



M.U.Ashurov, N.D.Mirzaxmedova

Turbo Pascal

dasturlash tili

O'ZBEKISTON RESPUBLIKASI
OLYI VA O'RTA MAXSUS TA'LIM VAZIRLIGI

NIZOMIY NOMIDAGI
TOSHKENT DAVLAT PEDAGOGIKA UNIVERSITETI

M.U.Ashurov, N.D.Mirzaxmedova

TURBO PASCAL DASTURLASH TILI



Ushbu metodik qo'llanmada dasturlash tillaridan biri Turbo Pascal dasturlash tilini imkoniyatlari haqida yoritib berilgan. Metodik qo'llanma bakalavriat ta'lim yunalishidagi talabalar va akademik litsey, kasb – hunar kollejlari o'quvchilari uchun mo'ljallangan bo'lib, qolaversa undan barcha qiziquvchilar ham foydalanishi mumkin.

Tuzuvchilar:

M.U.Ashurov

– Nizomiy nomli TDPUning “Informatika va TAT” kafedrası katta o'qituvchisi.

N.D.Mirzaxmedova

– Nizomiy nomli TDPUning “Informatika va TAT” kafedrası o'qituvchisi.

Taqrizchilar:

f-m.f.n. R. R.Boqiev

- Nizomiy nomli TDPUning “Informatika va TAT” kafedrası dotsenti.

p.f.d. F.M.Zokirova

- Toshkent axborot texnologiya universiteti, Kasb Ta'limi fakulteti dekani.

Metodik qo'llanma Nizomiy nomidagi TDPU Ilmiy kengashining 2011 yil
dagi _____-sonli qarori bilan nashrga tavsiya etilgan.

Kirish

Ma'lumki, kompyuter texnikasidan samarali foydalanish ikki qismning-texnik va dasturiy ta'minotining uzviyligi talab etiladi. Bu uzviylik kompyuter texnik ta'minotining jadal sur'atlar bilan takomillashib borishga mos dasturiy ta'minotni ham keskin sur'atlar bilan rivojlanishga sabab bo'ladi, va aksincha. Buning sababi ma'lum, mos dasturiy ta'minotsiz har qanday kompyuter "qimmatbaho o'yinchoq" bo'lib qoladi.

Ma'lumki, kompyuterda biror masalani hal qilish uchun avval uning modeli va algoritmi tuziladi. so'ng maskur algoritmlar ma'lum bir qonun-qoydalar asosida kompyuter tushunadigan ko'rsatma va buyruqlar shaklida yoziladi. Hosil bo'lgan matn kompyuter tilida yozilgan dastur deb ataladi. Demak, *dastur*-biror masalani echish uchun kompyuter bajarishi mumkin bo'lgan ko'rsatmalarning izchil tarkibi ekan.

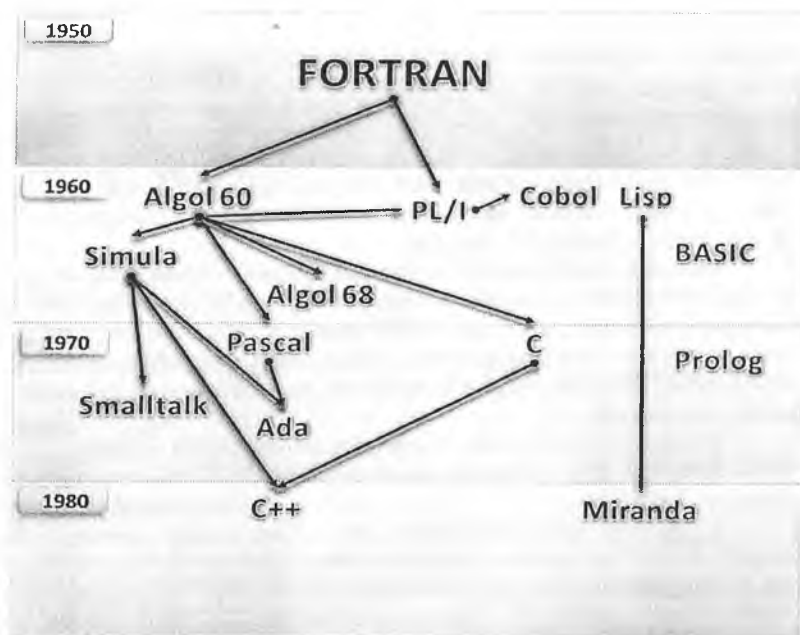
Kompyuter uchun dastur tuzish jarayoni *dasturlash* va dastur tuzadigan shaxs *dasturchi* deyiladi. Kompyuter tushinadigan "til" esa *dasturlash tili* deb ataladi.

Dasturlash tillari tarixidan: Fan va texnikaning keyingi yillardagi yutuqlarini juda qup hollarda dasturlash bilan bog'liq deb hisoblashadilar. Ammo bu to'g'ri emas. Birinchi va tuzilishi jihatidan murakkab bo'lgan ko'p maqsadli hisoblash quрилmalari XIX asrda yaratilgan bo'lib, ularga dastur tuzish muammosiga birinchi bo'lib Ada Lavleys (u CHARlz Bebbidjning «Difference Engine» hisoblash mashinasida ishlar edi) duch kelgan.

1949 yili birinchi dasturlash tili Short Code yaratilgan. Elektron hisoblash mashinalarining dastlabki avlodlarida dastur tuzuvchilar dasturlarni mashina kodlarida yaratishar edi. Tasavvur etish uchun «Hello, word!» kabi matnni chiqarish uchun yaratilgan dasturning bir qismini keltiramiz va siz uning naqadar dasturchi uchun noqulay ekanini anglab etasiz.

Mashina kodlari keyingi yillarda dasturlash olamiga kirib keluvchilar (elektronika sohasidagi nomutaxassislar) uchun noqulay bo'lgan. Ammo tez orada foydalanuvchila uchun tushunarli bo'lgan yuqori darajali dasturlash tillari yaratilib, dasturlash bilan shug'ullanuvchidar safi kengayishiga sabab bo'ldi.

Birinchi yuqori darajali dasturlash tillariga 20 asrning 50-yillarida bo'lib yaratilgan FORTRAN, Cobol va Algollarni keltirish mumkin. Bu tillar shu kunlarda ham «yashab kelmoqdalar» va keyingi yillarda yaratilgan minglab dasturlash tillarining avlodlaridan hisoblanishadi. Quyidagi keltirilgan sxemada dasturlash tillarining qisqa tarixini keltiramiz:



Dasturlash tillari ichida 1958 yilda yaratilgan Algol tili muhim o'rinni egallaydi. Uning yaratuvchilardan biri Djon Bekus FORTRAN tilining asoschilaridan biri bo'lgan. ALGORitmic Language nomi bu tilning algoritmlarni yozish uchun mo'ljallanganligidan dalolat beradi. Juda ham aniq mantiqiy tuzilishga ega bo'lgan Algol tili ilmiy va texnikaviy adabiyotlarda algoritmlarni yozish uchun qo'llanila boshlandi. Keyingi yillarda uning Algol 60 va Algol 68 versiyalari yaratildi. Algol 68 foydalanish uchun noqulay holda yaratilgan bo'lib shveysariyalik olim Niklaus Virt uning qabul qilinishiga o'zining noroziligini bildirdi SHu bilan birga bu versiya qator dasturlash tillarining yaratilishiga sababchi bo'ldi. 1967 yili Niklaus Virt Algol W nomli o'zining versiyasini yaratdi. Ammo muallifga shu kungacha mavjud bo'lgan dasturlash tillaridagi noqulayliklar yoqmas (jumladan, talabalarni dasturlar yaratish metodlari bilan tanishtirishda dasturlash tillarining imkoniyatlaridan foydalanish) va u 1968 yili o'zining dasturlash tilini yaratishga kirishdi. 1970 yili dasturlash olamida ikki olamshumul hodisa yuz berdi: birinchisi Unix operatsion tizimining yaratilishi bo'lsa, ikkinchisi Pascal dasturlash tilining yaratilishi. Niklaus Virt bu tilni XVII asrning buyuk faylasuf va matematigi Blez Paskal sharafiga Pascal dasturlash tili deb atadi. Dastlab, dasturlash tilining to'liq versiyasi SDC6000 kompyuteri uchun yaratilgan. O'zining aniqligi, mantiqiyliigi va boshqa qator xususiyatlari bilan dasturlashni o'rgatishdagi imkoniyatlari bilan bu dasturlash tili o'z o'imini topa oldi. 1975 yili Bill Geys va Pol Allen BASIC dasturlash tili versiyasini, Virt hamda Yensen esa «Pascal User Manual and Report» tilini yaratishdi. Keyingi yillarda bu dasturlash tillarining turli versiyalari

yaratila boshlandi. Jumladan, kompyuter texnologiyalari sohasidagi muhim hodisalaridan biriga aylangan 1983 yilda Borland firmasi yordamida Filip Kan tomonidan yaratilgan Turbo Pascal dasturlash tili shulardan biridir. Shu vaqtgacha bu tilning qator versiyalari ko'zga tashlandi (1992 yili, Borland 7.0). Jahondagi mashhur korporatsiyalardan biri bo'lgan Borland o'zining qator dasturlash tillari turli operatsion tizimi uchun yaratgan va ulardan ayrimlari keltiramiz:

Operatsion tizimlar	Dasturlash tillari
Ms-DOS	Turbo Pascal, Turbo S, Turbo Assembler
Windows	Delphi, C++ Builder, JBuilder,
Linux	Kylix

Dasturlash tili-matnlarni yozish qoidalari tizimidan iborat bo'lib, kompilyator tomonidan bu matnlar aniq instruktsiyalar va kattaliklar to'plami sifatida taqdim etilishi zarur. Barcha dasturlash tillari o'zlarining mashina tiliga bog'liqlik darajasi bilan baholanadilar. Mashina tiliga bog'liqlik tillarda instruktsiya va operandlar maxsus simvulli nomlar bilan ifodalanadilar. Masalan, Intel protsessori uchun assembler tilida yozilgan dasturdan lavha keltiramiz:


```
Mov cx, 100
Les bx, AOB
Dec bx
@@@Test : inc bx
          Cmp Byte PTR [bx],0
          Loopne @Test
```

Mashina tiliga bog'liqlik tillarning afzalligi yuqori darajadagi dasturlarni yaratishda bo'lib, bu dasturlar xotirada minimal hajmi egallab, hisoblashlarni maksimal tezlikda bajaradilar. Ammo bu tillar yordamida murakkab dasturlarni yaratish nihoyatda noqulay.

1-§. Turbo Paskal 7.0 integrallashgan muhiti.

Turbo Pascal 7.0 dasturlash tilini kompyuterga yuklagan vaqtda biz ekranda bir necha vazifalarni bajaruvchi integrallashgan muhitga ko'zimiz tushadi.

Muhitning birinchi satri – sarlavha satri deb yuritiladi



va odatda shu satrdan so'ng menyular qatori joylashadi. Menyular qatorida joylashgan



har bir ichki menyuda qator vazifalarni bajaruvchi buyruqlar joylashgan. Undan pastda asosiy ish maydoni va xolat satri joylashgan. Shuningdek, muxitning asosiy elementlaridan biri bu oynalar hisoblanadi.


Turbo Paskalning professional programmalash - 7 versiyasi (Turbo Pascal Professional) Borland International Ins firmasining bir nechta disketalarida beriladi. Shu disketalarning biri «INSTALLG'COMPILER» deb nomlanadi va bu disketada INSTALL.EXE programmasi mavjud. Turbo Paskalni muhitini o'rnatish uchun yuqorida ta'kidlangan disketani diskovodga qo'yib, programmani ishga tushiriladi. Programma Sizdan bir necha savollarga javob berishni taklif qiladi:

- qaysi diskovoddan Turbo Paskalni ishga tushirmoqchisiz?
- Turbo Paskalni vinchesterga o'rnatasizmi?
- qaysi kataloglarda Turbo Paskalning ishchi fayllarini joylashtirish lozim?

Agar sizda etarli asos bo'lmasa, taklif qilingan katalog ismlari bilan chegaralangan ma'qul. Bu holda sizdan faqatgina, disk ismini o'zgartirish talab etiladi. Shundan so'ng, INSTALL.EXE programmasi o'zining ishini davom ettiradi va uning so'rovi bo'yicha navbat bilan, ko'rsatilgan nomli disketalarni diskovodga kuyiladi. Natijada Turbo Paskal ishga tayyor holga keladi. Faraz qilaylik, Turbo-Paskalni urnatishda odatdagi «S» diskning o'rniga «D» diskni tanladingiz, bu holda sistema quyidagi kataloglarda joylashdi: D TP TURBO D TP TVDEMOS D TP TVIZION D TP UTILS Turbo Paskal dasturi qattiq diskda joylashgan TP\Bin\TPX.exe (Turbo.exe) fayli orqali ishga tushiriladi.

Turbo Paskal oynasi ikki xil holatda ishlaydi:

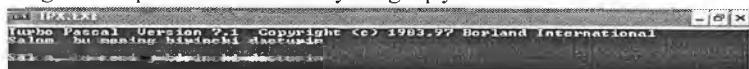
1. Turbo Paskal muhiti oynasining umumiy ko'rinishi.



```
TPX.EXE
File Edit Search Run Compile Debug Tools Options Window Help
NONAME00.PAS
Program M_1:
const
  st='Salom, bu mening birinchi dasturim';
begin
  writeln(st);
  readln;
end.
```

Matn muhariri oynasi dastur matnini kiritish va matnni tekshirish uchun birqancha qulayliklar yaratadi. Turbo.exe fayli ishga tushirilish bilan ekranda Turbo Paskal matn muhariri oynasi paydo bo'ladi.

2. Dastur natijasini ko'rish oynasi. Dastur natijasini olgandan keyin <Enter> tugmasi orqali matn muhariri oynasiga qaytib o'tish mumkin.



Turbo Pascal dasturida asosiy fayllar va funksional tugmalar

Turbo Paskalning asosiy fayllari $D : \backslash TP \backslash$ katalogida joylashgan bo'lib, ular quyidagilardir:

Turbo.exe - programmani yaratish uchun lozim bo'lgan integrallashgan muhit (Turbo Pascal Integrated Development Environment) fayli;

Turbo.hlp - tezkor yordam ma'lumotlarini saqlovchi fayl;

Turbo.tp - Turbo.exe dasturi foydalanadigan sistema konfiguratsiyasining fayli;

Turbo.tpl - Turbo Paskalning rezident modullari (Resident units);

Tptour.exe - integrallashgan muhitda ishlashni tanishtiruvchi dastur. Bulardan tashqari, $D : \backslash TP \backslash BGI$ katalogida Paskalning grafik rejim ishini ta'minlovchi fayllar mavjud:

Graph.tpu - Turbo Paskalning barcha grafik dasturlarini ishlashi uchun zarur bo'lgan modul;

BGI kengaytmali bir nechta fayl-videosistemalarning turli tiplari bilan ishlashni ta'minlovchi drayverlar;

CHR kengaytmali bir nechta fayl - vektorli shriftlarni o'z ichiga oluvchi fayllar.

Turbo Paskal muhitini boshqarish uchun funksional tugmalardan (<F1>, <F2>, ..., <F10>) foydalaniladi. Quyida funksional tugmalar keltirilgan:

<F1> - Turbo Paskal dasturi yordam xizmatidan foydalanish. (Help - Yordam);

<F2> - Dastur matnini faylga saqlash;

<F3> - Faylni ochish;

<F4> - Dasturni ishlashini tekshirish jarayonida qo'llaniladi. Dastur matnidagi kursor o'rnatildan satrgacha bo'lgan satrlarni bajarilishini satrma-satr tekshiradi;

<F5> - Faol oynani butun ekran bo'ylab yoyish;

<F6> - Keyingi tahrirlanuvchi oynaga o'tish;

<F7> - Dasturni ishlashini tekshirish jarayonida qo'llaniladi. Agar dasturda protsedura (funksiya) ga murojaat bo'lsa, protsedura (funksiya) ga kirib, birinchi operatordan oldin to'xtaydi.

<F8> - Dasturni ishlashini tekshirish jarayonida qo'llaniladi. Dastur matnini sarta-satr bajarilishini tekshiradi. Agar dasturda protsedura (funksiya) ga murojaat bo'lsa, protsedura (funksiya) ga kirmasdan dastur ishlashini tekshiradi;

<F9> - Dasturni Compilatsiya qiladi. lekin ishlashini ta'minlamaydi.

<F10> - Bosh menyu bo'limlari bilan ishlash;

<CTRL> <F9> - dasturni Compilatsiya qiladi, ishlashini ta'minlaydi va Turbo Paskal muhitiga qaytaradi;

<ALT>+<F5> - dastur matni oynasidan dastur natijalarni ko'rish oynasiga o'tish;

<ALT>+<X> - Turbo Paskal muhitidan chiqish;

Dastur matnini kiritishda boshqarish tugmalaridan foydalanish.

Dastur matnini kiritish jarayonida bir qancha boshqarish tugmalaridan foydalaniladi:

<Enter> - yangi (keyingi) satrga kursorni o'tkazish;

<PageUp> - bir sahifa yuqoriga;

<PageDown> - bir sahifa pastga;

<Home> - satr boshiga kursorni o'tkazish;

<End> - satr oxiriga kursorni o'tkazish;

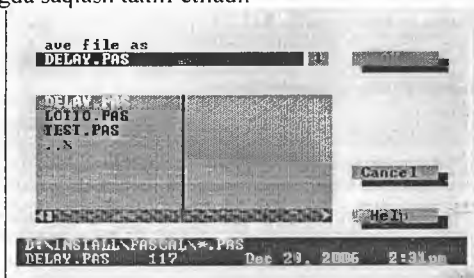
<CTRL>+<PageUp> - matn boshiga o'tish;

<CTRL>+<PageDown> - matn oxiriga o'tish;

<Delete> - kursor joylashgan belgini o'chiradi;

<Backspace> - kursordan chapda joylashdan belgini o'chiradi;

Turbo Paskal muhitida dastur mantini kiritib, natijalar olingandan so'ng, ushbu dasturdan keyinchlik foydalanish uchun dasturni hotirada faylga saqlash zarur bo'ladi. <F2> tugmasi orqali yangi faylni joylashish manzili, nomi ko'rsatiladi va <Enter> tugmasi bosiladi. Diskda saqlangan fayl *.pas kengaytmasini oladi. <F3> tugmasi orqali diskdagi avvaldan mavjud *.pas fayllarni ochish (yuklash) mumkin. Dasturni saqlashda TP katalogidagi tizimli fayllar bilan sizning fayllaringiz aralashib ketmasligi zarur, aks holda siz uchun kerakmas deb hisoblandan fayl Turbo Paskal dasturining tizimli fayli bo'lishi ehtimoli bor. SHuning uchun o'zingiz yaratgan fayllarni TP katalogidan boshqa katalogda saqlash taklif etiladi.



Turbo Paskal muhiti menyulari.

Fayl menyusi. Fayllar bilan ishlash imkonini beradi: New - Yangi fayl yaratish; Open - diskda saqlangan faylni ochish muloqotli oynasini ochadi. Ushbu amalni - F3 - tugmasi orqali bajarish mumkin; Save - Joriy taxrirlanayotgan dastur matnini faylga saqlash. Ushbu amalni <F2> tugmasi orqali bajarish mumkin; Save As - Joriy taxrirlanayotgan dastur matnini yangi nom va yangi manzili bilan saqlash muloqotli oynasini chaqiradi; Print - Dastur matnini chop etish; Exit - Turbo Paskal muhitidan chiqish; Fayl menyusidagi boshqa (Save All, Change dir,

Print Setup, Dos shell) buyruqlardan juda kam foydalanilishi uchun ushbu buyruqlarni o'tkazib yubordik;

Edit menyusi. Dastur matnini tahrirlash uchun bir qancha buyruqlarni taklif qiladi: Undo – Oxirgi amalni bekor qilish (ALT+Backspace); Redo – Bekor qilingan amalni qayta tiklash; Cut - beldilangan dastur qismini “Qirqish” (Shift + Del); Copy - beldilangan dastur qismini nusxalash (CTRL+Ins); Paste – Cut yoki Copy buyruqlari yordamida xotiraga olingan dastur qismini kursor turgan joyga o'rnatish; (Shift+Ins); Clear – Belgilangan matn qismini o'chirish (Shift+Ins); Show clipboard – Bufer xotiraga olingan dastur matni bo'laklarini ko'rish;

Search menyusi. Dastur matnidagi matn bo'laklarini qidirish, hatoliklar mavjud bo'lgan satrga o'tish imkoniyatini beradi: Find – Qidirilayotgan matn qismi va qidirish usulini kiritish uchun muloqotli oyna taklif qiladi; Replace – Biror matn bo'lagini boshqa matn bo'lagi bilan almashtirish uchun muloqotli oyna taklif qiladi; Search again – matn bo'lagini qaytadan qidirish; Goto line number – berilgan tartib raqamli satrda o'tish; Show last compile error – dastur matnidagi hato satrni va hatolik haqidagi ma'lumotni ko'rsatadi; Find error – Dastur matnidagi hato satrni aniqlaydi. Find procedure – dastur matnidagi protsedurani qidirish muloqotli oynasini taklif qiladi. Muloqotli oynada dastur matnida ishtirok etuvchi protsedura nomi kiritiladi.

Run menyusi. Dasturni ishga tushirishni amalga oshiradi. Run – Agar dasturning boshlag'ich matnida o'zgarish kiritilmagan bo'lsa, dasturni ishlashini ta'minlaydi. Agarda dastur matniga o'zgartirish kiritilgan bo'lsa, avval dasturni qayta Compilatsiya qilish zarurligi haqida so'raladi. Dasturni Compilatsiya qilinishini bekor qilish uchun <Ctrl+break> tugmalari bosiladi. Step over - Qism dasturga kirmasdan dasturning asosiy qismini satrma-satr bajarilishini tekshiradi (F8); Trace into – Dasturning asosiy qismida qism dasturga murojaat qilinganda qism dasturni satrma-satr bajarilishini tekshiradi (F7); Goto kursor – Kursor o'rnatilgan satrgacha dasturni bajarilishini ta'minlaydi; Program reset – YUqoridagi tekshirish jarayolarini bekor qiladi. (CTRL+F12);

Compile menyusi. Dastur matni kiritilgandan so'ng, uni Compilyatsiyaga yuborish zarur. Ushbu amalni ikki xil usulda amalga oshiriladi: Compile menyusidan Compile buyrig'ini tanlash; <ALT>+<F9> tugmalarini bosish; Avval Compilyator dastur matnining sintaktik xatolarini tekshiradi. Agar xatolik aniqlansa, Compilyator xatolikka yo'l qo'yigan satrda o'z ishini to'xtatadi va xatolikka mos keluvchi xabarni beradi. Aniqlangan xatoliklarni dastur tuzuvchi to'g'rilaydi va qaytadan Compilyatsiyaga yuboradi. Dastur matni Compilyatsiyadan muvaffaqiyatli o'tgandan so'ng dastur natijasini olib, natijalarni tahlil qilish imkoniyati paydo bo'ladi.

Debug menyusi. Dastur ishini teksirishda to'xtash nuqtalarini o'rnatish, dasturda ishlatiluvchi o'zgaruvchilarni kiritish va mos qiymatlarini ko'rish imkoniyatlarini beradi;

Tools menyusi: Turbo Paskal muhitidan chiqmasdan qo'shimcha uskunalardan foydalanish imkoniyatini beradi:

Options menyusi: Turbo Paskal dasturlash muhitini kerali paramtrlarini o'rnatish uchun buyruqlar taklif qiladi.

Window menyusi: Oynalarni ochish, yopish, faollashtirish va ekran maydoniga joylashtirishga yordam beradi.

Help menyusi. Turbo Paskal dasturi muhiti, operatorlari, funktsiya-protseduralari, kutubxonalari haqida ma'lumot olish (<ALT>+<H>); Index – Yordamchi tizimni tematik ko'rsatkich orqali ko'rsatadi; Contents - Yordamchi tizimni mundarijasini ochadi.

Bugungi kunda kasb-xunar kollejlari va oliy ta'lim muassasalarida Turbo Paskalning 5-7 versiyalari keng tarqalganligini va undan foydalanish qulayligini hisobga olingan holda, quyida Turbo Paskalning shu versiyalari uchun umumiy bo'lgan tomonlarini ko'ramiz. Turbo Paskal muhiti *Turbo.exe* faylidan ishga yuklanib, uning asosiy vazifasi Paskal tilidagi programmani taxrirlash va mashina kodiga o'tkazishdan iboratdir. Turbo Paskal muhitini yuklash natijasida bosh menyu hosil bo'lib, uni aktivlashtirish uchun, *F10* tugmachasidan foydalaniladi.

Bosh menyu quyidagi bo'limlarni o'z ichiga oladi:

File *.pas kengaytmali fayllari va Turbo Paskalga tegishli bulgan kataloglar ustida turli amallar bajaradi;

Edit yuqori menyudan matnlarni taxrirlash oynasiga qaytish;

Run programmani turli ko'rinishda ishga yuklaydi va natijalar oynasiga o'tishni ta'minlaydi;

Compile programmani kompilyasiya kilish va uni tekshirishni ta'minlaydi;

Options Turbo Paskal muhitini va undagi programmalarga tegishli hususiyatlarni taxrirlaydi;

Debug programmadagi proseduralarni izlash, hisoblash jarayonlari va o'zgaruvchilarni tekshirish, chaqirilgan stekni ko'rish va shunga o'xshash amallarni bajaradi;

Break + Watch agarda *Debug+ Integrated Debugging* ustanovkasi On xolatda bo'lsa, u holda *Break + Watch* menyusi *Watch window* (tekshirish oynasi) punktini oynaga o'rnatadi yoki olib tashlaydi; Quyida bosh menyuning mos ostki menyulariga qisqacha ta'riflar berilgan:

FILES BO'LIMI:

Load (F3) fayl yoki dasturni chaqirish uchun xizmat qiladi; *Pisk (ALT + F3)* tanlashni belgilaydi; *New* yangi dastur uchun oynani tozalaydi;

Save (F2) ekrandagi dasturni doimiy xotirada saqlaydi. *Write to* ishchi faylni yangi nom bilan saqlaydi; *Change dir* faol katalogni almashtiradi; *Os shell* vaqtincha OS ga chiqishni tashkil qiladi; *Quit (Alt + X)* TURBO PASCAL muhitidan chiqish.

RUN BO'LIMI:

Run (Ctrl + F9) yuklangan dasturni ishga tushiradi ; *Program reset (Ctrl + F2)*oldindagi parametrlarni taxrirlash; *Goto sursor (F4)* kursor turgan joyga qaytishni amalga oshiradi; *Trase into (F7)* dasturni qadamlab bajarish imkoniyatini hosil qiladi; *Setup over (F8)* proseduraga kirishni ta'minlaydi; *User screen(Alt-F5)*natijalar oynasiga o'tish.

COMPILE BO'LIMI:

Compile (Alt + F9) dasturni kompilyasiya qiladi; *Make* (F9) dasturni kompilyasiya qilishga tayyorlaydi; *Build* dasturni kompilyasiya qilishga tayyorlaydi (*Make* dan farqi, dasturga tegishli barcha fayllarni tekshiradi); *Destination* dasturni doimiy yoki tashqi xotiraga kompilyasiya bo'lishini ta'minlaydi; *Find error* dastur hatoligini izlaydi; *Primary files* qaysi *.pas fayl kompilyasiya qilinishini aniqlaydi; *Get info* ishchi oynadagi *.pas fayli haqidagi ma'lumotlarni beradi.

OPTIONS BO'LIMI:

Comptler dastur hususiyatlarini o'zgartiradi; *Linker* *.exe fayldagi ishlamaydigan kodlarni o'chiradi; *Environment* dasturchi uchun ishchi maydon hosil qiladi; *Direstories* turli tipdagi fayllarni qaysi katalogda ishlatish lozimligini o'zida saqlaydi; *Parameters* buyruqlar satrida kiritiladigan parametrlarni o'zida saqlaydi; *Save options* muxit hususiyatini tashqi xotiraga saqlab qo'yish; *Retrieve options* muhit hususiyatini tiklash.

DEBUG BO'LIMI:

Evaluate har qanday o'zgaruvchining qiymatini ko'rish imkoniyatini yaratadi; *Call stask* dastur bajarilish vaqtida joriy chaqiruvchi stekni ko'rsatadi; *Find procedure* prosedurani izlash; *Integrated debugging* dasturni qadamlab bajarilishi uchun yo'l ochib beradi; *Standalone debugging* dasturni *.exe faylga aylantirish uchun yo'l ochib beradi; *Display swapping* ekran hususiyatini o'rnatadi; *Refresh display* ekran hususiyatini aktivlashtiradi.

BREAK + WATSH BO'LIMI:

Add watsh (Ctrl + F7) ko'rish oynasiga zarur ma'lumotlarni qo'shish imkoniyatini hosil qiladi; *Delete watsh* ko'rish oynasidagi belgilangan punktini o'chiradi; *Edit watsh* ko'rish oynasini taxrirlaydi; *Remove all watshs* ko'rish oynasidagi barcha punktlarni o'chiradi; *Toggle breaskpoint* tekshiriluvchi nuqtalarni o'rnatish; *Clear all breaskpoint* barcha tekshiriluvchi nuqtalarni o'chirish; *View next breaskpoint* keyingi tekshiriluvchi nuqtaga o'tish.

Yuqoridagilarga qo'shimcha ravishda dastur matnini taxrirlash uchun funksional tugmachalardan ham foydalanish imkoniyati mavjud bo'lib, ular quyidagilardan iboratdir:

<i>Ctrl-K-B</i>	blok boshini belgilash;
<i>Ctrl-K-K</i>	blok ohirini belgilash;
<i>Ctrl-K-C</i>	kursor turgan joyga blokdan nusxa olish;
<i>Ctrl-K-V</i>	kursor turgan joyga blokni ko'chirish;
<i>Ctrl-K-Y</i>	Bloka olingan matnni o'chirish;
<i>Ctrl-Y</i>	kursor turgan satrni o'chirish;
<i>Ctrl-K-H</i>	Blokni olib tashlash yoki qo'yish.

Menyular tarkibidagi amallarni ma'lum klavishlarni bosish orqali ham bajarish mumkin. Quyida asosiy amallarni bajarishga mo'ljallangan klavishalar keltirilgan:

Klaviatura tugmalari:	Vazifalari:
F3	Ochish (tashqi hotiradagi faylni yuklash)
F2	Saqlash
Alt+F3	Aktiv oynani yopish
F6	Bir oynadan ikkinchi oynaga o'tish
Ctrl+F9	Dasturni ishga tushirish
Alt+F5	Dastur natijasini ekranda ko'rish
Alt+F9	Dasturni kompilyatsia qilish
Alt+x	chiqish

Yuqoridagilarga qo'shimcha qilgan holda dastur matnini taxrirlash uchun ishlatiladigan klavishlar quyidagilardan iboratdir:

<i>Yo'nalish klavishalari</i> (→ ← ↑ ↓)	YUrgichni kerakli yo'nalishda siljitish;
<i>Shift+(→ ← ↑ ↓)</i>	YUrgich turgan joydan boshlab tanlangan yo'nalishda dastur matnini qismini belgilash;
<i>Ctrl+Insert</i>	dastur matnini belgilangan qismi nusxasini hotiraga olish;
<i>Shift + Insert</i>	Hotirada olingan qismni dastur mantining yurgich turgan joyga o'rnatish;
<i>Shift + Delete</i>	dastur matnining belgilangan qismini o'chirish;

Dasturchilar deyarli menyular satriga murojaat etmasdan, yuqorida keltirilgan klavishlardan foydalanadi. Bu esa vaqtni tejash bilan birga dastur tuzuvchini e'tibori faqat dastur matnida bo'lishini ta'minlaydi.



Savol va topshiriqlar.

1. Install.exe programmasining vazifasini tushuntiring.
2. Turbo-Paskalning asosiy fayllarini yozib bering va ularning vazifalarini ayting.
3. Turbo-Paskalning qaysi versiyasi amalda ko'proq ishlatiladi?
4. Turbo.exe faylining vazifasini tushuntiring.
5. Turbo-Paskal muhitining yuqori menyu bo'limlari vazifasini tushuntirib bering.
6. Files bo'limining ostki menyulari vazifasini ayting.
7. Run bo'limining ostki menyulari vazifasini ayting.
8. Compile bo'limining ostki menyulari vazifasini ayting.
9. Opions bo'limining ostki menyulari vazifasini ayting.
10. Debug bo'limining ostki menyulari vazifasini ayting.

2-§. Turbo-Paskal dasturlash tili va uning alifbosi.

Paskal tili 1969 yili N.Virt tomonidan yaratilib mashhur olim Blez Paskal nomi bilan ataldi. Bu til N.Virtning o'ylashi bo'yicha programmalashning zamonaviy texnologiyasiga va uslubiga, strukturali programmalash nazariyasiga asoslangan va boshqa programmalash tillaridan muayyan yutuqqa ega til bo'lishi lozim edi. Mazkur til:

- Programmalashtirish kontsepsiyasini va strukturasini sistemali va aniq ifodalaydi;
- Programma tuzishni sistemali olib borish imkonini beradi;
- Programma tuzish uchun boy termin va struktura sxemalariga ega;
- Yo'l qo'yilgan xatoliklarni tahlil qilishning yuqori darajadagi sistemasiga ega.

1981 yili Paskal tilining halqaro standarti taklif etildi va IBM PC tipidagi shaxsiy kompyuterlarda Paskal tilining Borland firmasi tomonidan ishlab chiqilgan Turbo-Paskal oiladosh tili keng qo'llanila boshlandi. Hozirda Turbo-Paskalning bir qancha versiyalari yaratilib, yuqori darajadagi programmalar yaratish imkoniyatlari borgan sari kengaytirilib borilmoqda:

- 4.0 versiyasidan boshlab programma yozishni, taxrirlashni va natijalar olishni osonlashtirish uchun yangi integrallashgan muhit hosil qilindi;
- 5.5 versiyasining paydo bo'lishi bilan Turbo-Paskalda ob'ektili programmalash imkoniyati paydo bo'ldi;
- 6.0 versiyasidan boshlab esa Paskal programmasi ichiga quyi programmalash tili bo'lmish Assembler tilida yozilgan programmalarni qo'shish holati hosil qilindi. Shu bilan bir qatorda *tilning integrallashgan muhiti* ham bir qator o'zgarishlarga ega bo'ldi.

Integrallashgan muhit-dasturlashga yordamlashuvchi dastur bo'lib, u quyidagi asosiy vazifalarni bajarishi lozim:

- dastur matnini kiritish imkonini beradi;
- vaqt-vaqti bilan kiritayotgan dastur matnini diskda saqlab turishi;
- dasturni ishga tushirish uchun translyatorga ega bo'lishi;
- sintaktik xatoliklarni aniqlash vositasiga ega bo'lishi kerak.
- 7.0 integrallashgan muhiti sanab o'tilgan vazifalardan tashqari yana bir qancha vazifalarni ham amalga oshirish imkonini beradi.

Turbo- Paskal dasturlash tilining alifbosi

Ma'lumki, har qanday tilni o'rganish uning alfavitini o'rganishdan boshlanadi. Tilning alfaviti - shu tilgagina tegishli bo'lgan asosiy belgilari va tushunchalar to'plamidan iborat bo'ladi. Paskal tilining alfavitini tashkil etuvchi asosiy belgilar jamlamasini 3 guruhga ajratish mumkin: harflar, raqamlar va maxsus belgilar.

Til alfavitining metalingvistik (Bekus - Naur) formulasi quyidagicha bo'ladi:

<asosiy belgi>::=<harf>|<raqam>|<maxsus belgi>

Harf sifatida katta va kichik lotin harflari ishlatiladi. YA'ni 26 ta **lotin alifbosi harflari**: A, a, B, b, Cc, Dd, Ee, Ff, Gg, Hh, Ii, Jj, Kk, Ll, Mm, Nn, Oo, Pp, Qq, Rr, Tt, Uu, Vv, Ww, Xx, Yy, Zz. Lekin, matnlar va programmaga izohlar yozish uchun kirill alifbosining bosh va kichik harflarini ham alfavitga kiritilgan.

Raqaqlar sifatida oddiy **arab raqaqlari** olingan:

<raqam>::=0|1|2|3|4|...|9

Maxsus belgilar ko'p sonli va bir jinssiz bo'lganligi uchun ularni o'z navbatida 4 ta guruhga ajratamiz:

<maxsus belgi>::=<arifmetik amal belgisi>|<solishtirish amali belgisi>|<ajratgich>|<xizmatchi so'z>.

<arifmetik amal belgisi>::= * | / | + | -

Bu amallar mos ravishda ko'paytirish, bo'lish, qo'shish va ayirish belgilari hisoblanadi.

Solishtirish amallarining belgilari, ularning matematik ifodasi va amallarning ma'nosi 1-jadvalda o'z ifodasini topgan. Bu erda shu narsaga ahamiyat berish kerakki, ba'zi bir amallar ikkita belgi orqali ifodalangan.

1-jadval

Solishtirish amali belgisining yozilishi	Amalning matematik ifodasi	Amalning ma'nosi
=	=	Teng
◇	≠	Tengmas
<	<	Kichik
<=	≤	Kichik yoki teng
>	>	Katta
>=	≥	Katta yoki teng

Ajratgichlar guruhini quyidagi belgilar tashkil qiladi:

<ajratgich>::= . | , | : | ; | (|) | [|] | { | } | ' | " | @ | * | * | ! | ? | % | \$ | & | |

Ajratgichlarning vazifalarini tilni o'rganish davomida aniqlab boramiz.

Xizmatchi so'zlar guruhi juda keng, shuning uchun bu so'zlarni hammasini birdaniga yodlab, eslab qolish shart emas, balki ulardan foydalanish davomida ketma-ket eslab qolinaveradi:

<xizmatchi so'zlar>::=and | array | begin | case | const | div | do | downto | else | end | for | function | goto | if | in | label | mod | nil | not | of | or | packed | program | procedure | record | repeat | set | then | to | type | until | var | while | with

Mantiqiy amallar: AND («VA» - mantiqiy ko'paytirish amali);

OR («YOKI» - mantiqiy qo'shish amali);

NOT («EMAS» - mantiqiy inkor amali);

Barcha dasturlash dillari singari Paskal dasturlash tili ham o'zining imlosi, qonun va qoidalariga ega bo'lib, ular asosida yuqorida keltirilgan harflar, belgilar va amallar yordamida ko'rsatma va byuruqlar tuziladi. Har bir ko'rsatma yoki byuruq ":", (nuqtali vergul) belgisi bilan yakunlanadi.

Dastur yozishda quyidagilar qo'llaniladi:

Konstantalar (o'zgarmaslar)-dastur ishlashi davomida qiymati o'zgarmaydigan miqdorlar;

O'zgaruvchilar- dastur ishlashi davomida qiymati o'zgaradigan miqdorlar;

Algebraik ifodalar-arifmetik amallar bilan bog'langan o'zgarmaslar, o'zgaruvchilar va funtsiyalar;

Operatorlar-dasturlash tilining biror tugallangan amalini berish uchun mo'ljallangan buyrug'i;

Funksiyalar va protseduralar-o'z nomiga ega bo'lgan alohida dastur qismlari;

Operatorlar

Operator tushunchasi tilning eng asosiy tushunchalaridan biri bo'lib, har bir operator tilning yakunlangan jumlasiga hisoblanadi va ma'lumotlar tahtilining tugallangan bosqichini ifodalaydi.

Operatorlarni ikki guruhga ajratish mumkin. 1-guruh operatorlarining tarkibida boshqa operatorlar qatnashmaydi va bu operatorlarni asosiy operatorlar deb ataladi. Asosiy operatorlar jumlasiga quyidagi operatorlar kiradi: o'zlashtirish operatori, protsedura operatori, o'tish operatori, bo'sh operator. 2-guruh operatorlarining tarkibida esa boshqa operatorlar ham qatnashib, ular tarkibiy operatorlar deb ataladi. Ular jumlasiga quyidagi operatorlar kiradi: tashkiliy operator, tanlov operatori, takrorlash operatori, ulash operatori.

Masalani echish algoritmidagi yuqoridagi ikki guruh operatorlarining ketma-ketligi cheklanmagan miqdorda qatnashishi mumkin. Bu ketma-ketlikdagi operatorlar o'zaro ":" ajratish belgisi orqali ajratiladi, ya'ni programma matnining yozuvi alohida operatorlarga bo'linadi. SHunday qilib, S orqali ixtiyoriy yozish mumkin bo'lgan operatorni belgilasak, masala echilishining algoritmi quyidagi ketma-ketlik bo'yicha ifodalanishi mumkin:

S; S; ...;S.

Operatorlarning bu ketma-ketligi ularning programmada yozilish tartibi bo'yicha bajariladi. SHunday qilib, operatorning izdoshi undan keyin yozilgan operator hisoblanadi. Operatorlar bajarilishining bu tabiiy ketma-ketligini faqat o'tish operatori yordamida buzish mumkin. Tarkibiy operatorlarda esa operatorlarning bajarilish tartibi o'ziga xos qoidalar bilan aniqlanadi.

Nomlar va identifikatorlar

Ma'lumki, ma'lumotlarning tahlili jarayonini ifodalovchi algoritmlar turli xil ob'ektlar (o'zgaruvchilar, o'zgaruvchi miqdorlar, funktsiyalar va hokazo) ustida ish olib boradi. Bu ob'ektlarga ularning vazifasi va qabul qiladigan qiymatlariga qarab maxsus ismlar beriladi. SHu ismlarni odatda, identifikatorlar deb ataladi. Identifikator deb harf yoki "_" belgisidan boshlanuvchi, harf, raqam va "_" belgisining ixtiyoriy ketma-ketligiga aytiladi:

$\langle \text{identifikator} \rangle ::= \langle \text{harf} \rangle | \langle \text{identifikator} \rangle \langle \text{harf} \rangle | \langle \text{identifikator} \rangle \langle \text{raqam} \rangle$
Agar quyidagi oraliq tushunchani kiritsak:

$\langle \text{harf yoki raqam} \rangle ::= \langle \text{harf} \rangle | \langle \text{raqam} \rangle$
Yuqoridagi aniqlashni quyidagicha ham yozish mumkin:

$\langle \text{identifikator} \rangle ::= \langle \text{harf} \rangle \{ \langle \text{harf yoki raqam} \rangle \}$.

Xizmatchi so'zlardan identifikator sifatida foydalanish mumkin emas. Odatda identifikator so'zining o'rniga qulayroq va qisqaroq qilib ism deyish mumkin. Programmada qatnashuvchi ob'ektlarga ismlarni programma tuzuvchi o'z ixtiyoriga ko'ra tanlab olishi mumkin. Bir xil ism bilan bir necha xil ob'ektlarni nomlash mutlaqo mumkin emas. Turbo Pascal muhitida ismda qatnashuvchi belgilar soni (ism uzunligi) 63 ta belgidan oshmasligi kerak.

Ismlarga misollar:

Burchak, A1, Ahmad_Berdiev, C, Summa, Time, A, S1, ...

E'lonlar

Paskal tilining asosiy tushunchalaridan biri e'lon qilish hisoblanadi. Programmada qatnashuvchi barcha ob'ektlarning ismlari mos ravishda programmaning bosh qismida, ularning qanday tipdagi qiymatlar qabul qilishi mumkinligiga qarab, e'lon qilinib qo'yilishi kerak. Paskal tilida e'lon qilishning 5 xil turi mavjud:

- nishonlar e'loni;
- o'zgaruvchilar e'loni;
- tip aniqlash uchun e'lon;
- o'zgaruvchilar e'loni;
- protsedura va funktsiyalar e'loni.

Umuman olganda, yuqorida sanab o'tilgan e'lonlarning vazifalari ularning nomlaridan ham sezilib turibdi, e'lonning vazifalari esa keyinroq to'la ochib beriladi.

O'zgaruvchilar

O'zgaruvchi, programma ob'ekti bo'lib, turli xil qiymatlarni xotirada ma'lum nom bilan saqlab turish uchun ishlatiladi. O'zgaruvchi o'z qiymatini programmani bajarilish davomida, o'zlashtirish operatori yordamida qabul qiladi. qabul qilingan qiymat, o'zgaruvchiga boshqa yangi qiymat berilmaguncha saqlanib turiladi va yangi qiymat berilishi bilan eski qiymat butunlay o'chib, yo'q bo'lib ketadi. Har bir o'zgaruvchiga ma'lum bir tipga tegishli qiymatlarinigina qabul qilish huquqi beriladi. Boshqa tipdagi qiymatlarni o'zlashtirishga urinish programmaning xatoligini ta'minlaydi.

O'zgaruvchi - bu identifikatordir. Uning ismi o'zgaruvchining qiymatiga murojaat qilishda ishlatiladi. Boshqacha aytganda, programma matnidagi ism, shu o'zgaruvchining qiymatini ifodalaydi.

Funksiyalar va protseduralar

O'rta maktab kursidan funktsiya tushunchasi bizga yaxshi ma'lum. Algoritmik tillarda, faqat qiymatini hisoblash algoritmlari ma'lum bo'lgangina funktsiyalar ishlatiladi. Programma tuzuvchi programma uchun lozim bo'lgan kerakli funktsiyalarni o'z programmasiga kiritishi mumkin.

Xuddi funktsiyalar kabi hal qilinayotgan masalaning ma'lum bir tugallangan bosqichlarini hisoblash vazifasini protseduralar zimmasiga yuklasa ham bo'ladi. Funktsiyani hisoblash natijasida faqat, yagona natijaviy qiymatga erishiladi, protseduradan foydalanganda esa, natijaviy qiymatlar soni etarlicha ko'p bo'lishi mumkin.

Programmada aniqlangan funktsiya va protseduralar o'zgaruvchilarning e'loni bo'limida e'lon qilinib qo'yilishi kerak. Bunda har bir funktsiya va protseduraga, ularning bajaradigan vazifasiga mos ismlar berib qo'yiladi. Ularni aniqlashda formal parametrlardan foydalaniladi. O'z navbatida, bu parametrlarning tiplari funktsiya va protseduraning ichida aniqlanilib, e'lon qilinadi.

Programmada aniqlangan funktsiya va protseduralardan foydalanish uchun programma matnida ularning ismlari va formal parametrlarga mos, faktik parametrlari berilishi kerak.

Ma'lumki, matematika kursidagi elementar funktsiyalardan programma tuzishda juda ko'p foydalanishga tug'ri keladi. Bunday funktsiyalarni boshqa tillardagi kabi standart funktsiyalar deb ataladi va standart funktsiyalarning ismlaridan boshqa maqsadda foydalanish maqsadga muvofiq emas.

Programma matnini yozish qoidalari

Har bir algoritmik tilning programma matnini yozish qoidalari turlicha bo'ladi. Programmalash tillaridan eng soddasi Beysik tilining ma'lum versiyalarida, programmaning har bir operatori qat'iy aniqlangan qator nomerlari orqali yoziladi. Paskal tilida esa, operatorlar ketma-ket yozilib, o'zaro "-" belgisi bilan ajratib boriladi. Bundan tashqari, yozilgan programmaning o'qishga oson va undan foydalanish qulay bo'lishligi uchun programmada "matni ajratish" tushunchasidan foydalaniladi (bo'sh joy qatorni tugashi va izohlar).

Bo'sh joy (probel) grafik tasvirga ega emas belgi bo'lib, qatordagi bo'sh joyni anglatadi. Lekin, bo'sh joy belgisi o'zining sonli kodiga ega va programma matnidagi boshqa belgilar kabi kompyuterga kiritiladi.

Qator oxiri (tugashi) boshqaruvchi belgi bo'lib, u ham grafik tasvirga ega emas. Ma'lumki, programma matnini yozish davomida uni tabiiy ravishda yangi qatorlarga ajratilib yoziladi. CHunki, shu matnni yozmoqchi bo'lgan qog'ozning ham, kompyuter ekranining ham o'lchamlari cheklangan. Programma matnini alohida qatorlarga ajratmay yozish ham mumkin, lekin bir satrga 256 tadan ortiq belgi sig'maydi. Programma matnini alohida qatorlarga ajratish, programma tuzuvchining xohishiga qarab bajariladi. Ma'lum bir qator tugamay turib, yangi qatorga o'tish uchun "qator oxiri" tugmachasi bosiladi. Bu tugmacha ham o'zining maxsus sonli kodiga ega.

Izoxlar programmani o'qishga oson bo'lishi, uni qiynalmay tekshirib, yo'l qo'yilgan xatolarni to'g'rilash va programmada bajarilayotgan ishlarni tushuntirib borish uchun qo'yiladi. Izoxsiz yozilgan programmani hujjat sifatida qabul qilinmaydi. Muvaffaqiyatli qo'yilgan izoh programmaning va programma tuzuvchining katta yutug'i hisoblanadi. Izoxlar ixtiyoriy vaqtda programma matniga kiritilishi yoki olib tashlanishi mumkin. Bu bilan programmaning ishi o'zgarib qolmaydi. Izoxlarni "{" va "}" qavslari ichiga olinib yoziladi.

Programma "matn ajratgich"laridan foydalanishning quyidagi qoidalariga amal qilish lozim:

tilning ketma-ket yozilgan ikkita konstruktsiyasi orasiga albatta bo'sh joy yozilishi kerak;

ajratgichlarni xizmatchi so'zlar, sonlar va ismlar orasiga qo'yish maqsadga muvofiq emas.

Dasturning umumiy ko'rinishi

Paskalda dastur quyidagi ikki qismdan tashkil topadi: tasvirlash qismi; asosiy qismi.

program dastur ismi;

Uses (modullar ro'yxati);

Label(nishonlar ro'yxati);

Const (o'zgarmas miqdorlar);

var

(o'zgaruvchi miqdorilar)

protseduralar va funktsiyalarni e'lon qilish;

begin

(asosiy qismi)

end.

Programma tanasining asosiy qismi bu operatorlar bo'limidir. Har qanday programmada bu bo'lim albatta bo'lishi kerak. Programmaga qo'yilgan masalani echish shu bo'limda amalga oshiriladi. Boshqa bo'limlar esa yordamchi bo'limlar

bo'lib, tiplarni e'lon qilish bo'limlari deb ataladi. Bu yordamchi bo'limlar programmada qatnashishi yoki qatnashmasligi ham mumkin, lekin ularning yozilish ketma-ketligi saqlanib qolinishi zarur.



Savol va topshiriqlar.

1. Dasturlashning integrallashgan muhiti nima?
2. Til alfaviti nima?
3. Til alfavitining (Bekus - Naur) formulasini yozing.
4. Asosiy belgilarni sanab bering.
5. Hizmatchi so'zlarni yozing va ularning ma'nosini tushuntiring.
6. Til operatori qanday tushuncha?
7. Asosiy va tarkibiy operatorlarni ajratib sanab bering.
8. Operatorlar qanday tartibda bajariladi?
9. Programma operatorlariga ismlar qanday qo'yiladi va nima uchun qo'yiladi?
10. Ism (identifikator) ning BNF formulasini yozing.
11. E'lonlarning vazifasini tushuntiring.

3-§. O'zgarmas va o'zgaruvchi miqdorlar

Turbo Pascal dasturlash tilida o'zgarmas kattaliklar sifatida butun, haqiqiy va o'n oltilik sanoq sistemasi sonlari, mantiqiy doimiyliklar, simvollar, satriy kattaliklar, to'plam konstruktorelari va noma'lum ko'rsatgich belgisidan foydalanish yo'lga qo'yilgan.

Butun sonlar sifatida -2 147 483 648 sonidan +2 147 483 647 gacha bo'lgan butun sonlardan foydalaniladi va agar xisoblashlar jaryonida bu chegaralarlan «chiqish»ga yo'l qo'yilsa, dastur kompilyatori xatolik haqida xabar beradi.

Butun tiplar jadvalini keltiramiz:

Tip nomi	Uzunligi (baytlarda)	Qiymatlar diapazoni	Tipning quvvati
Byte	1	0 dan 255 gacha	256
ShortInt	1	-128 dan +127 gacha	256
Word	2	0 dan 65535 gacha	65536
integer	2	-32768 dan +32767 gacha	63536
LongInt	4	-2 147 483 648 dan +2 147 483 647 gacha	4294497296

Butun tiplardan tashqari haqiqiy tiplardan foydalanish tavsiya etiladi.

Haqiqiy sonlar juda ko'p hollarda qo'zg'aluvchi nuqtali-eksponentsial ko'rinishda ifodalanadilar. Juda ham kichik yoki juda xam katta sonlarni bunday ko'rinishda tasvirlash qulay hisoblanadi.

Masalan: 0,0000035000 sonini 3,50E-06 ko'rinishda ifodalash foydalanuvchi va dastur uchun qulay hisoblanadi.

Haqiqiy tiplar jadvalini keltiramiz:

Tip nomi	Uzunligi (baytlarda)	Qiymatlar diapazoni	Razryadlari miqdori
Real	6	-2.e39...1.7e38	11-12
Single	4	-1.5e45...3.4e38	7-8
Double	8	-5.0e324...1.7e308	15-16
Extended	10	-3.4e4932...1.1e4932	19-20
Comp	8	-9.2e18...9.2e18	19-20

O'n oltilik sanoq sistemasini sonlari bu sanoq sistemasining sonlari yordamida ifodalanadilar va faqat sonlarning oldiga dollar belgisi qo'yib yoziladi. Bu o'zgarmlar \$00000000 ... \$FFFFFFF diapazonda amal qiladilar.

Mantiqiy doimiyliklar. Paskal tilida mantiqiy tip boolean standart nomi bilan aniqlanadi. Mantiqiy tipli o'zgaruvchilar faqat ikki xil qiymat: True(rost) va False (yolg'on) larnigina qabul qilishi mumkin. Mantiqiy tipli qiymatlar ham tartiblangan, ya'ni False<True.

Paskal tilida asosan quyidagi uchta mantiqiy amaldan ko'proq foydalaniladi: not - rad etmoq, and - mantiqiy ko'paytirish, or - mantiqiy qo'shish.

Bu amallarni faqat mantiqiy o'zgarmlar ustidagina ishlatish mumkin va natijada yana mantiqiy o'zgarmlar hosil bo'ladi. quyida mantiqiy o'zgarmlar ustida amallar jadvali ko'rsatilgan:

Mantiqiy ko'paytirish	Mantiqiy qo'shish	Mantiqiy rad etmoq
True and true = true	true or true = true	not true = false
True and false= false	true or false= true	not false= true
False and true = false	False or true = true	
False and false = false	False or false = false	

Ixtiyoriy, qiymatlarni solishtirish amali ham mantiqiy qiymatni beradi:

Misol: 3>2 natijasi true

0<-1 natijasi false.

Simvoli doimiyliklarga '(apostrof) belgisi ichiga olingan klavyaturaning ixtiyoriy simvolini misol keltirish mumkin('a' yoki 'f' bunga misol bo'la oladi).

Agar apostrof simvolini o'zidan foydalanmoqchi bo'lsak uni dastur tarkibida quyidagicha yozish shart:

SHu holatda birgina apostrof belgisini chop etishga erishish mumkin.

Simvolni uning kodi yordamida ham yozish mumkin va bu holda simvol kodi oldidan # belgisi yoziladi.

Misol: #97-a simvoli;

#90-Z simvoli;

O'zgarmlar (const) - bu programmani ishlashi davomida o'zgarмай qoladigan miqdordir. Agar miqdor programmada ko'p marta ishlatilsa, uni programma matnida qayta - qayta yozgandan ko'ra, bu miqdorni o'zgarmlar deb aniqlab olib, programmadagi miqdorni o'rniga o'zgarmlar ismini yozish qulay bo'ladi. Masalan, hammaga ma'lum p ($p=3,1415926535\dots$) soni. Bu sonni bir necha marta takroran programmada yozish noqulay, shuning uchun, uni o'zgarmlar sifatida aniqlab olish maqsadga muvofiqdir.

O'zgarmlar *const* xizmatchi so'zidan keyin e'lon qilinadi (aniqlanadi):

<o'zgarmlar aniqlash>::=<o'zgarmlar ismi>=<o'zgarmlar>;

bu erda <o'zgarmlar>::=<skalyar miqdor>|<belgili qator>|<o'zgarmlar ismi>|<o'zgarmlar ismi>|<o'zgarmlar ismi>;

<o'zgarmlar bo'limi>::=<bo'sh>| *const* <o'zgarmlar aniqlash>;

Misol: *Program L1:*

const Pi 3.1415926535;

var A,B: real;

Begin

A:=Pi+Sin(Pi(Pi-1));*

B: sqrt(a);

end.

Satriy doimiyliklar - belilarning ixtiyoriy ketma-ketligi bo'lib ular apostroflar ichida beriladi. Lekin shuni aytish joizki CR kodli belgi bu tarkibda qatnasha olmaydi va u maxsus vazifani bajaradi. satriy doimiylik bo'sh ham bo'lishi mumkin.

Masalan:

'satriy doimiyliklar';

'G'ofur G'ulom bekati';

Turbo Pascal dasturlash tilida ayrim kattaliklarning qiymatlari o'zgaruvchilar yordamida saqlanishlari mumkin. O'zgaruvchi bu amalda axborotni saqlash uchun mo'ljallangan xotiraning bir qismi hisoblanadi.

O'zgaruvchilardan foydalanishda uning nomi va tipini ko'rsatish talab etiladi. Dasturda qatnashuvchi o'zgaruvchilarni tavsiflash o'zgaruvchilar bo'limi-var da amalga oshiriladi. Masalan, ikki butun sonning yig'indisini hisoblovchi dastur uchun o'zgaruvchilar bo'limini quyidagicha yozish mumkin:

```
var a,b:integer;{a va b-butun sonlarni ifodalovchi nom-kattaliklar}
    s:string; {S-satriy kattalik}
```

Satriy tip	
Tip nomi	Qiymatlar diapazoni
Char	ANSI kodlar jadvalidagi 0 dan 255 gacha bo'lgan kodlarga mos keluvchi belgilardan biri
String	Uzunligi 0 dan 255 gacha bo'lgan belgilar ketma – ketligi
String[n]	Uzunligi 0 dan n (n<255) gacha bo'lgan belgilar ketma – ketligi

Quyida o'zgaruvchilar va uzgarmaslarga nom berishdagi qoidalar bilan tanishamiz:

1. Nom berishda faqat 'A'. 'Z' va 'a'...'z' oraliqdagi ingliz tili alfaviti belgilaridan, 0 dan 9 gacha bo'lgan raqamlardan hamda _ (ostki chiziq) belgisidan foydalanish mumkin.

2. Nom faqat ingliz tili alfaviti belgilaridan hamda va _ (ostki chiziq) belgisidan boshlanishi mumkin.

3. Nom bir necha so'zlardan tashkil topgan bo'lsa ular ' _ ' belgisi bilan tutashtiriladi.

4. nom berishda foydalangan katta yoki kichik simvollar «farqlanmaydi»

5. Dasturda har bir nom faqat bir marotaba o'zgaruvchilar bo'limida qatnashish xuquqiga ega.

6. Nom sifatida xizmatchi so'zlardan foydalanish mumkin emas.

Masalan:

Tomonlari a,b qiymatlari ko'ra to'g'ri to'rtburchak perimetri, diagonali va yuzasini aniqlovchi dasturda qatnashuvchi o'zgaruvchilarni aniqlang.

```
var a,b,p:integer;
    D,uza:REAL;
```



Savol va topshiriqlar.

1. Miqdorlar qanday turdagi qiymatlarni qabul qilishi mumkin?
2. Simvulli doimiyliklarga deganda nimani tushinisiz? Misollar keltiring.
3. Satriy doimiyliklar nima. unga misollar keltiring.
4. Simvulli doimiyliklar bilan satriy doimiyliklarni farqi nimada?

5. Mantiqiy tipni nomi va qanday qiymatlar qabul qiladi?
6. O'zgaruvchilarni o'zgarmaslardan farqi nimada?
7. Dastur tuzish jarayonida o'zgaruvchi va o'zgarmaslar qaysi xizmatchi so'zlar bilan tavsiflanadi?
8. Sonli o'zgaruvchilar necha turga bo'linadi?
9. Butun sonli o'zgaruvchilarni turlariga misol keltiring.
10. Haqiqiy sonli o'zgaruvchilarni turlariga misol keltiring.
11. Satrli o'zgaruvchilar qanday tavsiflanadi? Misollar keltiring.

Mashqlar.

1. Quyidagi o'zgarmaslarni turlarini aniqlang.
 - a. 'g'; '!' ; '5';
 - b. 'axborot'; '-4563'; 'TAT'; '60%';
 - c. 'true'; 'false';
 - d. '9.567'; '56.3456'; '34566.6';
 - e. '0.45e-3'; '-6.8e6';
2. Quyidagi o'zgaruvchilarni turlarini aniqlang.
 - a. Omat: integer;
 - b. Hayol: real;
 - c. Sanoq: char;
 - d. Xamma: Boolean;
 - e. Men: single;
 - f. Bobodur: string [7];

4-§. Jadval ko'rinishidagi miqdorlar

Inson o'zining kundalik hayotida axborotlarni o'ziga qulay tarzda tasvirlashga odatlanib qolgan. Masalan, o'quvchining kundalik darslarini dars jadvali yordamida tasvirlash, futbol o'yinlari natijalarini qayd qilish kerakli ma'lumotlarni qidirib topish va ular ustida amallar bajarish jadval yordamida qulay. Siz yana karra jadvali, biror funktsiyani qiymatlari jadvali, kimyoviy elementlar jadvali, sinf o'quvchilarining navbatchilik jadvali kabi jadvallar bilan tanishsiz. Odatda jadvalni tashkil etuvchilar uning elementlari deb yuritiladi. Jadvallar birn o'lchovli, ikki o'lchovli, uch o'lchovli va ko'p o'lchovli bo'lishlari mumkin. Xayotda ko'proq bir o'lchovli (Chiziqli) va ikki o'lchovli (to'g'ri turtburchakli) jadvallar ko'proq qo'llaniladi.

Avtomobil soatiga 60km soat tezlik bilan harakatlansa, uning dastlabki 8 soatda bosib o'tgan yo'lini jadval yordamida tasvirlaydigan bo'lsak, uning ko'rinishi quyidagicha bo'ladi:

1	2	3	4	5	6	7	8
60	120	180	240	300	360	420	480

Jadval nomini V deb yuritadigan bo'lsak, $V[1]=60$, $V[2]=120, \dots$, $V[8]=180$ larga teng ekanligini kuzatamiz.

Ko'paytirish jadvalida birinchi o'lchov satr qiymatlaridan ikkinchi o'lchov ustun qiymatlaridan tashkil topgan.

	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9
2	2	4	6	8	10	12	14	16	18
3	3	6	9	12	15	18	21	24	27
4	4	8	12	16	20	24	28	32	36
5	5	10	15	20	25	30	35	40	45
6	6	12	18	24	30	36	42	48	54
7	7	14	21	28	35	42	49	56	63
8	8	16	24	32	40	48	56	64	72
9	9	18	27	36	45	54	63	72	81

Jadval nomini K deb yuritadigan bo'lsak, $K[1,1]=1$, $K[1,2]=2, \dots$, $K[9,9]=81$ larga teng ekanligini kuzatamiz.

Turbo Paskal dasturlash tilida jadval ko'rinishidagi miqdorlarni tasvirlash va ulardan foydalanish uchun massiv tushunchasi kiritilgan.

Massiv – jadval ko'rinishidagi miqdor bo'lib, u ma'lum sondagi bir turli va tartiblangan elementlar majmuasidan iborat.

Massiv nomga ega-bir xil tipga tegishli bo'lgan va tartiblangan elementlar to'plamidir. Odatda massivni bir necha usullarda berish mumkin va biz hozir ularning ayrimlari bilan tanishtirib o'tamiz:

1. Tiplar yordamida. Bu holda tip-massiv quyidagicha aniqlanadi:
 $\langle \text{tipning nomi} \rangle = \text{array} [\langle \text{indeksli tiplar ro'yxati} \rangle] \text{ of } \langle \text{tip} \rangle$

Bu erda: $\langle \text{tipning nomi} \rangle$ - to'g'ri nomli kattalik;

- Array, of- xizmatchi so'zlar;

$\langle \text{indeksli tiplar ro'yxati} \rangle$ - ixtiyoriy tartibli tip (longint dan boshqa)

Misol keltiramiz:

type

digit = array [0..9] of Char;

matrix = array [Byte] of Single;

var

m: matrix;

d: digit;

2. Massivni o'zgaruvchilar safida tip-massivdan foydalanmasdan ham berish mumkin.

Masalan:

Var

A,b:array [1..10] of real;

Bu erda birdaniga ikkita bir o'lchamli massiv (har biri 10 —elementdan iborat bo'lgan a va b massivlar) aniqlangan bo'lib, elementlarining tipi haqiqiy sonlardan iborat.

Yuqoridagi hollarda bir o'lchamli massivlar misol sifatida keltirilgan bo'lib, kvadrat qavslarda massiv elementlarining indeksleri, ya'ni tartib raqamlari ro'yxati keltirilgan.

A[1]-birinchi element

A [2]-ikkinchi element

...

A [10]- o'ninchi element

Agar massiv ko'p o'lchamli bo'lsa, uning indeksleri ro'yxati bir-biridan vergul bilan ajratilib yoziladi. Misollar keltiramiz:

D: array [1..5, 1..15] of real; (ikki o'lchamli)

kimel: array [0..25, 0..11] of string;(ikki o'lchamli)

yok: array [2001..2011,1..12,1..6] of byte;(uch o'lchamli)

Misol: Meteorolgik stantsiyada har bir soatda havo temperaturasi o'lchanib alohida jadvalga to'ldirilib boriladi:

O'lchash vaqti, s	1	2	3	4	5	6	23	24
Temperatura, S	12	13	11	15	10	12	11	11

Bu Chiziqli jadval bo'lib, 24 elementdan tashkil topgan bo'lib, uning elementlari 1 dan 24 gacha bo'lgan tartib raqamlariga ega. Mazkur jadvaldan 2 tartib raqamli element 13 ga, 23 tartib raqamli element 11 ga teng.

To'rtburchak jadval uchun gorizontaal va veritikal yo'nalishlar bo'yicha chegaralar ko'rsatilishi shart (qatorlar va ustunlar uchun). Bu turdagi jadval elementi uchun uning joylashgan joyiga ko'ra qator va ustun tartib raqamiga ko'ra aniq indekslardan foydalaniladi.

Xar qanday jadval ixtiyoriy tipdagi elementlardan iborat bo'lishi mumkin.

Misol:

Yo'lovchilarning 6-ta vagonida
15-chi -19 chi o'rinlarda joylashish-
lari jadvalini yaratamiz

Vagon-jadvali

	1-chi vagon	2-chi vagon	3-chi vagon	4-chi vagon	5-chi vagon	6-chi vagon
15	Mavjuda	Salima	Kamola	Menura	Aziza	Zulfiya
16	Sherzod	Salim	Ikrom	Qudrat	Aziz	Akram
17	Tolib	Sarvar	Coli	Karim	Olim	No'mon
18	Sojida	Gavhar	Lola	Gulrux	Zuhra	Fotima
19	Rustam	Mansur	Botir	Farrux	Azlar	Mahkam

Jadval ko'rinishdagi miqdorlarni, ya'ni massivlarni uning elementlari tashkil etadi. Oddiyroq qilib aytganda jadval kataklarida joylashgan qiymatlar jadval elementlari bo'lib, bu kataklar nomeri massivning indeksi deb yuritiladi. Dasturchi nuqtai nazaridan massivni jadval ko'rinishida tasavvur qilish nihoyat qulay. Masalan, M massiv 6 elemendan iborat bo'lib, quyidagi jadval yordamida elementlari berilgan bo'lsin:

1	2	3	4	5	6	- elementlar indeksi
12	2	0	-2	12	-18	- elementlar qiymati

U holda $M[1]=12$, $M[2]=2$, $M[3]=0$, $M[4]=-2$, $M[5]=12$, $M[6]=-18$ shu massiv elementlarini tashkil qiladi.

O'zbekiston futbol jamolarining turnir natijalari jadval ko'rinishida keltirilgan. Agar uni FF massivi deb yuritadigan bo'lsak:

```

FF[1] = 'Bunyodkor'
FF[2] = 'Paxtakor'
FF[3] = 'Nasaf'
FF[4] = 'Sho''rtan'
FF[5] = 'Mash''al'
FF[6] = 'Metallurg'
FF[7] = 'Andijon'
FF[8] = 'Qizilqum'
FF[9] = 'Navbahor'
FF[10] = 'Neftchi'
FF[11] = 'Olmalik'
FF[12] = 'Dinamo'
FF[13] = 'Iokomotiv'
FF[14] = 'Xorazm'

```

No	Jamoa
1	Bunyodkor
2	Paxtakor
3	Nasaf
4	Sho'rtan
5	Mash'al
6	Metallurg
7	Andijon
8	Qizilqum
9	Navbahor
10	Neftchi
11	Olmalik
12	Dinamo
13	Lokomotiv
14	Xorazm

Massiv elementlarini kiritishning bir necha yo'li mavjud va hozir shulardan o'zlashtirish operatori yordamida va klaviatura orqali elementni kiritish usuliga misollar ko'rib chiqamiz:

Massiv elementlarini o'zlashtirish operatori yordamida kiritish uchun massiv dasturning tavsiflash qismida e'lon qilinadi. Sungra dasturning asosiy qismida elementlar ketma-ket quyidagicha kiritiladi:

Massiv nomi[element indeksi]:=<elementning qiymati>:

Misol. «Hafta kunlari» massivini yaratish.

```

var Hafta_kuni:array[1..7] of string; i:byte;
begin
    Hafta_kuni[1]:='Yakshanba';
    Hafta_kuni[2]:='Dushanba';
    Hafta_kuni[3]:='Seshanba';
    Hafta_kuni[4]:='Chorshanba';
    Hafta_kuni[5]:='Payshanba';
    Hafta_kuni[6]:='juma';
    Hafta_kuni[7]:='Shanba';

end.

```

Klaviatura orqali elementni kiritishning operatorlaridan biri *Readln* operatori bo'lib, undan foydalanish qo'yidagicha:

Dasturning kerakli joyida elementni kiritish uchun quyidagi tartibda buyruqni kiritamiz:

Readln(Massiv nomi[element indeksi]):

```

Misol: Ishqoriy metallar massivini yarating.
var Ishqormet: array [1..5] of string;
begin
readln(Ishqormet [1]);
readln(Ishqormet [2]);
readln(Ishqormet [3]);
readln(Ishqormet [4]);
readln(Ishqormet [5]);
end.

```

Foydalanuvchi ishqoriy metallar nomlarini(litium, natriy, kaliy, rubidium, sezium) massiv elementlari sifatida kiritib chiqadi va ular mos ravishda quyidagi o'zlashtirish operatori bilan teng kuchli bo'lishadi:

```

Ishqormet [1]:= 'Litiy';
Ishqormet [2]:= 'Natriy';
Ishqormet [3]:= 'Kaliy';
Ishqormet [4]:= 'Rubidium';
Ishqormet [5]:= 'Seziy';

```

Ko'rib o'tganimizdek agar massiv elementlari soni ko'proq bo'lsa, ularni kiritish har ikki usulda ham qiyinchilik tug'diradi, ya'ni dastur hajmi «kattalashib» boradi. Bu holatni sikl operatorlari yordamida bartaraf etish mumkin. Shunday operatorlardan biri *for* operatori hisoblanadi. Bu operator bir buyruqdan takrorlanish yordamida bir necha marotaba foydalanish imkoniyatini beradi:

```


for i:=1 to 10 do
readln(a[i]);

```

Keltirilgan buyruqlar A-massivning 10 ta elementini kiritishga qulay bo'lib, readln(a[1]), readln(a[2]),..., readln(a[10]) buyruqlari bilan teng kuchli hisoblanadilar.

Dastur yordamida massiv elementlari yoki ular bilan bajarilgan amallar natijasini ekranda ko'rish uchun *writeln* operatoridan foydalanish mumkin. Masalan, writeln:(a[5])-buyrug'i a-massivning beshinchi elementini ekranda chop etadi.

Misol. 10 ta butun sonlardan A-massiv elementlarini klaviaturadan kiritish va ularni chop etish dasturini ko'rib chiqamiz:

Dastur tuzilishi	Natija:
<pre> var a:array [1..10] of integer; i:1..10; begin for i:=1 to 10 do readln(a[i]); for i:=1 to 10 do writeln('a[',i,']=',a[i],'); readln; end. </pre>	

Izoh: siz o'zlashtirish operatori, ma'lumotlarni klaviatura yordamida kiritish, ularni ekranga chiqarish va **For** sikli bilan keying darslarda batafsil tanishib chiqamiz.

1- Masala. Elementlari tasodufiy yaratilgan butun sonlardan iborat bo'lgan 15 elementli A va B massivlardan quyidagi shartni qanoatlantiruvchi uchinchi C massivni yarating.

$C[i] := A[i] + B[i]$, bu erda $i = 1, 2, 3, \dots, 15$;

Javob:
var a,b,c:array [1..15] of integer; i:byte;
begin
for i:=1 to 15 do a[i]:=random(30);
for i:=1 to 15 do b[i]:=random(30);
for i:=1 to 15 do c[i]:=a[i]+b[i];
for i:=1 to 15 do writeln(c[i]);
end.

2-Masala. 5-sondan iborat bir o'lchamli D massiv berilgan. Bu massivning har bir mos elementining 10 foaidan iborat bo'lgan yangi massiv yarating.

```

var d,d1:array [1..5] of real;
i:byte;

begin

```

```

for i:=1 to 5 do
readln( d[i]);
for i:=1 to 5 do
d1[i]:=(d[i]/100)*10;
for i:=1 to 5 do
writeln(d1[i]);
readln;
end.

```



Savol va topshiriqlar.

1. Hayotda uchraydigan jadval ko'rinishdagi miqdorlarga misol keltiring.
2. Chiziqli massiv qanday o'lchovlarda bo'ladi?
3. Ikki o'lchovli massiv qanday ko'rinishda bo'ladi?
4. Massivda indeks nima uchun kerak?
5. Massiv elementlarining indeksleri qanday qiymatlar qabul qilishi mumkin?
6. Jadval ko'rinishidagi miqdorlarning turlarini qanday ajiratish mumkin?

Mashqlar.

1. «yil oylari» massivini yarating.
2. Quyida keltirilgan sodda o'zgaruvchilar ketma-ketligi qanday massivlarni ifodalaydi:
 - a. $A[0], A[1], A[2], A[3], \dots, A[49]$;
 - b. $B[0,0], B[0,1], B[0,2], \dots, B[3,5]$;
 - c. $C[0,0,0], C[0,0,1], \dots, C[1,1,1]$;
3. Guruxingiz a'zolari haqida jadval tuzing. (ismi, tug'ilgan kuni, turur joy va pasport nomeri). Massivni dasturda tavsiflang va elementlarini tahlil qiling.
4. Dasturda bittadan bir o'lchovli butun va belgili jadvallarni hamda ikki o'lchovli haqiqiy turdagi jadvalni tavsiflang.
5. Agar var a : *array [0..21] of integer*; bo'lsa, a massivda nechta element mavjud?
6. Quyidagilarni tavsiflang va elementlarini tahlil qiling.
 - a. Var A: *array [0..99] of integer*;
 - b. Var B: *array [0..9, 0..19] of real*;
 - c. Var C: *array [0..19] of integer*;

5-§. Standart funktsiyalar va algebraik ifodalar.

Kattaliklarning tiplari ular foydalanadigan qiymat va ular bilan bajariladigan amallar bilan aniqlanadilar. Butun va haqiqiy tiplarda kattaliklar uchun qo'iydagi amallar qo'llaniladi:

Amallar	Amalning vazifasi	Butun va haqiqiy tiplarda
+	qo'shish	Qo'shish amali bajariladi, amalning natijasi qo'shiluvchilar tipiga mos
-	ayirish	Aayirish amali bajariladi, amalning natijasi qo'shiluvchilar tipiga mos
*	ko'paytirish	Ko'paytirish amali bajariladi, amalning natijasi qo'shiluvchilar tipiga mos
/	bo'lish	Bo'lish amali bajariladi, amalning natijasi faqat haqiqiy tip

Bundan tashqari faqat butun tiplar bilan bajariluvchi quyidagi amallar mavjud:

div	Butunga bo'lish	amalning natijasi faqat butun tip
mod	Bo'lish amalidagi qoldiq	

Quyidagi jadvarda Turbo Paskal dasturlash tilida foydalanadigan standart (matematik) funktsiyalar keltirilgan.

Функция номи	Аргумент тури	Киймат тури	Изох
Abs(x)	Хакикий	Хакикий	x
Sin(x)	Хакикий	Хакикий	Sinx
Cos(x)	Хакикий	Хакикий	Cosx
Arctan(x)	Хакикий	Хакикий	Arctgx
Sqrt(x)	Бутун\Хакикий	Хакикий	\sqrt{x}
Sqr(x)	Хакикий	Хакикий	x^2
Exp(x)	Хакикий	Хакикий	e^x (e=2.718282.....)
Ln(x)	Хакикий	Хакикий	Ln x
Frac(x)	Хакикий	Хакикий	"x"ning kasr qismi {x}
Int(x)	Хакикий	Хакикий	"x"ning Butun qismi {x}
Random	-	Хакикий	{0.1} oralikdagi tasodifiy son
Random(n)	Word	Word	{0.n} oralikdagi tasodifiy son

Ifodalar

Ifodalar asosan operator va operandlardan tashkil topadi. Operand sifatida o'zgaruvchi, doimiylik, funktsiya yoki boshqa biror ifodadan foydalanish mumkin. Asosiy operatorlarni keltirib o'tamiz. Bular $+$, $-$, $*$, $/$, div va mod .

$+$, $-$, $*$, $/$ operatorlari bizga tanish.

div operatori bir soni ikkinchisiga bo'lishda natijaning butun qismini beradi

Masalan, $23 \text{ div } 5$ ifodaning qiymati 4 ga teng

mod operatori bir soni ikkinchisiga bo'lishda qoldiq qiymatini beradi.

Masalan, $23 \text{ mod } 5$ ifodaning qiymati 3 ga teng

$*$, $/$, div va mod operatorlari $+$ va $-$ larga ko'ra, $*$, $/$, div va mod operandlari yuqori darajali xisoblanishadi va ular ifoda qiymatini xisoblashda birinchilardan bo'lib bajariladi.

Quyidagi natijalarga ko'ra xulosalar chiqaring:

A)

$$19 \text{ div } 4=4;$$

$$19 \text{ mod } 4=3;$$

$$12 \text{ div } 4=3;$$

$$12 \text{ mod } 4=0;$$

$$-21 \text{ div } 4=-5;$$

$$-21 \text{ mod } 4=-1;$$

$$-7 \text{ div } (-4)=1;$$

$$-7 \text{ mod } (-4)=-3.$$

B)

$$4^2 = \text{sqr}(4) = 16;$$

- $x=13$, $x^2 = \text{sqr}(x) = \text{sqr}(13) = 169;$

- agar $d=2$ va $e=5$ bo'lsa,

$$(d+e)^2 = \text{sqr}(d+e) = \text{sqr}(2+5) = \text{sqr}(7) = 49;$$

- agar $x=3$ bo'lsa,

$$x^4 = \text{sqr}(\text{sqr}(x)) = \text{sqr}(\text{sqr}(3)) = \text{sqr}(9) = 81.$$

V)

- $\text{abs}(12) = 12$; $\text{abs}(-12) = 12$;

- agar $x=3$ va $y = -5$ bo'lsa.

$$\text{abs}(x+y) = \text{abs}(3+(-5)) = \text{abs}(-2) = 2;$$

- agar $x=3$ va $y=-5$ bo'lsa

$$\text{abs}(x) + \text{abs}(y) = \text{abs}(3) + \text{abs}(-5) = 3 + 5 = 8.$$

Shuningdek, amallar tartibini belgilashda ifoda tarkibida o'rinli qavslardan foydalanish mumkin.

Butun va haqiqiy tiplar uchun darajaga ko'tarish amalini $\text{Sqr}(n)$ funksiyasi yordamida amalga oshirish mumkin. Masalan, a^{11} -qiymatini yuqoridagi fikrga ko'ra $z:=\text{sqr}(a)$; $v:=\text{sqr}(z)$; $u:=\text{sqr}(v)$; $y:=u*z*a$; ifodalar yordamida hisoblash mumkin.

a	z	v	u	y
a	a^2	a^4	a^8	a^{11}

Haqiqiy x -sonini absolyut qiymatini p darajaga ko'tarish uchun ($|x^p|$ -ni hisoblash uchun) $\exp(p*\ln(\text{abs}(x)))$ ifodaning qiymatidan foydalaniladi.



Savol va topshiriqlar.

1. Standart funksiyalarning odatdagi va Paskalda yozilishining qanday farqi bor?
2. Standart funksiyalarning qaysilarini bilasiz? Sanab bering.
3. Ifodalar nimalardan tashkil topgan?
4. Arifmetik amallarning bajarish tartibi qanday?
5. Biror ifodada bir xil amallar qatnashsa, ularning bajarilish tartibi qanday bo'ladi?
6. Amallarning bajarish tartibini o'zgartirish uchun nimalardan foydalaniladi?
7. Ifodada amal ishoralaridan ikkitasi ketma-ket kelishi mumkinmi? Mumkin bo'lsa, ular qanday yoziladi?

Mashqlar

1. Berilgan funksiya qiymatini ratsional usulda (eng kam amallar soni orqali) hisoblashni taklif qiling va dastur tilida ifodalang.

- a. $y = x^8 (y - (x^2)^2 * x)$, ya'ni 3-ta amal yordamida;
- b. $y = x^{10} (y - (x^2)^2 - (x^2 * x)^2)$, ya'ni 3-ta amal yordamida;
- c. $y = x^8 (y - ((x^2)^2)^2)$, ya'ni 3-ta amal yordamida.
- d. $y = x^{10}$
- e. $y = x^9$

2. Quyidagi ifodalarni Pascal dasturlash tilida yozing.

a) $v = 1 + \sin^2(x + y)$	d) $u = \frac{\sqrt[3]{8 + x - y ^2 + 1}}{x^2 + y^2 + 2}$
v) $w = \cos x - \cos y $	e) $y = \sqrt[8]{\frac{\pi}{8}} \cdot \sqrt{\frac{\sqrt{a^2 + b^2 + c^2}}{a + b + c}} - \left(\frac{1}{2a} \cdot e^{- x-a }\right) \cdot (c - b - a)$
s) $t = \frac{2 \cos\left(x - \frac{\pi}{6}\right)}{0.5 + \sin^2 y}$	o) $u = (1 + z) \cdot \frac{x + y}{a - 1 - x^2} \cdot \frac{z}{1} + \frac{\sin^2 x}{x^2 + y^2}$

6-§. O'zlashtirish va ma'lumotlarni ekranga chiqarish operatorlari.

Dasturning bajariluvchi qismi odatda bir yoki bir necha *operatorlardan* iborat bo'ladi. Operator dastur bajarilishi lozim bo'lgan biror amalni ifodalaydi. SHunday operatorlardan biri juda ko'p foydalanadigan o'zlashtirish operatori hisoblanadi. Agar dasturda xisoblashlar bajarilsa u holda albatta o'zlashtirish (qiymat berish) foydalaniladi. Umumiy holda o'zlashtirish quyidagi ko'rinishga ega:

o'zgaruvchi := *ifoda*;

Bu erda:

- o'zgaruvchi dastur bajarilishi natijasida unga o'zlashtirgan qiymatni ifodalaydi.

:= qiymat berish belgisi

ifoda - bu ifodaning qiymati nomi ko'rsatilgan kattalikka (o'zgaruvchiga) o'zlashtiriladi.

ifoda va o'zgaruvchi tiplari mos kelishi shart:

Misollar:

D:=5*g-12.5;

Ff:=f1+(a-t);

Perimetr:=2*a+2*b;

Z:= False;

Dfg:= 'diametr';

O'zlashtirish operatoridan foydalanishga qator misollar keltiramiz:

Tomonlari 5 sm,7 sm va 8 sm teng bo'lgan uchburchak perimetrini hisoblash uchun quyidagi o'zlashtirish operatorlaridan dastur tarkibida foydalanish mumkin.

$$A:=5; B:=7; C:=8; P:=A+B+C;$$

Katetlari 3.5 sm,7.8 sm bo'lgan to'g'ri burchakli uchburchak perimetrini hisoblash uchun quyidagi o'zlashtirish operatorlaridan dastur tarkibida foydalanish mumkin.

$$A:=3.5; B:=7.8; C:=\text{sqrt}(\text{sqr}(a)+\text{sqr}(b)); P:=A+B+C;$$

$y=ax^2+bx+c$ funktsiya qiymatini $a=2, b=4, c=5, x=45$ larda hisoblash uchun quyidagi o'zlashtirish operatorlaridan foydalanishimiz mumkin.

$$a:=2; b:=4; c:=5; x:=45;$$

$$y:=a*\text{sqr}(x)+b*x+c;$$

Radiusi 8 sm bo'lgan aylana uzunligini aniqlashda o'zlashtirish operatorlari quyidagich bo'lishi mumkin:

$$R:=8; c:=2*\text{pi}*r;$$

yoki

$$R:=8; d:=2*r; c:=d*\text{pi};$$

Izoh: $\text{pi}=3,141592654\dots$ ga teng bo'lib, Turbo Paskal dasturlash tilida doimiylik sifatida aniqlangan, shuningdek bu tilda "R" va "r" dastur tarkibida bir kattalikni ifodalaydi.

Ma'lumotlarni ekranga chiqarish operatorlari

Ma'lumotlarni ekranga chiqarish Write va Writeln protseduralari yordamida amalga oshiriladi. Write so'zidan keyin qavslarda apostrof belgisi bilan chegaralangan matnli ma'lumotlar va o'zgaruvchilar ro'yxatini vergul orqali ajratib berish mumkin. Misollar:

Write('Teglama ildizlari');

Write(Ko'paytma);

Write(rr);

Write('to'rtburchak tomonlari. A=', A,'B=',B);

O'zgaruvchi nomidan so'ng ikki nuqta belgisini qo'yib o'zgaruvchi qiymatini monitorida aks ettirish formatini belgilash mumkin. Integer tipidagi kattalik uchun format-bu qiymatni aks ettiruvchi chop etish maydonining kengligi(ekrandagi pozitsiyalar miqdori) hisoblanadi. Masalan, Write(a:7) instruktsiyasi a o'zgaruvchi qiymatini chop etilishiga 7-ta pozitsiyadan foydalanish belgilanganligini ko'rsatadi. Agar qiymatni tasvirlash uchun ko'rsatilgan pozitsiyadan kamroq pozitsiya zarur bo'lsa, u holda bu sonning oldiga format qiymatini to'ldirgunga qadar probellar joylashiriladi.

Real-tipidagi kattalik uchun format-bu o'zgaruvchi ikki nuqta va ikki nuqta bilan ajratilgan ikki butun sondan iborat. Masalan, Write(a:7:2); Bu erda birinchi son chop etish maydonining kengligini ifodalasa, ikkinchisi kasriy qismdagi sonlar miqdorini aniqlaydi. Agar faqat kenglik qiymatigina formatda berilsa va shuningdek, formatda ko'rsatilgan kenglik o'zgaruvchi qiymatidan kamroq bo'lgan holda ham o'zgaruvchi qiymati "suzib yuruvchi" nuqtali son ko'rinishida bo'ladi.

Write instruktsiyasi orqali monitordagi axborot chop etilgach kursor oxirgi pozitsiyadan keyingi joyni egallaydi va keyingi Write instruktsiyasining natijasi bu pozitsiyadan chop etiladi.

Writeln instruktsiyasining Write instruktsiyasidan farqi shundaki har safar bu instruktsiya bajarilgach, kursor keyingi satr boshiga o'tkaziladi. Bu holatlarni quyidagi dasturlar yordamida kuzatishimiz mumkin.

Dasturlardan namunalar	Natija:
<pre>var a,b,c:integer; begin a:=4; b:=6;write ('a+b=');write(a+b);readln; end</pre>	a+b=10
<pre>var a,b,c:integer; begin a:=4;b:=6;c:=a+b;write(a,b,c);readln; end.</pre>	4610
<pre>var a,b,c:integer; begin a:=4;b:=6;c:=a+b;write(a:5,b:5,c:5);readln; end.</pre>	4 6 10
<pre>var a,b,c:integer; begin a:=4;b:=6;c:=a+b; writeln (a);writeln(b);writeln(c);readln; end.</pre>	4 6 10



Testlar

1. d-o'zgaruvchi qiymatini aniqlang.

```
var a,b,c,d:integer;
begin a:=5; b:=55; c:=555; d:=a+b+c;
writeln(d);
end.
```

a) 615 b) 5505 d) 1000 e) 1555

2. dastur natijasini aniqlang

```
var a,b,c,d:integer;
begin
a := 5; b := 55;
write(a-b);
write(#9);
c := 555;
d := a * b * c;
writeln(d-a);
```

end.

a) -50 615 b) -50615 d) -50 e) -50 615

3. $\pi=3.141592..$ qiymatni hosobga olib dastur natijasini aniqlang.

Begin

writeln(pi:5:4);readln;

end.

a) 3.141592.. b) 3,1416 d) 3.1416 e) 3.1415

4. Berilgan dastur uchun formatli chop etish natijasini aniqlang:

var t: real;

begin t:=12.345;

writeln(t:3);readln;

end.

a) 1.2E+01 b) 1.2E-01 d) 12 e) 1234

5. Dastur natijasini aniqlang:

var x1,x2:integer;

begin

x1:=-5; x2:=-15;

writeln(abs(x1)+abs(sqrt(x2)));readln;

end.

a) 230 b) -20 d) 20 e) 2020

6. To'g'ri javobni aniqlang.

a) 7 b) 0 d) 0 7 e) 70
0 7

7. f3 –qiymatini aniqlang:

var f1,f2,f3:boolean;

begin

f1:=false; f2:=True; f3:=(f1 and f2); writeln(f3); readln;

end.

a) TrueFalse b) True d) False

c) FalseTrue

8. Keltirilgan dastur uchun to'g'ru javobni aniqlang:

var a,b,c,y,x:real;

begin

a:=1; b:=2; c:=3; x:=-1; y:=a*sqrt(x)+b*x+c; writeln(y:2:2);

end.

a) 2 b) 6.00 d) 0 e) 2.00



Topshiriqlar.

1. Ifodalar qiymatini hisoblang:
 - a. $y = |x| + x^4$, agar $x = -3$ va $x = 3$ bo'lsa;
 - b. $a = |x| + 4x^3 - 7x^2$, agar $x = 2$ va $x = -2$ bo'lsa;
 - c. $z = |x - 2| - 3x^8$, agar $x = -2$ va $x = 1$ bo'lsa;
 - d. $a = 6b^2 + |b - 3|^3 - 15$, agar $b = 9$ va $b = -3$ bo'lsa.

2. Quyidagi ifodalar qiymatlarini xisoblash uchun dastur tuzing:
 - a. $y = (3x^3 + 18x^2) * x + 12x^2 - 5$;
 - b. $a = (d + c + b) * e - 5k - 1$;
 - c. $d = 3c^3 + |c^2 - 4c + 7|^3 - 5c$ bu erda $c = |x + 4| - |x^2 - 3x + 6|$.

3. Ekranda buuyk mutafakkir va shoir Alisher Navoiyning quyidagi ruboiysini chop etish dasturini tuzing:

*G'urbatda g'arib shodmon bo'lmas emish,
El anga shafiqu mehribon bo'lmas emish.
Oltun qafas ichra gar qizil gul butsa,
Bulbulg'a tikondek oshyon bo'lmas emish.*

begin

writeln('G'urbatda g'arib shodmon bo'lmas emish,');

writeln('El anga shafiqu mehribon bo'lmas emish.');

writeln('Oltin qafas ichra gar qizil gul butsa,');

writeln('Bulbulg'a tikondek oshyon bo'lmas emish.');

readln;

end.

7-§. Ma'lumotlarni xotiraga muloqot usulida kiritish.

Dasturlarning muhim xossaligidan biri bu ularning foydalanuvchiga qator qulayliklar yaratib berishida bo'lib bu imkoniyatlar muloqot usulida dasturdan foydalanishda yaqqol ko'zga tashlanadi. Paskal tilining read, readln, write, writeln protseduralari yordamida muloqot tipidagi dasturlarda ma'lumotlarni kiritish va chiqarish etarli emas. masalan, read va readln protseduralari yordamida klaviatura yordamida faqat tiplashirilgan kattaliklarni kiritish (bunda ham har bir kiritilgan simvol ekranda o'z aksini topadi) imkoniyatini beradi. Turbo Paskal dasturni klaviatura yordamida samarali boshqarishning sodda va aniq vositlariga ega.

Juda ko'p hollarda read va readln protsedurlaridan foydalanishda klaviaturadan kiritilgan simvolni ekranda aks etilishi zarurati bo'lmay qoladi. Bundan tashqari bu protseduralar faqat raqamlar, simvollar, ayrim belgilarni kiritish va qisman maxsus klavishalardan foydalanish (masalan, Backspace dan oxirgi kiritilgan simvolni o'chirish uchun) imkoniyatini beradilar. Bu protseduralar funksional yoki tahrirlovchi klavishalarni yoki Ctrl, Alt, Shift tugmalrining boshqa klavishalar bilan bo'lgan kombinatsiyasini «tanish» qobiliyatiga ega emaslar va shuning uchun ham ulardan jiddiy dasturlarda kam foydalanish maqsadga muvofiq.

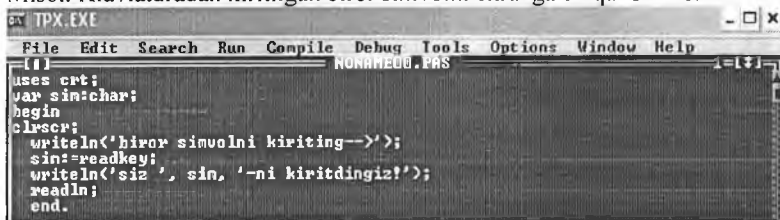
ReadKey standart funksiyasi. bu funktsiya Srt modulining funksiyasi bo'lib, klaviatura yordamida dasturni boshqarishda qulay hisoblanadi. Undan parametrsiz foydalanadilar, u simvolli tip qiymatini beradi va bu qiymatdan dasturda foydalanish mumkin. ReadKey funksiyasining ishlash tartibi quyidagicha:

dastur bajarilishi biror tugmani bosilishini «kutish»ni tashkil qilish bilan vaqtincha to'xtatiladi.

biror tugma bosilgach, fnktsiya o'z ishini gugma kodini «aniqlab berish» bilan tugatadi.

ReadKey funksiyasidan foydalanishga doir sodda misol keltiramiz va uning boshqa imkoniyatlari bilan siz kelgusi mashg'ulotlarda albatta tanishasiz.

Misol. Klaviaturadan kiritilgan biror simvolni ekranga chiqarish dasturi.



```
TPX.EXE
File Edit Search Run Compile Debug Tools Options Window Help
NONHEAD.PAS 1=171
uses crt;
var sin:char;
begin
clrscr;
writeln('biror simvolni kiriting-->');
sin:=readkey;
writeln('siz ', sin, '-ni kiritdingiz!');
readln;
end.
```

Keltirilgan dasturning natijasi.



```
TPX.EXE
biror-simvolni kiriting-->
siz U-ni kiritdingiz!
```

Dasturda foydalanilgan Slrscr protsedurasi Srt moduliga tegishli bo'lib, har safar dastur ishga tushirilganda ekranni "tozalash" vazifasini bajaradi.

Yuqoridagi dasturda sim:=readkey buyrug'i foydalanuvchi tomonidan biror simvol kiritilganicha dastur bajarilishini to'xtadi (masalan ko'rsatilganidek "W") va faqat biror simvol kiritilgach, dastur bajarilishi davom ettiriladi.

Umumiy holda barcha tugmalar va ularning kombinatsiyasini ikkita to'plamga ajratilgan bo'lib, ularni asosiy va kengaytirilgan to'plam nomlari bilan yuritiladi.

Asosiy to'plamga harflar va raqamlarni ifodalovchi tugamalar, razdeliteley, znakov prepiniyalar taalluqli bo'lsa kengaytirilgan to'plamga bu tugmalarning Shift(yoki CapsLock tugmasidan foydalanishi ham), Backspace, Enter va Esc tugmalarini birgalikda (kombinatsiyasi) ishlatilishi mansubdir. Ko'rsatib o'tilgan tugmalarning biri bosilganda readKey funktsiyasi shu tugmaga mos ASCII-kodini aniqlash imkonini beradi.

KeyPressed standartnaya funktsiyasi klaviatura bilan o'zaro aloqada bo'luvchi ikkinchi funktsiya imsoflanadi. farqli bosilgan tugmani kodini parametrsiz foydalaniladi va ixtiyoriy tugma bosilganda True , aksincha False qiymatlarini beradi. KeyPressed bosilgan tugma kodi bilan biror amalni bajarmaydi ammo bu kod zarur hollarda readKey funktsiyaisi yordamida aniqlanadi. DOS tizimida bosilgan tugmalar kodini operatsim tizim tomonidan joylashtiadigan bufer (maksimum 15 kod uchun) «joy» mavjud bo'lib, tugmalar kodi bosilishiga qarab quyidagi tartibda joylashadi: birinchi bosilgan tugma kodi, ikkinchi bosilgan tugma kodi,... oxirgi bosilgan tugma kodi.

ReadKey funktsiyasi bu kema-ketlikni boshdan «o'qiy» boshlaydi va faqai bufer bo'sh bo'lgan holda dastur bajarilishini «to'xtatib» turadi.



Savol va topshiriqlar.

1. Read, Write so'zlari qanday ma'noni anlatadi?
2. Ma'lumotlarni muloqot holdida kiritish operatorining vazifasini ayting.
3. Ma'lumotlarni kiritish operatorining necha xil ko'rinishi bor? Ularning farqi nimada.
4. O'zgaruvchilarga qiymat berishda o'zlashtirish operatorining qulaylik tomonini tushuntiring.
5. O'zgaruvchilarga qiymat berishda kiritish operatorining qulaylik tomonini tushuntiring

Mashqlar

1. $a=19$, $v=6$, $d=190$ bo'lganda quyidagi ifodalarning qiymatini hotiraga qulay usulda kiritib hisoblash dasturini tuzing.

- A) $y=a+b^2+ad$
- b) $h=\text{sqrt}(b+d)-(d-a)$
- c) $s=\text{cosa}+\text{sind}$
- d) $p=\text{pi}*d^2+a*b$.

2. $y=5x+34$ funksiyaning qiymatini $(-6:6)$ oraliqdagi butun son qiymatlarida qulay usulda kiritib hisoblash dasturini tuzing.

3. $y=ax^2+bx+c=0$ tenglamaning a. b. c ning quyidagi qiymatlarida qulay usulda kiritib hisoblash dasturini tuzing.

- a) $a=1$, $b=1$, $c=3$

- b) $a=1, b=4, c=4$
- c) $a=1, b=-4, c=4$
- d) $a=1, b=5, c=3$

4. $y=21x^2+7x+1963$ funksiyaning qiymatini x ning $(-6;6)$ oraliqdagi butun son qiymatlarida qulay usulda kiritib hisoblash dasturini tuzing.

5. Tomonlari $a=3, b=5, c=6$ bo'lgan uchburchakning yuzini topish dasturini tuzing.

8-§. Ekran bilan ishlash oreratorlari.

Ekran bilan ishlash uchun Turbo Paskal tilida turli protseduralar, funksiyalar va operatorlarni qamrab olgan SRT modulidan foydalaniladi. SRT moduli sodda ammo samarali imkoniyatlarga ega. Kompyuter tarkibida foydalanadigan axborotni vizual akslantiruvchi vosita-displey ikkita asosiy qismdan iborat: monitor va boshqarish bloki(adapter yoki displey adapteri). Bu ikki qurilma bir-biri bilan hamkorlikda (ayrim hollarni hisobga olmaganda, masalan rangli monitor va monoxrom adapter bilan ishlay olishi) "faoliyat" ko'rsatishadi.

Birinchi adapter 1981 yili monoxrom adapter (MDA) nomi bilan IBM PC kompyuterlarida qo'llanilgan bo'lib, matnli ma'lumotlarni faqat 20 qatorda 40 tadan belgi yoki 25 qatorda 80 tadan baeli bilan chiqaradigan formatlarga ega edi. Bunda yorqin belgilar qora fonda akslanar edi va shriftlar kengligi har ikki holda ham bir xil bo'lib, ular "oddiy"(obo'chno'y), hamda "tagiga chizilgan" (podecherknuto'y) ko'rinishga ega bo'lishlari mumkin edi.

1982 yili Hercules firmasi HGC adapteri yaratdi va u MDA adapteri xosalarini takrorlash bilan birga 700x350(piksellarda) hajmdagi grafik tasvirlarni ham ekranga chiqara olar edi. Aynan shu vaqtda IBM kompaniyasi CGA(Color Graphics Adapter-rangli grafik adapter) nomli adapterni yaratdi. Bu adapter matnli(40x25 yoki 80x25) va grafik ma'lumotlarni (320x200) ekranga chiqarish imkoniyatiga ega edi. MDA adapteri kabi simvollarni "tagiga chizilgan" rejimda chiqara olmasada "negativ" (qora rangli simvollar yorqin fonda) tasvirda chiqarish imkoniyatiga ega bo'lgan. Rangli rejimda simvollar uchun 16 rangdan fon uchun esa 8 rang qo'llanilgan. CGA adapterining matnli imkoniyatlari IBM kompaniyasining keyinchalik yaratgan EGA, MGCA, VGA va SVGA adapterlarida ham saqlangan. SRT moduli imkoniyatlari bu adapterlarga qo'llanilishi nuqtai nazaridan qarab chiqiladi.

TextMode protsedurası. Bu protsedura adaptarning matn bilan ishlash rejimlaridan birini aniqlash imkoniyatini beradi. Uning sarlavhasi quyidagicha:

```
procedure TextMode (mode:word);
  bu erda mode-matnli rejim kodi.
```

SRT modulida mode uchun quyidagi doimiyliklardan foydalaniladi.

```
const
  BW0 =0:          {40x25 hajmdagi oq-qora rejim}
```

```
Co40=1;      {40x25 xajmdagi rangli rejim }
BW80  =2;    {40x25 xajmdagi oq-qora rejim }
CO80  =3;    {80x25 xajmdagi rangli rejim }
```

```
Mono    =7;      {MDA dan foydalaniladi}
Font8x8 =256;   { EGA yoki VGA adapterlarida 80x43 yoki 80x50
rejimlariga xos, yuklanuvchi shrift uchun tanlanadi }
```

TextMode protsedurasi yordamida o'ralgan rejim kodi modulning Last-Mode global o'zgaruvchisida saqlanish orqali dastlabki holatga qaytarilishi mumkin.

Misollar keltiramiz.

1-Misol. BW40 doimiylikidan foydalanish.

```
uses crt; var lm,t:word; const BW40=0;
begin lm:=lastmode; textmode(1);
  writeln('123456789x123456789x123456789x123456789x');
end.
```

Dastur natijasi



(bir satrda 40 ta simvol aks ettirilgan)

2-Misol. (col80+font8x8 –rejimi uchun)

```
uses crt; var lm,t:word; const col80=3;
begin lm:=lastmode; textmode(col80+font8x8);
write('123456789x123456789x123456789x123456789x');
write('123456789x123456789x123456789x123456789x');
end.
```

Dastur natijasi



(bir satrda 80 ta simvol aks ettirilgan)

Vazifa. Mustaqil ravishda TextMode protsedurasining qolgan rejimlarida matnning yozilishini kuzating.

TextColor protsedurasi Kiritiladigan matn belgilari rangini aniqlaydi. Undan foydalanish tartibi quyidagicha: TextColor(Color: Byte):

Misol. textcolor(4) yoki textcolor(red) operatorlari matnning rangi qizilbo'lishini ta'minlaydi.

TextBackground protsedurasi. Fon rangigi belgilash uchun ishlatiladi. Undan foydalanish tartibi quyidagicha :TextBackground(Color: Byte);

Misol, TextBackground (1) yoki textcolor(blue) operatorlari fon rangini to'q ko'k bo'lishini ta'minlaydi.

TextColor va TextBackground protsedurasi parametrlarining tipi Byte, tipi bilan aniqlanadilar va ular tegishli rangni ta'minlaydilar.

Rang	Kod	
Black	0	Qora (CHyorno'y)
Blue	1	To'q ko'k (Tyomno-siniy)
Green	2	To'q yashil (Tyomno-zelyono'y)
Cyan	3	Moviy (Biryuzovo'y)
Red	4	Qizil (Krasno'y)
Magenta	5	Binafsha rang (Fioletovo'y)
Brown	6	Jigarang rang (Korichnevo'y)
LightGray	7	Och ko'k (Svetlo-sero'y)
DarkGray	8	To'q kul rang (Tyomno-sero'y)
LightBlue	9	Ko'k (Siniy)
LightGreen	10	Och yashil (Svetlo-zelyono'y)
LightCyan	11	Och moviy (Svetlo-biryuzovo'y)
LightRed	12	Och qizil(atirgul) (Rozovo'y)
LightMagenta	13	Malina (och malina rang) (Malinovo'y)
Yellow	14	Sariq (Jyolto'y)
White	15	Oq (Belo'y)
Blink	128	belgi yonib-o'chishi

ClrScr protsedurasi. Bu protsedura ekran va Window protsedurasi yordamida yaratilgan oynani tozalaydi, berilgan fon rangi bilan to'ldirib, kursorni ekraning yuqori chap nuqtasiga joylashtiradi. Undan foydalanish tartibi quyidagicha: clrscr;

Window protsedurasi. Bu protsedura matn uchun mo'ljallangan oyna (kichik ekran) ni yaratish uchun mo'ljallangan. Bu protseduradan foydalanganda kursor oynaning yuqori chap nuqtasiga joylashadi va oyna fon rangi bilan to'ldiriladi. Matn chop etilishida matn shu oynaning o'ng chegarasiga etgach, yangi qatorga tushishi va oyna to'lgach matnni «tepaga qarab siljitish» ta'minlanadi. Undan foydalanish tartibi quyidagicha:

Window(X1,Y1,X2,Y2: Byte); Bu erda (X1,Y1)-oynaning yuqori chap, (X2,Y2) –o'ng past nuqtalari koordinatalari.

GotoXY protsedurasi. Bu protsedura kursorni ekran yoki oynada yangi koordinatali nuqtaga o'tkazish uchun ishlatiladi. Undan foydalanish tartibi

quyidagicha: GotoXY(X,Y: Byte); Misol, GotoXY(4,8); (X,Y-qiymatlari ekran yoki oyna chegarasidan chiqadigan bo'lsa, buyruq bekor qilinadi)

WhereX va WhereY funktsiyalari. Bu funktsiyalar yordamida kursorning joriy koordinatalari aniqlanadi: WhereX –uning gorizontal, WhereY - vertikal koordinatalari qiymatini aniqlashda qo'llaniladi.

Misol: writeln(wherex,wherey);

1-masala. Textbackground protsedurasi yordamida binafsha rangli fonda matnni chop etuvchi dastur tuzing.

```
uses crt;
const col80=3;
begin  textmode(col80+font8x8);
textbackground(5);
write('textbackground(5);-matn fonini binafsha rang bilan berilishini
ta'minlaydi');
end.
```

Dastur natijasi:



2-masala. Textcolor protsedurasi yordamida oq rangli fonda qizil rangli matnni chop etuvchi dastur tuzing.

```
uses crt;
const co80=3;
begin  textmode(co80+font8x8);
textbackground(15); textcolor(4);
write('oq fonda qizil rangli matn yozilgan');
end.
```

3-masala. Ixtiyoriy tugma bosilguncha ekran rangi qizil bo'lishi va tugma bosilgach. uning rangi qora bo'lishini ta'minlovchi dastur yarating.

```
Uses CRT;
var
k: Char;
begin
TextBackground(red) ;
ClrScr;
Writeln('ixtiyoriy tugmani bosing...');
k:= ReadKey;
TextBackground(Black) ;
ClrScr
end.
```

4-masala. Ekranida Window protsedurasi yordamida ikkita oyna ochi bunda to'g'ri burchakli uchburchaklarning o'xshshaligi to'g'risidagi teoremlarni chop etuvchi dastur yarating.

```
Uses CRT;
Var k,lm: integer;
begin
  lm:=lastmode;TextBackground(Blue);
  Window(5,2,35,12);TextColor(Yellow);writeln('To"g"ri          burchakli
uchburchak-');
  writeln('larning bittadan burchagi mos ');writeln('ravishda teng bo"lsa, ular
');
  writeln('o"xshash bo"ladi ');Writeln(' enter tugmasini bosing...');ReadLn;
  ClrScr; TextBackground(Red); TextColor(White);Window(40,2,70,12);
  writeln('To"g"ri burchakli uchburchak-');writeln('larning katetlari mos rav-
ishda ');
  writeln('proporsional bo"lsa, ular ');writeln('o"xshash bo"ladi ');
  Writeln(' enter tugmasini bosing...');ReadLn;clrscr;
end.
```

5-masala. Kursorni ekranning (20,20) koordinatali nuqtasiga o'rnatib shu nuqtadan boshlab.'TURBO PASCAL' matnini chop etgandan so'ng kursor joylashgan nuqtaningkoordinatalarini aniqlovchi dastur tuzing.

```
uses crt;
var x,y:byte;
begin clrscr; x:=20; y:=20; gotoxy(x,y); write('TURBO PASCAL');
writeln(' yangi_X=',whereX,' yangi_y=',WhereY);readln;
end.
```

6-masala. Quyida keltirilgan ikki dasturni bajarib, clreol protsedurasi vazifasini aniqlang. (amaliy ish)

```
A) uses crt;
begin clrscr;
  writeln('tarkibida 2,14 foizdan ko"p temir va uglerod bo"lgan qotishma-
cho"yandir'); readln; gotoxy(1,1);
  writeln('tarkibida 2,14 foizdan kam uglerod tutgan temir va uglerod qotish-
ma-po"latdir');readln;
end.
```

```
B) uses crt;
var x,y:byte;
begin clrscr;
  writeln('tarkibida 2,14 foizdan ko"p temir va uglerod bo"lgan qotishma-
cho"yandir');
  readln; gotoxy(24,1); clreol;
```

```
write(' kam uglerod tutgan temir va uglerod qotishma-po"latdir');  
readln;  
end.
```

7-masala. LowVideo, NormVideo, HighVideo protseduralari yordamida
matn yorqinligini turli holatlarda ta'minlovchi dastur yaratng.

```
Uses CRT;  
begin  
  clrscr;  
  LowVideo;WriteLn('Past yorqinlik');  
  NormVideo;WriteLn('nirmal yorqinlik');  
  HighVideo;WriteLn('yuqori yorqinlik');  
  readln;  
end.  
uses crt;  
var  
  lm,t:word;  
const  
  col80=3;  
begin  
  lm:=lastmode; textmode(col80);  
  for t:=1 to 15 do  
    begin  
      textcolor(t);  
  
      writeln('123456789x123456789x123456789x123456789x');  
    end;  
    {print('rejim 40*25') }  
end.
```



Savol va topshiriqlar.

1. Paskalda ekran bilan ishlash uchun qanday moduldan foydalaniladi?
2. Paskalda asosan necha hil rang ishlatilishi mumkin?
3. Paskalda ishlatilishi mumkin bo'lgan qanday ranglarni bilasiz?
4. Matn rangini qanday operator orqali o'zgartiriladi?
5. Matn foni qanday operator orqali o'zgartiriladi?
6. ClrScr protsedurasi nima ish bajaradi?
7. Window protsedurasi nima ish bajaradi?
8. GotoXY protsedurasi nima ish bajaradi?
9. WhereX va WhereY funksiyalari nima ish bajaradi?
10. Ekran matn holatida necha satr va ustundan iborat?



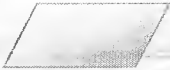
Mashqlar.


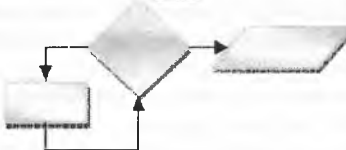



1. Ekranida ism va familiyangizni oq, foni rangini qora rangda chiqaring.
2. Ekranida oila a'zolaringizni ismlari turli ranglarda va ekranni och moviy rangida chiqaring.
3. Ekranida "O'zbekiston mustaqil davlat" matnini turli fon rangida va turli rangda chiqaring.
4. "REFERAT" matnini ekranning o'rta qismidan 20 qatorda ko'k rangda sariq fonda chiqaring?

9-§. Chiziqli dasturlar tuzish

Chiziqli dasturlar tuzilish jihatidan aniq ketma-ketlikdagi amallardan iborat. Bu amallarning bajarilishi dasturda keltirilish tartibi bilan bog'liq, ya'ni yozilgan ketma-ketlikda bajariladi. Har bir masalani echishdan oldin algoritimi tuzib olinadi. Algoritim tuzishda quyidagi geometrik figuralardan foydalaniladi.

Blok – sxema jadvali

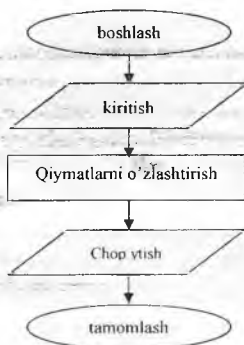
Shakl nomi	Geometrik figura	Vazifasi
Ishga tushirish , to'xtatish.		Algoritmning boshlash va tugatish;
Jarayon		Arifmetik ifodalarni hisoblash;
Kiritish		Boshlang'ich ma'lumotlarni kiritish;

Yechim		Boshqarishni shart asosida o'zgarishi;
Takrorlanish jarayoni		Takrorlanish jarayoni tasvirlash;
Avvaldan ma'lum jarayon		Qism dasturlarga murojaat qilish;
Chiqarish		Natijalarni tashqi qurilmalarga chiqarish;
Yurish		Amallarni bajarish yo'nalishi;

Chiziqli algoritmlarni blok-sxemasida

Chiziqli algoritmlar va dasturlar odatda juda sodda masalalarni yechishda qo'llaniladi. Bu masalalar yechimi biror shartga yoki siklik amallar bajarilishiga bog'liq emas.

Hech qanday shart tekshirilmaydigan va tartib bilan faqat ketma-ket bajariladigan algoritmlar chiziqli algoritmlar deb yuritiladi.



Chiziqli dastur deganda masalaning algoritmiga mos keluvchi buyruqlar ketma-ketligi dasturda uchrash tartibiga mos ravishda bajariladigan dasturlarni tushuniladi. Demak, dasturni bajarish boshlanganda, dastlab 1-buyruq, keyin 2-

buyruq va x.k. tarzida bajariladi. Bunda *begin* va *end* larga alohida e'tibor beriladi. Ular operatorlar qavsi hisoblanadi va amallarni bajarish paytida ustunlikka ega bo'ladi. Dasturning bitta satrida bir nechta buyruqlar kelishi mumkin. Bunda ular bir-biridan «;» belgisi bilan ajratiladi. Paskal tilidagi dastur qo'yilgan masalaning yechish algoritmidagi buyruqlarni kompilyatorga «tushunarli» bo'lgan ko'rinishda yuqoridagi umumiy sxemaga muvofiq ifodalash natijasida hosil bo'ladi.

Masalan, to'g'ri to'rtburchakning tomonlariga ko'ra uning perimetri, diagonali va yuzasini hisoblashni (a, b – tomonlar qiymatiga ko'ra) quyidagicha tashkillashtirish mumkin.

Algoritmni matn shakli	Blok sxemasi	Dasturi
1 Boshlansin; 2 Kiritilsin a,b; 3 Hisoblansin $p=2a+2b$; 4 Hisoblansin $d=\sqrt{(a^2+b^2)}$; 5 Hisoblansin $s=ab$; 6 Chiqarilsin p, d, s; 7 Tamomlansin.	<pre> graph TD Start([bosh]) --> Input[a,b-lar qiymatini kiritish] Input --> P["p := 2*a + 2*b"] P --> D["d := sqrt(sqrt(a)+sqrt(b))"] D --> S["s := a*b"] S --> Output[/p, d, s/] Output --> End([tam]) </pre>	<i>Program to'rtburchak yuzi;</i> <i>Var a, b: Integer;</i> <i>P, d, s: real;</i> <i>Begin</i> <i>Write ('a,b tomonlarni</i> <i>qiymatlari kiritilsin');</i> <i>ReadLn(a,b);</i> <i>P:=2*a+2*b;</i> <i>D:=sqrt(sqrt(a)+sqrt(b));</i> <i>S:=a*b;</i> <i>WriteLn('to'rtburchak</i> <i>perimetri=',p);</i> <i>WriteLn('to'rtburchak</i> <i>dioganperli=',d);</i> <i>WriteLn('to'rtburchak yu-</i> <i>zasi=',S);</i> <i>End.</i>

Dastur natijasi

```

a, b tomonlarni qiymatlari kiritilsin
1
2
3
to'rtburchak perimetri= 1.8000000000E+01
to'rtburchak dioganali= 6.4031242374E+00
to'rtburchak yuzasi= 2.0000000000E+01
  
```

Yuqoridagi masalani dastur natijasini olish uchun, avval Turbo Paskal muxitini olinga tushiramiz. So'ngra dastur matnini kiritamiz. Bu matnni xatoliklarga tekshiramiz, ya'ni kompilyasiya qilamiz. Buning uchun *ALT + F9* yoki *ALT + C* tugmalaridan foydalanamiz. Matnni kiritishda yo'l qo'yilgan xatoliklar mavjud bo'lsa, ularni bartaraf etamiz. Xatoliklar qolmagandan so'ng, dasturni bajarishga

ko'rsatma beramiz, buning uchun *CTRL + F9* yoki *ALT + R* tugmalaridan foydalanish mumkin.

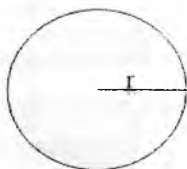
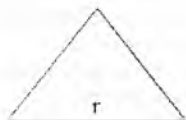


Savol va topshiriqlar.

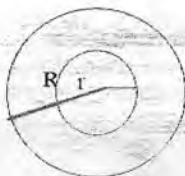
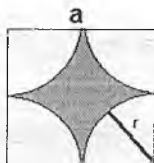
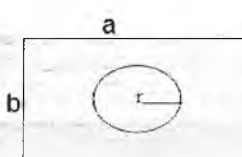
1. Chiziqli dastur deganda nimani tushinasiz?
2. Qanday operatorlardan foydalanib muloqot dasturlari tuzish mumkin?
3. Chiqarish formati qanday operator orqali beriladi?
4. Chiqarish formati xaqida so'zlab bering.
5. Ekrandagi natijani ko'rish uchun qaysi klavishlar bosiladi?

Mashqlar.

1. Kvadratning tomoni a ga teng, uning perimetri va yuzasini xisoblovchi dastur tuzing.
2. Tomonlari mos ravishda a, b, c bo'lgan ixtiyoriy uchburchakning yuzini Geron formulasi orqali hisoblash dasturini tuzing va $a=3, b=4, c=5$
3. Tomonlari r bo'lgan teng tomonli uchburchak, radiusi r ga teng doiraning yuzlarini, perimetrini va doirani uzunligini hisoblash dasturini tuzing va $r=5$ hol uchun.



4. Quyida berilgan shakllarning bo'yalgan sohalarini yuzasini xisoblash dasturini tuzing.



5. Ekranda "Kamil inson bo'lishimiz shart" degan matn chiqarish dasturini tuzing.
6. Boshlang'ich tezligi v_0 bo'lib, a tezlanish bilan tekis xarakat qilayotgan moddiy nuqtaning t vaqt ichida bosib o'tadigan yo'lni aniqlash dasturini tuzing.
7. Erkin tushayotgan jismning t vaqtda ichida bosib o'tgan yo'lni hisoblash dasturini tuzing.
8. Dastur tuzing.

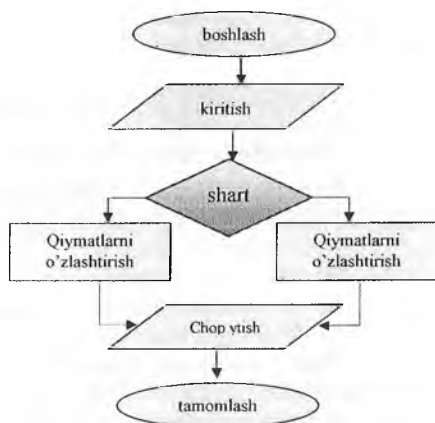
$$w = |\cos x - \cos y|^{(1-2 \sin^2 y)} \left(1 + z + \frac{z^2}{2} + \frac{z^3}{3} + \frac{z^4}{4} \right)$$

$$x=0.4 \times 10^4, y=-0.875, z=-0.475 \times 10^{-3}$$

10-§. Tarmoqlanish va o'tish operatorlari.

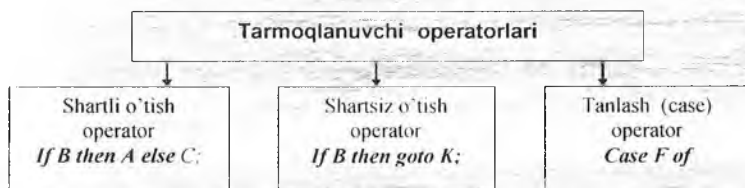
Turli masalalarni yechganda ko'rsatmalarni bajarish tartibi biror bir shartning bajarilishiga bog'liq holda bajariladi. Ya'ni algoritim tarmoqlanadi.

Blok-sxemasi



Shartga muvofiq bajariladigan ko'rsatmalar bilan tuziladigan algoritmlar *tarmoqlanuvchi algoritmlar* deyiladi.

Ma'lum bir shartni bajarilishi yoki bajarilmasligiga qarab, tarmoqlanuvchi jarayon holatlari aniqlanadi. Tarmoqlanuvchi jarayonlarni hisoblash uchun turbo paskal tilida quyidagi operatorlardan foydalanish mumkin:

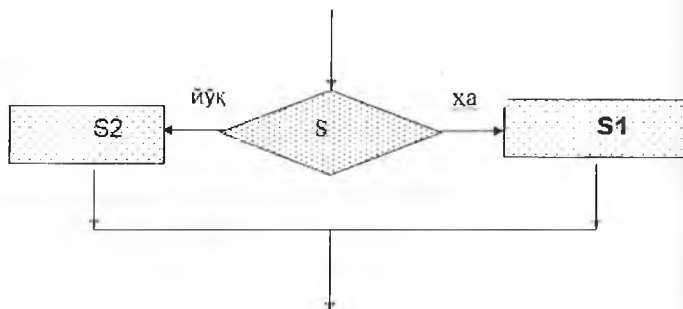


Shartli o'tish operatori.

Shartli operator ikki xil ko'rinishda bo'ladi:

- to'liq shartli operator;
- to'liqmas shartli operator.

To'la shartli operatorning algoritmik sxemasini quyidagi ko'rinishga ega:



To'liq shartli operator quyidagi formada yoziladi:

if <mantiqiy ifoda> *then* <operator> *else* <operator>

bu erda *if* (agar), *then* (u xolda), *else* (aks xolda) xizmatchi so'zlar.

Shunday kilib, to'liq shartli operatorni quyidagicha yozish mumkin:

if S *then* S1 *else* S2;

bu erda S - mantiqiy ifoda;

S1 – S mantiqiy ifoda rost qiymat qabul qilganda bajariluvchi operator;

S2 -S mantiqiy ifoda yolg'on qiymat qabul qilganda bajariluvchi operator.

Shartli operatorning bajarilishi unda yozilgan S1 yoki S2 operatorlaridan birini bajarilishiga olib keladi, ya'ni agar S mantiqiy ifoda bajarilishidan so'ng *true* (rost) qiymati hosil bo'lsa S1 operatori, aks holda esa S2 operatori bajariladi.

To'liq shartli operatorga doir misollar:

if a=2 *then* d:= x+2 *else* d:= x-2;

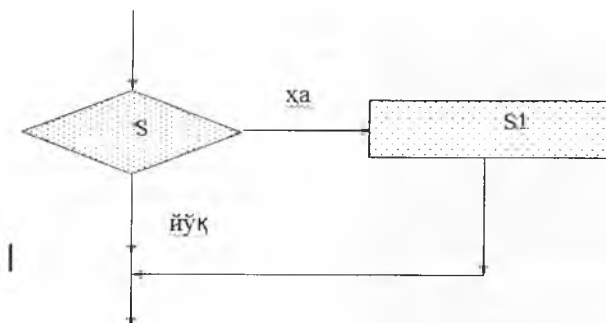
if (x<y) and (z>5) *then begin* y:= x * sin(x);

t:= x * cos(x) *end else begin* y:= 0; t:= 1 *end*;

if x<0 or x=3 *then* y:= x*x+1 *else if* x<2

then y:= sqrt(abs(x-1)) *else* y:= x*x;

Qisqa (to'liqmas) shartli operatorning algoritmik sxemasini quyidagi ko'rinishga ega:



Qisqa (to'liqmas) shartli operatorning yozilishini quyidagicha ifodalanadi:
 if S then S1;

bu erda S - mantiqiy ifoda, S1 - operator.

Agar S ifoda qiymati *true* (rost) bo'lsa S1 operatori bajariladi, aks holda esa boshqarish shartli operatoridan keyin yozilgan operatorga uzatiladi.

Shartli operatoridan foydalanishga misollar keltiramiz.

1-misol. Kiritilgan ixtiyoriy butun sonni juft yoki toqligini aniqlovchi dastur yarating.

```

program r1;
var x:integer; javob:string;
begin
  readln(x);
  if x mod 2 =0 then javob:='juft' else javob:='to'q' ;
  writeln('kiritilgan son ',javob);
  readln;
end.
  
```

2-misol. Kiritiladigan ixtiyoriy a,b,c sonlar uchun $a+b>c$, $a+c>b$, $b+c>a$ tengsizliklarning barchasi bajarilganda «shartlar qanoatlantirilgan» deb javob beruvchi dastur yarating.

```

program r2;
uses crt;
var a,b,c,d:integer;
    javob:string;
begin
  clrscr;
  readln(a,b,c);
  if (a+b>c) and (a+c>b) and (b+c>a) then
    writeln('shartlar qanoatlantirilgan');
  readln;
end.
  
```

Yuqorida keltirilgan ikki misoda masala shartiga ko'ra shartli operatorning to'liq va to'liq bo'lmagan holatlaridan foydalanildi.

Tarmoqlanish operatoridan foydalanishda quyidagi qoidalarga amal qilish shart:

IF operatoridan foydalanganda ELSE dan oldin «;» (nuqta-vergul) qo'yilmaydi.

Shartli operator Then va ELSE xizmatchi so'zlaridan keyin bir necha operator (amal yoki buyruq) ishlatilishi zarurati bo'lsa, u holda bu buyruqlar begin va end qavslari ichiga joylashtirishi shart.

(Shartsiz o'tish operatorini o'rganishda $ax^2+bx+c=0$ tenglamaning echimlarini aniqlovchi dastur keltirilgan, shu holatga e'tibor bering).

Shartsiz o'tish operatori.

Shartsiz o'tish operatori *goto* quyidagicha yoziladi:

goto belgi;

Bu erda *goto* xizmatchi so'z bo'lib, belgi operator boshqarishni uzatishi zarur bo'lgan (belilangan) «manzili» hisoblanadi. Belgi sifatida Turbo Paskal dasturlash tilida 0 dan 9999 gacha bo'lgan butun sonlardan va simvollar birikmasidan(xizmatchi so'zlardan tashqari) foydalanish mumkin. Belgilar dasturning tavsiflash qismining Label (nishonlar ro'yxati) bo'limida beriladi, masalan: Label 12, bel, r1;

Yuqoridagi operatorlardan foydalanib, $ax^2+bx+c=0$ tenglamaning echimlarini aniqlovchi dastur yaratamiz:

```
uses crt;
```

```
label 12,13,aa,2;
```

```
var a,b,c:integer; x, d:real;
```

```
begin
```

```
  clrscr;
```

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

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

```
    if d=0 then goto aa else goto 12;
```

```
  aa:x:=-b/(2*a);
```

```
    writeln('x=',x:2:0); goto 13;
```

```
  12: if d>0 then
```

```
    begin
```

```
      writeln('x1=',(-b+sqr(d))/2*a:2:0);
```

```
      writeln('x2=',(-b-sqr(d))/2*a:2:0);
```




```
    end
```

```
  else
```

```
    writeln('tenglama haqiqiy echimlarga ega emas');
```

```
  13:end.
```

Keltirilgan dasturni turli holatlarda tenglamaning ildizlarini aniqlashini ko'rib chiqamiz.

Tenglamaning ko'rinishi	Dastur natijasi
$x^2 - 8x + 16 = 0$ ($A=1, b=-8, c=16$) qiymatning readln(a,b,c) buyrug'i orqali kiritilishiga e'tibor qarating.	
$x^2 + 5x - 6 = 0$ ($A=1, b=5, c=-6$) qiymatning readln(a,b,c) buyrug'i orqali kiritilishiga e'tibor qarating.	
$5x^2 + x + 6 = 0$ ($A=5, b=1, c=6$)	

Goto operatoridan foydalanishda quyidagi qoidalarga amal qilish shart:

Goto operatori boshqarishni uzatuvchi belgi nomi albatta tavsiflash bo'limida ko'rsatilishi va u dasturning kerakli joyida «:» bilan ajratilgan holda aniqlanishi shart. (ko'rsatilgan misoldagi 12: if d>0 then... kabi)

Goto operatori boshqarishni uzatuvchi belgi nomi tavsiflash bo'limida ko'rsatilishi va u dasturning asoiy qismida foydalanmaslik mumkin. (ko'rsatilgan misolda, label 12,13,aa,2; da aniqlangan «2» belgisidan dasturda foydalanilmagan).



Savol va topshiriqlar.

1. Tarmoqlanuvchi algoritm deb nimaga aytiladi?
2. O'tish operatori nima uchun qo'llaniladi?
3. O'tish operatorining umumiy ko'rinishi qanday?
4. Dasturda nishonlar nima uchun ishlatiladi?
5. Nishonlar ishlatilgan dasturda o'tish operatori ishlatilmasligi mumkinmi?
6. O'tish operatori ishlatilgan dasturda nishonlar ishlatilmasligi mumkinmi?
7. Tarmoqlanuvchi operator nima uchun qo'llaniladi?
8. Tarmoqlanuvchi operatorida operatorlar ketma-ketligi ishtirok etsa, ular qanday so'zlar orasida yoziladi?
9. Tarmoqlanuvchi operatorning qisqa va to'liq ko'rinishlari haqida nimalar bilasiz?
10. Qaysi operatoridan keyin nuqtali vergul yozilmaydi?



Testlar.

- Quyidagilardan xato yozilganini toping.
 - Goto 10;
 - Goto 30;
 - Goto -5;
 - GoTo _5;
 - Goto sin;
 - Goto 2 _5;
 - GOTO a _5;
- Tarmoqlanish operatori uchun quyidagi shartlardan xato yozilganini toping.
 - $a < b$;
 - $a < -b$;
 - $a < b$;
 - $-a > 0$;
 - $-1 > 0$;
 - $a > b$;
 - $a = b$;
- Quyidagilardan xato yozilganini toping.
 - If $a = b$ then $a := a + 1$; else $b := a$;
 - If $a = 1$ then $a := a + 1$; else $b := a$;

Mashqlar.

- $\begin{cases} 1, & ax^2 + px > 0 \\ x^2, & ax^2 + px \leq 0 \end{cases}$ funkcionini hisoblovchi dastur tuzing.
- Uchta son berilgan ular ichida manfiy sonlarning kubini hisoblovchi dastur tuzing.
- Uchta a, b, c son berilgan $a < b < c$ tengsizlikni bajarilishini tekshiruvchi dastur tuzing.
- Berilgan uchta kesmadan uchburchak hosil qilish mumkin yoki mumkin emasligini aniqlovchi dastur tuzing.

11-§. Tanlash (Case) operatori.

Bu operator bir necha yo'nalish bo'yicha tarmoqlanishni ta'minlab beruvchi (tanlashni amalga oshiruvchi) operator hisoblanadi. Uning umumiy ko'rinishi quyidagicha:

Case <tanlash indeksi-kalid> *of* <tanlash holatlari ro'yxati-elementlari> *else* <operatorlar >*end*;

Bu erda *Case*, *of*, *else* va *end* paskalning xizmatchi so'zlari; <tanlash indeksi-kalid> - sonli, belgili yoki matiqiy o'zgaruvchi yoki ifoda; <tanlash holatlari ro'yxati-elementlari> - tanlash indeksi-kalitiga mos qiymatlar.

Tanlash indeksi sifatida haqiqiy tipdan foydalanish mumkin emas va bu indeks tanlash holatlari ro'yxatidagi mos buyruqlni bajarilishini ta'minlaydi. *Case* operatoridan foydalanishni quyidagi misollarda ko'rib chiqamiz:

1-misol. «Sadaf» kichik tadbirkorlik firmasi bir kecha kunduzda W kVt/soat elektr energiyasini sarflaydi. Bu firmaning 2011 yining kerakli oylari uchun elektr energiyasini sarflash miqdorini aniqlang (Tanlash indeksi sifatida butun tipdan foydalanish).

const yil=2011;

var W,R:real; j:word;

begin

writeln('Oyning tartib raqami va bir kecha-kunduzdagi');

writeln('sarflanadigan energiya miqdorini kiriting!');

```

readln(j,W);
case j of
  1,3,5,7,8,10,12: r:=31*W;
  4,6,9,11: r:=30*w;
  2: if (yil mod 4=0) then r:=29*W else r:=28*W;
else writeln('oy tartib raqami xato kiritilgan')
end;
if (j>0) and (j<13) then
begin
  writeln(j, '-nchi oyda ', r:6:2,'kvt/s miqdorda');
  writeln(' elektr energiyasi sarflangan');
end;
end.

```

Keltirilgan dasturning bir qismiga izoh keltiramiz:

<pre> case j of 1,3,5,7,8,10,12: r:=31*W; 4,6,9,11: r:=30*w; 2: if (yil mod 4=0) then r:=29*W else r:=28*W; else writeln('oy tartib raqami xato kiritilgan') end; </pre>	<p>Kiritilgan oy tartib raqami j-ga ko'ra tanlash boshlanadi(oydagi kunlar miqdoriga nisbatan) va shunga mos R-ning qiymati xisoblanadi</p>
--	---

2-misol. Uchburchak tomonlarini ifodalovchi uchta son kiritilib bu sonlar uchburchak tomonlari bo'lganda uning perimetrinida aks holda ularning uchburchak tomonlari bo'la olmasligi haqidagi ma'lumotni beruvchi dastur yaratilgan(Tanlash indeksi sifatida mantiqiy tipdan foydalanish).

```

Var t:boolean;
    a,b,c:real;
begin
  writeln ('uchburchak tomonari uzunligini kiriting');
  readln(a,b,c);
  t:=(a+b>c) and (a+c>b) and (b+c>a);
  case t of
    true: writeln('bunday uchburchak mavjud va uning perimetri (a+b+c):4:0.' ga teng');
    false :writeln('bunday uchburchak mavjud emas');
  end;
  readln;
end.

```

Dastur natijasi:

```
TPX.EXE
Turbo Pascal Version 7.1 Copyright (c) 1983,97 Borland International
uchburchak tomonari uzunligini kiriting
10
11
12
bunday uchburchak mavjud va uning perimetri 33 ga teng
```

3-misol. Kiritilgan a va b haqiqiy sonlar uchun qo'shish, ayirish, ko'paytirish va bo'lish belgisini kiritilganda bu sonlar ustida mos arifmetik amallarni bajaruvchi dastur yarating (Tanlash indeksi sifatida char tipidan foydalanish).

```
uses crt;
var
  t:char;
  a,b:real;
begin
  writeln ('ikkita sonni kiriting');
  readln(a,b);
  writeln( 'arifmetik amallarga moc belgini kiriting' );
  readln(t);

  case t of
    '+': writeln(a+b);
    '-': writeln(a-b);
    '*': writeln(a*b);
    '/': writeln(a/b);
    '^': writeln(a/b)
    else writeln ('arifmetik amallarga moc belgini kiriting');
  end;
end.
```

Dastur natijasi:

```
TPX.EXE
ikkita sonni kiriting
2011
2011
arifmetik amallarga moc belgini kiriting
+
2011.00*2011.00=4044121.00
```



Savol va topshiriqlar.

1. Tanlash operatorini vazifasini aytib bering.
2. Tanlash operatorining tarmoqlanish operatoridan farqini aytib bering.
3. Tanlash operatorining else qismi bo'lgan va else qismi bo'lmagan shakllari haqida so'zlab bering.

4. Tanlash operatorida operatorlar ketma-ketligi tanlansa, ular qanday so'zlar orasida yoziladi?
5. Tanlash operatorini tarmoqlanish operatori bilan almashtirish mumkinmi? Javobingizni izoxlang.

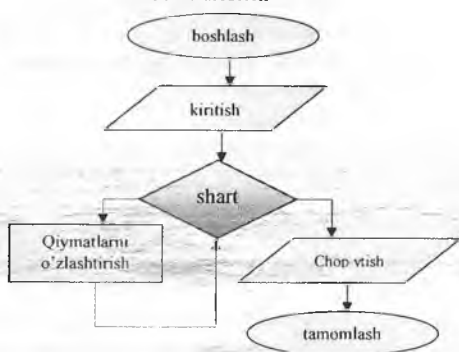
Mashqlar.

1. Butun sonli *xafta kuni* o'zgaruchisining qiymatiga qarab hafta kuning nomini chiqaruvchi dastur tuzing.
2. kiritilgan tartib raqamiga ko'ra yildagi oylar nomini ekranga chiqaruvchi dastur tuzing.
3. Oila a'zolaringizga mos ularning tug'ilgan yili va yoshini aniqlovchi dastur tuzing.
4. Kiritilgan hafta kuniga qarab, shu kunning dars jadvalini ekranga chiqaruvchi dastur tuzing.
5. Kiritilgan havo haroratiga ko'ra "yomg'ir yog'moqda", "qor yog'moqda", "yog'ingarchilik bo'lmayapdi", "issiq" va "sovuq" javoblardan birini ekranga chiqaruvchi dastur tuzing.

12-§. Parametrik takrorlash operatori.

Yuqorida keltirilgan masalalarning umumiy xususiyati shundan iboratki, bu masalalarda amallar bir marta bajariladi yoki umuman bajarilmaydi. Lekin ba'zi bir masalalarda biron-bir amalni bir-necha marta bajarishimizga to'g'ri keladi. Ya'ni jarayon takrorlanadi.

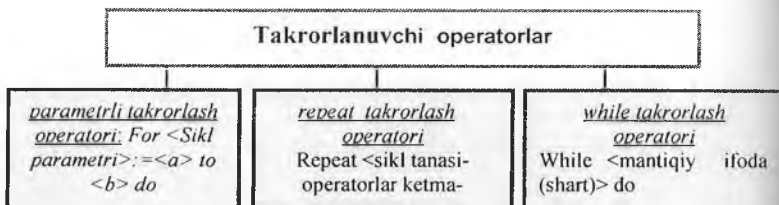
Blok-sxema



Masalani echish jarayonida bir hil amallarni takrorlanishini talab qiluvchi algoritmlar **takrorlanuvchi algoritmlar** deyiladi.

Yechilayotgan masalaning mohiyatiga qarab, dasturchi tuzuvchi o'zi uchun qulay bo'lgan takrorlash operatorini tanlab olishi mumkin.

Takrorlash operatorlarining 3 xil turi mavjud:



Dasturlash jarayonida ayrim hollarda bir yoki bir necha amallarni bir necha marotaba takrorlab bajarish zarurati tug'iladi. Masalan, $1+2+\dots+2011$ yig'indini hisoblashimiz hisoblash uchun quyidagi dastur tuzadigan bo'lsak,

```
..
a:=1; S:=s+a;
a:=2; S:=s+a;
a:=3; S:=s+a;
```

a:=4; S:=s+a; va hokazo, ya'ni dasturimiz «uzundan-uzun» ko'rinishga ega bo'lar edi. E'tibor bilan qaraydigan bo'lsak a-o'zgaruvchi har safar 1-ga ortib borib, ega bo'lgan qiymati S-ga qo'shilmogda. Aynan shunday hollar uchun parametrlil sikllardan foydalanish dasturchining ishini engillashtiradi.

Parametrik sikllarning umumiy ko'rinishi quyidagicha:

```
For <Sikl parametri>:=<a> to <b> do
< operator yoki operatorlar>
```

Bu erda a-parametr bosh qiymatiga, b-parametr oxirgi qiymatiga teng.

Misol uchun:

```
For i:=1 to 23 do
  s:=s+1/i;
```

Siklning bu holatida parametr i-ning qiymati dastlab 1-ga teng bo'lib, so'ngra siklning har bir qadamida '+1'-ga orta boradi va 2,3,...,23 ga teng bo'ladi. Zarur hollarda parametrning qiymatini '-1' orttirish mumkin bo'lib, bunda «to» o'rniga «downto» ishlatiladi.

Misol uchun:

```
For k:=30 downto 1 do
begin W:=W+sqrt(k); R:=r+sqrt(k); end;
```

<operator> ko'rinishidagi sikl bo'lib, ayrim xollarda ham ishlatiladi. Sikl parametrining qiymati faqat butun sonlardan iborat va sikl qadami doimo birga teng.

Parametrik sikllarning o'ziga xos xususiyatlari quyidagilardan iborat:

For siklidan takrorlanishlar soni aniq bo'lgan hollarda foydalanish maqsadga muvofiqdir.

Sikl parametri qiymati +1 yoki -1 ga avtomatik tarzda oshiriladi («to» yoki «downto» ishlatilishiga ko'ra).

Sikl parametri sifatida butun, belgili, mantiqiy yoki sanoq tiplaridan foydalanish mumkin.

Sikl bir necha amalni bajarishga mo'ljallangan bo'lsa, sikl tanasida bu amallar «begin» va «end» qavslari ichida berilishi shart (1-misolga qarang).

Parametrik takorlanishlar «ichma-ich» joylashishlari ham mumkin va bu holat juda ko'p masalalarni echishda qo'llaniladi.


Masalan:

```
for t:=1 to 3 do
  for k:=1 to 5 do
    writeln(t,k);
```

Bu sikllarni aniqroq tasavvur yitish uchun quyidagi dastur va uning natijasini taqqoslaymiz:

t-parametrlning qiymati 1-ga teng bo'lganda, k-parametr 1,2,3,4,5 qiymatlarni qabul qiladi.

t-parametrlning qiymati 2-ga teng bo'lganda, k-parametr yana 1,2,3,4,5 qiymatlarni qabul qiladi va hokazo.

<i>Dasturni tuzilishi</i>	<i>Natija</i>
<pre>uses crt; var t,k:byte; begin clrscr; writeln('t',' ',k'); writeln('----'); for t:=1 to 3 do for k:=1 to 5 do writeln(t,' ',k); end.</pre>	

1-misol. 'A' dan 'Z'-gacha va 'z' dan 'a'-gacha bo'lgan barcha simvollarni chop etuvchi dastur tuzing.

Dastur tuzilishi:

```
var i:char;
begin
  for i:='A' to 'Z' do
    write(i,' ');
  writeln;
  for i:='z' downto 'a' do
    write(i,' ');
  readln;
end.
```

Dastur natijasi:



```
TPX.EXE
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
a b c d e f g h i j k l m n o p q r s t u v w x y z
```

2-misol. Raqamlari yig'indisi 8-ga teng bo'lgan barcha ikki xonali sonlarni aniqlab, chop etuvchi dastur yarating.

Dastur tuzilishi:

```
uses crt;
var i:10..99;
    a,b:0..9;
begin
  clrscr;
  for i:=10 to 99 do
  begin
    a:=i div 10;
    b:=i mod 10;
    if a+b=8 then write(i,' ');
  end;
end.
```

Dastur natijasi:



```
TPX.EXE
17 26 35 44 53 62 71 80
```



Savol va topshiriqlar.

1. Takrorlash operatorlarining necha xili mavjud?
2. Takrorlanuvchi algoritimga misol keltiring.
3. Parametrlilik takrorlash operatorining ko'rinishi qanday?
4. Parametrlilik takrorlash operatori qachon qo'llaniladi?
5. Takrorlash parametri qanday qiymatlarni qabul qiladi?
6. Takrorlash parametrining qiymatlari chegaralanganmi?
7. Takrorlash operatorini ishlashini tushintiring.
8. Qanday holda Do yoki Downto xizmatchi so'zlar ishlatiladi?

Mashqlar.

1. Quyidagi operatorlardagi takrorlanishlar sonini aniqlang.
a) For i:=1 to 100 do b:=1

- b) For i:=231 to 63100 do t:=2
- c) For i:=-29 to 1 do h:=3
- d) a:=4; b:=18; For i:=a+10 to b-1 do z:=z+2;
- e) a:=2; b:=20; For i:=a*a to 3*b+1 do z:=z+2;

2. Funktsiyaning qiymatini X ning -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5 qiymatlarida hisoblash dasturini tuzing.
3. Ekranga 1dan 50 gacha bo'lgan sonlarni o'sish tartibida chiqaruvchi dastur tuzing.
4. A[1...17] massiv berilgan. Massivning nolga teng elementlarini indeksini chiqaruvchi dastur tuzing.

13-§. Shart bo'yicha takrorlash operatorlari.

Parametrli takrorlanuvchi operatorida takrorlanishlar soni aniq edi. Shunday masalalar ham borki, unda biror amallar ma'lum bir shart bajarilmaguncha takrorlash kerak bo'ladi. Bunday hollarda shart bo'yicha takrorlash operatorlaridan foydalaniladi. Shart bo'yicha takrorlash operatorlari ikki xil ko'rinishda bo'lib ular quyidalardan iborat:

a) repeat sikli (takrorlanadigan amallar kamida bir marotaba bajarilib so'ngra shart tekshiriladi).

bu takrorlashning tuzilishi quyidagicha:

Repeat <sikl tanasi-operatorlar ketma-ketligi>

Until <shart (mantiqiy ifoda)>

Bu erda <operatorlar ketma-ketligi> bajarilishi lozim bo'lgan amallar yoki sikl tanasida joylashgan operatorlar majmui, <shart> takrorlanishi, bajarilishi yoki to'xtatilishini boshqaruvchi shartdan iborat. Bu xil ko'rinishdagi sikl hech bo'lmaganda bir marotaba bajariladi, negaki operatorlar ketma-ketligi shartni tekshirishdan oldin yozilgan.

Izoh : *Repeat* va so'zlari qo'shimcha tarzda *begin, end* qavslar vazifalarini ham bajarishadi.

Repeat takrorlash operatorini bajarilishini quyidagi masala yordamida ko'rib chiqamiz:

Masala. $y=ax^2$ funktsiya qiymatlarini $x=0$ dan $x=5$ gacha 0,5 qadam bilan xisoblovchi dastur yarating.

Masala shartiga ko'ra, foydalanuvchi faqat a -ning qiymatini kiritishi dastur esa $y=ax^2$ funktsiya qiymatini 0,5 qadam bilan hisoblashi zarur. $a=2$ qiymat uchun natija quyidagicha bo'lishi zarur, ya'ni dastavval $x=0$ da funktsiya qiymati hisoblanishi (chop etilishi), so'ngra x -ning qiymati 0,5 ga o'ttirilishi va hosil bo'lgan qiymat 5-dan katta bo'lmasligi tekshirilishi zarur (quyidagi jadvalga e'tibor bering).

X	0	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
Y	0	0.5	2	4.5	8	12.5	18	24.5	32	40.5	50

Repeat takrorlash operatorini blok-sxemasi va dastur tuzilishiga etibor bering.

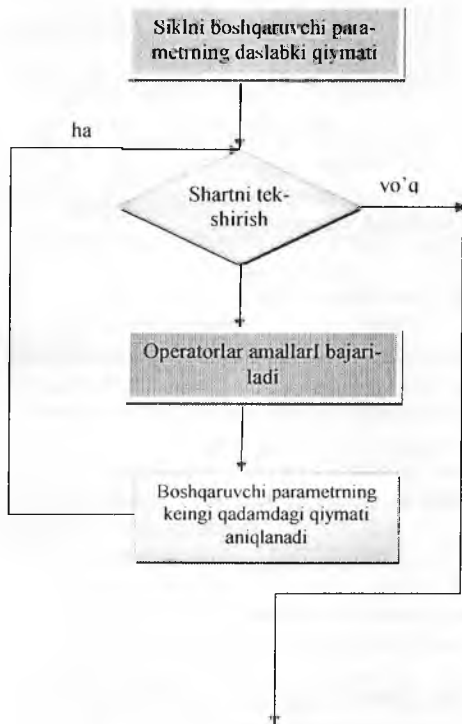
<i>Repeat takrorlash operatori blok-sxemasi</i>	<i>Dastur</i>	<i>Dastur natijasi:</i>
	<pre> var x,a,y:real; begin readln (a); x:=0; repeat y:=a*sqr(x); writeln(y:5:2); x:=x+0.5; until (x>5) end. </pre>	

b) while sikli(takrorlanadigan amallar bajarilishi uchun avval shart tekshiriladi). Bu takrorlashning tuzilishi quyidagicha:

While <mantiqiy ifoda (shart)> do
 <operator – sikl tanasi>

Bu erda mantiqiy ifoda (shart) qiymati True bo'lguncha sikl tanasidagi operatorlar bajariladi, aks hollarda sikl tanasidagi amallar bajarilmaydi.

While shartli tarorlash blok-sxeması



Dastur-misol

```

var x:integer;
begin
x:=-5;

```

```

while (x<15) do

```

```

begin
write(sqrt(x), ' ');

```

```

x:=x+2;

```

```

end;

```

```

end.

```

Dastur natijasi:



Savol va topshiriqlar.

1. Shart bo'yicha takrorlash operatorlarining parametrik takrorlash operatoridan farqi nimada?
2. Shart bo'yicha takrorlash operatorlarini sanab bering.
3. Repeat operatorining ishlashini tushintiring.
4. While operatorining ishlashini tushintiring.
5. Repeat operatorining While operatoridan farqi nimada?
6. Qanday holda takrorlash tanasida begin va end yozish shart emas?

Mashqlar.

1. Kvadratlari berilgan natural sondan katta bo'lmagan sonlarni chiqaruvchi dastur tuzing.
2. N ta haqiqiy son berilgan. Ularni orta borish tartibida joylashtiring.
3. Barcha uch xonali natural sonlardan raqamlari yig'indisi berilgan n ($1 \leq n \leq 27$) ga tenglarini chop eting.
4. Quyida funktsiyalarning qiymatlarini berilgan oraliqda berilgan qadam bilan hisoblash dasturini tuzing.

$$A) y = \frac{e^x + e^{-x}}{2} \quad [-9; 9], h=0.2$$

$$B) y = \cos x \quad [-\pi/18; 4\pi], h=10^\circ$$

$$V) y = (1 + 2x^2)e^{x^2} \quad [-5; 10], h=0.3$$

$$G) y = \frac{1+x^2}{2} \arctg x - \frac{x}{2} \quad [-9; 9], h=0.2$$

$$D) y = e^{2x} \quad [-8; 9], h=0.1$$

$$E) y = \left(\frac{x^2}{4} + \frac{x}{2} + 1 \right) e^{2x} \quad [-1; 9], h=0.4$$

5. Quyidagi yig'indining qiymati berilgan natural sondan ortiq bo'lguncha hisoblash dasturini parametrlri, shart bo'yicha takrorlash operatorlari yordamida tuzing.

$$A) y = x - \frac{x^3}{3} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1}$$

$$B) y = 1 - \frac{3}{2}x^2 + \dots + (-1)^n \frac{2n^2 + 1}{(2n)!} x^{2n}$$

$$V) y = 1 + 2\frac{x}{2} + \dots + \frac{n^2 + 1}{n!} \left(\frac{x}{2} \right)^n$$

$$G) y = 1 + \frac{x^2}{2!} + \dots + \frac{x^{2n}}{(2n)!}$$

14-§. Belgili miqdorlar bilan ishlash funktsiyalari.

Kompyuter hotirasidagi har bir belgi o'zining kodi orqali ifodalaniladi. U ASCII (American Standard Code for Information Interchange) kodi deb yuritiladi. Paskal dasturlash tilida ham. barcha dasturlash tillari kabi belgilarning ASCII

kodlarini aniqlovchi va aksincha, ASCII kodiga mos belgini aniqlab beruvchi standart funktsiyalar qo'llaniladi.

Shar belgili tipi maxsus jadvalning elementlari to'plamiga tegishli bo'lib, dastur tarkibida apostroflar yoradmida tasvirlanadi. Masalan, 'A', ' ' (bo'shliq) , '\n'. Belgili tipni ifodalovchi o'zgaruvchi xotirada 1 bayt joyini egallaydi. Bu o'zgaruvchilar dasturning o'zgaruvchilar bo'limida tavsiflanadilar:

Var <o'zgaruvchi nomi> : char;

Misol.

Var Ch : char;

Belgi, Symbol: char;

Quyidagi standart funktsiyalar belgili tipilar bilan amallar bajarishda foydalanadilar:

1. Chr(x) - byte tipiga mansub x-o'zgaruvchini unga mos qo'yilgan simvolga «aylantirib» beradi;

program sim;

var x:byte;

begin x:=84; writeln(char(x));readln;end.

Dastur natijasi- T . 84 ga 'T' sivolga mos qo'yilganini anglatadi.

2. Ord(belgi) –belgili tipga mansub «belgi» unga mos kodga «aylantiradi» va uning qiymatini beradi.

program sim;

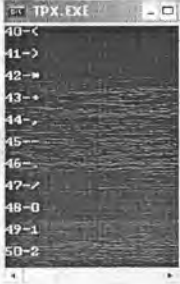
var belgi:char;

begin belgi:='T';writeln(ord(belgi));readln;end.

Dastur natijasi-84. 'T' sivolga mos qo'yilgan kod 84 ekanini anglatadi.

3. UpCase(belgi) – kichik harfda berilgan «belgi»ni bosh harfga o'tkazishda foydaniladi. Masalan: UpCase('d')='D'

ASCII jadvali belgilar- raqamlar, harflar va boshqa alomatlar ro'yxatini qamrab oladi. Quyidagi dasturni ishga tushirish orqali siz bu ro'yxat bilan to'liq tanisha olasiz:

Dastur tuzilishi	Dastur natijasining bir qismini keltirib o'tamiz:
<pre>var k:word; begin for k:=0 to 255 do begin write(k,'-',chr(k)); writeln; writeln; end; end.</pre>	

Jadvaldagi belgi nomeri uning kodini anglatadi. Dastur orqali kerakli belgini '<belgi>' - apostroflar ichida yozish bilan, yoki # belgisidan keyin kodini yozish bilan tasvirlash mumkin. Masalan, write('z');writeln(#49); buyruqlarining natijasi "z1" ga teng. Belgili tip Turbo Paskalda char tipi bilan aniqlanadi va u xotirada bir bayt joyni egallaydi. qorida keltirilgan dasturda chr –funksiyasi berilgan kod bo'yicha belgini aniqlash imkonini beradi.

1-masala. Dastur bajarilishi jarayonida belgili tipdan foydalanib 1dan 100 gacha bo'lgan butun sonlar yig'idasini xisoblovchi dastur tuzing(belgi sifatida "h!" dan foydalaning).

```
uses crt;
var sim:char; s:integer; i:1..100;
begin
clrscr;
writeln('1 dan 100 gacha bo'lgan sonlar yig'idisini aniqlash uchun');
writeln(' h-tugmasini bosing');
sim:=readkey; if sim='h' then
begin
s:=0;
for i:=1 to 100 do
s:=s+i;
writeln('natija= ',s);
end
else writeln('siz boshqa tugmadan foydalandiz');
end.
```

Dastur natijasi:

```
1 dan 100 gacha bo'lgan sonlar yig'idisini aniqlash uchun
h-tugmasini bosing
natija= 5050
```

Paskalda satrlar "massiv xususiyati" ga ega. YA`ni satr-massiv deb, satrdagi belgilar esa massivning elementlari deb qaraladi. Masalan, s-satrlı o'zgaruvchi bo'lsa, s[3]-shu satrning 3-belgisini bildiradi. YA`ni s:= 'monitor' bo'lsa, s[3]-"n".



Savol va topshiriqlar.

1. Chr funksiyasi qanday vazifani bajaradi?
2. Chr funksiyasining argumenti qanday qiymatlarni qabul qilishi mumkin? Javobingizni izohlang.
3. Ord funksiyasi qanday vazifani bajaradi?
4. Ord funksiyasining qiymatlar chegarasi qanday? Javobingizni izohlang.

5. *UpperCase* funksiyasi qanday vazifani bajaradi?
6. Satrli o'zgaruvchilarning "massiv xususiyati" nimani anglatadi?

Mashqlar.

1. Lotin "A" harfining ASCII kodi 65 ekani ma'lum bo'lsa, "ASAD" so'zini undagi harflarning ASCII kodlari orqali tasvirlang.
2. Kiritilgan satrni "alifboni o'nga bitta surish" usuli yordamida kodlash dasturini tuzing.
3. Alifboni o'nga bitta surish usuli bilan kodlangan matnning aslini aniqlovchi dastur tuzing.
4. Kiritilgan so'zning belgilarini ASCII kodini 16 lik sanoq sistemasida ifodalovchi dastur tuzing.
5. Kiritilgan belgining ASCII kodini aniqlab, ekranga chiqaruvchi dastur tuzing.
6. Berilgan satrni massiv, satrdagi belgini massivning elementlari deb qaralganda, satrdagi belgilarning ASCII kodlarini ekranga chiqaruvchi dastur tuzing.

15-§. Satrli miqdorlar bilan ishlash funksiyalari va protseduralari.

Turbo Paskal dasturlash tilida String tipidan matnli ma'lumotlar bilan ishlashda foydalaniladi. Bu tipning har bir elementi xotirada 1 bayt joy egallovchi belgilardan iborat bo'ladi. Satriy o'zgaruvchida belgilar miqdori 0 dan 255 tagacha bo'lishi mumkin. Satriy kattalik o'zgaruvchilar (Var) bo'limida quyidagicha e'lon qilinishi mumkin.

```
var d:string;
    satr:string [15];
```

Bu erda d –kattalikning uzunligi 0 dan 255 tagacha, satr-kattalikning uzunligi esa 0 dan 15 gacha bo'lishi nazarda tutilgan. Har qanday satriy tipidagi kattalikda birinchi baytning indeksi 0 ga teng bo'lib, unda satriy kattalik uzunligi haqidagi ma'lumot joylashadi. Satriy kattalikning birinchi belgisi ikkinchi baytni egallaydi va uning indeksi birga teng.

Satrli miqdorlaning elementlari kattalik nomi va kvadrat qavslarga olingan elementning tartib nomeri yordamida aniqlanadilar.

Misol:

Satr:='Geologiya' bo'lsa, satr[1]='G'-ga, satr[2]='e'-ga va h.

Quyidagi dastur yordamida satriy kattalikning uzunligini uning birinchi baytiga ko'ra, ya'ni 0-indeks orqali aniqlaymiz.

```
var t:string;i:byte;
```

```
begin
```

```
t: 'informatika';i:=ord(t[0]);writeln(i);end.
```

Dasturdagi ord(t[0]) protsedurasi t-kattalikning uzunligini aniqlashda yordam beradi. Bu qiymatni Length(t) funksiyasi yordamida ham aniqlash mum-

kin. Uzunlik qiymati mazkur kattalik uchun 11-ga tengligi dastur natijasidan aniqlanadi. Quyidagi dasturni yordamida satriy kattalikning har bir elementini alohida «ko'rishimiz» mumkin:

```
var t:string; i,k:byte;
begin
t:='informatika';i:=ord(t[0]);
for k:=1 to i do
writeln(k,t[k]);
end.
```

Endi satriy kattaliklar ustida ayrim funktsiya va protseduralar yordamida bajaraladigan amallar bilan tanishamiz:

Concat(s1,[s2,...,sN]) – funktsiyasi (tipi String) s1,s2,...,sN satriy kattaliklarni «yig'indisini» beradi (bu funktsiya vazifasini '+' amali ham bajaradi)

Misollar keltiramiz:

Ifoda	Natija
'r1+'r2+'r3+'...+'rn'	'r1+r2+r3+...+rn'
Concat('Rezistor ', '-qarshilik ',degan ma''noni ',anglatadi')	'Rezistor-qarshilik degan ma''noni anglatadi'

Copy (st, t1, d) – funktsiyasi (tipi String) st satriy kattalikning t1 tartib raqamli belgisidan d-ta simvolni (t1-dan boshlab) nusxalash imkonini beradi.

Misollar keltiramiz:

St-qiymati	Ifoda	Natija
Elektr qarshilik o''tkazgich uzunligiga bog''liq'	Copy(St,8,10)	'qarshilik'
'Mol-modda miqdori'	Copy(St,11,6)	'miqdor'

Delete (st, t1, d) - protsedurasi st satriy kattalikning t1 tartib raqamli belgisidan d-ta simvolni (t1-dan boshlab) o'chirish imkonini beradi. Delete protsedurasidan foydalanganda uning uzunligi kamayadi.

Misollar keltiramiz:

St-qiymati	Operator	Natija St-ning yangi qiymati
'abcdefg'	Delete(S,3,2)	'abefg'
'abcdefg'	Delete(S, 2,6)	'a'

length(st) - funktsiyasi (tipi Integer) st satriy kattalikning uzunligini aniqlaydi.

Misollar keltiramiz:

St-qiymati	Ifoda	Natija
'Zomin tog''-o''rmon qo''riqxonasi'	Length(St)	30
''''	Length(St)	1
'mediana'	Length(St)	6
'(a+b)*c'	Length(St)	7

Pos (st1, st) - funksiyasi (tipi Integer) st satriy kattalikning tarkibida st1 satriy kattalik "joylashgan" bo'lsa, uning tartib raqamini beradi, aks holda 0 qiymatni beradi. St1 satriy kattalik bir necha marotaba uchragan holda ham birinchisining tartib raqamini beradi.

Misollar keltiramiz:

st1 qiymati	Ifoda	Natija
'abcdef'	Pos('cd',S2)	3
'abcdcdef'	Pos('cd',S2)	3
'abcdef'	Pos('k',S2)	0

Val (a2, x,d) - protsedurasi a2 catriy kattalikni mumkin bo'lgan holda, x-ga haqiqiy yoki butun tipidagi son sifatida, aks holda x-ga 0-qiymatini o'zlashtiradi.

Misollar keltiramiz:

1-dastur	Natija	Hulosa
<pre> program val1; var a2:string;d,x:integer; begin a2:='rrr'; Val(a2,x,d); writeln(x); end. </pre>	0	A2-satriy miqdorni butun x-soniga o'zlashtirishida uni son ko'rinishida ifodalay olmasligi tyfayli, x-ga 0 qiymatni o'zlashtirdi
2-dastur	Natija	Hulosa
<pre> program val2; var a2:string;d,x: integer; begin a2:='2011';Val(a2,x,d); writeln(x/20:4:1); end. </pre>	100.6	A2-satriy miqdorni butun x-soniga o'zlashtirdi, uni ko'rsatilgan formatda 20 ga bo'lib chop

Quyidagi proseduralarni mustaqil o'rganing va misollar ko'rib chiqng:

Insert (st1, st, t1) - protsedurasi st1 satriy kattalikning t1 tartib raqamli belgisidan st satriy kattalik tarkibiga joylashtiradi.

Str (x, st) – protsedurasi x haqiqiy yoki butun tipidagi son bo'lsa, uni st satriy kattalikka aylantiradi.


Yana ikkita satriy kattalik bilan =, <, <=, >, >=, <=, >= munosabat amallari ham bajariladi. Bu munosabatlar ikkita satriy miqdorlarni taqqoslash uchun ishlatiladi. Taqqoslash natijasi true yoki false qiymatlarni qabul qiladi va bu amal Concat amaliga nisbatan past darjali hisoblanadi. Satriy miqdorlarni taqqoslash quyidagicha bajariladi: chapdan o'ngga elementlar taqqoslanib boriladi va bu teng indeksli elementlar "mos" kelmaguncha davom ettiriladi. Mos kelmagan elementlarning qaysi biri ASCII jadvalida katta qiymatli indeksga ega bo'lsa, shu satriy miqdor katta hisoblanadi. Agar satriy miqdorlarning uzunligi har xil bo'lib, bosh qismlari bir xil elementlardan iborat bo'lsa, ikkita satriy miqdordan "uzuni" katta hisoblanadi. Bir xil uzunlikka ega va mos indeksli belgilari bir xil bo'lgan satriy miqdorlar o'zaro teng hisoblanadilar.

Misolalar:

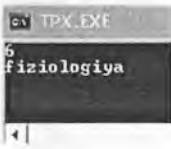
Ifodalar	Natija
'True1' < 'True2'	True
'Mother' > 'MOTHER'	True
'Geografiya' > 'Geologiya'	False
'Cat' = 'Cat'	True
'uzunlik' > 'balandlik'	True

Yuqorida keltirilgan funktsiya va protseduralardan dasturlarda qo'llashga misollar keltiramiz:

1-misol.

Dastur:	Dastur natijasi:
<pre> uses crt; var s1,s2,s3,s4,s5:string; begin clrscr; s1:='satriy'; s2:='kattaliklar'; s3:=concat(s1,' ',s2); writeln(s3); s4:=s1+' '+s2; writeln(s4); writeln(length(s3)); s5:=copy(s3,length(s3)-2,3); writeln(s5); end. </pre>	

2-misol.

<i>Dastur:</i>	<i>Dastur natijasi:</i>
<pre> uses crt; var s1,s2,s3,s4,s5:string; i:integer; begin clrscr; s1:='SAMARQAND'; i:=pos('QAND',s1); writeln(i); s2:='biologiya'; s3:='fizika'; delete(s2,1,2); delete(s3,5,2); insert(s2,s3,5); writeln(s3); readln; end. </pre>	

Kiritilgan ixtiyoriy jumla tarkibidagi “A” belgisini “O” belgisiga almashtirish chop etuvchi dastur tuzing.

1-usul	2-usul
<pre> PROGRAM sim1; VAR jumla:string;i,d: INTE- GER; SIM:STRING; BEGIN read(Jumla);d:=length(jumla); writeln(d); FOR I:=1 TO D DO BEGIN IF COPY(JUMLA,I,1)='A' THEN JUMLA[I]:='O'; END; WRITELN(JUMLA); readln; END. </pre>	<pre> PROGRAM sim1; VAR jumla:string;i,d: INTE- GER; SIM:STRING; BEGIN read(Jumla);d:=ord(jumla[0]); writeln(d); FOR I:=1 TO D DO BEGIN IF JUMLA[I]='A' THEN JUMLA[I]:='O'; END; WRITELN(JUMLA); readln; END. </pre>

3-misol: Burchak qiymati gradus o'lchovida kiritilgan holat uchun sinus, kosinus, tangens funksiyalari qiymatini mos belgini tanlash yo'li bilan hisoblovchi dastur tuzing.

```

PROGRAM char1;
uses crt;
VAR sim:char; i,grad:INTEGER;javob,radian:real; label 15,20;
BEGIN
  writeln('burchak qiymatini gradus o'lchovida kiriting');
  readln(grad); radian:=(pi*grad)/180;
  writeln('sinus ', grad,' ni hisoblash uchun "s" tugmasini tanlang');
  writeln('kosinus ', grad,' ni hisoblash uchun "c" tugmasini tanlang');
  writeln('tangens ', grad,' ni hisoblash uchun "t" tugmasini tanlang');
  sim:=readkey;
  case sim of
    's': javob:=sin(radian);
    'c': javob:=cos(radian);
    't': if cos(radian)=0 then goto 15 else javob:=sin(radian)/cos(radian);
  end;
  WRITELN(javob:5:2);goto 20;
  15:writeln ('qiymat mavjud emas');
  20: readln;
END.

```

4- misol: Tarkibida raqamlar bo'lgan s1-satriy kattalikning barcha raqamlri dan yangi s2 –satriy kattalik yaratuvchi dastur tuzing.

```

Program Satriy kattaliklar;
Var s1, s2: string; i: byte;
begin
  writeln('Tarkibida raqamlar bo'lgan satriy kattalikni kiriting');
  readln(s1);
  s2:='';
  for i:=1 to length(s1) do
    if (s1[i]>='0') and (s1[i]<='9')
      then s2:=s2+s1[i];
  writeln('natija ',s2);
end.

```

Dasturni quyidagi qiymat uchun tekshirib to'g'ri tuzilganiga ishonch hosil qiling:

```

S1:=`Balandligi 375 m bo`lgan poytaxtdagi teleminora`;
Natija: 375

```

– masala

5-misol: S-satriy miqdorning boshida joylashgan barcha bo'shliq belgilarini o'chiruvchi dastur tuzing.

Program Sinov1;

Var s: string[80];

Begin

writeln('Boshida bir necha bo''shliq belgisi qatnashgan s-satriy miqdor ni kirining');

readln(s);

while (pos(' ',s)=1) and (length(s)>0) do

delete(s,1,1);

write(Natija',s);

end.



Savol va topshiriqlar.

1. Paskal dasturlash tilida matnli ma'lumotlar bilan ishlashda qanday tipidan foydalanila?
2. Concat funksiyasini vazifasi nimadan iborat? Misollar keltiring.
3. Copy funksiyasi nima uchun qo'llaniladi?
4. Delete protsedura nima va unga misol keltiring.
5. Length funksiyasi nima ish bajaradi, misol bilan tushuntiring.
6. Pos funksiyasi qanday vazifani bajaradi va qachon qiymati nolga teng bo'ladi.
7. Val protsedurasini vazifasini tushuntiring.
8. Paskalda berilgan satr ichiga boshqa satrni joylashtirishning imkoni bormi? Javobingizni tushuntiring.
9. Sonli miqdorni satrli miqdorga o'tkazib bo'ladimi? Javobingizni izoxlang.

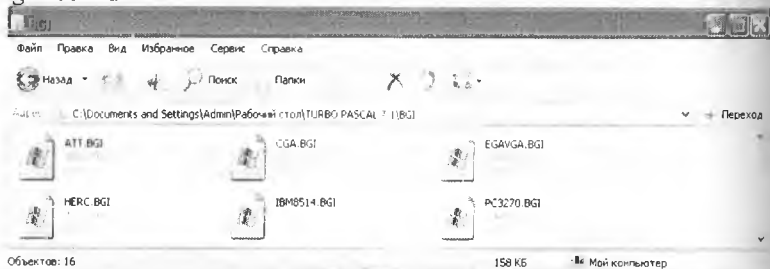
Mashqlar.

1. Quyidagi funksiyalar bajarilgandagi natijasini aniqlang.
 - a) Concat('davlat', 'bayrog'');
 - b) Copy('sabab', 1,2)+('daftar', 1,3);
 - c) A:= 'mustaqillik'; length(a);
 - d) Pos('l', 'istiqlol');
2. Berilgan o'zgaruvchi S1 va S2 ning qiymatlari mos ravishda "telekommunikatsiya" va "takrorlandi" bo'lsa, quyidagi amallar bajarilgach, uning qiymatlarini aniqlang:
 - a) Delete(s1, 5, 11);
 - b) Delete(s1, 1, 4);
 - c) Delete(s2, 7, 5);
 - d) Insert(s2, ma. 10);
 - e) Insert(s2, mas. 10);
3. Kiritilgan so'z palindrom bo'lishini tekshiruvchi dastur tuzing.

4. Kiritilgan so'zni teskarisiga o'girib beruvchi dastur tuzing.
5. Berilgan so'zning belgilari orasiga bittadan probel chiquvchi dastur tuzing.
6. Berilgan satrdagi barcha "a" harflarini o'chirib beruvchi dastur tuzing.
7. Berilgan butun sonning raqamlari ko'paytmasini hisoblovchi dastur tuzing.

16-§. Paskalda ekranni grafik holatga o'tkazish.

Turbo Paskal dasturini ishga tushirish bilan ekran matnli holatda bo'ladi va agarda yaratilayotgan dasturda grafikli ob'ektlardan foydalanish nazarda tutilsa displey adapterini xrafikali holatga moslashish zarur bo'ladi. Turbo Paskal dasturlash tilining 4.0 versiyasidan boshlab uning paketi tarkibiga grafikli ob'ektlar bilan ishlovchi qism dasturlarni qamrab olgan Graph moduli qo'shilgan. Bu modulning qism dasturlaridan foydalanish uchun kerakli drayverni (daryver maxsus dastur bo'lib, kompyuterga ulangan texnik qurilmalar faoliyatini boshqarishga mo'ljallangan) dastur bajarilishi uchun jalb etish zarur. Grafikali drayver displey grafikali holatda displey adapterini boshqaradi. Borland firmasi barcha tipdagi adapterlar uchun bunday drayverlarni ishlab chiqqan. odatda bu drayverlar dastur paketining BGI (Borland Graphics Interfase) nomli papkasida joylashgan bo'ladi.



Grafikli holatda ekranni bir-biriga juda yaqin joylashgan nuqtalar kabi tasavvur etish mumkin va bu nuqtalar turli ranglarda «yoritilib» tasvir hosil bo'lishiga xizmat qilishadi. Ayrim adapterlarning grafikali holatlari bilan tani-shamiz:

SGA adapteri (Color Graphics Adapter-rangli grafikali adapter). Bu adapter 5 ta grafik holatni qo'llab-quvvatlaydi. Ularning 4-tasi ekranning quyi imkoniyatlari (gorizontal yo'nalish bo'yicha 320 ta piksel (nuqta), vertikal yo'nalish bo'yicha 200 ta piksel (nuqta) joylashgan holat)ga mos keladi va ular bir-biridan palitra-ranglar to'plami bilan farqlanadilar. Har bir palitra 3 xil ranglardan iborat. 5-chi holat yuqori imkoniyatli bo'lib ekran echimi 640x200 ga teng.

EGA adapteri (Enhanced Graphics Adapter-takomillashtirilgan grafikali adapter). U SGA adapterini takrorlash bilan birga quyi imkoniyatli echimni (640x200; 16 xil rang) va yuqori imkoniyatli echimni (650x480; 16 xil rang) qo'llash imkoniyatini beradi.

MCGA adapteri (Multi-Color Graphics Adapter- rang-barang grafikali adapter). U SGA adapterini takrorlash bilan birga quyi imkoniyatli echimni (640x480, 2 xil rang) qo'llash imkoniyatini beradi.

VGA adapteri (Video Graphics Array grafikali videomassiv). U SGA va EGA adapterlarini takrorlash bilan birga ularni yuqori imkoniyatli echim (640x480, 16 xil rang) bilan to'ldiradi.

Bundan tashqari SVGA, EGAVGA kabi qator adapterlar va ularning drayverlari mavjud bo'lib, siz ular bilan kelgusi faolyatlaringizda batafsil tanishasiz.

Paskal dasturlash tili Graph modulining ayrim qism dasturlari bilan tanishamiz.

InitGraph-protselurasi adapterning grafikali holatga o'tishini ta'minlaydi. Uning sarlavhasi quyidagicha:

```
procedure InitGraph (var Driver, mode: integer; Path: string);
```

Bu erda: Driver- integer tipidagi o'zgaruvchi bo'lib, grafikali drayver tipini aniqlaydi, mode- xam integer tipli o'zgaruvchi bo'lib, grafikali drayverning ishchi holatini aniqlaydi. Path-string tipli o'zgaruvchi bo'lib, drayver fayli joylashgan manzil-yo'lni aniqlash uchun mo'ljallangan ifodadan iborat. Bu protseduradan foydalanish uchun biror axborot tashuvchi diskda albatta grafikali drayver fayli joylashgan bo'lishi shart. Dastur bajarilish jarayonida bu drayver protsedura yordamida operativ xotiraga yuklanadi va adapterning grafikali holatga o'tishini ta'minlaydi. Drayver tipi adapter tipi bilan mos kelishi shart. Graph modulida drayver tipini ko'rsatish uchun quyidagi doimiyliklar aniqlangan:

```
Const  
detect          =0; {tipni avtomatik aniqlash}  
CGA             =1;  
MCGA           =2;  
EGA            =3;  
EGA64         =4;
```

...

Juda ko'p adapterlar turli holatlarda ishlay oladilar va bu holatlar mode-o'zgaruvchisi yordamida aniqlanadi.

Protseduradan foydalanishga misol keltiramiz:

CGA.BGI drayveri S diskning TP\BGI papkasida joylashgan bo'lib, ekranning 320x200 echimli holatidan foydalanish nazarda tutilgan bo'lsa protseduraga murojaat qilishni quyidagi tartibda amalga oshiramiz:

```
uses Graph;  
var  
driver, mode: integer;  
begin  
driver := CGA;  
mode := CGAC2;  
InitGraph (driver, mode, 'c:\tp\BGI');  
end.
```

Agar kompyuterning adapter tipi noma'lum bo'lsa yoki yaratilayotgan dastur ixtiyoriy tipdagi adapter uchun mo'ljallangan bo'lsa, protseduraga murojaat qilishda drayver tipini avtomatik aniqlashni talab qilish mumkin, ya'ni bu holda

```
driver:=detect;
```

```
InitGraph (driver,mode,'c:\tp\BGI');
```

buyruqlaridan foydalanish mumkin. Bunda bir necha grafikali holat ishlovchi adapterlar uchun mode-ning maksimal qiymati avtomatik tanlanadi.

GraphResult funksiyasi (grafikali protseduralarga murojaat qilinganda) integer tipidagi qiymatni qaytaradi: 0-qiymatda xatolik yo'qligi va -14 dan -1 gacha bo'lgan qiymatlarda xatoliklarga yo'l qo'yilgani haqida «xabar» beradi.

GraphErrorMsg funksiyasi String tipidagi ma'lumotni-xatolikni berish uchun foydalaniladi.

GetDriverName funksiyasi String tipidagi ma'lumotni- yuklangan grafikali drayver nomini aniqlashda foydalaniladi.

GetModName funksiyasi String tipidagi ma'lumotni- ekranning echim holatlari (gorizontal va vertikal yo'nalishlar bo'yicha piksellar miqdorini) va adapterning ish holati nomini parametr bo'yicha aniqlash imkonini beradi.

GetMaxMode funksiyasi integer tipidagi kattalik bo'lib, adapterning grafikali holatlarining maksimal miqdorini aniqlaydi.

CloseGraph protsedurasi adapterning grafik holatdagi ish faoliyatini tugatib ekranning matnli holatini tiklaydi.

RestoreCRTMode qisqa muddatga ekranning matnli holatga qaytishini ta'minlaydi. CloseGraph protsedurasidan farqli o'rnatilgan grafikali holat parametrlari saqlanishini ta'minlaydi, ya'ni grafik drayver joylashgan xotiraning qismi tozalanmaydi.

GetGraphMode funksiyasi integer tipidagi qiymatni beradi va bu qiymat grafik adapterning o'rnatilgan grafikali holatini aniqlaydi.

SetGraphMode protsedurasi adapterning yangi grafikali holatini belgilaydi.

Quyida yuqorida keltirilgan funktsiya va protseduralardan foydalangan holda Turbo Paskal dasturlash tilida ekranni grafik holatga o'tkazish va kerakli hollarda uni matnli holatga vaqtincha o'tkazuvchi dasturimiz:

```
uses Graph;  
var  
Driver, Mode, Error: integer;  
begin  
Driver:= Detect;  
InitGraph(Driver, Mode, 'c:\tp\bgi');  
Error:= GraphResult;  
if Error <> grOk then  
Writeln(graphErrorMsg(Error))  
else  
begin  
Writeln('Bu grafikali holat');  
Writeln('Enter"-ni bosing...':20);
```

```

Readln;
RestoreCRTMode;
Writeln('Bu matnli holat...');
Readln;
SetGraphMode(GetGraphMode);
Writeln('bu yana grafikali holat...');
Readln;
CloseGraph
end
end.

```

Dasturda keltirilgan «grok»(qiymati 0 ga teng bo'lgan doimiylik) Graphresult funksiyasini natijasi bo'lib, protseduralarga murojaat qilishda xatolikga yo'l qo'yilmaganligini tekshirishda foydalaniladi.

1-masala. Dastur tarkibida GetDriverName funksiyasidan foydalanib, yuklangan grafikali drayver nomini aniqlang.

```

uses Graph;
var
Driver, Mode, Error: integer;  dr: string;
begin
Driver:= Detect;
InitGraph(Driver, Mode, 'c:\tp\bgi');
dr:=GetdriverName;
Writeln(dr, ' ');
Writeln("'Enter" tugmasini bosing...':50);
Readln;
CloseGraph ;
end.

```

Dastur natijasi yuklangan grafikali drayverga bog'liq va u quyidagicha bo'lishi mumkin:



2-masala. GetMaxMode funksiyasidan foydalanib, adapterning grafikali holatlarining maksimal miqdorini aniqlovchi dastur yarating.

```

uses Graph;
var
Driver, Mode, Error: integer;  d: integer;
begin
Driver:= Detect;

```



```

InitGraph(Driver,Mode,'c:\tp\bgi');
d:=GetMaxMode;
restorecrtMode;
Writeln(d);
Writeln("'Enter" tugmasini bosing...':50);
Readln;
CloseGraph ;
end.
Dastur natijasi:

```



3-masala. GetModName funksiyasi yordamida ekranning echim holatlarini va adapterning ish holati nomini parametr bo'yicha aniqlang.

Dastur ko'rinishi:

```

uses Graph;
var
Driver, Mode,Error,i:integer; ds:string;
begin
Driver:= Detect;
InitGraph(Driver,Mode,'c:\tp\bgi');
writeln (getdrivername);
for i:=0 to 3 do
begin
ds:=GetModeName(i);
Writeln(ds);
end;
Writeln("'Enter" tugmasini bosing...':50);
Readln;
CloseGraph;
end.

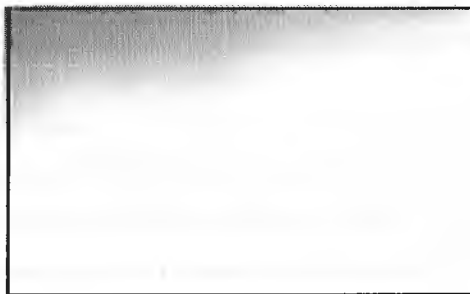
```

Koordinatalar, oynalar va sahifalar.

Turbo Paskalning juda ko'p grafikali protseduralari va funksiyalari ekranda ko'rinib turuvchi kursordan farqli bo'lgan "ko'zga tashlanmas" ikkinchi kursor ko'rsatgichidan foydalanishadi. Bu kursorning joylashishi odatda (0,0) koordinatali nuqtaga nisbatan belgilanadi. Umuman, chiziladigan sath - maydon alohida nuqtalar-piksellardan tashkil topgan. Nuqtaning o'rnini uning gorizontal (X) va vertikal (Y) koordinatalari bilan belgilanadi. Yuqori chap nuqtaning koordinatalari (0,0) dan iborat. Koordinatalarning qiymatlari vertikal yo'nalish bo'yicha tepadan

pastga va gorizontal yo'nalish bo'yicha chapdan o'ngga ortib boradi. GetMaxX va GetMaxY funksiyalari qiymatlari Word tipiga mansub bo'lib, ekranning joriy ish holatdagi koordinatalarining maksimal qiymatini aniqlashda foydalaniladi.

0,0)



(GetMaxX,
GetMaxY)

Kursor-ko'rsatgich o'rnini integer tipidagi qiymatlarni beruvchi GetX va GetY funksiyalari aniqlash imkonini beradilar.

SetViewport(x1,y1,x2,y2:integer;Clipon:Boolean)-protsedurasi ekranda yuqori chap nuqtasi koordinatalari (x1,y1) va quyi o'ng nuqtasi koordinatalari (x2,y2) bo'lgan to'g'ri to'rtburchak shaklidagi oynani yaratishda qo'llaniladi. Boolean tipidagi ifoda qiymatiga ko'ra mazkur oynaga sig'mayotgan tasvir elementlarini «qirqish» uchun beriladi. Bu ifodaning True qiymatida mazkur protsedura yordamida aniqlangan sohaga tasvir elementlari sig'masa, ular qirqiladilar, aksincha False qiymatida soha chegaralari «bekor» qilinadi.

MoveTo(x,y:integer)-protsedurasi ekranda ko'rsatgichni koordinatalari (x,y) bo'lgan nuqtaga joylashtiradi va bu joylashtirish oyna o'rnatilmagan holda (0,0) nuqtaga nisbatan amalga oshiriladi.

MoveRel(dx,dy:integer)-protsedurasi ekranda ko'rsatgichni yangi nuqtaga joriy koordinatalariga nisbatan dx,dy-ortirmalar bo'yicha joylashtirish imkonini beradi.

SetbkSolor(n)- protsedurasi ekran fonini ko'rsatilgan n-parametrga mos keluvchi (0 dan 15 gacha) rang bilan berilishini ta'minlaydi.

ClearDevice- protsedurasi grafik ekranni tozlab, uning fonini SetbkSolor bilan aniqlangan rang bilan «bo'yalishi»ga xizmat qiladi.

ClearDevice- protsedurasi yaratilgan grafik oynani, agar u «to'liq ekran» holatida aniqlanmagan bo'lsa tozalaydi va ko'rsatgich oynaning chap yuqori nuqtasiga joylashadi.



Savol va topshiriqlar.

1. Paskalda Graph moduli qanday maqsadda qo'llaniladi?
2. Grafikli drayver nima va u qanday vaifani bajaradi?
3. Ekranni grafik holatga o'tkazish uchun Paskalda qanday ko'rsatma beriladi?

4. Grafik xolatdan chiqish uchun qaysi operatoridan foydalaniladi?
5. Ekranni grafik holatga o'tkazilganda yurgichning ko'rinishi qanday bo'ladi?
6. Grafik xolatda ekranning qaysi nuqtasi joriy nuqta bo'ladi?
7. GetMaxX va GetMaxY funksiyalarini vazifalarini aytib bering.

Mashqlar.

1. Ekranni grafik holatga o'tkazuvchi va <enter> tugmasini bosganda yana matnli holatga qaytaruvchi dastur tuzing.
2. Ekranni burchaklarida qizil rangli nuqta hosil qiluvchi dastur tuzing.
3. Nuqtalar yordamida ekranni o'rtasidan bo'luvchi gorizontal chiziq hosil qiling.
4. Random funksiyasidan foydalanib turli rangli nuqtalarni hosil qiling.

17-§. Paskalning shakllar chizish operatorlari.

Grafikali muhitda ob'ektlarni chizish uchun sodda shakllarni chizuvchi protseduralar va ulardan foydalanish uchun funksiyalardan foydalanish zarur. Quyida ularni keltirib o'tamiz:

PutPixel(x,y:integer;color:word) –protsedurasi ko'rsatilgan color rangli (x,y) koordinatali nuqtani ekranda tasvirleydi.

Masalan: PutPixel(100,100,red)-protsedurasi ekannda (100,100)-koordinatali nuqtada qizil rangli nuqta tasvirini yaratadi.

Bu protseduradan dastur tarkibida foydalanib, samarali "effektlar" olishimiz mumkin.

Masalan, quyidagi dastur yordamida ekranda

Setviewport (20,20,200,200,true) protsedurasi bilan belgilangan oynada 32000 ta nuqtalarni

putpixel(random(getmaxx),random(getmaxy),random(255));
protsedurasi yorlamida turli ranglar bilan bo'yab ajoyib tasvir yaratishimiz mumkin.

Dastur ko'rinishi:

```
uses Graph;
var
  Driver, Mode,d,m:integer;

begin
  Driver:= Detect;
  InitGraph(Driver,Mode, 'c:\tp\bgi');
  setbkcolor(black);
  Setviewport(20,20,200,200,true);
  for d:=1 to 32000 do
```

```

putpixel(random(getmaxx),random(getmaxy),random(255));
readln;
CloseGraph ;
end.

```

GetPixel(x,y:integer):word funksiyasi (x,y) koordinatali nuqtaning rangini aniqlashda qo'llaniladi. Bu funksiyadan foydalanishga misol keltiramiz:

```

uses graph;
var
gd,gm:integer; a:word;
begin
gd:=detect; initgraph(gd,gm,""); a:=getpixel(25,25);writeln(a);readln;
end.

```

Line(x1,y1,x2,y2:integer)-protsedurasi (x1,y1) va (x2,y2) koordinatali nuqtalarani tutashtiruvchi to'g'ri chiziq chizish uchun ishlatiladi.

Misol:

```

uses graph;
var
gd,gm, x1,x2,x3,y1,y2,y3:integer;
begin
gd:=detect; initgraph(gd,gm,"");
x1:=45;y1:=45;x2:=160;y2:=120;x3:=100;y2:=80;
line(x1,y1,x2,y2); line(x3,y3,x2,y2); line(x1,y1,x3,y3);readln;
end.

```

misol: line(10,100,340,100);

LineTo(x,y:integer)-protsedurasi ko'rsatgich turgan nuqtadan koordinatalri (x,y) bo'lgan nuqtagacha to'g'ri chiziq chizilishini ta'minlaydi. Bu protseduradan foydalangandan so'ng ko'rsatgich koordinatasi shu nuqtaga ko'chib o'tadi.

misol:

```

uses graph;
var gd,gm, i:integer;
begin gd:=detect;initgraph(gd,gm,"");
for i:=1 to 20 do
lineto(random(getmaxx),random(getmaxy));readln;
end.

```

LineRel(dx,dy:integer)-protsedurasi ko'rsatgich turgan nuqtaga nisbatan koordinatalari mos ravishda (dx,dy) qiymatlarga orttirilgan nuqtagacha to'g'ri chiziq chizilishini ta'minlaydi.

```

uses graph;
var gd,gm:integer;
begin

```

```
gd:=detect; initgraph(gd,gm,"");
moveto(100,100); Linerel(100,40);readln;
end.
```

SetLineStyle (type, Pattern,Thick:word)-protsedurasi chizilishi kerak bo'lgan chiziqning shaklini o'rnatish uchun qo'llaniladi.

Bu erda:

Type- chiziqning shaklini

Pattern- chiziqning namunasi

Thick- chiziqning qalinligini aniqlovchi parametrlar.

CHiziqning tipini quyidagi doimiylar orqali berish mumkin:

const

Solidln =0; {uzluksiz chiziq}

Dotteln =1; {nuqtali chiziq}

Centerln =2; {shtrix-punktirli chiziq}

Dashedln =3; {punktirli chiziq}

Userbtn =4; {chiziq ko'rinishi foydalanuvchi tomonidan belgilanadi}

Pattern-parametri foydalanuvchi tomonidan belgilanadigan chiziq uchun aniqlanadi. Pattern parametri uzunligi 16 piksel bo'lgan kesmani aniqlaydi. Masalan, bu parametr qiymati 100 ga teng bo'lsa, 16 pikselning shu qismi "yoritiladi" qolgan qismi fon rangi bilan ifodadanadi. dastur chizilishi zarur bo'lgan chiziqni ana shu 16 pikselda ketm-ket tasvirlaydi.

Thick-parametri quyidagi qiymatlarni qabul qiladi:

const

NormWidth =1; {chiziq qalinligi 1 pixel}

ThickWidth =3; {chiziq qalinligi 3 pixel}

dastur tarkibida kattaliklar-doimiylikalr nomini o'zgartirishimiz mumkinligini hisobga olgan holda, SetLineStyle protsedurasidan foydalanishga misol keltiramiz:

Misol:

```
uses graph;
```

```
const uzluksiz=0; nuqtali=1;Shtrix_punktir=2; punktir=3;foychiz=4;
chizqal=1;
```

```
var
```

```
gd,gm, i:integer;
```

```
begin
```

```
gd:=detect;initgraph(gd,gm,"");
```

```
setLineStyle(nuqtali,0,chizqal);moveto(100,100);Linetto(400,100);
```

```
setLineStyle(uzluksiz,0,chizqal);moveto(100,110);Linetto(400,110);
```

```
setLineStyle(Shtrix_punktir,0,chizqal);moveto(100,120);Linetto(400,120);
```

```
setLineStyle (punktir,0,chizqal);moveto(100,130);Linetto(400,130);
```

```
setLineStyle (foychiz,80,chizqal);moveto(100,140);Linetto(400,140);
```

```
readln;
```

```
end.
```

Izoh: domiyliklar nomi o'rniga ularning qiymatidan ham foydalanishimiz mumkin.

SetLineStyle- protsedurasi yordamida o'rnatilgan chiziqning shakli yordamida to'rtburchak, ko'pburchak va boshqa shakllarni chizishda foydalanish mumkin.

rectangle(x1,y1,x2,y2:integer); -protsedurasi chap yuqori burchagi (x1,y1) va o'ng quyi burchagi (x2,y2) nuqtalarda joylashgan to'gri to'rtburchak chizish uchun ishlatiladi.

masalan:

```
uses graph;
```

```
var gd,gm:integer;
```

```
begin
```

```
gd:=detect;
```

```
initgraph(gd,gm,'c:\tp\bgi');
```

```
rectangle(5,5,getmaxx-5,getmaxy-5);
```

```
readln;
```

```
end.
```

SetColor (color:word); - protsedurasi chizilishi va chop etilishi zarur bo'lgan chiziqlar va simvollar rangini belilaydi. Graph modulida ranglarni ifodalash uchun Srt modulidagi doimiyliklardan foydalaniladi.

SetColor :word- funktsiyasidan joriy rang kodini aniqlashda foydalaniladi.

GetMaxColor :word- funktsiyasi *SetColor* protsedurasida foydalanishi mumkin bo'lgan maksimal kodni aniqlashda ishlatiladi.

Matnli holatdan farqli fon rangi faqat "qora" rangda emas, ixtiyoriy bo'la oladi. Grafikali ekranda rangni aniqlash orqali butun ekran rangini o'zgartirish mumkin, ya'ni bu ekranni turli sohalarini turli ranglar bilan "bo'yash"ning iloji yo'q.

Circle(x,y:integer;r:word) : - protsedurasi markazi (x,y) va radiusi *r* - piksel bo'lgan aylana chizish ishlatiladi. Bu protseduradan foydalanishda chiziq qalinligini parametr orqali o'zgartirish mumkin, ammo chiziq shakli faqat uzluksiz chiziqdan iborat bo'ladi.

Masalan:

```
...
```

```
setlinestyle(1,0,3);
```

```
circle(50,50,45);
```

```
.....
```

Yuqoridagi protseduralar yordamida ekranda markaz koordinatalari va radiuslari tasodufiy qiymatga ega bo'lgan 120 ta aylana chizish dasturini keltiramiz:

```
uses graph,crt;
```

```
var gd,gm, i:integer;
```

```
begin
```

```
gd:=detect;initgraph(gd,gm,"");
```

```

for i:=1 to 120 do
begin
setcolor(random(getmaxcolor));
circle(random(getmaxx),random(getmaxy),random(100));
end;
end.

```

Arc (x,y:integer;BegA,EndA,r:word) : - protsedurasi markazi (x,y) va radiusi *r* –piksel bo'lgan aylana yoyining *BegA* va *EndA* burchaklariga mos keluvchi qismini chizish uchun shlatiladi. *BegA* va *EndA* graduslarda beriladi va soat mili yo'nalishiga qarshi ravishda hisobga oldinadilar.

Masalan:

```
arc(200,200,90,180,35);
```

ellipse(x,y:integer; BegA, EndA, RX,Ry:word) : - protsedurasi markazi (x,y) va gorizantal radiusi *RX*, vertikal radiusi *Ry* –piksel bo'lgan ellips yoyining *BegA* va *EndA* burchaklariga mos keluvchi qismini chizish uchun shlatiladi.

Masalan:

```
ellipse(100,100,150,60,145,100);
```

SetFillStyle(fill,color:word);-protsedurasi biror sohani to'ldirishning stili va rangini belgilaydi. To'ldirish tipi uchun quyidagi kattaliklardan foydalaniladi:

const

EmptyFill	=0;{fon rangi bilan to'ldirish}
SolidFill	=1;{yaxlit-uzluksiz to'ldirish }
LineFill	=2;{- - - - simvollar bilan to'ldirish }
LtSlashFill	=3;{ / / / / / simvollar bilan to'ldirish }
SlashFill	=4;{ / / / qalin simollar bilan to'ldirish }
BkSlashFill	=5;{ \ \ \ \ qalin simvollar bilan to'ldirish }
LtBkSlashFill	=6;{ / \ / \ / \ / \ / simvollar bilan to'ldirish }
HatchFill	=7;{ + + + + + simvollar bilan to'ldirish }
XHatchFill	=8;{ x x x x simvollar bilan to'ldirish }
InterLeaveFill	=9;{ kataklar bilan to'ldirish }
WideDotFill	=10;{kam sonli-zich bo'lmagan nuqtalar bilan to'ldirish }
CloseDotFill	=11;{zich bo'lgan nuqtalar bilan to'ldirish }
UserFill	=12;{foydalanuvchi tomonidan aniqlangan shakl bilan to'ldirish }

floodfill(x,u;integer; Border:word):-protsedurasi ixtiyoriy chegaralangan sohani oldindan aniqlangan sohani to'ldirishning stili va rangi bilan to'ldiradi.

masalan: parametrlari (150,150,getmaxx-150,getmaxy-150) bo'lgan to'rtburchak sohani turli stillarda to'ldiruvchi dasturni keltiramiz:

```
uses graph;
var gd,gm:integer; i:byte;
begin
gd:=detect; initgraph(gd,gm,'c:\tp\bg');setbkcolor(black);
setcolor(red);setlinestyle(0,0,1);
for i:=0 to 12 do
begin
setfillstyle(i,green);rectangle(150,150,getmaxx-150,getmaxy-150);
floodfill(160,160,red);
readln;
end;
readln;
end.
```

Bar (*x1,y1,x2,y2:integer*):-protsedurasi chap yuqori burchagi (*x1,y1*) va o'ng quyi burchagi (*x2,y2*) nuqtalarda joylashgan to'g'ri to'rtburchakli ekran sohasini to'ldiradi.

masalan: bar(340,125,480,220);

Bar va setfillstyle protseduralari yordamida 12 ta to'g'ri to'rtburchakli sohani turli stillarda to'ldirib, to'ldirish stilini namoyish etuvchi dastur yarating.

```
uses graph,crt;
const
stil:array[0..11]of string=('0','1','2','3','4','5','6','7','8','9','10','11');
var
gd,gm,a,b:integer; i:byte;
begin
gd:=detect;initgraph(gd,gm,"");setbkcolor(black);
a:=5; b:=100;
for i:=0 to 11 do
begin
setfillstyle(i,4);bar(a,100,a+50,200);outtextxy(a+15,75,stil[i]);a:=a+50;
end;
readln;
end.
```

Bar3d(*x1,y1,x2,y2,depth:integer.top:boolean*):-protsedurasi oldingi hududi chap yuqori burchagi (*x1,y1*) va o'ng quyi burchagi (*x2,y2*) nuqtalarda joylashgan parallelepiped tasvirini yaratishda ishlatiladi.

bu erda :

-depth-uch o'lmali tasvirning uchunchi o'lehami

- top- parallelepipedning yuqori hududini aks ettirish uchun ishlatiladi, ya'ni "true" qiymtida yuqori hudud aks etirilsa, aksincha holda "ko'rinmaydi".

Masalan: Bar3d protsedurasi yordamida ekranda parallelepipedning yuqori hududini aks ettirish va aks etirilmasligini namoyish etuvchi dasturni keltiramiz.

```
uses graph,crt;
var gd,gm:integer;
begin
gd:=detect;initgraph(gd,gm,"");
setbkcolor(black);setcolor (magenta);
begin
bar3d(100,100,200,200,25,true);
bar3d(300,100,400,200,25,false);
end;
readln;
end.
```

FillEllipse (x,y,Rx,Ry:integer)-markazi (x,y nuqtada bo'lgan gorizontal radiusi Rx-ga, vertikal radiusi Ry-ga teng ellipsni to'ldirish uchun ishlatiladi.(Rx va Ry-piksellarda)

Masalan: fillellipse(340,220,100,50);

Sector(x,y:integer;BegA,EndA,Rx,Ry:word)-protsedurasi markazi (x,y) nuqtada bo'lgan gorizontal radiusi Rx-ga, vertikal radiusi Ry-ga teng ellipsning boshlang'ich burchagi BegA-ga, oxirgi burchagi EndA ga teng sektorini aniqlangan shakl bilan to'ldiradi.

Masalan: sector (200,200,180,200,55,300);

PieSlice(x,y:integer;BegA,EndA,R:word)-protsedurasi markazi (x,y) nuqtada bo'lgan radiusi R-teng aylananing boshlang'ich burchagi BegA-ga, oxirgi burchagi EndA ga teng sektorni berilgan shakl bilan to'ldiradi.

Masalan: PieSlice(300,300,145,360,100);



Savol va topshiriqlar.

1. PutPixel operatori haqida so'zlab bering.
2. Ekranda biror shakl chizish uchun uning rangi qaysi operator yordamida tanlanadi?
3. Paskalda kesma chizish imkoniyatini amalda ko'rsatib bering.

4. Qaysi protsedura chiziqning shaklini o'rnatish uchun qo'llaniladi?
5. *SetLineStyle*- protsedurasi yordamida qanday shakllarni chizishda foydalaniladi?
6. Aylana qaysi operator yordamida chiziladi?
7. *SetFillStyle* operatorini vazifasini aytib bering.
8. Bar va Bar3D operatorlarini vazifasi nimadan iborat?
9. *FillEllipse* operatori qanday shakl chizadi?

Mashqlar.

1. Ekranni o'rtasidan bo'luvchi chiziq chizuvchi dastur tuzing.
2. Ekraning o'rtasida radiusi 75 ga teng qizil rangli aylana chizing.
3. Ekranida tomonlari 50 ga teng uchburchak va to'rtburchak chizuvchi dastur tuzing.
4. Mashina rasmini chizing.
5. Ekranni ko'k vertikal chiziqlar bilan to'ldiring.
6. Svetofor rasmini chizing 3 xil rangda bo'lsin.
7. 6 xil bo'yash usulini namoyish etuvchi 50x50o'lchamili 6t akvadrat chizing.
8. Spektir(kamalak) nurlarini hosil qiluvchi dastur tuzing.

18-§. Fayllar bilan ishlash.

Fayl deb kompyuter tashqi xotirasining nomlangan soxasiga aytiladi va uning uchta xarakterli xususiyatlari mavjud. Birinchidan, uning nomi mavjud bulib, bu nomdan dasturda foydalaniladi. Ikkinchidan uning komponentlari bir tipga mansub va bu fayl tipidan boshqa barcha tiplar bulishi mumkin. Uchinchidan yangi yaratiluvchi faylning uzunligi haqida uning e'lon qilish vaqtida "fikir yuritilmaydi" va bu faqat tashqi xotira elementining xajmiga bog'liq.

Fayl tipi quyidagi uch yo'llarning biri bilan yaratiladi:

<nom>=file of<tip>; <nom>=Text; <nom>=File;

Bu erda <nom>- fayl tipining nomi (to'g'ri nomlangan identifikator), file, of xizmatchi so'zlar, <tip> bu fayl tipidan boshqa barcha tiplar.

Misol: Type

```
Product=record      Name:String;      Code:Word;End:
Text80=file of String [80];
```

Var

```
F1: File of Char;  F2: Text;  F3: File;      F4: Text80;
F5: File of Product;
```

Fayllarni e'lon qilish usullariga ko'ra, ularni uch turga ajratish mumkin: tiplashirilgan fayllar(File of ... bilan beriladi, yuqoridagi misolda. F1, F4,

F5);

matnli fayllar(TextFile tipi bilan aniqlanadi, yuqoridagi misolda. F2); tiplashirilmagan fayllar(File tipi bilan beriladi, yuqoridagi misolda. F3);

Faylning turi uning saqlanish usulini aniqlaydi va umuman Turbo Pascal tilida oldindan yaratilgan faylni nazorat qilish vositalari mavjud emas va bu vazifani dasturchi o'z zimmasiga olishi lozim. Fayllar bilan ishlash faqat faylni ochish protsedurasi bajarilgandan so'ng bajarilishi mumkin. Bu odindan e'lon qilingan fayl o'zgaruvchisini yaratilgan yoki yaratilishi lozim bo'lgan fayl nomi bilan bog'lash protsedurasi bo'lib, undan so'ng fayldan o'qish yoki unga yozish yo'nalishi beriladi. Har qanday fayllar (yoki mantiqiy qurilmalar) faylni (mantiqiy qurilmani) ochish maxsus protsedurasi yordamida dasturga u bilan ishlash imkoniyatini beradi. Fayl o'zgaruvchisi avvaldan yaratilgan fayl nomi bilan quyidagi standart protsedura yordamida bog'lanadi:

Assign (<fayl o'zgaruvchisi>,<fayl nomi>) protsedurasi fayl o'zgaruvchisini fayl nomi bilan bog'laydi.

AssignFile (<fayl o'zgaruvchisi >,<fayl nomi >);- bu protseduraning umumiy ko'rinishi bulib, bu erda fayl o'zgaruvchisi - dasturda e'lon qilingan fayl tipidagi o'zgaruvchi, fayl nomi esa, fayl nomini yoki ungacha bo'lgan yo'lni ifodalovchi matn.

Fayl initsializatsiyasi deb, bu faylga ma'lumotlarni jo'natish yoki undan olish yo'nalishiga aytiladi.

Faylni o'qish uchun fayl Reset protsedurasi yordamida initsializatsiyu qilinadi va bu protseduraning ko'rinishi quyidagicha:

Reset (<fayl uzgaruvchisi >);

Izoh: fayl o'zgaruvchisi -avval Assign protsedurasi yordamida mavjud fayl bilan bog'langan bo'lishi lozim.

Bu protsedura bajarilishi natijasida fayl o'qish uchun tayyorlanadi va natijada maxsus ko'rsatgich bu faylni boshiga, ya'ni 0-tartib nomerli komponentni ko'rsatib turadi.

Delphi dasturlash tilida Reset protsedurasi yordamida ochilgan tiplashtirilgan fayllarga read protsedurasi bilan murojaat qilish mumkin. Reset protsedurasi yordamida ochilgan matnli fayllar uchun Write yoki Writeln protseduralaridan foydalanib bo'lmaydi.

Rewrite (< fayl o'zgaruvchisi >) protsedurasi fayl o'zgaruvchisi bilan bog'langan faylga yozish uchun beriladi va bunda mavjud fayldagi barcha ma'lumotlar avval "o'chiriladi" va yangi ma'lumotlar buyruqlarga ko'ra faylga yoziladi.

Append (<fayl uzgaruvchisi >) protsedurasi mavjud faylni kengaytirish ya'ni unga qushimcha ma'lumotlarni yozish imkoniyatini beradi.(faqat Text tipidagi fayllar uchun qo'llaniladi)

Close (< fayl uzgaruvchisi >) protsedurasi faylni yopish uchun qo'llaniladi. ammo fayl o'zgaruvchisi bilan bog'lanish o'z kuchini saqlaydi.

Quyida keltirilgan ikki protseduradan foydalanish uchun Reset, Rewrite yoki Append protseduralari yordamida ochilgan fayllar avval yopilgan bo'lishlari shart.

Rename (< fayl uzgaruvchisi >, < yangi nom >) protsedurasi faylni qayta nomlash uchun ishlatiladi.

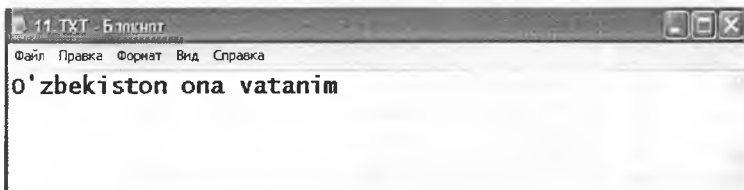
Erase (< fayl uzgaruvchisi >) fayli o'chiradi.

Endi yuqoridagi protseduralardan dasturlarda foydalanishga misollar ko'rib chiqamiz:

1-misol. S-mantiqiy diskda "11.txt" faylini yaratish va unda 'O'zbekiston ona vatanim' matnini joylashtiruvchi dastur yarating.

```
var f:text;
begin
assign ( f,'c:/11.txt');
rewrite(f);
writeln(f,'O"zbekiston ona vatanim');
close(f);
end.
```

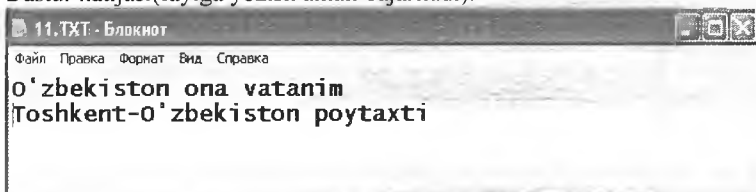
Dastur natijasi(faylga yozish amali bajariladi):



2-misol. S-mantiqiy diskda joylashgan "11.txt" faylidagi 'O'zbekiston ona vatanim' matnini "Toshkent- O'zbekiston poytaxti" matni bilan to'ldirish.

```
var f:text;
begin
assign ( f,'c:/11.txt');
rewrite(f);
writeln(f,'O"zbekiston ona vatanim');
close(f);
end.
```

Dastur natijasi(faylga yozish amali bajariladi):



3-misol.

```
var f:text;
x,i:integer;
begin
assign ( f,'c:/12.txt');
rewrite(f);
for i: 1 to 15 do
```

```

begin
  write(f,i);
  write(f,' ');
  write(f,sqrt(i));
  writeln(f);
  end;
  close(f);
  end.

```

Dastur natijasi(faylga yozish amali bajariladi):

```

1 1.000000000E+00
2 1.4142135624E+00
3 1.7320508076E+00
4 2.000000000E+00
5 2.2360679775E+00
6 2.4494897428E+00
7 2.6457513111E+00
8 2.8284271247E+00
9 3.000000000E+00
10 3.1622776602E+00
11 3.3166247904E+00
12 3.4641016151E+00
13 3.6055512755E+00
14 3.7416573868E+00
15 3.8729833462E+00

```

Fayldan o'qish amali bajarilishi kuzatish uchun avval S mantiqiy diskda (yoki boshqa manbada) o'qish uchun faylni tayyorlaymiz. Buning uchun «Bloknot» dasturida «15.pas» faylini quyidagi ko'rinishda tayyorlab olamiz:

```

1 2 3 4 5 6 7 8 9 10

```

4-misol. S mantiqiy diskdagi «15.pas» fayldan 10 ta soni o'qib ularning kvadratlarni monitorida aks ettiruvchi dastur yarating.

```

uses crt;
var f:text;
x,i:integer;
begin
  assign ( f,'c:\15.txt');
  reset(f); clrscr;
  for i:= 1 to 10 do

```

```

begin
  read (f,x);
  writeln(i,'-ning kvadrati ',sqr(x),' ga teng');
  end;
  close(f);
end.

```

Dastur natijasi:



```

TPX.EXE
1 -ning kvadrati 1 ga teng
2 -ning kvadrati 4 ga teng
3 -ning kvadrati 9 ga teng
4 -ning kvadrati 16 ga teng
5 -ning kvadrati 25 ga teng
6 -ning kvadrati 36 ga teng
7 -ning kvadrati 49 ga teng
8 -ning kvadrati 64 ga teng
9 -ning kvadrati 81 ga teng
10 -ning kvadrati 100 ga teng

```

Fayldan o'qishda faylning oxirini aniqlovchi EOF(f) mantiqiy funktsiyasidan foydalanishga misol keltiramiz. Buning uchun siz quyidagi kabi sinfdoshlar-ingizni 4 fandan olgan test natijalarini aks ettiruvchi «15.txt» hujjatni yaratib kerakli joyda saqlashingiz zarur.



```

15.txt - Блокнот
Файл Правка Формат Вид Справка
axmedov
14
15
20
25
baqoev
13
17
22
25
rustamov
12
20
11
22

```

5-misol. O'quvchilarning 4 fandan olgan natijalarini fayldan o'qib, har bir o'quvchining familiyasi va ballari yig'indisini ekranda chop etuvchi dastur yaratish.

Dastur ko'rinishi:

```

uses crt;
var f:text;
fam:string;
a,b,c,d:integer;

```

```

begin
assign ( f,'c:/15.txt');
reset(f); clrscr;
while not eof(f) do

begin
  readln (f,fam, a,b,c,d);
  writeln(fam,a+b+c+d);
  end;
close(f);
end.

```

(dasturni mustaqil bajarib, xulosa chiqaring)

Mustaqil bajarib ko`ring.

```

1.
var f:text;
x,i:integer;
begin
mkdir ('c:/v1');{yangi katalog ochish uchun}
assign ( f,'c:/v1/12.txt');
rewrite(f);
writeln(f,sqrt(2011));
close(f);
end.

```

```

2.
var f:text;
x,i:integer;
begin
assign ( f,'c:/v1/12.txt');
rewrite(f);
writeln(f,sqrt(2011));
close(f);
rename(f,'c:/v1/y1.txt');
end.

```



Savol va topshiriqlar.

1. Fayl turdagi to`zgaruvchi deganda nimani tushinasiz?
2. Paskalda matnli faylni ifodalovchi xizmatchi so`zni ayting.
3. Assign operatori vazifasini aytib bering.
4. Rewrite operatori vazifasini aytib bering.
5. Rewrite operatori bilan ochilayotgan fayl tashqi xotirada avvaldan mavjud bo`lsa qanday xodisa yuz beradi?
6. Close operator nima uchun qo`llaniladi?

7. Append protsedurasini vazifasini aytib bering.
8. Fayldagi ma'lumotlarni ochish uchun qaysi operator yordamida ochiladi.
9. EOF funksiyasini vazifasini aytib bering.

Mashqlar.

1. Guruxning familiyasi va ismlaridan tashkil topgan "gurux.txt" nomli matnli fayl hosil qiluvchi dastur tuzing.
2. I-misolda hosil qilingan "gurux.txt" fayli ochib, undagi familiya va ismini davomidan otasini ismini qo'shib qo'ying.
3. Hafta kunlarining nomlarini kiritib, ularni "hafta.txt" faylda saqlab qo'yadigan dastur tuzing.
4. "hafta.txt" fayli ochib, davomidan hafta kunlarining ingliz tilidagi nomlarini yozuvchi dastur tuzing.
5. "hafta.txt" faylida berilgan hafta kunlarining nomlarini ekranga chiqaruvchi dastur tuzing.
6. "gurux.txt" faylida berilgan familiyalar ichidan "S" harifi bilan boshlanadiganlarini ekranga chiqaruvchi dastur tuzing.

19-§. Qism dasturlar. Funksiyalar. Protseduralar

Funksiya va protseduralar maxsus tuzilish va nomga ega bo'lgan dasturning mustaqil qismlari hisoblanishadi. Dastur matnida bu nomlardan foydalanish funksiya va protseduralarga murojlat qilishni anglatadi. Funksiya va protseduralarni umumiy nom qism dasturlar deb yuritishadi. Bu qism dasturlar yordamida har qanday dasturni bir-biriga bog'liq va bo'liq bo'lmagan qimlarga ajratish mumkin. Bu holat dastur xotirani tejash bilan birga, dasturchi uchu qator qulayliklar yaratadi. Har bir qism dasturda bir marotaba uchraydi, ammo undan murojlat qilish usuli bilan birnecha marotaba foydalanish mumkin. (yaratilgan qism dasturlar yordamida modullar tarkibida foydalanish uyqoru samaralar berishi bilan k.....)

Dastur tarkibida qim dasturlar joylashishuni quyidagicha tasvirlash mumkin:

Dastur

Dasturosti A

Dasturosti A1

Dasturosti A2

Dasturosti B

Dasturosti B1

Dasturosti B2

Dasturosti B21

Dasturosti B22

Funksiya bu dasturning bir qismi bo'lib, *function* so'zi bilan boshlanib, quyidagi tuzilishga ega:

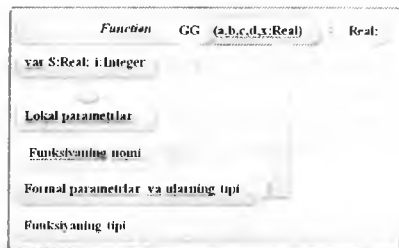
function <funksiyaning nomi> (<formal parametrlar ro'yxati>): <tip>;
bu erda tip funksiya qiymatining tipi.

Bu tip tartibli, haqiqiy, String va Point tipida bo'lishi mumkin. Formal parametrlar quyidagi tuzilishga ega:

<Bir nechta tipga tegishli parametrlar>: <tipning nomi>;

Misollar.

1)



2) *function* RR1(x1,y1,x2,y2: byte): Real;

Dasturda funksiyaga murojaat qilish haqiqiy parametrlar yordamida amalga oshiriladi, masalan

GG (6,5,6,-1,0) yoki RR1(a,b,c,d) va h.

3) *function* Aro(s: String): String;

4) *function* NNC(L: Word; c: Char): String;

Funksiya – dasturning bir qismi bo'lib, unga dasturda bir nechta marotaba murojaat qilinishi mumkin. Umuman:

a) agarda parametrlar qiymatlari yordamida yagona natija olinadigan bo'lsa, funksiyadan foydalanish mumkin;

b) qism dasturning yakuniy natijasi albatta funksiya nomiga o'zlashtirilishi shart.

1-masala.

Funksiya yordamida to'rtburchak tomonlariga ko'ra uning perimetri va yuzasini hisoblovchi dastur tuzing.

Program rr;	dastur nomi
Var d,h:integer;	
Function dper (a,b:integer):integer;	funksiya nomi, formal parametrlar va funksiyani tipi
Begin	
dper:=a+a+b+b;	—funksiyaning yakuniy natijasi uning nomiga o'zlashtiriladi
end;	
Function ss(a,b:integer):integer;	— funksiya nomi, formal parametrlar va funksiyani tipi
Begin	
ss:=a*b;	—yakuniy natija ss funksiya nomiga o'zlashtiriladi
end;	
Begin	
Readln (d,h);	
Writeln ('to'g'ri burchak perimetri');	
Writeln (dper (d,h));	—d,h haqiqiy parametrga bog'liq dper va ss funksiya qiymatlari hisoblanadi
Writeln (ss(d,h));	
Readln;end.	

2-masala.

Funksiya yordamida a, b, c tomonlari berilgan uchburchak yuzasini hisoblovchi dastur tuzing.

```

program ss4;var p:real;
function geron(a,b,c:real):longint;
begin
p:=(a+b+c)/2;
geron:=sqrt(p*(p-a)*(p-b)*(p-c));
end;
var x,y,z :read;
begin
x:=3; y:=4; z:=5;
writeln(geron(x,y,z));
readln;
end.

```

3-masala.

$m! \cdot k!$ ni hisoblovchi dastur tuzing.

```

program funksiya1;
    var f,m,k:integer;
function fact(n:integer):integer;
    var p,i:integer;
    begin p:=1; for i:=2 to n do
        p:=p*i; fact:=p;
    end;
begin    writeln('m!-n! hisoblash uchun ');
        writeln('m va n qiymatini kirit');
        read(m,k); f:=fact(m)-fact(k);
        writeln('f=';f:5); readln;readln;
end.

```

Funksiyaning natijasi uning nomiga o'zlashtirilishi shart, ammo qism dasturning natijalari parametrlar ro'yxatida ham berilishi mumkin. Bu holatda funksiyadan emas, balki protsedura nomli qism dasturdan foydalaniladi. Protседura sarlavhasi quyidagi tuzilishga ega:

Procedure <protседura nomi> (<formal parametrlar ro'yxati >);

Protседura nomi natijani belgilash uchun ishlatilmaydi. Formal parametrlar ro'yxatida natijalar var xizmatchi so'zidan keyin tiplari bilan birgalikda beriladi.

Protседura funksiyaga nisbatan umumiyroq qismdastur bo'lib, funksiya yani protseduraga aylantirish mumkin va protseduraning natijaviy parametrlari bir nechta bo'lishi mumkin (hatto 0 ga teng bo'lishi ham mumkin).

Protседuraning ko'rinishlariga misol keltiramiz:

1) Procedure Uch (a,b,c: Real; var ha, hb, hc, ma, mb, mc: real);

2) Procedure Aylana(R: Real; var c,d,s: real);

3) Procedure Max(Var A,B,C: Real);

4) Procedure Beep(H,T: Word).

Protседuraga murojaat qilish quyidagi ko'rinishga ega:

<Protседura nomi> (<haqiqiy parametrlar ro'yxati >)

Misol.

1) Uch(3,5,7); Uch(x,y,z);

2) Aylana (5.1);

Funksiya va protsedurada formal va haqiqiy parametrlari orasida quyidagi mosliklar bo'lishi lozim:

a) tiplari mos bo'lishi, agar parametr tuzilmalar shaklida bo'lsa unda ham formal va haqiqiy parametrlar tiplari bir xil bo'lishi shart:

b) ular son jihatdan teng bo'lishi mumkin:

v) berilish kema-ketligi mos bo'lishi.

(1 formal parametr 1-haqiqiy parametr bilan, 2-formal parametr 2-haqiqiy parametr bilan va h.).

4-masala.

Berilgan a, b, c –sonlar uchburchak tomonlari bo'la olishsa, True aksincha False qiymatlarini beruvchi funktsiya yarating.

```
function aniqlash(aa,bb,cc:real):boolean;  
begin  
  if (aa+bb>cc) and (bb+cc>aa) and (cc+aa>bb) then  
    aniqlash:=true else  
    aniqlash:=false;  
end;
```

5-masala.

Tomonlari a, b, c bo'lgan uchburchakning perimetrini hisoblash uchun qism-dastur funktsiya yarating va undan shu uchburchakning uzasini hisoblashda foydalaning.

```
var a,b,c:real;  
function per(x,y,z:real):real;  
begin  
  per:=x+y+z;  
end;  
  
PROCEDURE UZa(x,y,z:real);  
  var p,s:real;  
begin  
  p:=per(x,y,z)/2;  
  s:=sqrt(p*(p-x)*(p-y)*(p-z));  
  writeln(s);  
end;  
begin  
  uza(3,4,5);  
END.
```

6-masala.

Ixtiyoriy uchburchak uchun $c^2 = a^2 + b^2 - 2ab \cos C$, $a^2 = c^2 + b^2 - 2ab \cos A$, $b^2 = a^2 + c^2 - 2ab \cos B$, tenglik matematika fanidan o'rinli ekanligi ma'lum. Siz funktsiya yordamida $\cos C$, $\cos A$, $\cos B$ qiymatlarini xisoblovchi dastur yarating,

```
var a,b,c:real;  
  javob:boolean;
```

```

    KosinA,kosinB,kosinC:real;
    label 200,201;
function aniqlash(aa,bb,cc:real):boolean;
begin
    if (aa+bb>cc) and (bb+cc>aa) and (cc+aa>bb) then
        aniqlash:=true else
        aniqlash:=false;
    end;
function burchcos(aa,bb,cc:real):real;
begin
    burchcos:=(sqr(aa)+sqr(bb)-sqr(cc))/(2*aa*bb);
end;

begin

    readln(a,b,c);
    javob:=aniqlash(a,b,c); writeln(javob);
    if javob=false then goto 200 ;

    KosinC:=burchcos(a,b,c);
    writeln(KosinC);
    KosinA:=burchcos(b,c,a);
    writeln(KosinA);
    KosinB:=burchcos(c,a,b);
    writeln(KosinB); goto 201;
    200: writeln('bunday uchburchak mavjud emas');
    201: readln;
end.

```

7-masala.

Prosedura tarkibda umuman parametrdan foydalanmaslik mumkin va shu holat uchun misol keltiramiz:

```

procedure A:
begin writeln('*****');end;
begin A;end.

```

8-masala.

X,y haqiqiy sonlar.Qism dastur-prosedura yordamida ularga berilgan qiymatlarni almashiruvchi dastur yarating.(Qiymat berishda x = 5 va y = 10 ga teng , bo'sa dastur bajarilishida x= 10 va y=5 ga teng bo'lsin).

```

var x,y:real;
    procedure almashish(xx,yy:real);

```

```

    var orkat:real;
    begin
    orkat:=xx;      xx:=yy;      yy:=orkat;      writeln('x=',xx:4:2);
writeln('y=',yy:4:2);
    end;
    begin
    readln(x,y);  writeln('x=',x:4:2);  writeln('y=',y:4:2);  almashish(x,y);
    readln;
    end.

```

9-masala.

Dastur tarkibidagi qism-dasturlar vazifalarini aniqlang.

var m,n:real;

```

Function Max(a,b:real):real;
begin
if a>b then Max:=a else max:=b;
end;
Function Min(a,b:real):real;
begin
if a>b then Min:=a else Min:=b;
end;

begin
readln(m,n);
Writeln('Max=',max(m,n));
    Writeln('Min=',min(m,n));
readln;
end.

```

10-masala.

$Ax^2 + bx + c = 0$ tenglama echimlarini aniqlashda prosedura va uning tarkibida funksiyadan foydalaning.

var m,n,d:real;

```

Procedure KVT(a,b,c:real);
var d,x1,x2,x:real;
function disk(bb,aa,cc:real):real;
begin disk:=sqrt(bb)-4*aa*cc; end;
begin
d:=diskr(b,a,c);

```

```

if d>0 then
  begin
    x1:=(-b+sqrt(d))/(2*a); writeln('x1=',x1:4:3);
    x2:=(-b-sqrt(d))/(2*a); writeln('x2=',x2:4:3);
  end
else
  if d=0 then
    begin x:=(-b+sqrt(d))/(2*a); writeln('x=',x:4:3);
    end
  else
    if d<0 then
      begin
        writeln('yechimga ega emas');
      end;
    end;
begin
  readln(m,n,d);

  KVT(m,n,d);
  readln;

end.

```

11-masala.

200 ta aylananani ekranda tarkibida markazi tasodufiy nuqtalarda joylashgan va radiuslari 0..44 qiymatlardan birini qabul qilishni ta'minlovchi prosedura yordamida tasvirlovchi dastur yarating.

```

uses graph;
var i,gd,gm:integer;

PROCEDURE Aylana:
  var p,s:real;
begin
  circle(random(getmaxx),random(getmaxy),random(45));
end;
begin
  gd:=detect;
  initgraph(gd,gm,"");
  for i:=1 to 200 do
  begin
    setcolor(random(2000));
  end;
  aylana;
end;

```

```
readln;  
END.
```

12-masala.

Tarkibida formal parametri satriy kattalikni "teskarilovchi" finksiyadan foydalnib, kiritilgan ixtiyoriy so'zni teskari tartibda yozuvchi dastur yarating.

```
var s:string;  
function teskari(t:string):string;  
    var w:string; D,i:INTEGER;  
    begin w:=""; D:=LENGTH(T);  
          FOR I:=D downTo 1 DO  
              w:=w+t[i];  
              teskari:=w;  
          end;  
begin  
    readln(s);  
    writeln(teskari(s));  
  
END.
```

13-masala.

1*2*3*4*...*N- ko'paytma qiymatini hisoblovchi funktsiya yarating.

```
function fakt(n:integer):integer;  
    var p,i:INTEGER;  
    begin p:=1;  
          FOR i:=1 To n DO  
              P:=P*i;  
              fakt:=p;  
          end;
```

14-masala.

Yaratilgan fak-funksiyasi yordamida quyidagi yig'indini hisoblash dasturini tuzing:

$1+1*2+1*2*3+1*2*3*4+1*2*3*4*5$
($1!+2!+3!+4!+5!+?$)

```
var k,s:integer;  
function fakt(n:integer):integer;  
    var p,i:INTEGER;  
    begin p:=1;  
          FOR i:=1 To n DO
```



```

    P:=P*i;
    fakt:=p;
end;
begin
    for k:=1 to 5 do
        s:=s+fakt(k);
        writeln(S);
        readln;
    
```

END.

15-masala.

Prosedura yordamida qo'shish, ayirish, ko'paytirish hamda bo'lish amallari berilgan a va b —haqiqiy sonlar uchun bajaruvchi dastur yarating.

```

var a,b:real;
procedure Amallar(x,y:real);
    var P,S,F,B:real;
    label 15,20;
begin
    S:=x+y; Writeln(x:5:2,'+',y:5:2,'=',x+y:5:2);
    F:=x-y; Writeln(x:5:2,'-',y:5:2,'=',x-y:5:2);
    P:=x*y; Writeln(x:5:2,'*',y:5:2,'=',x*y:5:2);
    if y=0 then goto 15 else
    B:=x/y; Writeln(x:5:2,'/',y:5:2,'=',x/y:5:2); goto 20;
    15:writeln(x:5:2,'-ni 0 ga bo"lish mumkin emas');
    20: end;
Begin
    readln(a,b);
    amallar(a,b);
    readln;
end.

```

16-masala.

x va y- haqiqiy sonlar uchun x^y qiymatini hisoblovchi dastur tuzing.

```

var a,b:real;
label 5;
function daraja(a,b:real):real;
begin
    if a > 0 then daraja:=exp(b*ln(a)) else
    if a < 0 then daraja:=(-a*abs(a))*exp(b*ln(abs(a))) else

```

```

if b=0 then daraja:=1 else
    daraja:=0
end;
Begin
5: readln(a,b); if (a=0) and (b=0) then goto 5;
   writeln(daraja (a,b));
   readln;
end.

```

```

PROGRAM HISOBLAGICH ;
VAR FF:text;
    N, P,son: INTEGER;
BEGIN
N:=0; P:=0;
Assign(ff,'c:/musman.txt');
RESET (ff); (*faylni ochish va uning birinchi komponentasini o'qish*)
WHILE NOT EOF (ff) DO
BEGIN
read(ff, son);
writeln(son);

IF son < 0 THEN N:=N+1 ELSE P:=P+1;

END;
WRITELN ('musbat sonlar:', P, ' ta');
WRITELN ('manfiy sonlar:', N, ' ta');
readln;
END.

```



Savol va topshiriqlar.

1. Qism dastur - funktsiyaning berilishi va funktsiyaning nomi.
2. Funktsiyaning o'ziga xos xususiyatlari
3. FUNCTION ning formal va haqiqiy parametrlari.
4. Parametr-qiyamat, parametr-o'zgaruvchi, parametr-doimiylik tushunchalari.
5. Qism dastur protseduraning berilishi va protseduraning nomi.
6. PRECEDURE ning formal va haqiqiy parametrlari.
7. Protسدuraning funktsiyadan farqi.
8. Protسدuraning o'ziga xos xususiyatlari

Mashqlar.

1. [10, 50], [56, 98], [150, 200] oraliqdagi barcha butun sonlarni yig'indisini topish dasturini tuzing.

2. Uchburchakning berilgan tomonlari bo'yicha, uning bissektitsiyasini aniqlovchi dastur tuzing.

3. Uchburchakni uchlarini koordinatalari berilgan: 1) (120,20), (80,170), (140,150), 2) (200,100), (500,156), (210,180), 3) (300,190), (200,390), (415,220). SHu uchburchakni mos ravishda yashil, sariq va qizil ranglarda chizish dasturini tuzing.

4. Uchta sondan kichigini topish dasturini tuzing. Buning uchun ikkita sondan kichigini topish funksiyasini tuzib, undan foydalaning.

5. $Z = \cos x$ funksiyani qiymatini x ning $[-10, 8]$ oraliqdagi butun sonlar qiymatlarida hisoblash dasturini tuzing.

6. $F = \cos 2x$, $z = \text{abs}(F)$ uchun $y_1 = 2 * z^3$ va $y_2 = z^4 - 2 * F$ funksiyalarning qiymatlarini hisoblash dasturi tuzilsin. Bunda x o'zgaruvchi y_1 funktsiya uchun 2dan 10 gacha 3 qadam bilan va y_2 funktsiya uchun 5 dan 15 gacha 4 qadam bilan o'zgaradi. Natija ekranga chiqarilsin.

Mustaqil ishlash uchun mashqlar.

1. *Quyidagi murakkab ifodalarni dasturlash tilida yozing :*

a)
$$y = \frac{1}{\cos x} + \ln \left| \lg \frac{x}{2} \right| + \frac{Ax^2 + Bx - C}{A + B - C} - \cos^2 2 \frac{\pi - x}{3} + \frac{AB}{Cx}$$

b)
$$y = a^{x^2} + (x + a\sqrt{b})^{3x} + \left| \sqrt{\frac{3a - 4b}{2a - 3b - 4}} \right|^2 + e^x$$

c)
$$y = \frac{\sqrt{|\ln|x - ab| - x^3|}}{\sin \alpha + \sin \beta} + \sqrt[3]{\frac{3VH^2}{\pi r^2}} + \frac{1}{3} \pi r^2 H + \text{ctg} \cdot \frac{3\pi}{4}$$

d)
$$y = \frac{3.75a - b}{a^2 + b^2} - \frac{1.25a - b}{a^3 + b^3} + \frac{3a + b}{7ab} + \frac{\sin a + a}{\cosh b + b} \cdot \frac{\sqrt{|2a - b|}}{\ln|a - b|} \cdot e^{\frac{5a}{2b}}$$

Chiziqli masalalarni dasturlash.

2. Temperaturasi t_1 bo'lgan v_1 litr suv t_2 temperaturali v_2 litr suvga qo'shildi. Hosil bo'lgan suvning temperaturasi va hajmini aniqlang.

3. Radiusi r bo'lgan aylanaga tashqi chizilgan muntazam n -burchakning perimetrini toping.

4. x , y va z haqiqiy sonlar berilgan bo'lsin. a va b larning qiymatlarini berilgan ifodalardan foydalanib toping.

$$a = \frac{\sqrt{|x-1|} - \sqrt[3]{|y|}}{1 + \frac{x^2}{2} + \frac{y^2}{4}}, \quad b = x(\operatorname{arctg} z + e^{-(x+3)})$$

5. a) ;

$$6. \quad a = \frac{3 + e^{x-1}}{1 + x^2 |y - \operatorname{tg} z|}, \quad b = 1 + |y - x| + \frac{(y-x)^2}{2} + \frac{|y-x|^3}{3 + e^2};$$

$$7. \quad a = (1+y) \frac{x+y/(x^2+4)}{e^{-x-2} + 1/(x^2+4)}, \quad b = \frac{1 + \cos(y-z)}{x^3/2 + \sin^2 z};$$

8. Teng tomonli uchburchakning tomoni berilgan bo'lsin. Uning yuzi va perimetrini toping.

9. Uzunligi l bo'lgan mayatnikning tebranish davrini aniqlang.

10. To'g'ri burchakli uchburchakning gipotenuzasi hamda bitta kateti berilgan bo'lsin. Uning yuzi va ichki chizilgan aylanasining radiusini toping.

11. Aylananing uzunligi l bo'lsin. Shu aylana bilan chegaralangan doira yuzini aniqlang.

12. Agar halqaning ichki radiusi 20, tashqi radiusi esa undan x miqdorga katta bo'lsa, halqaning yuzini toping.

13. Asoslari a va b , a asosga yopishgan burchagi α bo'lgan teng yonli trapetsiya yuzini toping.

14. Uchburchakning tomonlari berilgan. Shu uchburchakning:

a) balandliklari topilsin;

b) medianalari topilsin;

c) bissektrisalari topilsin;

d) ichki va tashqi chizilgan aylanalarning uzunliklari topilsin.

15. Uchburchak burchaklari va tashqi chizilgan aylanasining radiusi bilan berilgan bo'lsin. Uning tomonlarini aniqlang.

16. Uchburchak uchlarning koordinatalari orqali berilgan bo'lsin. Uning perimetri va yuzini toping.

17. Yon sirti S , asosining yuzi Q bo'lgan silindrning hajmi hisoblansin.

18. Asosining tomoni a , balandligi h bo'lgan muntazam oltiburchakli to'g'ri prizmaning to'la sirtini hisoblang.
19. TSilindr yon sirtining yoyilmasi tomoni a bo'lgan kvadratdan iborat. TSilindrning hajmini aniqlang.
20. Asosining tomonlari a , b va balandligi h bo'lgan to'rtburchakli kesik piramida berilgan bo'lsin. Uning hajmi topilsin.
21. Asosining radiusi R va yasovchisi L bo'lgan konus hajmi va to'la sirtini toping.

Tarmoqlanuvchi jarayonlarni dasturlash.

1. x va y haqiqiy sonlar berilgan bo'lsin. Hisoblang:

- a) $\max(x, y)$;
 b) $\min(x, y)$;
 c) $\max(x, y) + \min(x, y)$.

2. x , y va z haqiqiy sonlar berilgan bo'lsin. Hisoblang:

- d) $\max(x, y, z)$;
 e) $\min(x, y, z)$, $\max(x, y, z)$;;
 f) $\max(x - y + z, xyz)$;
 g) $\min(x - y - 2 + z/3, x - 2y + z, x - y - z)$.

3. a , b va c haqiqiy sonlar berilgan bo'lsin. $a < b < c$ munosabat o'rinlimi ?

4. a , b va c haqiqiy sonlar berilgan bo'lsin. $a \geq b \geq c$ munosabat o'rinli bo'lsa bu sonlarning har birini ikkilantiring, aks holda ularni modullari bilan almashtiring.

5. x va y haqiqiy sonlar berilgan bo'lsin. z ni hisoblang:

$$z = \begin{cases} x - y, & \text{agar } x > y \\ y - x - 1, & \text{aks holda} \end{cases}$$

6. Ikkita haqiqiy son berilgan. Agar ularning birinchisi ikkinchisidan kichik bo'lsa uni nol bilan, aks holda berilgan sonlarning o'рта arifmetigi bilan almashtiring.

7. Ikkita haqiqiy son berilgan. Bu sonlarning kichigini yarin yig'indisi bilan, katasini esa ko'paytmasi bilan almashtiring.

8. Uchta o'zaro har xil sonlarning yig'indisi birdan kichik bo'lsa, berilgan sonlarni

ing eng kichigi, aks holda eng kattasi topilsin.

9. Uchta a , b va c haqiqiy sonlar berilgan bo'lsin. Tomonlari shu sonlarga teng uchburchak mavjudmi? Mavjud bo'lsa, uning perimetri va yuzi topilsin.

10. Ikki a va b haqiqiy sonlar berilgan bo'lsin. $ax+b=0$ tenglamaning yechimlari sonini aniqlang.

11. Uchta a , b va c haqiqiy sonlar berilgan bo'lsin. SHu sonlar uchun $ax^2+bx-c=0$ tenglamaning haqiqiy echimlari mavjudmi? Agar mavjud bo'lsa, bu echimlarni aniqlang.

12. h haqiqiy son berilgan bo'lsin. $ax^2+bx-c=0$ tenglamaning haqiqiy echimlari mavjudmi? Bu yerda

$$a = \sqrt{\frac{|\sin 8h| + 17}{(1 - \sin 4h \cosh^2)}}; \quad b = 1 - \sqrt{\frac{3}{3 + |\operatorname{tg} ah^2|}}; \quad c = a + bh^3 \cos ah$$

Agar haqiqiy echimlari mavjud bo'lsa, ularni aniqlang.

13. a_1 , a_2 , b_1 , b_2 , c_1 va c_2 haqiqiy sonlar berilgan bo'lsin.

$$\begin{cases} a_1x + b_1y + c_1 = 0 \\ a_2x + b_2y + c_2 = 0 \end{cases}$$

tenglamalar sistemasining yechimlari mavjudmi? Agar yechimlari mavjud bo'lsa, ularni aniqlang.

14. Ikki a va b haqiqiy sonlar berilgan bo'lsin. $ax^3+b=0$ tenglamaning haqiqiy yechimlari sonini aniqlang.

15. a , b va c haqiqiy sonlar berilgan bo'lsin. $ax^4 \cdot bx^2 \cdot c = 0$ ($a \neq 0$) bikvadrat tenglamani to'la tekshiring. Haqiqiy echimlari bo'lsa, ularni aniqlang, aks holda bu haqda malumot bering.

16. a haqiqiy soni berilgan bo'lsin. Berilgan $f(a)$ funktsiyaning qiymatini toping:

$$\text{a) } f(x) = \begin{cases} x^2, & \text{agar } x \leq 0 \\ 4, & \text{aks holda} \end{cases} \quad \text{b) } f(x) = \begin{cases} |x|, & \text{agar } -2 \leq x \leq 0 \\ 2x^2 + 4, & \text{aks holda} \end{cases}$$

$$\text{c) } f(x) = \begin{cases} -1, & \text{agar } x < 0 \\ 0, & \text{agar } x = 0 \\ 1, & \text{agar } x > 0 \end{cases} \quad \text{d) } f(x) = \begin{cases} 0, & \text{agar } x \leq 0 \\ x^2 - x, & \text{agar } 0 \leq x \leq 10 \\ x^3 + \sin^2 \frac{\pi}{x}, & \text{aks holda} \end{cases}$$

17. N natural soni berilgan bo'lsin. Uning juft yoki toqligini aniqlang.

18. Kunning K ($K \leq 86400$) soniyasi o'tib bormoqda. Tushlikkacha qancha vaqt qolganligini soat va minutlarda aniqlang. Tushlik vaqti 12.00.00 hisoblanishi va uni o'tib ketgan bo'lishi mumkinligini nazarda tuting.

Takrorlanuvchi jarayonlarni dasturlash.

1. n natural soni berilgan bo'lsin. Hisoblang :

a) $1+2+3+\dots+n$

b) $2n!$

c) $\left(1 + \frac{1}{1^2}\right)\left(1 + \frac{1}{2^2}\right)\left(1 + \frac{1}{3^2}\right)\dots\left(1 + \frac{1}{n^2}\right)$

d) $\frac{1}{\sin 1} + \frac{2}{\sin 1 + \sin 2} + \dots + \frac{n}{\sin 1 + \sin 2 + \dots + \sin n}$

e) $\frac{\sqrt{2 + \sqrt{2 + \dots + \sqrt{2}}}}{n \text{ marta}}$

f) $\frac{\cos 1 \cdot \cos 1 + \cos 2 \cdot \dots \cdot \cos 1 + \cos 2 + \dots + \cos n}{\sin 1 \cdot \sin 1 + \sin 2 \cdot \dots \cdot \sin 1 + \sin 2 + \dots + \sin n}$

g) $\sqrt{3 + \sqrt{6 + \dots + \sqrt{3(n-1) + \sqrt{3n}}}}$

2. x haqiqiy son va n natural soni berilgan bo'lsin. Hisoblang:

a) $\underbrace{((\dots((x+2)^2+2)^2+\dots+2)^2+2)}_{n \text{ marta}}$

b) $\sin x + \sin^2 x + \dots + \sin^n x$

c) $\sin x + \sin x^2 + \dots + \sin x^n$
 $\sin x + \sin \sin x + \dots + \sin \sin \dots \sin x$

d)
$$\frac{(x-2)(x-4)\dots(x-2^n)}{(x-1)(x-3)\dots(x-2^n+1)}$$

e) $(x-1)(x-3)\dots(x-2^n+1)$

3. N natural soni berilgan bo'lsin.

a) Bu sonda qancha raqam qatnashgan ?

b) N soning raqamlari yig'indisi nimaga teng ?

c) N sonini yozishda 3 raqami qatnashganmi ?

d) N sonidagi birinchi va oxirgi raqamlarni aniqlang.

4. N va M natural sonlari berilgan bo'lsin. N sonining oxirgi M ta raqamlari yig'indisini toping.

5. N va M natural sonlari berilgan bo'lsin. Bu sonlarning eng katta umumiy bo'luvchisini toping.

Ko'rsatma: N va M sonlari teng bo'lib qolmaguncha, kattasidan kichigini ayiradi va ayirmaga kattasining nomi berilaveradi.

6. N va M natural sonlari berilgan bo'lsin. N/M kasrni qisqarmaydigan ko'rinishga keltiring.

7. N natural soni berilgan bo'lsin.

a) N ning bo'luvchilarini aniqlang ;

b) N ning bo'luvchilarining yigindisini toping ;

c) N ning tub yoki tub emasligini aniqlang .

8. N natural soni berilgan bo'lsin. $N !!$ ni hisoblang.

$$N!! = \begin{cases} 2 \cdot 4 \cdot \dots \cdot N, & \text{agar } N \text{ soni juft bulsa} \\ 1 \cdot 3 \cdot \dots \cdot N, & \text{agar } N \text{ soni tok bulsa} \end{cases}$$

Eslatma :

9. $a_0 = 1, a_k = ka_{k-1} - 1/k, k = 1, 2, \dots$ bo'lsin. a_n ni toping.

10. $x_0 = c, x_1 = d, x_k = qx_{k-1} + rx_{k-2} + b, k = 2, 3, \dots$ bo'lsin. c, d, q, r, b haqiqiy sonlari va

n natural soni berilgan bo'lsa, x_n ni toping.

11. $u_1 = u_2 = 0$, $v_1 = v_2 = 1$ hamda

$u_i = \frac{u_{i-1} - u_{i-2}v_{i-1} - v_{i-2}}{1 + u_{i-1}^2 + v_{i-1}^2}$; $v_i = \frac{u_{i-1} - v_{i-1}}{|u_{i-2} + v_{i-2}| + 2}$; $i = 3, 4, \dots$ bo'lsin. Agar n natural soni berilgan bo'lsa, v_n ni hisoblang.

12. $a_1 = b_1 = 1$; $a_k = \frac{1}{2} \left(\sqrt{b_{k-1}} + \frac{1}{2} \sqrt{a_{k-1}} \right)$; $b_k = 2a_{k-1}^2 + b_{k-1}$, $k = 2, 3, \dots$

bo'lsin. n natural soni berilgan. Hisoblang: $\sum_{k=1}^n a_k b_k$.

13. $a_1 = b_1 = 1$; $a_k = 3b_{k-1} + 2a_{k-1}$; $b_k = 2a_{k-1} + b_{k-1}$, $k = 2, 3, \dots$

bo'lsin. Hisoblang: $\sum_{k=1}^n \frac{2^k}{(1 + a_k^2 + b_k^2)k!}$.

14. a , x va ε musbat haqiqiy sonlar berilgan bo'lsin. y_1, y_2, \dots ketma-ketlik

$$y_0 = a; y_i = \frac{1}{2} \left(y_{i-1} + \frac{x}{y_{i-1}} \right), i = 1, 2, \dots$$

formula bilan hosil qilinadi. SHu ketma-

ketlikning $|y_i^2 - y_{i-1}^2| \leq \varepsilon$ tengsizlikni qanoatlantiruvchi dastlabki hadi topilsin.

15. a haqiqiy soni berilgan bo'lsin. x_0, x_1, x_2, \dots ketma-ketlik

$$x_n = \begin{cases} \min(2a, 0.95), & \text{agar } a \leq 1 \\ \frac{a}{5}, & \text{agar } 1 < a < 25 \\ \frac{a}{25}, & \text{boshqa hollarda} \end{cases} \quad x_n = \frac{4}{5}x_{n-1} + \frac{a}{4x_{n-1}}$$

formular yordamida aniqlanadi. SHu ketma-ketlikning $\frac{5}{4}a|x_{n+1} - x_n| \leq 10^{-4}$ shartni qanoatlantiruvchi birinchi hadini toping.

16. K va N ($N \geq K \geq 0$) natural sonlari berilgan. Hisoblang

$$\frac{N \cdot (N-1) \cdot \dots \cdot (N-K+1)}{K!}$$

17. Hisoblang $1 - \frac{1}{2} + \frac{1}{3} - \dots + \frac{1}{9999} - \frac{1}{10000}$.

$$\frac{1}{1 + \frac{1}{3 + \frac{1}{5 + \dots}}}$$

18. Hisoblang $101 + \frac{1}{103}$.

19. x haqiqiy soni berilgan bo'lsin. ($x \neq 0$). Hisoblang

$$\frac{x}{x^2 + \frac{x^2}{x^2 + \frac{x^2}{x^2 + \dots}}}$$

20. Hisoblang:

a) $\sum_{i=1}^{100} \frac{1}{i^2}$

b) $\sum_{i=1}^{128} (-1)^i \frac{1}{i!+1}$

c) $\prod_{i=1}^{50} \frac{i^2}{i^2 + 2i + 3}$

d) $\prod_{k=3}^{103} \left(1 - \frac{1}{k!}\right)^2$

21. n ($n > 3$) natural soni berilgan bo'lsin. Hisoblang:

a) $\sum_{k=1}^n \frac{1}{(2k+1)^2}$

b) $\sum_{k=0}^n (-1)^k \frac{k+1}{(2k)!}$

$$c) \prod_{k=3}^n \left(\frac{k}{2k+1} - \cos^k |k| \right)$$

$$d) \prod_{k=1}^n \frac{(1-k)^2 + 1}{((k-1)!+1)^2}$$

22. x haqiqiy son hamda $0 < \varepsilon < 10^{-4}$ sonlari berilgan. Berilgan cheksiz yig'indilarning qiymatini ε aniqlikda hisoblang. Talab qilingan aniqlik dastlabki k ta hadlarning yig'indisi va dastlabki $k+1$ ta hadlarning yig'indisi orasidagi farqning absolyut qiymati ε dan kichik bo'lganda erishilgan deb faraz hisoblansin.

$$a) \sum_{k=1}^{\infty} \frac{x}{k^2}$$

$$b) \sum_{k=1}^{\infty} \frac{x+k^2}{k!}$$

$$c) \sum_{k=1}^{\infty} (-1)^k \frac{x^2 - kx + 0.25}{(2k+1)!}$$

$$d) \sum_{k=1}^{\infty} (-1)^{k-1} \frac{kx}{k! + 2^k + 3^{k-1}}$$

23. x_1, x_2, \dots, x_{50} haqiqiy sonlar berilgan bo'lsin. SHu sonlarning eng kattasini toping.

24. x_1, x_2, \dots, x_{50} haqiqiy sonlar berilgan bo'lsin. SHu sonlarning hammasi qaysi oraliqqa tegishli bo'ladi?

25. Fibonachchi sonlari $u_1 = u_2 = 1, u_i = u_{i-1} + u_{i-2}, i=3, 4, \dots$ formula yordamida aniqlanadi.

a) n natural soni berilgan bo'lsin. Fibonachchi sonlarining dastlabki n ta hadlarining yig'indisi topilsin;

b) $K > 0$ son berilgan bo'lsin. Fibonachchi sonlarining K dan katta bo'lgan birinchi hadini toping;

c) $K > 0$ son berilgan bo'lsin. Fibonachchi sonlarining K dan kichik hadlari ichida eng kattasini toping.

26. n natural soni va a_1, a_2, \dots, a_n hamda x haqiqiy sonlar berilgan bo'lsin. Gornor shemasi bo'yicha

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

ko'phadning qiymatini hisoblang.

Grafiklar bilan ishlash.

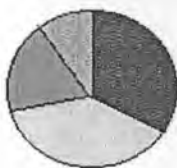
1. Quyidagi geometrik figuralarni yasang:

- a) Uchlari (100, 100), (150, 100) va (100, 150) nuqtalarda yotgan uchburchak ;
- b) Uchlari (80, 80), (170, 80), (170, 150) va (80, 150) nuqtalarda yotgan to'g'ri to'rtburchak ;
- c) Uchlari (120, 100), (140, 120), (140, 140), (120, 160), (100,140) va (80, 150) nuqtalarda yotgan oltiburchak.

2. Markazi ekran markazida joylashgan, tomonlari ekran tomonlariga parallel yoki perpendikulyar, tomonlarining uzunligi 70 va 100 piksel bo'lgan to'g'ri to'rtburchak yasang va bo'yang.

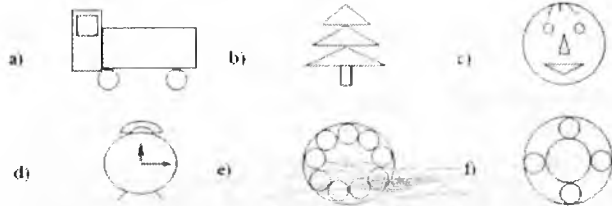
3. Markazi ekran markazida joylashgan, radiusi 125 piksel bo'lgan doira tasvirini yasang va bo'yang.

4. Sektorli diagramma – bu sektorlarga bo'lingan doira bo'lib, uning har bir sektorining yuzi berilgan sonlarga proporsional. Sektorlar bir - birlaridan ranglari bilan ajralib turadi. a_1, a_2, a_3, a_4, a_5 va a_6 haqiqiy sonlar berilgan bo'lsin. SHu sonlar uchun sektorli diagramma yasang.



5. Boshi ekran markazida joylashgan, vertikal va gorizontal strelkalar kesishuvidan hosil bo'ladigan Dekart koordinatalar tekisligi tasvirini yasang.

6. Quyidagi tasvirlarni yarating:



7. Quyidagi funksiyalarning grafiklarini yasang:

a) $y = \frac{x^2 + 3x - 4}{x - 2}$

d) $y = \frac{e^x}{3x^2 + 2x + 1}$

b) $y = 3 - \frac{e}{\sin x} - \frac{3}{x^2}$

e) $y = \frac{2x + e}{x + 2x + 3}$

c) $y = \sin x$

f) $y = \cos x + |x|$

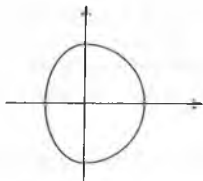
8. Parametrik ko'rinishda berilgan quyidagi egri chiziqlarning grafiklarini yasang:

- a) Markazi koordinatalar boshida yotgan r radiusli aylana :

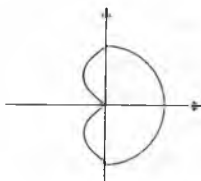
$$x = r \cos t, y = r \sin t, t \in [0, 2\pi]$$

b) Katta va kichik yarim o'qlari r_1 va r_2 bo'lib, koordinata o'qlariga parallel bo'lgan ellips: $x = r_1 \cos t, y = r_2 \sin t, t \in [0, 2\pi]$

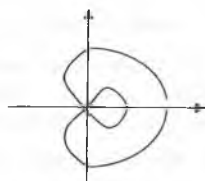
c) Paskal shilliqurtlari. Bunda $b \geq 2a, a \cdot b \leq 2a$ va $a \geq b$ bo'lgan hollarni ko'rib chiqishni unutmang. $a > 0, b > 0, t \in [0, 2\pi]$



$$b > 2a$$



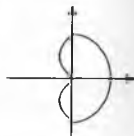
$$a < b < 2a$$



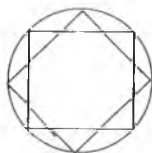
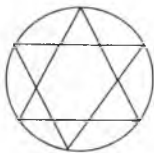
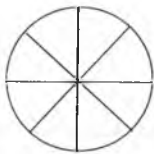
$$a > b$$

d) Kardiodida
 $y = a \sin t(1 + \cos t), a > 0, t \in [0, 2\pi]$

$$x = a \cos t(1 + \cos t),$$



9. Quyidagi tasvirlarni hosil qiling.



10. Aylana va unga ichki chizilgan muntazam oltiburchak tasvirini yasang.

Fayllar bilan ishlash.

1. Fibonachchi sonlari ketma-ketligi

$$u_0, u_1, u_2, \dots, u_i, u_{i+1}, u_{i+2}, \dots$$

formula bilan aniqlanadi. K natural soni berilgan bo'lsin. K dan kichik bo'lgan barcha Fibonachchi sonlarini f fayliga yozing.

2. N natural soni berilgan bo'lsin. Undan kichik bo'lgan barcha tub sonlarni g fayliga yozing.

3. *type seriya file of real.*

a seriyadagi manfiy sonlar miqdorini aniqlash dasturini yozing.

4. $\text{type narx} = \text{record sum}:0..maxint; \text{tiyin}:0..99 \text{ end};$

$\text{baho} = \text{file of narx};$

t o'zgaruvchiga baholar faylidagi eng arzon narxni qiymat qilib beruvchi dastur yozing.

5. Elementlari haqiqiy sonlardan iborat q fayli berilgan bo'lsin. Shu faylda sonlar o'sish tartibida yozilganmi yoki yo'qmi?

6. Elementlari haqiqiy sonlardan iborat p va q fayllari berilgan bo'lsin. Bu fayllar bir-biriga tengmi ?

7. Haqiqiy sonlarning h fayli berilgan bo'lsin.

a) h faylidagi eng kichik va eng katta sonni toping;

b) h faylidagi sonlarning yig'indisini aniqlang;

c) h faylidagi sonlarning ko'paytmasini aniqlang;

d) h faylida manfiy sonlar ko'pmi yoki musbatlarimi ;

e) $a_1, a_2, a_3, a_4, \dots, a_k, \dots$ sonlar h faylining elementlari bo'lsa,

$$a_1 - a_2 + a_3 - a_4 + \dots + (-1)^{k-1} a_k - \dots$$

f) h faylida eng oxirgi manfiy son nechanchi o'rinda uchraydi?

8. Butun sonli g fayli berilgan bo'lsin. Undagi juft sonlarning yig'indisi hamda toq sonlarning ko'paytmasini hisoblang.

9. Haqiqiy sonli f fayli berilgan bo'lsin. Undagi manfiy sonlarni g fayliga, musbatlarini esa h fayliga ko'chiring.

10. Haqiqiy sonli g fayli berilgan. Undagi toq nomerli elementlarni f fayliga, juft nomerlilarni esa h fayliga ko'chiring.

11. Haqiqiy sonli f va g fayllari berilgan. f dagi elementlarni g fayliga, g faylidagi elementlarni esa f fayliga ko'chiring. Bunda yordamchi h faylidan foydalanish mumkin.

12. f matnli faylida haqiqiy sonlar bir-biridan bo'sh joy belgisi bilan ajratilgan holda saqlanadi. Undagi eng katta va eng kichik elementlarni aniqlang.

13. Talabalarning familiyasi, ismi hamda informatika va matematik tahlil fanlaridan olgan baholari saqlanayotgan f fayli berilgan bo'lsin.

a) familiyasi x , ismi y bo'lgan talabaning informatika va matematik tahlil fanlaridan olgan baholarini aniqlang;

b) alochi talabalarning ro'yxatini aniqlang;

c) talabalarning har bir fan bo'yicha o'rtacha o'zlashtirish darajasini aniqlang.

14. Avtomobillarning nomi, qayd qilish raqami hamda egasining familiyasi va ismi haqidagi ma'lumotlar g faylida saqlanayotgan bo'lsin.

a) Qayd qilish raqami k bo'lgan barcha avtomobillarning nomi va egasi aniqlansin:

b) Familiyasi va ismi x va y bo'lgan shaxsga tegishli barcha avtomobillar ro'yxatini aniqlang;

c) q nomli avtomobilning qayd qilish nomeri va egasini toping.

15. Telefonlarning f faylida abonentning familiyasi, ismi, telefon nomeri va yashayotgan shaxri va ko'chasi haqidagi malumotlar saqlanadi.

a) Familiyasi p , ismi q bo'lgan abonent mavjudmi ?

b) k nomerli telefon egasini aniqlang ;

c) t shaharning q ko'chasidagi abonentlar ro'yxatini aniqlang.

16. Har birida N tadan butun son saqlanayotgan f va g fayllari berilgan bo'lsin. Bu fayllardagi bir xil o'rinda turgan elementlarning eng katta umumiy bo'luvchilarini h fayliga yozing.

17. Matnli f fayli berilgan bo'lsin. Bu matnni g fayliga shunday ko'chiringki, g ning har bir satri yoki nuqta bilan tugasin yoki 60 ta belgidan iborat bo'lsin

18. f va g fayllarida haqiqiy sonlar o'sish tartibida yozilgan. O'sish tartibini saqlagan holda bu fayllardagi malumotlarni h fayliga ko'chiring.

Protsedura va funktsiya.

1. x ning qiymatlari -2.34 , 0 , 5.6 bo'lgan hollar uchun quyidagi dastur natijalarini aniqlang:

function sign(t:real):integer;

begin if t>0 then sign: =1

else if t=0 then sign: =0 else sign: =-1;

end;

2. Berilgan ikkita natural sonlarning eng katta umumiy bo'luvchisini topish uchun protsedura-funktsiya yozing.

3. Quyidagi dastur yordamida qanday masala echilgan?

function max(m,n:real): real;

begin if m > n then max: = m else max: = n;

end;

4. Quyidagi dasturlar matnida mavjud xatoliklarni toping.

a) *function f(a: 'a'..'z'): integer;*

begin

f := ord(a)-ord('a');

if f < 0 then f := -1

end;

b) function $g(k: \text{integer}): 0.. \text{maxint}$;

var $i, s: 0.. \text{maxint}$;

begin

$s := 0$;

for $i := 1$ to k do $s := s + \text{sqr}(i)$

end;

c) function $h(x: \text{integer}): \text{integer}$;

begin

$h(x) := (\text{sqr}(x) + x) / 2$

end;

5. x haqiqiy son berilgan bo'lsin. Quyidagi funktsiyaning qiymatini hisoblang:
 $sh(x) \text{tg}(x+1) - \text{tg}^2(2+sh(x-1))$.

6. s va t haqiqiy sonlar berilgan. Hisoblansin

$f(t, -2s, 1.17) + f(2.2, t, s-t)$

Bu erda $f(a, b, c) = \frac{2a - b - \sin c}{5 + |a - b - c|}$.

7. s va t haqiqiy sonlar berilgan bo'lsin. Hisoblansin

$[g(1.2, s) + g(t, s) - g(2s-1, st)] / g(2t, 3s)$

Bu erda $g(a, b) = \frac{2a^2 + 3b}{a^2 + 2ab + 3b^2 + 5a^2b^3 + e^a - e^b}$.

8. y haqiqiy soni berilgan bo'lsin. Quyidagi funktsiyaning qiymatini toping:

$\frac{1.7t(0.25) + 2t(1+y)}{6 - t(y^2 - 1)}$, by erda $I(x) = \frac{\sum_{k=0}^{10} x^{2k+1} (2k+1)!}{\sum_{k=0}^{10} x^{2k} (2k)!}$.

9. a, b, c haqiqiy sonlar berilgan bo'lsin. Hisoblang

$\frac{\max(a, a+b) + \max(a, b+c) + \max(a, a+c)}{1 + \max(a+bc, 1.15)}$.

10. a, b, x, y haqiqiy sonlar berilgan. Quyidagi ifodaning qiymatini hisoblang:

$\frac{\cos(a+bi) + \sin^2(x+yi)}{\cos(3ax + 2byi) + \sin(x+yi)} \cdot \text{ctg}(a+bi)$

Bu erda

$$\cos(c + di) = \cos c \cdot \frac{e^d + e^{-d}}{2} + \sin c \cdot \frac{e^d - e^{-d}}{2} i$$

$$\sin(c + di) = \sin c \cdot \frac{e^d + e^{-d}}{2} + \cos c \cdot \frac{e^d - e^{-d}}{2} i$$

11. a, b, c, d haqiqiy sonlar berilgan. Hisoblang :

$$\frac{3e^{(a+bi)} + 4e^{c+di}}{5e^{2ac+3bdi}} \quad \text{Bu erda } e^{x-yi} = e^x (\cos y + i \sin y).$$

12. a va b haqiqiy sonlar berilgan. Agar $u = \min(a, b)$ hamda $v = \min(2ab, 3a+b)$ bo'lsa, $\min(u + v^2, 3.14)$ ni hisoblang.

13. n va m natural sonlari hamda $a_1, \dots, a_n, b_1, \dots, b_m, c_1, \dots, c_{30}$ haqiqiy sonlari berilgan bo'lsin. Hisoblang

$$t = \begin{cases} \min(b_1, \dots, b_m) + \min(c_1, \dots, c_{30}) & \text{agar } \max(a_1, \dots, a_n) \geq 0 \\ 1 + (\max(a_1, \dots, a_n))^2 & \text{bo'shqa hollarda} \end{cases}$$

14. k, l va m natural sonlari hamda $x_1, \dots, x_k, y_1, \dots, y_l, z_1, \dots, z_m$ haqiqiy sonlari berilgan bo'lsin. Hisoblang

$$t = \begin{cases} (\max(x_1, \dots, x_k) + \max(z_1, \dots, z_m)) / 2 & \text{agar } \max(x_1, \dots, x_k) \geq 0 \\ \min(y_1, \dots, y_l) + \max(z_1, \dots, z_m) & \text{bo'shqa hollarda} \end{cases}$$

15. s va t haqiqiy sonlar berilgan bo'lsin. Hisoblang

$$h(s, t) + \max(h^2(s-t, st), h^4(s-t, s+t)) + h(1+s, 1+t).$$

$$\text{Bu erda } h(a, b) = \frac{a}{1+b^2} + \frac{b}{1+a^2} - \frac{a+b}{ab} + 2$$

16. a_0, \dots, a_6 haqiqiy sonlar berilgan bo'lsin. $x=1, 2, 3, 4$ lar uchun $p(x-1) - p(x)$ funksiyaning qiymatini hisoblang. Bu erda

$$p(y) = a_6 y^6 + a_5 y^5 + \dots + a_1 y + a_0.$$

17. a, b, c va d natural sonlari berilgan. Bu sonlar uchun ab va cd kasrlarni qisqarmaydigan kasr ko'rinishiga keltiring. (Ikki natural sonning eng katta umumiy bo'luvchisini topish protsedura-funksiyasidan foydalaning.)

18. $x_1, y_1, \dots, x_{10}, y_{10}$ haqiqiy sonlar berilgan. O'n burchak uchlariining koordinatalari mos ravishda $(x_1, y_1), \dots, (x_{10}, y_{10})$ bo'lsin. Shu o'n burchakning perimetrini hisoblang. (Koordinatalari berilgan ikki nuqta orasidagi masofani topish protsedura-funksiyasidan foydalaning.)

19. To'rtburchak uchlariining koordinatalari berilgan bo'lsin. Koordinatalar boshi

bo'lgan $(0, 0)$ nuqta shu to'rtburchak ichida yoxudmi? (Uchburchak kengligini va yuzini malum bo'lgan uchburchak yuzini topish protsedura funktsiyasini yozing.)

20. Uchta natural soni berilgan bo'lsin. Ularning eng katta umumiy bo'luvchani toping. (Ikki natural sonning EKUB ini topish protsedura funktsiyasidan foydalaning.)

Foydalanilgan adabiyotlar ro'yxati

1. С.А.Немнюгин. TURBO PASCAL G Практикум, 2-сиздание, Москва... – “питер” – 2007
2. Немнюгин С. А. Turbo Pascal. Учебник. – М.,2002
3. Кульгин Н. Программирование в Turbo Pascal 7.0 и Delphi, 2-е издание, С. Петербург, БХВ-С. Петербург, 1999 г.
4. В. Boltayev, М. Maxkamov, А. Azamatov “Paskal dasturlash tili” Toshkent- 2007.
5. Ш.Назиров. Дастурлаш тиллари Т.2007 й..
6. Р. Бокиев ва бошқалар. Paskal дастурлаш тили. Т.2007у
7. Фараонов В.В. Программирование на языке высокого уровня Delphi – М., 2003.
8. М.Ашуров, И.Мирмахмудов, Ш.Сапаев. Замонавий дастурлаш тиллари Фанидан лаборатория ишлари – Т. 2008.

Mundarija

Kirish.....	3
1-§. Turbo Paskal 7.0 integrallashgan muhiti.....	6
2-§. Turbo-Paskal dasturlash tili va uning alifbosi.....	13
3-§. O'zgarmas va o'zgaruvchi miqdorlar.....	20
4-§. Jadval ko'rinishidagi miqdorlar.....	24
5-§. Standart funktsiyalar va algebraik ifodalar.....	31
6-§. O'zlashtirish va ma'lumotlarni ekranga chiqarish operatori.....	34
7-§. Ma'lumotlarni xotiraga muloqot usulida kiritish.....	38
8-§. Ekran bilan ishlash oratorlari.....	41
9-§. Chiziqli dasturlar tuzish.....	47
10-§. Tarmoqlanish va o'tish operatorlari.....	51
11-§. Tanlash (Case) operatori.....	56
12-§. Parametrik takrorlash operatori.....	59
13-§. Shart bo'yicha takrorlash operatorlari.....	63
14-§. Belgili miqdorlar bilan ishlash funktsiyalari.....	66
15-§. Satrli miqdorlar bilan ishlash funktsiyalari va protseduralari....	69
16-§. Paskalda ekranni grafik holatga o'tkazish.....	76
17-§. Paskalning shakllar chizish operatorlari.....	82
18-§. Fayllar bilan ishlash.....	89
19-§. Qism dasturlar. Funktsiyalar. Protseuralar.....	95
Mustaqil ishlash uchun mashqlar.....	106
Foydalanilgan adabiyotlar ro'yxati.....	121

