ma'lumotlar almashinuvini ta'minlaydi.

RIP (Routing information protocol) – ma'lumotlarni kerakli manzilga yetkazish uchun eng maqbul yo'lni tanlovchi qaydnoma;

ARP (Address resolution protocol) – tarmoqdagi kompyuterning sonli manzilini aniqlaydi.

DNS (Domain name system) – tarmoqdagi kompyuterning domen manzilini aniqlaydi.

FTP (File transfer protocol) – fayllar uzatilishini ta'minlaydi.

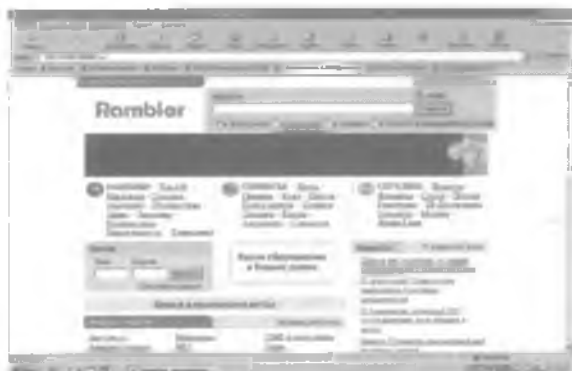
NFS (Network information service) – lokal tarmoqlarda mavjud bo'lgan katalog va fayllardan foydalanishni ta'minlaydi.

SMTP (Simple mail transfer protocol) va POP (Post Office Protocol) – elektron pochta ma'lumotlarining uzatilishini ta'minlovchi qaydnomalar.

Internet Explorer brauzeri, elektron pochta

Internet Explorer 1995 yildan ishlab chiqarila boshlangan bo'lib, dunyodagi ko'pchilik abonentlar mazkur brauzerdan foydalanishadi. Odatda u *Пуск-Все программы-Internet Explorer* menyular ketma-ketligi orqali ishga tushiriladi.

Natijada Internet Explorer brauzeri oynasi hosil bo'ladi va bunda uy sahifasi ham avtomatik yuklanadi (59-rasm).



59-rasm. Internet Explorer brauzeri oynasi

Internet Explorer brauzerining ba'zi buyruqlari quyidagilar:

Файл menyusi:

Новое окно(Ctrl+N) – yangi oyna ochish;

Открыть(Ctrl+O) – kerakli saytni ochish;

Сохранить(Ctrl+S) – joriy sahifasini saqlash.

Печать(Ctrl+P) – joriy sahifasini chop qilish.

Правка menyusi:

Копировать(Ctrl+S) – sahifaning belgilangan qismidan nusxa olish;

Выделить все(Ctrl+A) – sahifani to'liq belgilash;

Найти(Ctrl+F) – sahifadan kerakli ma'lumotni qidirish;

Вид menyusi:

Панели обозревателя → **Избранное** (Ctrl+Shift+I) – tanlangan sahifalarni ko'rish;

Панели обозревателя → **Журнал** (Ctrl+Shift+H) – hafta davomida va undan oldingi ochilgan barcha sahifalar va fayllar ro'yxatini ko'rish va ularni ochish;

Переход → **Назад**(Alt+←) – oldingi ochilgan sahifaga o'tish;

Переход → **Вперед**(Alt+→) – keyingi ochilgan sahifaga o'tish;

Переход → **Домашняя страница** (Alt+Home) – uy sahifasiga o'tish;

Остановить (Esc) – qidiruvni to'xtatish;

Обновить (F5) – qidiruvni yangilash;

Источник – sahifaning HTML tilidagi kodini ko'rish;

Сервис menyusi:

Удалить журнал обозревателя – jumaldan keraksiz fayl va sahifalarni o'chirish;

Свойства обозревателя – uy sahifasini o'rnatish, internetga ulanish, brauzerining ishlash parametrlarini o'zgartirish kabi ko'plab ishlarni

bajarish mumkin.

Odatda Internet Explorer brauzeri ishga tushirilganda qidiruv tizimlaridan biri ham avtomatik yuklanadi. Hozirgi kunda juda ko'p qidiruv tizimlari mavjud. Masalan, rus tilida muloqot qila oladigan abonentlar uchun www.rambler.ru, www.mail.ru, www.vandex.ru, www.google.ru, www.aport.ru, ingliz tilida muloqot qila oladigan abonentlar uchun www.yahoo.com, www.altavista.com, www.hotbot.com, www.britanisa.com, o'zbek tilida muloqot qila oladigan abonentlar uchun www.google.uz qidiruv tizimlari xizmat ko'rsatadi. Bu tizimlarning har biri qulay interfeyslarga ega bo'lib, ma'lumotlarni qidirishning turli usullarini taklif etadi. Ko'pgina qidiruv tizimlari ma'lumotlarni qidirishdan tashqari boshqa saytlarga kirish, ixtiyoriy sohadagi Web-sahifalarni ochish, elektron pochta bilan ishlash kabi ko'plab qulayliklarni yaratadi.

Elektron pochta (E-mail) – internet tarmog'ining asosiy xizmat turaridan biri bo'lib, ma'lumotlarni masofadan turib elektron ko'rinishda uzatish usulidir. Hozirgi kunda 120 milliondan ortiq elektron pochta abonentlari mavjud. Elektron pochta oddiy pochta tizimining avtomatlashtirilgan ko'rinishidir. Uning oddiy pochtaga nisbatan o'ta tezkor ekanligi asosiy ahamiyatga ega. Bundan tashqari oddiy pochta ishlaydigan ko'plab kishilarning ishini elektron pochta bir necha kishi boshqarib turishi hisobiga kattagina ishchi kuchlari va mablag'lar tejab qolinmoqda. Shu o'rinda ta'kidlab o'tish kerakki, elektron pochta oddiy pochta tizimining yo'qolib ketishiga olib kelmaydi. Elektron pochta oddiy pochta yana bir farqli jihati shuki, bunda nafaqat matnli ma'lumotlarni, balki ixtiyoriy ko'rinishdagi OLE turidagi va katta hajmdagi axborotlarni jo'natish imkoniyati ham mavjud.

Elektron pochta bilan ishlaydigan ko'plab dasturlar mavjud. *MS Exchange*, *MS Mail*, *MS Outlook Express* dasturlari shular jumlasidandir. Bu dasturlar elektron pochta bilan ishlash uchun juda qulay bo'lib, ajoyib interfeysga ega. Masalan, manzillar kitobi, jo'natilgan va qabul qilingan xatlar ro'yxati kerakli parametrlari bilan birgalikda tartiblangan holatda bo'ladi. Bu dasturlar orqali Internet Explorer brauzeri bo'lmaganda ham elektron pochta foydalanish mumkin.

Odatda ko'pchilik bevosita Internet qidiruv tizimlaridan biridagi elektron pochta tizimiga murojaat qilishadi. Masalan, www.rambler.ru, www.mail.ru, www.vandex.ru, www.yahoo.com qidiruv tizimlari qulay elektron pochta muhitlarini taqdim etadi.

Elektron pochta abonentlari o'zlarining maxsus adreslariga ega bo'lishi lozim. Bu adreslarning har biri dunyo bo'yicha takrorlanmas va original ko'rinishda bo'ladi. Adreslar maxsus dasturlar yordamida tekshirib boriladi

hamda bir xil tuzilishga ega manzillar yaratilishiga yo'l qo'yilmaydi.

Elektron pochta adreslarining umumiy tuzilishi quyidagicha:

<nom>@<tashkilot>.<domen>

Elektron pochta manzili yaratish uchun qidiruv tizimlaridan birortasiga kirib, E-mail bo'limidagi **Registratsiya** buyrug'i bosiladi. Natijada elektron pochta abonentining kerakli ma'lumotlarini kiritish uchun so'rovnoma paydo bo'ladi. Bu so'rovnomada yangi abonentning familiyasi, ismi, login, parol, manzili kabi ma'lumotlar kiritilib, Registratsiya buyrug'i bosiladi. Natijada elektron pochta muvaffaqiyatli yaratilganligi yoki ma'lumotlar kiritishda xatolikka yo'l qo'yilganligi to'g'risida xabar paydo bo'ladi.

Elektron pochta muhitiga kirish uchun qidiruv tizimidagi E-mail bo'limidan login va parol kiritilib, OK tugmasi bosiladi. Hosil bo'lgan elektron pochta interfeysida quyidagi bo'limlar eng ko'p ishlatiladi:

Входящие – qabul qilingan xatlar;

Отправленные – jo'natilgan xatlar;

Написать письмо – yangi xat yozish va jo'natish.

Yangi kelgan xatlarni ko'rish uchun **Входящие** buyrug'i bosiladi. Natijada **Прочитанные** va **Непрочитанные** bo'limlari paydo bo'ladi. **Непрочитанные** buyrug'ini bosib, ro'yxatdagi xatlarni tanlash yordamida ochib ko'rish mumkin.

Yangi xat yozish va uni jo'natish uchun **Написать письмо** buyrug'i bosiladi. Bunda xat yozish uchun qulay interfeys paydo bo'ladi. Oynaning **Кому** darchasiga jo'natilayotgan adres yoziladi. **Прекрепить** buyrug'i yordamida xatga ixtiyoriy fayllarni ilova qilish mumkin. Xat tayyor bo'lgandan so'ng **Отправить письмо** buyrug'i bosiladi. Agar xat manzilga yetib borgan bo'lsa, **Письмо успешно отправлено** xabari paydo bo'ladi. Pochta bilan ishlab bo'lgandan so'ng har doim **Выход** buyrug'i yordamida elektron pochta interfeysidan chiqish lozim.

WWW, HTML till operatorlari

WWW (World wide web – butunjahon o'rgimchak to'ri) tizimi internet tarmog'ining eng asosiy xizmat turlaridan biridir. WWW tizimi 1989-1991 yillarda yaratilgan bo'lib, oddiy foydalanuvchilar uchun ham tarmoqdan samarali foydalanish imkoniyatlarini yaratadi. Bunda maxsus HTTP (Hyper text transfer protocol – gipermatnlarni uzatish protokoli) deb nomlangan protokol yaratilgan bo'lib, TCP/IP protokoli tarkibiga kiritildi.

WWW tizimi barcha ma'lumotlarning foydalanuvchilar uchun Web-sahifalar ko'rinishida ekranga chiqishini ta'minlaydi. Bu tizimda ma'lumotlar **gipermatnlar** ko'rinishida namoyon bo'ladi. Hozirgi kunga kelib

internet tarmog'ida ma'lumotlarning mul'timedia turlari ortib bormoqda. Ya'ni, matnli ma'lumotlardan tashqari rasm, grafik, animatsion obyekt, audio hamda video ma'lumotlar paydo bo'lmoqda. Gipermatnli ma'lumotlar tizimi mul'timedia tizimlari bilan takomillashtirilib, *gipermedia* tizimini hosil qiladi. Web tizimida ma'lumotlar brauzerlar tomonidan o'qiladi. Brauzerlarga joylashtirilgan gipermedia ma'lumotlari *Web sahifalar* deb ataladi. Biror sohaga oid Web sahifalar majmuasi *saytlar* deb ataladi. Foydalanuvchilarga turli interaktiv xizmatlar ko'rsatuvchi yirik saytlar *portallar* deyiladi. Saytlar va portallar maxsus kerakli qurilmalar va dasturlar bilan jihozlangan kompyuterlar – *Web serverlarga* joylashtiriladi. Provyayderlar Web serverlarning uzluksiz ishlashini ta'minlab turadi.

Saytlar Front Page, PHP, Dreamweaver kabi dasturlash vositalari orqali yoki bloknot yordamida sof HTML tilida yaratiladi. HTML (Hyper text markup language – gipermatnni belgilash tili) ma'lumotlarni gipermatn va gipermedia ko'rinishlarida tayyorlashga mo'ljallangan maxsus tildir.

Bloknot yordamida HTML tilida sayt tuzish uchun *Пуск – Все программы – Стандартные – Блокнот* menyular ketma-ketligi bosiladi. Natijada bloknot matn muharriri hosil bo'ladi. Bu fayl **Файл** menyusining **Сохранить как** buyrug'i yordamida .htm yoki .html kengaytma bilan saqlanadi. Bu holda fayl saqlangan joyda brauzer yordamida ochiladigan Web sahifa fayli hosil bo'ladi. Web sahifa ochilib, **Вид** menyusining **Источник** yoki **В виде HTML** buyrug'i bosiladi. Natijada dastlabki yaratilgan bloknot Web sahifaga bog'langan holda ochiladi. Bu yerda HTML tilida kerakli buyruqlar va ma'lumotlar kiritiladi. Bajarilayotgan amallarning natijalarini ko'rib borish uchun **Файл** menyusining **Сохранить** (Strl+S) buyrug'i bosilib, brauzer oynasiga o'tiladi hamda **Вид** menyusining **Обновить**(F5) buyrug'i bosiladi.

HTML tilida yoziladigan dastur strukturasi quyidagicha bo'ladi:

```
<html>
```

```
<head>
```

```
Sarlavba satriga oid operatorlar va ma'lumotlar
```

```
< /head >
```

```
<body>
```

Dasturning tana qismi bo'lib, sahifa ko'rinishi uchun kerakli operatorlar va ma'lumotlar joylashtiriladi.

```
</body>
```

```
</html>
```

Bunda < (ochilgan "teg") va > (yopilgan "teg") belgilari buyruqlarni (operatorlarni) ajratib turish uchun ishlatiladi. Bu buyruqlar toki shu buyruq teg ichida "/" (slesh) belgisi yordamida yopilmaguncha amal qiladi.

Teglar har doim juft (ochilgan va yopilgan) holatda bo'ladi hamda operatorlarning bosh va kichik harflarda yozilishi farqlanmaydi.

HTML tilining ba'zi operatorlari:

`<title> ... </title>` – sarlavha satridagi yozuvni aniqlaydi.

Masalan, `<title>Bosh sahifa</title>`

Dastur tana qismida dastlab ochilgan `<body>` tegi ichida sahifa foni (bgcolor) va unda yoziladigan ma'lumotlar rangi (text) parametrlarini ko'rsatib o'tish mumkin. Masalan, `<body bgcolor=silver text=red>`.

Bu parametrlar yozilmasa, sahifa oq, yozuv qora rangda bo'ladi.

Bunda sahifa foni sifatida biror rasmni o'rnatish ham mumkin. Masalan, `<body background="D:\Photo\1-rasm.jpg">`.

`<h1> ... </h1>` – eng katta o'lchamli shriftdagi sarlavhani aniqlaydi. Bundan tashqari `<h2>`, `<h3>`, ..., `<h6>` kattaliklardagi shriftlar ham mavjud bo'lib, shrift o'lchamlari mos ravishda kichiklashib boradi. Bu teglarda tartiblash (align) parametrini ko'rsatish ham mumkin.

Masalan, `<h1 align="center"> TTYMI </h1>`

Align parametrining center (markazga) qiymatidan tashqari left (chap tomondan), right (o'ng tomondan), justify (ikki tomondan ham tekislash) qiymatlari ham mavjud.

`<p>...</p>` – matnni alohida abzatsdan yozishni ta'minlaydi. Bunda abzatsni tartiblash parametrini kiritish ham mumkin.

Masalan, `<p align=justify> Matn </p>`.

`
` – kursor pozitsiyasini yangi qatorga o'tkazadi.

`<hr>` – gorizontal chiziq qo'yadi. Bunda teg ichida chiziq rangi (color), qalinligi (size), uzunligi (width), tartiblanishi (align) parametrlarini ko'rsatib o'tish mumkin.

Masalan, `<hr color= green size=2 width=80% align=center >`

Bu misolda chiziq sahifa kengligining har doim 80 foiz qismini egallaydi. Sahifa kengligini aniq piksel hisobida berish ham mumkin.

Masalan, `<hr width=800>`.

`<i> ... </i>` – matnni kursiv ko'rinishda yozadi.

` ... ` – matnni qalin shriftda yozadi.

`<u> ... </u>` – tagiga chizilgan matnni aniqlaydi.

`...` – teg orasiga yozilgan matn shrifti, o'lchami, rangi, turini aniqlaydi.

Masalan, `` Bu matn 7 kattalikdagi shriftda, yashil, Arial shriftda yozilgan

`<center> ... </center>` – qator markazidan yozishni ta'minlaydi.

`_{...}` – quyi indeksga yozishni bildiradi.

` ` – tegsiz yoziladigan bu buyruq bo'sh joy tashlanishini bildiradi.

`<input> ... </input>` – ma'lumotlarni kiritish uchun darcha ochadi.

`<marquee> ... </marquee>` – yuruvchi matnni aniqlaydi. Bunda ekranga nisbatan foiz hisobida yoki piksellar hisobida yuruvchi matn sohasi o'ldhamini ko'rsatish mumkin.

Masalan, `<marquee width=50% height=10> Saytimizga hush kelibsiz! </marquee>`.

`` – sahifaga rasm joylashtirishni ta'minlaydi. Rasm sifatida .gif, .jpg, .jpeg, .bmp kengaytmali fayllardan foydalanish mumkin.

Masalan, ``.

Bunda rasming kengligi (width), balandligi (height), ramka qalinligi (border), tartiblanishi parametrlarini ko'rsatish ham mumkin.

Masalan, ``

`<a>...` – sahifaga giper murojaat o'rnatish. Bunda web-sahifada teg o'rtaida yozilgan matn yoki obyekt bosilganda ko'rsatilgan ixtiyoriy web-sahifa ochiladi yoki faylga mumkin bo'lgan tarzda murojaat qilinadi.

Masalan,

` 2-sahifaga kirish`;

` Informatika `

` ToshTYMI sayti `

`<bg sound>` – sahifa joriy holatida .mid yoki .wav kengaytmali audio tovushlarni namoyish etish imkonini beradi.

Masalan, `<bg sound src="D:\music\m1.wav">`

`<table> ... </table>` – jadvalni chizishni ta'minlaydi. Bunda `<table>` tegi ichida jadval chegaralari qalinligi (border), kengligi (width), balandligi (height), tartiblanishi (align), yacheykalar orasidagi masofa (cellpadding), yacheyka chegarasidan ichidagi yozuv gacha bo'lgan masofa (cellspacing) parametrlarini kiritish mumkin.

Masalan, `<table border=2 width=80% height=30% align=center cellpadding=5 cellspacing=10> ... </table>`.

Jadval kengligi va balandligini piksel hisobida berish ham mumkin.

Masalan, `<table width=600 height =400> ... </table>`.

`<caption> ... </caption>` – jadval sarlavhasini aniqlaydi.

`<tr> ... </tr>` – jadvaldagi bitta satrni aniqlaydi. Yopiluvchi `</tr>` tegi zarur emas. Bunda navbatdagi `<tr>` tegi keyingi qatorni bildiradi.

`<th>` – jadval ustuni sarlavhasini aniqlaydi. Bu teg `<tr> ... </tr>` teglari orasida kelishi zarur.

`<td> ... </td>` – jadval qatoridagi bitta yacheykani aniqlaydi. Yopiluvchi `</td>` tegi zarur emas. Bunda navbatdagi `<td>` tegi keyingi yacheykani bildiradi.

Yacheyka o'Ichamini jadval o'Ichamini buzgan holda o'zgartirish mumkin. Masalan, `<td width=150 height=50>`.

Yacheyka fonini ochiluvchi `<td>` tegida ko'rsatish mumkin.

Masalan, `<td bgcolor=yellow> 1-yacheyka </td>`.

Agar ustundagi bir nechta yacheykani birlashtirish lozim bo'lsa, `<td>` tegi ichida `rowspan` parametrini qo'llash lozim.

Masalan, `<td rowspan=3>yacheyka1</td>`.

Agar satrdagi bir nechta yacheykani birlashtirish lozim bo'lsa, `<td>` tegi ichida `colspan` parametrini qo'llash lozim.

Masalan, `<td colspan=4>yacheyka2</td>`.

Misol. Bizga quyidagi jadval berilgan bo'lsin.

Yacheyka1		Yacheyka3	Yacheyka4	Yacheyka5
	Yacheyka2	Yacheyka6		
Yacheyka7		Yacheyka8	o'ngga tartiblash	

Bu jadvalni ifodalash uchun dastur tanasida quyidagilarni kiritish lozim bo'ladi:

```
<table border="1">
  <tr>
    <td rowspan=2>Yacheyka1</td>
    <td rowspan="3">Yacheyka2</td>
    <td>Yacheyka3</td> <td>Yacheyka4</td>
    <td rowspan="2">Yacheyka5</td>
  </tr>
  <tr>
    <td colspan=2>Yacheyka6</td>
  </tr>
  <tr>
    <td>Yacheyka7</td> <td>Yacheyka8</td>
    <td colspan=2 align=right>o'ngga tartiblash</td>
  </tr>
</table>
```

Bu misoldan ko'rinib turibdiki, `<tr>` tegi bilan keyingi qatorga o'tilganda `<td>` tegi yangi qatordagi o'zlashtirilmagan yacheykani aniqlaydi.

Brauzer oynasida bir nechta Web-sahifalar joylashtirish lozim bo'lsa, u holda freymlardan foydalanadi. Freymlar dastur tanasi o'roida ishlatilib, `<frameset>` tegi bilan boshlanadi. Teg ichida freymlarning sonini, o'Ichamini ko'rsatish lozim.

Masalan, `<frameset rows="60%, 40%">` yozilsa, bu teg ekranning ikki

qatorli freymga bo'linishini, 1-freym ekranning 60 foizini, 2-freym 40 foizini egallashini anglatadi.

Masalan, `<frameset cols="30%, 50%,20%">` yozilsa, bu teg ekranning uch ustunli freymga ajratilishini, 1-freym ekranning 30 foizini, 2-freym 50 foizini, 3-freym 20 foizini egallashini anglatadi.

Masalan, `<frameset cols="150, *, 100">` yozilsa, bu teg ekranning uch ustunli freymga ajratilishini, 1-freym 150 piksel kenglikda, 3-freym 100 piksel kenglikda, 2-freym ekranning o'rtadagi qolgan qismini egallashini anglatadi.

`<frameset>` yordamida freymlar strukturasi yozilgandan so'ng `<frame>` tegidan foydalanib, freymning har biri uchun joriy freymda ochiladigan sahifa yoki fayl nomi ko'rsatib o'tiladi.

Masalan, `<frame src="f2.htm"> <frame src="f3.htm">`.

`<frameset>` yordamida ochilgan freymlar strukturasi `</frameset>` tegi yordamida yopilishi lozim.

Umuman olganda `<frameset> ... </frameset>` teglari ichida yana `<frameset>` strukturasi qo'llash mumkin.

Foydalanilgan adabiyotlar

1. Степанов А.Н. Информатика: Учебник для вузов. 4-изд. – СПб.: Питер, 2006. 684 с.
2. Романова Ю.Д. и др. Информатика и информационные технологии: Учебное пособие. –М. Эксмо, 2008. 592 с.
3. Фаронов В.В. Delphi 2005. Язык, среда, разработка приложений. СПб. Питер. 2005г.
4. Кандзюба С.П., Громов В.Н. Delphi 5. База данных и приложения. изд. Диасофт. Киев. 2001 г. 592 с.
5. Гарнаев А.Ю. Использование MS EXCEL и VBA в экономике и финансах. Санк-Петербург.1999 й.
6. Арипов М.М., ва б. Информатика Ахборот технологиялари. 2-кисм. Тошкент,2003. 425бет.
7. Арипов М.М.,Мухаммадиев Ж.У. Информатика ва Информацион технологиялар. Тошкент. 2004й. 278 бет.
8. Холматов Т.Х., Тайлоков Н.И., Назаров У.А. Информатика. Тошкент. 2003й. 256 бет.
9. Shodimetov X.M., Xolboev O'.N. Axborot texnologiyalari. Uslubiy ko'rsatma. T. 2007.
10. Shodimetov X.M., Xolboev O'.N. Excel elektron jadvali va unda masalalar yechish. Uslubiy ko'rsatma. Toshkent. 2009 y. 80 b.
11. Eshdavlatov B., Abduqayumov B., Xolboyev O'.N. Informatika. Uslubiy ko'rsatma. Toshkent 2011 y.
12. Xodjayeva N.Sh., Xolboyev O'.N., Qodirova Ye.V. Delphi muhitida dasturlash. Uslubiy ko'rsatma. Toshkent. 2006 y.
13. www.ph4s.ru sayti.
14. www.wikipedia.org sayti.
15. www.ziyonet.uz sayti.

Informatika asoslari, EHMLar va shaxsiy kompyuterlar.....	3
Kompyuterning dasturiy ta'minoti	10
Windows operatsion tizimi	11
Sanoq sistemalari.....	14
Algoritmlar.....	18
Takrorlanuvchi algoritmlarga doir misollar	21
Delphi dasturlash tili va muhiti, Delphida konsol ilova, strukturasi, elementlari, alifbosi va nomlar	23
Turlar, o'zgarmaslar, o'zgaruvchilar, standart funksiyalar.....	26
Operatorlar, o'zlashtirish operatori, tarkibiy va bo'sh operator, ma'lumotlarni kiritish va chiqarish protseduralari	29
Shartli operator, o'tish operatori	34
Tanlash operatori.....	37
Parametrli takrorlanish operatori	39
Sharti oldin va sharti keyin beriladigan takrorlanish operatorlari.....	40
Massivlar, bir o'lchovli massivlar.....	45
Ikki o'lchovli massivlar	49
Dinamik massivlar.....	50
Protседura va funksiyalar	54
Funksiya qism dasturi.....	56
Fayllar bilan ishlash.....	56
Delphi ning vizual imkoniyatlari, Delphi dasturlash muhiti.....	60
Sinflar va komponentlar	63
Label, Edit, Memo matnli komponentlari, Button tugmasi	67
Label, Edit, Memo va Button komponentlari yordamida dasturlar tuzish	70
Grafik-tasvirlar yaratishga mo'ljallangan sinf va buyruqlar.....	72
Image, Shape, Paintbox, Timer komponentlari	75
Image va Timer komponentlari yordamida dasturlar tuzish	77

Chiziqli algebraik tenglamalar sistemasini yechishning Gauss usuli	79
Eng sodda taqribiy integrallash formulalari	85
Differensial tenglamalarni taqribiy yechish usullari	93
Axborot texnologiyalari	99
Microsoft Word matn muharriri, dastur interfeysi	99
PowerPoint dasturi, slaydlar va taqdimotlar	107
PowerPoint dasturida taqdimot yaratish	108
Excel elektron jadvali, Excel interfeysi	114
Ma'lumotlarni kiritish va tahrir qilish	118
Excelda formula va funksiyalar	121
Adreslar	123
Excelda diagrammalar qurish va ularni tahrirlash	128
MBBT, Access dasturi, Access interfeysi	130
Accessda jadval tuzish	133
Access da forma yaratish	137
Accessda so'rov tuzish	140
Accessda ma'lumotlarni izlash, saralash, filtrlash	143
Accessda hisobot tayyorlash	144
Kompyuter tarmoqlari, Internet	145
Internet xizmatlari, Internet protokollari	147
Internet Explorer brauzeri, elektron pochta	150
WWW, HTML tili operatorlari	153
Foydalanilgan adabiyotlar	159

Xolmatvoy Maxkamboevich Shodimetov,

Botir Niyazkulovich Abdukayumov,

O'tkir Nurullaevich Xolboev

INFORMATIKA VA AXBOROT TEXNOLOGIYALARI

O'quv qo'llanma

Muharrir: Yunusova Z.K.

Texnik muharrir va sahifalovchi: Tashbaeva M.X.

Nashrga ruxsat etildi 17.12.2012 y.

Qog'oz bichimi 60×84/16. Hajmi 10 b.t.

Adadi 50 nusxa. Buyurtma №6-2/2012

ToshTYMI bosmaxonasida chop etildi

Toshkent sh., Odilxo'jaev ko'chasi, 1 uy

Toshkent temir yo'l muhandislari instituti, 2012y.