

**O'ZBEKISTON RESPUBLIKASI
XALQ TA'LIMI VAZIRLIGI**

**SAMARQAND XALQ TA'LIMI XODIMLARINI QAYTA TAYYORLASH VA
MALAKASINI OSHIRISH HUDUDIIY MARKAZI**

**ANIQ VA TABIIY FANLAR METODIKASI
KAFEDRASI**

Aminov I.B., Tugalov R.G'.

PASKAL DASTURLASH TILIDA MASALALAR TO'PLAMI

Uslubiy ko'rsatma

Samarqand - 2020 yil

Aminov I.B., Tugalov R.G' Paskal dasturlash tilida masalalar to'plami. Uslubiy ko'rsatma. - SamXTXQTMOHM, Samarqand, 2020-yil. - 32 bet.

Uslubiy ko'rsatma umumta'lim maktablarining informatika fani o'qituvchilari uchun mo'ljallangan bo'lib, u dasturlash asoslarining boshlang'ich tushunchalari, asosiy elementlari, operatorlari, ularning turlari va qo'llanilish o'rni, dasturlash ko'nikma va malakalarini shakllantirish bo'yicha nazariy va amaliy bilimlarni o'z ichiga olgan.

Taqrizchilar:

B.I.Usmonov - SamXTXQTMOHM Tabiiy va aniq fanlar metodikasi kafedrasini mudiri, dotsenti.

F.Nomozov-SamDU Ta'limda kompyuter texnologiyalari kafedrasini dotsenti.

So'z boshi

Bugungi kunda mustaqil respublikamizda «Kadrlar tayyorlash Milliy dasturi» ni amalga oshirish borasida bir qancha tadbirlar amalga oshirilishi rejalashtirilgan bo'lib, shulardan biri, kadrlar tayyorlash tizimi va mazmunini mamlakatning ijtimoiy va iqtisodiy istiqbollardan, jamiyat ehtiyojlaridan, fan, madaniyat, texnika va texnologiyalarning zamonaviy yutuqlaridan kelib chiqqan holda qayta qurishdan iboratdir.

Ta'lim tizimida har bir fanning o'qitish mazmunini qayta tahlildan o'tkazish va uni tubdan isloh qilish nuqtayi nazaridan yondashuv zarur bo'ladi. Rivojlangan davlatlar tajribasidan shu narsa ma'lumki, bozor iqtisodi sharoitida axborot tizimining qanchalik suratda takomillashuvi davlat tizimining ravnaq topish darajasini belgilashi ma'lum. Shu ma'noda, axborotlarning ishonchliligi, ularning o'z vaqtida tayyorlanishi ko'p jihatdan kompyuter dasturchilarning mukammal darajada bilishligiga bog'liq, chunki bu uch vosita bir-biri bilan chambarchas bog'langan holda faoliyat ko'rsatadi.

Shu narsa quvonchliki, Respublikamizni mustaqillikka erishuvi natijasida hozirgi kunda barcha o'quvchilar va talabalar zamonaviy kompyuterlardan foydalanish imkoniyatiga ega bo'ldilar. Shu bois ham, "Informatika va axborot texnologiyalari" fani barcha o'quv yurtlarida o'qitiladigan kompyuter savodxonligi borasidagi uzluksiz dastur tizimiga kiritilgan Davlat ta'lim konsepsiyasiga asoslanadi.

Uslubiy ko'rsatmadan maktab o'qituvchilari "Informatika va hisoblash texnikasi asoslari" fanidan dasturlash asoslariga doir mavzularni o'rganishda va masalalar yechishda foydalanishlari mumkin.

Bundan tashqari uslubiy ko'rsatmadan Paskal dasturlash tilini o'rganishga qiziqqan barcha maktab, akademik litsey va kasb-hunar kolleji o'quvchilari ham foydalanishlari mumkin.

Pascal dasturlash tili to'g'risida qichqacha ma'lumotlar

Turbo - paskal tizimi va u bilan ishlash qoidalari

Pascal dasturlash tilining birinchi versiyasi **1968 yilda** Stenford universitetida shved olimi Niklaus Virt tomonidan yaratilgan. 1982-yilda Amerikaning Borland firmasi tomonidan Turbo Pascal tizimining yaratilishi, Pascal tilining keng doirada qo'llanila boshlanishiga asos bo'ldi. 1992-yilda Borland firmasi Turbo Pascal 7.0 - versiyani taqdim etdi. Bu versiyada kompilyatorning ishi ancha yaxshilangan, obyektga mon'ljallangan dasturlash tamoyili to'liq ishga tushirilgan va uning imkoniyatlari kengaytirilgan.

Dasturni ishga tushirish.

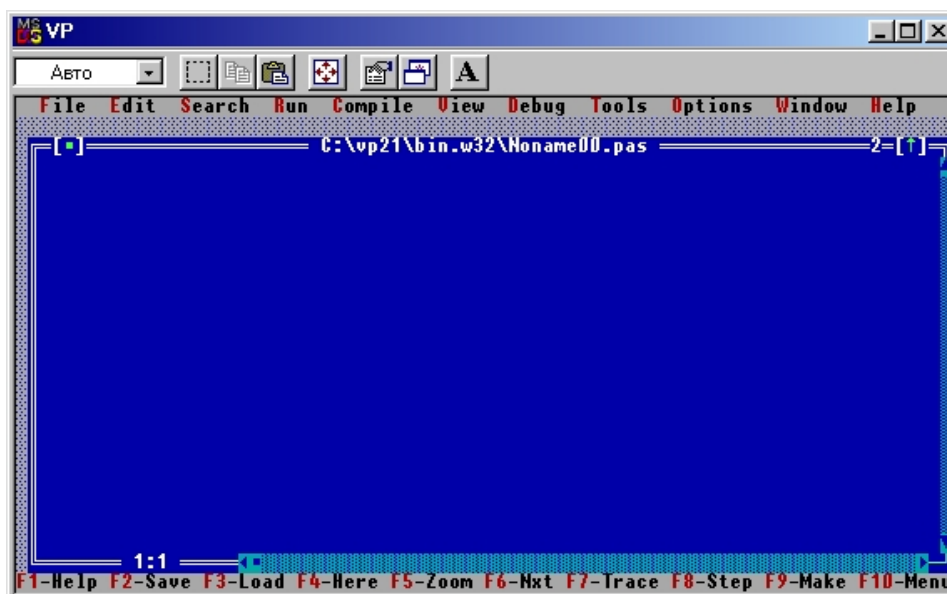
IDE (Integrated Development Environment) - integrallashgan vositalar qobig'i (**IVQ**) - Turbo Pascal muhiti deb yuritiladi. Turbo Pascal muhiti **turbo.exe** fayli orqali **MS DOS** operatsion tizimining buyruqlar satrida yoki **Windows** muhitining ishchi stolida ishga tushiriladi.

Turbo Pascal tili dasturlashning muloqot (dialog) tizimidan iborat. Bu tizim - matn muharriri, kompilyator, jamlovchi va yuklovchi qismlardan iborat Turbo Pascal tizimidan chiqish [**ALT+X**] tugmachalarini birgalikda bosish orqali amalga oshiriladi.

Turbo Paskal tizimining bosh "menyu" si

Turbo Paskal muhiti ishga tushirilgandan keyin kompyuter ekranida uning oynasi paydo bo'ladi. Oyna sarlavhalar satri, bosh menyu va ishchi soha (matn muharriri) va buyruqlar satridan iborat.

Sarlavhalar satri matn muharririda yozilgan dastur nomidan iborat bo'ladi. Pascal tilida tuzilgan dastur bilan ishlash uchun, asosan, bosh menyudan foydalaniladi. Bosh menyuga chiqish uchun F10 tugmacha bosiladi.



File Edit Seatch Run Compile Debid Tools Options Winlow Help

Bosh “menyu” quyidagi asosiy bo‘limlardan iborat:

1. File – fayllar bilan ishlash amallarini bajarishga mo‘ljallangan. **File** bilan ishlash amallari:

- **Open (F3)** – mavjud faylni muharrirlash uchun yuklash. Bu bo‘lim tanlangandan keyin ekranda muloqat oynasi hosil bo‘ladi. Oynada yuklash lozim bo‘lgan fayl nomi ko‘rsatilib, **Enter** tugmachasi bosiladi. Fayl yuklangandan keyin u muharrirlash va uni bajarish uchun tayyor bo‘ladi.

- **Save(F2)** – faylni saqlash. Bu bo‘lim ishga tushirilganda muharrir xotirasidagi fayl diskka yoziladi. Agar muharrirlanayotgan fayl standart **Noname.pas** nomiga ega bo‘lsa, u holda saqlashda fayl nomini o‘zgartirish imkoniyati mavjud bo‘ladi.

- **Save as** – muharrirlanayotgan faylni boshqa nom bilan saqlash.

- Ushbu buyruq amalga oshirilgandan keyin muharrirlanayotgan faylga boshqa nom beriladi va u kompyuter xotirasiga yangi nom bilan saqlanadi.

- **New** – yangi faylni yaratish. Bu buyruq berilgan integrallashgan muhit muharriri yangi fayl yaratish tartibiga o‘tadi. Yaratilayotgan faylga **Noname.pas** nom beriladi. Faylni saqlashda unga ixtiyoriy nom berish mumkin bo‘ladi.

- **Exit (Alt +x)** – dasturlash muhitida ishni tugallash. Bu buyruq berilgandan keyin integrallashgan muhitda ish tugallaniladi va undan chiqiladi.

2. Edit – fayllarni muharrirlashni qo‘shimcha amallari **“menyu”**si bo‘lib, matnni muharrirlashning asosiy funksiyalarini bajaradi. Bu amallar quyidagilardan iborat:

- **matn bo‘laklarini nusxalash;**

- **matn bo‘laklarini o‘rnatish;**

- **matn bo‘laklarini o‘chirish;**

- **muharrirlanayotgan faylning boshlang‘ich variantini tiklash ;**

3. Search – matn bo‘laklarini izlash va almashtirish **“menyu”**si bo‘lib, matn qismlarini izlashni, lozim bo‘lganda ularni yangisi bilan almashtirishni amalga oshiradi.

4. Run – Pascal dasturini bajarish **“menyu”**si bo‘lib, ishchi sohada joylashgan dasturni ishga tushiradi. Bundan tashqari ushbu dasturni qadamma-qadam hamda lozim bo‘lganda esa ayrim qismini bajarishni amalga oshiradi.

Bu buyruq berilganda dastur ishga tushadi, agar dasturda xatoliklar mavjud bo'lmasa, o'zgaruvchilarga sonli qiymatni berishni talab etuvchi muloqat oyna hosil bo'ladi. Qiymatlar tartib bilan berilgandan keyin **Enter** tugmachasi bosilib, dastur ishga tushiriladi. Dastur natijasini ko'rish **Alt +F5** tugmachalarini bosish orqali amalga oshiriladi.

5. Compili – kompilyasiya “**menyu**”si bo'lib, ishchi sohada joylashgan dasturni kompilyasiya qilish uchun ishlatiladi. Bu buyruq berilganda dastur kompilyasiya qilinadi. Agar kompilyasiya vaqtida dasturda xatolik mavjud bo'lsa, u holda bu haqda kerakli xabar kompyuter ekraniga chiqariladi. Xatoliklar tuzatilgandan keyin dastur yana kompilyasiya qilinadi. Kompilyasiya qilingan dastur bajariladi.

6.Debug – dasturni otladka qilish amallari “**menyu**”si bo'lib, dasturda mavjud xatoliklarni topishni yengillashtiruvchi buyruqlardan iborat.

7. Tools – instrumental vositalar “**menyu**”si bo'lib, Turbo Pascal muhitidan chiqmasdan turib ayrim dasturlarni bajarishni amalga oshiradi.

8. Options- muhit parametrlarini o'zgartirish “**menyu**”si bo'lib, Turbo Pascal muhitida ishlash jarayonida lozim bo'lgan kompilyator parametrlarini o'rnatish uchun ishlatiladi.

9. Window – oynalarni boshqarish “**menyu**”si bo'lib, oynalar bilan ishlashning barcha asosiy amallarini bajaradi. Bu amallar: *oynalarni ochish, oynani yopish, oynani kerakli joyga joylashtirish va oyna o'lchovini o'zgartirishdan iborat.*

10. Help – yordam olish “**menyu**”si bo'lib, muhitda ishlash bo'yicha kerakli ma'lumotlarni olish uchun ishlatiladi.

Yuqorida ta'kidlaganimizdek, **bosh “menyu”ga** qarashli bo'lim – **ostki “menyu”** deb yuritiladi. Ostki “**menyuga**” buyrug'ini tanlash uchun uning nomi ustiga kursor ko'rsatkichi joylashtirilib, **Enter** tugmachasi bosiladi. Ostki “**menyu**” dan chiqish **Esc** tugmachasini bosish orqali amalga oshiriladi.

Turbo-Paskal matn muharriri buyruqlari

Turbo Pascal muhitida ishlash uchun matn muharriri mavjud bo'lib, unda ishlash eng quyi darajadagi matn muharririga o'xshab ketadi. Turbo Pascal muhiti matn muharriri Pascal tilidagi dasturlarni kiritish va ularni muharrirlash uchun ishlatiladi.

Turbo Pascal muhitida ishlashda ma'lum bir cheklashlar mavjud:

- 1. Faylda umumiy belgilar soni 64535 tadan oshib ketmasligi kerak.*
- 2. Dasturda bitta satrning uzunligi 126 ta belgidan ortiq bo'lmasligi kerak.*
- 3. Satrni 127- chidan 249- chi o'rniga bo'lgan joy izohlarni joylash uchun ishlatiladi.*

Matn muharririda ishlashda har bir satrning oxirida **Enter** tugmachasi bosilib, keyingi yangi satrga o'tiladi. **Enter** tugmachasini bosish orqali bitta satrni bir nechta satrga ajratish mumkin.

Agar kursor satrning oxirida turgan bo'lsa, **Del** – tugmachasini yoki agar kursor ikkinchi satrning boshida turgan bo'lsa, **BkSp** tugmasini bosish orqali ikkita satr bitta satrga birlashtiriladi.

Turbo Pascal muhitida matnlar bilan ishlashda bajarilgan amallarni bekor qilish, ya'ni matnni boshlang'ich variantiga qaytish uchun **Edit** bo'limidan **Undo ([alt+ BkSp])** tiklash uchun **Redo** buyrug'i bajariladi.

Matn bo'laklarini ajratib olish uchun quyidagi amallardan birortasini bajarish kerak:

1) *sichqoncha ko'rsatgichini birinchi belgiga o'rnatib, uning chap tugmachasini bosib turgan holda oxirgi belgiga harakatlantirish;*

2) *Shift tugmachasini bosib turgan holda matn kursorini siljitish.*

Ajratib olingan matn bo'laklarini ko'chirish, nusxalash va o'chirish uchun **Edit** menyusini buyruqlaridan foydalaniladi.

Turbo Pascal muhiti matn muharririda ishlashda asosan quyidagi buyruqlardan foydalaniladi:

1) **Kursorni joylashtirish:** **PgUp**-kursorni bir varaq yuqoriga; **PgDn**-kursorni bir varaq pastga; **Home**-satr boshiga; **End**-satr oxiriga; **Ctrl+PgUp**-matn boshiga; **Ctrl+PgDn**-matn oxiriga;

2) **Muharrirlash buyruqlari:** **Backspace** - kursordan chap tomonda turgan belgini o'chirish; **Del** - kursordan o'ng tomonda turgan belgini o'chirish; **Ctrl+Y** - kursor joylashgan satrni o'chirish; **Enter** - mavjud satrni bo'lish, ya'ni satr o'rnatish; **Ctrl + Q** - joriy satrni tiklash;

3) **Bloklar bilan ishlash:** **Ctrl+B**-sichqoncha yordamida bo'lakni ajratish blokini boshiga o'rnatish; **Ctrl+K**-blok oxiriga o'rnatish; **Ctrl+H**-ajratish blokini olib tashlash va tiklash; **Ctrl+Y**-blokni o'chirish; **Ctrl+Del**-blokni nusxalash; **Ctrl+Ins**-blokni o'rnatish; **Ctrl+R**-diskli fayldan blokni o'qish; **Ctrl+P**- blokni chop etish;

Paskal tili alfaviti va asosiy elementlari

Paskal tilining alfavitini qo'llanish maqsadlariga qarab bir necha guruhga bo'lish mumkin. Asosiy belgilar:

1) *lotin va rus alfaviti harflari;*

2) *raqamlar: 0,1,2,3,4,5,6,7,8,9;*

3) *arifmetik amal belgilari: +, -, *, /, div, mod;*

4) *mantiqiy amal belgilari : and, or, not;*

5) *munosabat belgilari: <, >, <=, >=, =, <>;*

6) *ta'minlash belgisi : := ;*

7) *ajratuvchilar: ., ;, :, ' ;*

8) qavslar : (,), {, }, [,];

9) boshqa belgilar: !, @, #, %, ?, & ;

Xizmatchi soʻzlar: *and, array, begin, case, const, div, do, downto, else, end, file, for, function, goto, if, in, label, mod, nil, not, of, or, packed, procedure, programm, record, repeat, set, string, then, to, type, until, var, while, with.*

Standart identifikatorlar:

1) oʻzgarmaslar: **false, true, maxint;**

2) turlar: **Boolean, integer, char, real, string, text;**

3) fayllar: **output, input;**

4) funksiyalar va proceduralar.

Soʻz. Soʻz dasturlash tilining alohida belgilaridan tashkil topgan boʻlib, maʼlum maʼnoni anglatadi. Dastur matnida soʻzlar bir-biridan boʻshliq yoki maxsus belgilar bilan ajratiladi. Foydalanilishiga qarab soʻzlar xizmatchi soʻzlar va nomga ajratiladi.

Nom yoki identifikatorlar oʻzgaruvchilar, oʻzgarmaslar, turlar, proceduralar va funksiyalarning nomlarini belgilash uchun ishlatiladi. Paskal tilida standart va ixtiyoriy beriladigan nomlar mavjud. Nom harf va raqamdan iborat boʻlib, doimo harfdan boshlanishi kerak. **Masalan: YY, GAMMA, Hoch, XX22.**

Paskal tilida nom sifatida xizmatchi soʻzlar va standart nomlardan, yaʼni standart oʻzgarmaslar, funksiyalar, fayllar, proceduralar va turlarning nomlarini qoʻllash mumkin emas. **Standart nomlar quyidagilardan iborat:** *abs, argtan, boolean, char, chr, cos, dispose, eos, eof, eoln, exp, false, forwar, get, input, integer, ln, maxint, new, odd, ord, output, pack, pagen, pred, put, read, readln, real, reset, rewrite, rounds, sin, sqr, sqrt, succ, text, true, string, trunc, unpack, write, writeln.*

Son. Paskal tilida sonlar oʻnlikda tasvirlanadi. Sonlar butun va haqiqiy boʻlishi mumkin. Bunda (+) musbat belgisi tashlab yozilishi ham mumkin. Haqiqiy son qoʻzgʻaluvchan vergulli va qoʻzgʻalmas vergulli formalarda tasvirlanishi mumkin. **Misol: butun sonlar: 5, 05, +73,-72678; haqiqiy sonlar: 0.65, -5E-02, 150.0, 1.7E2, -5.0E2**

Bundan tashqari Paskal tilida belgi va satrlardan ham foydalaniladi. Satrlar apostrof ichiga olingan belgilar ketma-ketligidan yoki bitta belgidan iborat boʻladi: **'Hikmat' 'fakultet', 'summa ', 'q', '>'**.

Arifmetik amallar. Paskal tilida ham sonli maʼlumotlar ustida bir nechta amallarni bajarish mumkin. Quyida Paskal tilida mavjud amallarni keltiramiz.

<i>Paskal tilida arifmetik amallar</i>			
<i>amallar</i>	<i>belgilash</i>	<i>amallar</i>	<i>belgilash</i>
<i>ko'paytirish</i>	$a*b$	<i>ayirish</i>	$a-b$
<i>bo'lish</i>	a/b	<i>butunga bo'lish</i>	$a \text{ div } b$
<i>qo'shish</i>	$a+b$	<i>bo'linma qoldig'i</i>	$a \text{ mod } b$

Bajarilayotgan amallarda operandlardan hech bo'lmaganda bittasi haqiqiy turga mansub bo'lsa, u holda hosil bo'ladigan natija ham haqiqiy turga tegishli bo'ladi. Bo'linma natijasi har doim haqiqiy son bo'ladi.

Paskal tilida darajaga ko'tarish amali yo'q. Shuning uchun sonlarni butun darajaga ko'tarish ularni ko'p marta bir biriga ko'paytirish yo'li bilan amalga oshiriladi. Haqiqiy darajaga ko'tarish logarifmlash orqali bajariladi: $x^n = e^{n \ln x}$ yoki $x^n = 10^{n \lg x}$.

Paskal tilida butun sonli berilmalar uchun **div va mod** amallari mavjud bo'lib, butunga bo'lish va bo'linma qoldig'i amallarini anglatadi.

Masalan: $11 \text{ div } 3 = 3$; $11 \text{ mod } 3 = 2$.

Munosabat va mantiqiy amallar. Paskal tilida ikkita mantiqiy konstanta true(chin) va false(yolg'on) ishlatiladi. Bular miqdorlarni taqqoslashda va ba'zi shartlarni tekshirishda qo'llaniladi. Ma'lumotlarni taqqoslash uchun quyidagi munosabat amallaridan foydalaniladi: $<$ (*kichik*), $<=$ (*kichik yoki teng*), $=$ (*teng*), $<>$ (*teng emas*); $>$ (*katta*), $>=$ (*katta yoki teng*).

Munosabat amallari sonli ma'lumotlar uchun qo'llanilganida mantiqiy qiymatlar hosil bo'ladi. **Masalan:** $96 > 73$ (*chin*), $-8 > -3$ (*yolg'on*), $4=7$ (*yolg'on*), $16 <> 22$ (*chin*).

Mantiqiy ma'lumotlar uchun quyidagi mantiqiy amallardan foydalanish mumkin: **or**- qo'shish (**va**); **and**- ko'paytirish (**yoki**); **not** - inkor (**yo'q**). **Or** va **and** mantiqiy amallar ikkita kattalik bilan, **not** amali esa bitta kattalik bilan bajariladi.

Mantiqiy qiymatlar ustida amallar natijasi quyidagicha:

<i>a</i>	<i>b</i>	<i>not a</i>	<i>a or b</i>	<i>a and b</i>
<i>true</i>	<i>true</i>	<i>false</i>	<i>true</i>	<i>true</i>
<i>true</i>	<i>false</i>	<i>false</i>	<i>true</i>	<i>false</i>
<i>false</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>false</i>
<i>false</i>	<i>false</i>	<i>true</i>	<i>false</i>	<i>false</i>

Misol. Agar $a = 25$ va $b = 20$ bo'lsa:

a) $(a > 10) \text{ or } (b < 12)$ (*true*) b) $(a > 10) \text{ and } (b < 3)$ (*false*)

c) $(a > 10) \text{ or } (b > 12)$ (true) d) $(a > 10) \text{ and } (b > 12)$ (true)

Standart funksiyalar. Paskal tilida quyidagi standart funksiyalar mavjud.

<i>Paskal tilida standart funksiyalar</i>			
Funksiya	Belgilash	Funksiya	Belgilash
absolyut qiymat	<i>abs(x)</i>	kvadratga ko'tarish	<i>sqr(x)</i>
argtangens	<i>arctan(x)</i>	yaxlitlash	<i>round(x)</i>
kosinus	<i>cos(x)</i>	argument butun qismi	<i>trunc(x)</i>
sinus	<i>sin(x)</i>	juftlikni tekshirish	<i>odd(x)</i>
natural logarifm	<i>ln(x)</i>	oldingi qiymat	<i>pred(x)</i>
kvadrat ildiz	<i>sqrt(x)</i>	navbatdagi qiymat	<i>succ(x)</i>
eksponenta	<i>exp(x)</i>		

Odd(x) funksiyasining argumenti toq bo'lsa **true**, aks holda **false** natijani beradi. ***Pred(x)*** va ***succ(i)*** funksiyalar oldingi (***i-1***) va navbatdagi (***i+1***) qiymatlarni beradi. **Masalan.** $pred(7)=6$; $succ(23)=24$

Ifoda. Paskal tilida ifodalar o'zgarma-lardan, o'zgaruvchilardan, funksiyalardan, amallar ishorasidan va qavslardan tuzilgan bo'ladi. Ifodalarda qatnashayotgan o'zgarma-lar, o'zgaruvchilar va funksiyalar dasturda e'lon qilingan bo'lishi yoki standart nomga ega bo'lishi kerak. Sonli qiymatlarni hisoblash uchun arifmetik ifoda qo'llaniladi. Arifmetik ifodada bo'lish va ko'paytirish amallari qo'shish va ayirish amallaridan oldin bajariladi. **Masalan:** $3 + 2 * 5 - 3$ ifodaning bajarilishidan 10 hosil bo'ladi.

Bir xil o'rinli amallar ishtirok etgan ifodalarda ular chapdan o'ngga qarab bajariladi. **Masalan.** $4 * 6 / 3$ ifoda natijasi 8 ga teng.

Qavs ichiga olingan ifodalar birinchi navbatda bajariladi.

Masalan. $(7-2) * (5+4)$ ifoda natijasi 45 ga teng

Arifmetik ifodada haqiqiy va butun turdagi ma'lumotlar qatnashishi mumkin, natija ham mos ravishda haqiqiy yoki butun bo'ladi.

Oddiy yozuvda	Paskal tilida
$\frac{a+12,686}{c_2-1,87 \cdot 10^3} + \gamma$	$(a+12.686)/(s2-1.87E3)+gamma$
$\frac{ 2\sin 4x + x }{3x}$	$exp(abs(2*\sin(4*x)+x)/(3*x)))$
$\frac{2^2 + \ln 1,13}{tg 2x}$	$(2*2*2+\ln(1.13))/(sin(2*x)/cos(2*x))$
$\frac{a^2 + 2^{1.17}}{\sqrt{x+y}}$	$(sqr(a)+exp(1.17*\ln(2)))/sqrt(x+y)$

Mantiqiy ifoda. Mantiqiy ifodalar mantiqiy qiymatlar, mantiqiy va munosabat amallari, arifmetik hamda belgili qiymatlar qatnashib, natijasi **true va false** qiymat bo'lishi mumkin. Unda amallar quyidagi tartibda bajariladi:

1. not (yo'q);
2. * (ko'paytirish), / (bo'lish), div, mod, and (va)
3. + (qo'shish), - (ayirish), or (yoki);
4. < (kichik), <= (kichik yoki teng), = (teng), <> (teng emas), >= (katta yoki teng), > (katta);

Bitta satrda berilgan amallar bir xil bajarilish o'rniga ega. Mantiqiy ifodalarda faqat oddiy qavslar yoziladi.

Masalan.

- 1) $(A > 3)$ and $(B = A + 6)$ or not $(c = 4)$
- 2) $(A <= x)$ and $(x <= B)$ or $(c <= x)$ and $(x <= d)$

Paskal tilining asosiy operatorlari

1. Ta'minlash (o'zlashtirish) operatori

Bu operator o'zgaruvchilarga tegishli qiymatlarni dasturning o'zida berish uchun ishlatiladi. Operatorning unumiy ko'rinishi quyidagicha:

o'zgaruvchi: = ifoda.

Bu yerda «:=» belgisi ta'minlash amalini bajaradi.

Masalan. $X:=2$; $z:=x*x-2$; $t:=SQR(x+y)$; $z:=true$; $t:=(z>z1)$ and $(x<0)$;

2. Ma'lumotlarni kiritish operatori

Bu operator dasturda boshlang'ich qiymatlarni berish uchun ishlatiladi: **read(b1,b2,...,bn); readln(b1,b2,...,bn).** Bu yerda **b1,b2,...,bn** qiymatlari kiritilishi talab etilgan o'zgaruvchilarning nomi.

Masalan. $Read(a,b,c); readln(a,b,c).$ $readln(x1, x2, x3);$

3. Ma'lumotlarni chiqarish operatori

Dasturda hosil bo'lgan natijalarni chiqarish uchun chiqarish operatori ishlatiladi: `write (a1, a2, ..., an); write; writeln (a1, a2, ..., an);`. Bu yerda a_1, a_2, \dots, a_n qiymati chiqarilishi kerak bo'lgan o'zgaruvchilarning nomi.

Masalan. `write(x,y); writeln(x1,x2); write('matn=', x); writeln(x,'"n,y); writeln(x:5:3,'"3,y:5:2);`.

4. O'tish operatori

Bu operator kerak bo'ganda dasturning tabiiy bajarilishini buzib, boshqaruvni dasturning boshqa joyiga uzatish uchun xizmat qiladi:

goto<nishon>; Bu yerda **goto** (... ga o'tish) operator nomi, nishon – 4 ta raqamdan oshmagan natural son. **Masalan:** `goto 25; 647;`

5. Shartli o'tish operator

Paskal tilida shartli o'tish operatorning to'liq va qisqartirilgan ko'rinishlari mavjud.

a) if B then S1 else S2; Shartli operator quyidagicha bajariladi:

if operatoridan keyin turgan **B** mantiqiy ifoda chin qiymatni qabul qilsa, **S1** operator, aks holda, **S2** operator bajariladi.

Masalan:

1) `if X<=0 then i:=i+1 else i:=i-1;`

2) `if (x<=0) and (a<=0) then S:=s+1 else s:=s*i;`

b) if B then S; bu yerda mantiqiy ifoda **B** chin qiymatga ega bo'lsa, **S** operator bajariladi, aks holda hech qanday amal bajarilmasdan **S** operatoridan keyingi operatorga o'tiladi.

Masalan: `if x<y then begin x:=y; y:=x; end;`

6. Sikl operatorlari

1) **Parametrli sikl operatori.** Uning umumiy ko'rinishi quyidagicha: **for V:=X₀ to X_n do S;** bu yerda **for** (uchun) **to** (gacha), **do** (bajar)-parametri, X_0 va X_n sikl parametrining boshlang'ich va oxirgi qiymatlari, **S**-sikl tanasi deb ataluvchi operatorlar ketma-ketligi. V-sikl parametri.

Masalan. 1) `for i:=1 to 10 do X:=sqr (i+1); S:=S+X;`

2) `for n:=1 to a+b do S:=S+sqr (n*n+2);`

Parametrli sikl operatorini parametrning qiymati kamayib boruvchi shaklida ham ifodalash mumkin: **for x:=X_n downto X₀ do S;**

Masalan: 1). `for i:=10 downto 1 do X:=Sqr (i+1); S:=S+X;`

2). `for n:=a+b downto 0 do S:=S+Sqr (n*n+1);`

2). **Sharti oldin qo'yilgan sikl operatori.** Bu operatorning umumiy ko'rinishi quyidagicha: **while B do S;** bu yerda while (toki), do (bajarmoq) xizmatchi so'zlar, B-mantiqiy ifoda, S-operator. S operator

bajarilmasligi yoki bir martadan ortiq bajarilishi mumkin, chunki S operatorning har bir bajarilishidan oldin B ifodaning qiymati tekshiriladi. Agar ifoda chin (true) qiymat qabul qilsa, S operator bajariladi, aks holda, ya'ni B yolg'on (false) qiymat qabul qilsa, sikl operatorining bajarilishi tugaydi.

Masalan:

1) $y:=1; k:=0; m:=5; \text{while } y \leq m \text{ do begin } y:=y+3; k:=k+1; \text{end};$

2) $x:=3; y:=1; v:=1; \text{eps}:=0,0001;$

$\text{while } \text{abs}(v) \geq \text{eps} \text{ do begin } V:=(x/y-y); y:=y+v; \text{end};$

3) **Sharti keyin qo'yilgan sikl operatori.** Operatorning umumiy ko'rinishi: **repeat s_1, s_2, \dots, s_n until B**, bu yerda repeat (takrorlanmoq), until (gacha) xizmatchi so'zlar s_1, s_2, \dots, s_n - operatorlar ketma-ketligi, B- mantiqiy ifoda.

Operatorning bajarilishida repeat va until operatorlari orasida joylashgan operatorlar ketma-ketligi bir marta va undan ortiq bajarilishi mumkin. Bu jarayon qo'yilgan mantiqiy ifoda B chin (true) qabul qilguncha davom etadi.

Masalan: $i:=1; s:=0; n:=10; \text{repeat } s:=s+i; i:=i+2 \text{ until } i>n;$

Massiv elementlari bilan ishlash

1) **Bir o'lchovli massivlar.** Bir o'lchovli massivlar quyidagicha aniqlanadi: array [indekslarning quyi va yuqori chegaralari] of <elementlar turi>; Indekslar turi real va integer dan tashqari boshqa oddiy turlar ham bo'lishi mumkin.

Masalan:

array [1..100] of real integer;

array [-109..-1] of integer;

Massivlarning turlar yoki to'g'ridan - to'g'ri o'zgaruvchilar bo'limida tavsiflash mumkin.

Masalan:

a) type

Massiv = 1..100;

Vektor = array [massiv] of real;

Vektor2 = array [-109..-1] of integer;

Var A, b, vektor; C, d: vektor2;

b) var a, b: array [1..100] of real;

c, d: array [-109..-1] of integer;

Massiv elementlarini kiritish va chiqarish, read va write operatorlari orqali amalga oshirilishi ham mumkin.

Masalan:

a) massiv elementlarini kiritish:

for i: = 1 to 100 do read (A [i]);

b) massiv elementlarini chiqarish:

for j: = 1 to 100 do write(B [j]);

2) **Ikki o'lchovli massivlar.** Ikki o'lchovli massivlarning element-lari ikkita indeks orqali ifodalanadi. Bu indekslarning biri shu elementni satr bo'yicha, ikkinchi indeks esa ustun bo'yicha o'rnini ko'rsatadi. Ikki o'lchovli massivlar quyidagicha tavsiflanadi:

array [< indeks turi>, < indeks turi>] of < elementlar turi >;

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

array [1..5,1..3] of integer;

Ikki o'lchovli massivlarni tavsiflash turlar yoki to'g'ridan-to'g'ri o'zgaruvchilar bo'limida amalga oshirish mumkin.

Masalan:

a) type

matritsa = array [1..4,1..5] of real;

matritsa 2 = array [dush.. yak] of - 20..20;

var

a, b: matritsa ; c, d: matritsa2;

b) var

a, b: array [1..4,1..5] of real;

c, d: array [dush..yak] of -20..20;

Paskal tilida qism dastur

Dasturlash jarayonida shunday holatlar mavjud bo'ladiki, bunda hisoblash jarayonining ayrim bo'laklarini parametrlarining turli qiymalari uchun takror-takror bajarishda to'g'ri keladi. Bunday hollarda har bir bo'lak uchun alohida dastur, ya'ni qism dastur tuzib kerak bo'lganda, unga murojat qilish ishni ancha engillashtiradi.

Paskal tilida qism dasturining ikki xil ko'rinishi qo'llaniladi: funsiya (function) va prosedura (procedure). Har qanday dasturda bir necha prosedura va funksiyalar bo'lishi mumkin. Prosedura va funsiya tavsiflash bo'limida o'zgaruvchilaridan keyin beriladi.

1) Funksiya (function) yordamida dastur tuzish

Funksiya (function) quyidagicha ko'rinishda ifodalanadi:

Function (< formal parametrlar va turlari >): funsiya turi >;

<tavsiflash bo'limi> begin

<buyruqlar bo'limi> - end;

Masalan: *a) function MIN (n: integer; x, a: real): real;*

b) function STA (n, i: integer; b, c: real): real;

Funksiyaga murojaat qilish standart funksiyalarga o'xshash amalga oshiriladi. Qayerda ifodani hisoblash kerak bo'lsa, o'sha joyda funksiya nomi yoziladi, undan keyin faktik parametrlar beriladi. Agar parametrsiz funksiya chiqarilsa, u holda faqatgina funksiyaning nomi ko'rsatiladi.

Masalan: a) *ALB(4,5,6)*; b) *STA(2,3,3,2)*; d) *vit*; f) *Ster*;

2) **Procedura (procedure)** yordamida dastur tuzish.

Prosedura quyidagi ko'rinishda beriladi:

procedure nom (<formal parametrlar>);

<tavsiflash bo'limi>; begin

<buyruqlar operatorlar bo'limi>; end;

Masalan: a) *procedure sutta (A,B,C: integer; var s: real);*

b) procedure FAK (n: integer; var F: real);

Proseduralarni bajarish uchun dasturning kerakli joyida murojaat qilish kerak bo'ladi. Proseduraga murojaat qilinganda formal parametrlar faktik parametrlar bilan almashadi. Bu quyidagicha bo'ladi:

<prosedura nomi> (<argumentlar ro'yxati>):

Masalan: a) *Summa (5,m,7,S1)*; b) *FAK(20,F1)*;

Agar prosedura parametrsiz bo'lsa, u holda ularga murojaat bo'lganda faqat prosedura nomi ko'rsatiladi.

Masalan: a) *Summa*; b) *FAK*; c) *Step*;

Paskal tilida satrli ma'lumotlar bilan ishlash

Belgilar ketma-ketligidan iborat bo'lgan satrlarni tavsiflash uchun **String** so'zi ishlatiladi. Satrlar **0 dan 255** tagacha belgidan iborat bo'lishi mumkin. Agar satr o'zgaruvchilari qabul qiladigan satr uzunliklarini ko'rsatish lozim bo'lsa, **String** so'zidan keyin kvadrat qavslarda belgilar soni ko'rsatiladi: **String[50]**; **String[100]**; va hokazo. Satrlar bilan ishlash.

Ana shu funksiya va proseduralardan ba'zilarini keltiramiz.

1. **Concat funksiyasi.** Bu funksiya satrlar ketma-ketligini birlashtirish (qo'shish) vazifasini bajaradi. Uning umumiy ko'rinishi quyidagicha:

concat (S1,S2,.....,Sn);

Bu funksiyaning bajarilishi natijasida **S1** satr qolgan satrlar bilan birlashtiriladi va ular funksiyaning parametrlari hisoblanadi.

Misol.

var a,b,c: string;

begin a:='infor'; b:='matika';

c:=concat(a,b); writeln(s); end.

Dastur lavhasining bajarilish natijasida a va b satrlar ulanib, **informatika** so'zi hosil bo'ladi.

2. Copy funksiyasi. Bu funksiya satrlardan satr bo'laklarini ajratib olish uchun ishlatiladi. Uning umumiy ko'rinishi quyidagicha:

copy(S:string;n,m:integer):string;

Bu yerda **s** - berilgan satr; **n** - esa satr bo'lagi ajratib olinadigan belgi nomeri; **m** - esa ajratib olinadigan belgilar soni.

Masalan. var a,b,c : string;

begin a:='informatika'; b:=copy(a,3,5); c:=copy(a,8,3);
writeln(b,c); end.

Bu dasturning bajarilishi natijasida '**forma**' va '**tik**' so'zlari hosil bo'ladi.

3. Delete prosedurasi. Bu prosedura satrlardan belgilarni yoki so'z bo'laklarini o'chirish uchun ishlatiladi. Uning umumiy ko'rinishi quyidagicha:

delete(var S:string; n:integer; m:integer);

Bu yerda **S** berilgan satr; **n** - **S** satrda o'chirish boshlanadigan belgining nomeri; **m** - o'chiriladigan belgilar soni;

Masalan. var a,b : string;

begin a:='informatika'; delete(a,6,5); writeln(a);
delete(a,3,3); writeln(a); end.

Bu dasturning bajarilishi natijasida '**infora va**' '**nimatika**' so'zlari hosil bo'ladi.

4. Insert prosedurasi. Bu prosedura berilgan satrga biror satr bo'lagini qo'shish uchun ishlatiladi. Uning umumiy ko'rinishi quyidagicha bo'ladi: **insert(S1:string; var S:string;n:integer);**

Bu yerda **S** - berilgan satr; **S1** - berilgan satrga qo'shiladigan satr bo'lagi; **n** - **S** satrga qo'shish amalga oshiriladigan belgining nomeri.

Masalan. var a,b:string;

begin a:='infora'; b:='matik';
insert(b,a,6); writeln(a); end.

Dastur natijasi '**informatika**' kabi bo'ladi.

5. Length funksiyasi. Bu funksiya berilgan satrlarning yoki satr bo'laklarining uzunliklarini aniqlash uchun ishlatiladi. Uning umumiy ko'rinishi quyidagicha yoziladi:

length(S:string):integer;

Bu yerda **S** - berilgan satr. Funksiyaning bajarilish natijasi sondan iborat bo'ladi, ya'ni funksiyaning qiymati satrdagi belgilar soniga teng bo'ladi.

Masalan. var S:string; T:integer;

begin S:='INFORMATIKA';
T:= Length(S); writeln('T=',T);
end.

Bu dasturning bajarilishi natijasida T=11 hosil bo'ladi, ya'ni berilgan satr 11 ta belgidan iborat ekanligi aniqlanadi.

Turbo Paskal modullari

Modullar. Turbo Paskalda modullar yaratib va ularni kompilyasiya qilib masalalarni yechish mumkin. O'z vaqtida Turbo Pascal yaratuvchilari IBM kompyuterlarining barcha imkoniyatlarini o'z ichiga olgan funksiya va proseduralarni modullar ko'rinishida tavsiflaganlar va yaratganlar.

Bu modullarga quyidagilar kiradi:

System – standart funksiya va proseduralardan iborat. **System** moduli barcha dasturlarda avtomatik qo'llaniladi .

Dos – **MS DOS** operatsion sestemasi vositalaridan foydalanish imkonini beruvchi funksiya va proseduralardan iborat.

Crt - IBM dinamikalari, klaviatura va ekran bilan ishlash uchun mo'ljallangan proseduralar majmuasidan iborat.

Graph - kompyuter grafik imkoniyatlaridan foydalanishni amalga oshirish uchun ishlatiladigan proseduralar to'plami.

Printer - chop etish qurilmasi bilan ishlash uchun kichik modul.

Graph3 – Turbo Paskalning 3-versiyasi uchun to'liq grafik dasturlardan iborat.

Modullardan foydalanish uchun dastur boshida quyidagi satrlar yoziladi:

Program- dastur nomi;

Uses modul nomi;

Agar dasturda bir nechta modullardan foydalanilsa, quyidagicha yoziladi:

Uses 1- modul nomi, 2- modul nomi, ..., N- modul nomi;

1) CRT moduli

CRT moduli quyidagi funksiya va proseduralardan iborat:

1. **AssignCrt** - prosedurasi- **CRT** qurilmasi uchun matn faylini belgilaydi.

2. **ClrEol** prosedurasi- ekranda kursor turgan joydan satr oxirigacha belgilarni o'chiradi.

3. **ClrScr** prosedurasi- ekranni tozalaydi va kursorni yuqori chap burchagiga joylashtiradi.

4. **Delay**- prosedurasi – berilgan songa teng millisekund ish bajarilishini to'xtatib turadi.

5. **Delline** prosedurasi- kursor turgan satrni olib tashlash;

6. **GOTOXY**- kursorni berilgan satr va ustunga joylashtirish;

7. **HighVideo** –chiqarilayotgan belgilarni yuqori yorug'likda hosil qiladi;

8. **Inline** prosedurasi- kursor turgan o'rinda bo'sh satrni hosil qiladi;

9. **KeyPressed** funksiyasi- biror klavisha bosilsa **true** qiymat hosil qiladi, aks holda **false** hosil bo'ladi.

10. **LowVideo** – belgilar quyi yorug'likda hosil qilinadi;

11. **NormVideo** – belgilar normal yorug'likda hosil qilinadi;

12. **Nosound**-ovoz hosil qilish dinamikasini o'chiradi;

13. **RedKey** – klaviaturadan belgini o'qiydi;

14. **Sound** – ovoz hosil qilish dinamikasini ishga tushiradi va berilgan chastota bo'yicha ovoz hosil qiladi.

15. **textbackground** – fon rangini o'rnatadi ;

16. **textcolor** – matn rangini o'rnatadi;

17. **textmode** – matn tartibini o'rnatadi;

18. **WhereX** –x koordinatani hosil qiladi;

19. **WhereY** –y koordinatani hosil qiladi;

20. **Window** – ekranda matn oynasini hosil qiladi: **Window (x1, y1, x2, y2)**; Bu yerda **(x1, y1)**- oynaning chap yuqori burchagi koordinatlari **(x2;y2)**- oynaning o'ng quyi burchagi koordinatalari.

2) Graph moduli

Turbo Paskal muhitida standart ish tartiboti matnli hisoblanadi. Bu tartibotda ekranga matnlar bilan ish olib boriladi.

Grafik tartibotida displey ekrani bir-biriga juda yaqin joylashgan nuqtalar majmuasi aylanadi. Ekranda mavjud nuqta **piksel** yoki **piksel' (pixel)** deb yuritiladi. Piksellar to'g'ri burchakli jadvalni hosil qiladi. Nuqtalar shu jadval elementlari hisoblanadi va ular ekranning yuqori chap burchagidan boshlab nomerlanadi. Bu nuqtaning koordinatasi **(0,0)** dan iborat bo'ladi.

Matnli ma'lumotlarning belgilarini tasvirlash fikirlangan sondagi piksellardan iborat bo'lgan matritsadan foydalaniladi. Har bir belgining o'rni to'g'ri burchakli o'lchovdan, *masalan*, **8x8**, **8x14**, **8x16** va hokazolardan iborat bo'ladi. Belgi o'rniga dastur yordamida belgining tasviri joylashtiriladi.

Mavjud videoadapterlarning grafik holatlari monitoring imkoniyatlari bilan, ya'ni ekran piksellarining va hosil qilinishi mumkin bo'lgan ranglarning umumiy soni bilan aniqlanadi.

Zamonaviy monitorlar **800x600 ta**, **1024x768 ta** va undan ko'proq nuqtalar bilan ishlash imkoniyatini beradi. Monitoring imkoniyati qancha yuqori bo'lsa, unda hosil qilinayotgan tasvirlar shunchalik sifatli bo'ladi. Ko'plab videoadapterlar bir nechta grafik sahifalar bilan ishlashi

mumkin. Grafik sahifa deganda - tezkor xotiraning grafik tasvirlarni hosil qilish uchun ajratilgan sohasi tushuniladi.

Kompyuterda ishlash jarayonida mavjud grafik tartibotlardan birini tanlagan holda, nuqtalar soni va mumkin bo'lgan ranglar sonini o'zgartirish mumkin.

Drayver deganda - kompyuterning biror bir qurilmasi ishini boshqaruvchi dastur tushuniladi. *Grafik drayver* grafik tartibotida displey adapterini boshqaradi. Grafik drayverlar diskda ishchi katalogining **BGI** qism katalogida fayl ko'rinishida joylashgan bo'lib, kengaytmasi. **BGI (Borland Graphic Interfase)** dan iborat bo'ladi.

Turbo Pascal muhitida **CGA, EGA, VGA, MCGA, PC 3270** kabi videoadapterlarning grafik imkoniyatlarini ta'minlovchi standart kutubxona moduli-**GRAPH moduli** ishlatiladi.

Graph kutubxonasi **80** ga yaqin grafik prosedura va funksiyalar, shu bilan birga o'nlab o'zgarmlar va ma'lumotlar turining tavsiflaridan iborat. **Graph** modulining ishga tushirish **uses** xizmatchi so'zi orqali amalga oshiriladi.

Graph moduli-Graph.tpu faylidan iboratdir. Grafik tartibotida ishlash uchun bu fayl kompilyator uchun ishchi fayl bo'lishi kerak. Dastur boshida **Graph.tpu** fayli joylashgan katalogga yo'l ko'rsatilishi kerak.

a) Grafik holatini o'rnatish va undan chiqish tartiblari

Kompyuter ekranining oddiy ish tartiboti matnli hisoblanadi. Matn tartibotidan grafik tartibotiga o'tish uchun **Graph** modulining **InitGraph** prosedurasi ishlatiladi. Uning umumiy ko'rinishi quyidagicha:

InitGraph (D, M, F); - *ekranni grafik holatga o'tkazish*. Bu yerda **D**- drayver nomeri, **M**- tartibot nomeri, **F**- esa kerakli drayver mavjud bo'lgan faylga yo'l. Agar **F** o'zgaruvchi bo'sh (**F=""**) satrdan iborat bo'lsa, drayver joriy katalogdan izlanadi. **D** va **M** lar o'zgaruvchi parametrlardir. **InitGraph** proseduralari ishga tushirayotgan paytda **D** ning qiymati **0** ga teng bo'lsa kerakli drayver va shu drayver uchun optimal grafik tartibot avtomatik tarzda aniqlanadi. **Graph** modulida **0** ga teng bo'lgan **Detect** o'zgarmlari ishlatiladi.

Grafik holatidan chiqish yoki monitorni boshlang'ich ish holatiga o'tkazish uchun **CloseGraph** prosedurasi ishlatiladi.

CloseGraph- *grafik holatidan chiqish*.

b) Graph modulining funktsiya va proseduralari

Quyida **Graph** moduliga tegishli prosefura va funksiyalarni qarab chiqamiz.

Koordinatalarni o'rnatish prosedura va funksiyalari

Ko'plab grafik prosedura va funksiyalarda ekranda joriy o'rinni ko'rsatuvchi **ko'rsatkich**dan foydalaniladi. **Ko'rsatkich**ning matn kursoridan farqi shundaki, u ekranda ko'rinmaydi. Ana shu ko'rsatkichning holati, shu bilan birga umuman grafik holatda har qanday koordinata ekranning yuqori chap burchagi **(0,0)** koordinataga nisbatan beriladi. Shunday qilib, ekranning gorizontali koordinatasi chapdan o'ngga qarab, vertikal koordinatasi esa yuqoridan pastga qarab o'sib boradi.

SetMaxX va GetMaxY funksiyalari. Joriy ish tartibotida mos ravishda maksimal gorizontali va vertikal koordinatalarini aniqlaydi.

GetX va GetY funksiyalari. Integer turidagi qiymatlar bo'yicha ekranning gorizontali va vertikal koordinatalarini o'rnatadi. Agar oyna o'rnatilmagan bo'lsa, koordinatalar ekranning chap yuqori burchagiga nisbatan olinadi.

SetViewPort prosedurasi - grafik ekranda to'g'ri burchakli oyna hosil qiladi. Uning umumiy ko'rinishi:

Procedure SetViewPort (X1, Y1, X2, Y2: integer; ClipOn: boolean);

Bu yerda **X1...Y2** lar oyna burchagining koordinatalari. **(X1, Y1)**- yuqori chap va **(X2, Y2)** o'ng quyi burchak koordinatalari, **ClipOn** – **Boolean** tipidagi ifoda.

MoveTo prosedurasi – ko'rsatkichni yangi joriy holatga o'rnatadi. Uning umumiy ko'rinishi:

Procedure MoveTo (X,Y:integer); Bu yerda **X,Y** ko'rsatkichning gorizontali va vertikal yo'nalishlar bo'yicha yangi koordinatalari. Koordinata ekranning yuqori chap burchagiga asosan olinadi.

ClearDevice prosedurasi – grafik oynani tozalaydi.

Uning umumiy ko'rinishi:

Procedure ClearDevice; Bu prosedura bajarilganda ekran tozalanaadi, ko'rsatkich ekranning yuqori chap burchagiga joylashadi va ekran fon rangi bilan bo'yaladi.

Ko'pburchaklar hosil qilish funksiya va proseduralar

Rectangle prosedurasi – ko'rsatilgan burchak koordinatalari bo'yicha to'g'ri to'rtburchak hosil qilish. Uning umumiy ko'rinishi:

Procedure Rectangle (X1, Y1, X2, Y2: integer); bu yerda **X2 - X1... to'g'ri to'rtburchak burchaklarining koordinatalari. (X1, Y1)** – yuqori chap to'rtburchak va **(X2, Y1)** quyi o'ng burchak koordinatalari. To'g'ri burchak joriy rang va joriy chiziq qalinligi bo'yicha hosil qilinadi.

DrawPoly prosedurasi – egilish nuqtalarining koordinatalari bo'yicha ixtiyoriy sinq chiziqlarni chizadi. Uning umumiy ko'rinishi:

Procedure DrawPoly(N:Word;Var Points); Bu yerda **N** – egilish nuqtalarining soni; **Points** – nuqta koordinatalarini tashkil etuvchilar. Chizish jarayonida joriy rang va joriy chiziq qalinligi tanlanadi.

Yoy, aylana va ellipslar hosil qilish funksiya va prosedurasi

Circle prosedurasi – aylana chizadi. Uning umumiy ko‘rinishi quyidagicha:

Procedure Circle (X, Y: integer; R: word); bu yerda **X, Y** – aylana markazining koordinatalari, **R** – aylana radiusi. Aylana joriy rangda hosil qilinadi. Chiziqning qalinligi joriy holatda beriladi.

Arc prosedurasi – aylana yoyini chizadi. Uning umumiy ko‘rinishi quyidagicha:

Procedure Arc (X, Y: integer; BegA, EndA, B:Word); Bu yerda **X, Y** – markaz koordinatalari. **BegA, EndA** – yoyning boshlang‘ich va oxirgi burchaklari. **R**- radius. Burchaklar soat strelkasiga teskari holda o‘lchanadi va graduslarda ko‘rsatiladi. Nol burchak vektorning chapdan o‘ngga gorizontal yo‘nalishiga mos keladi. Agar boshlang‘ich burchak **0** va oxirgi burchak **359** bo‘lsa, u holda to‘liq aylana hosil bo‘ladi.

Ellipse prosedurasi-ellips yoyini chizadi. Uning umumiy ko‘rinishi:

Procedure Ellipse(X,Y:integer,BegA, EndA, RX, RY:Word); bu yerda **X, Y** – markaz koordinatalari; **BegA, EndA** – yoyning mos ravishda boshlang‘ich va oxirgi burchaklari; **RX, RY** – ellipsning gorizontal va vertikal yarim o‘qlari.

Rang berish va sohani bo‘yash funksiya va proseduralari.

SetColor prosedurasi- hosil qilinayotgan chiziq va belgilar uchun joriy rangni o‘rnatadi. Uning umumiy ko‘rinishi:

Procedure Setcolor (Color: word); Bu yerda **color**- joriy rang.

SetbkColor prosedurasi- fon rangini o‘rnatadi. Uning umumiy ko‘rinishi:

Procedure SetbkColor (Color: word); Bu yerda **Color**- fon rangi.

Setfillstyle prosedurasi- sohani to‘ldirish turi va rangni o‘rnatadi. Uning umumiy ko‘rinishi quyidagicha:

Procedure Setfillstyle (fill,color: word);

Bu yerda **fill** to‘ldirish turi; **color**-to‘ldirish rangi. To‘ldirish yordamida tasvirning qandaydir bo‘lagini doimiy takrorlanuvchi **naqishlari** bilan to‘ldirish mumkin.

FloodFill prosedurasi- ixtiyoriy yopiq figurani to‘ldirishning joriy usulidan foydalangan holda to‘ldiradi. Uning umumiy ko‘rinishi quyidagicha:

Procedure FlodFill (X,Y: integer, border: Word);

Bu yerda **X,Y**- yopiq figura ichidagi ixtiyoriy nuqta koordinatalari; **Border**- chegaraviy chiziq rangi. Agar figura yopiq bo'lmasa, u holda to'ldirish butun ekran bo'yicha yoyiladi.

Bar prosedurasi- ekranning to'g'ri burchakli sohasini bo'yaydi. Uning umumiy ko'rinishi:

Procedure Bar (x1, y2, x2, y2: integer); Bu yerda **x1, ..., y2** lar bo'yaladigan soha koordinatalari bo'lib, **(x1,y1)** yuqori chap burchak va **(x2,y2)** quyi o'ng burchak koordinatalari hisoblanadi. Prosedura sohani **SetfillStyle** prosedurasi orqali o'rnatilgan **naqsh** va rang bilan bo'yaydi.

Bar3d prosedurasi- parallelepipedning uch o'lchovli tasvirini chizadi va uning oldingi tomonini bo'yaydi. Uning umumiy ko'rinishi quyidagicha:

Procedure Bar3d (x1, y2, x2, y2, Depth: integer; top: boolean); Bu yerda **x1, ..., y2** oldingi yoq koordinatalari bo'lib, **(x1,y1)** yuqori chap va **(x2, y2)** quyi o'ng burchak koordinatalari hisoblanadi. **Depth**- uch o'lchovli tasvirning uchinchi o'lchovi («chuqurligi»); **Top**- yuqori yoqning tasvirlash usuli.

FillPoly prosedurasi-bo'yalgan ko'pburchak chizadi. Uning umumiy ko'rinishi: **procedure Fillpoli (N:word; vat Cords);** **N**- yopiq ko'pburchak uchlarining soni; **Cords-Pointtype** tilidagi o'zgaruvchi bo'lib, uchlarining koordinatalaridan iborat bo'ladi.

Chiziqning shakli va rangi **SetLineStyle** va **SetColor** proseduralari orqali, bo'yash rangi va turi esa **SetFillStyle** prosedurasi bilan beriladi.

FillEllipse prosedurasi- bo'yalgan ellipsni hosil qiladi. Uning umumiy ko'rinishi quyidagicha:

Procedure Fillellipse (X,Y,RX,RY:integer); Bu yerda **X,U**- ellips markazining koordinatalari; **RX, RY**- ellipsning **gorizontal** va **vertikal** radiuslari.

Sector prosedurasi- ellips sektorini chizadi va uni bo'yaydi. Uning umumiy ko'rinishi:

Procedure Sector (X,Y: integer; BegA, endA, RX, RY: word);

Bu yerda **begA, endA** - mos ravishda ellips sektorining boshlang'ich va oxirgi burchaklari. Boshqa parametrlar **FillEllipse** prosedurasi parametrlari kabi aniqlanadi.

PieSlice prosedurasi- aylana sektorini chizadi va bo'yaydi. Uning umumiy ko'rinishi:

procedure Pieslice (x,y: integer; BegA, endA, R: word);

Uning **Sector** prosedurasidan farqi, unda faqatgina bitta **R** radius ko'rsatiladi. Proseduradan aylanaviy diagrammalarni yasashda foydalanish ancha qulay hisoblanadi.

Algoritm va dastur tuzishga doir amaliy mashg'ulotlar.

Chiziqli algoritm va dastur tuzish

Chiziqli dastur deganda masalaning algoritmgiga mos keluvchi buyruqlar ketma-ketligi dasturda uchrash tartibiga mos ravishda bajariladigan dasturlar tushuniladi. Dasturning bitta satrida bir nechta buyruqlar kelishi mumkin. Bunda ular birbiridan «;» 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.

1-masala. Berilgan radius bo'yicha aylana uzunligi va u bilan chegaralangan doira yuzini toping.

<i>dasturi</i>
<pre>program yuza; var L,R,S:real; begin writeln('aylana radiusini kiriting'); readln(R); write('aylana uzunligi-');L:=2*Pi*R; writeln ('L=';L); write('doira yuzasi-'); S:= Pi* R* R; writeln(' S =', S); end.</pre>

Ushbu dasturni kompyuterda bajarish uchun, avval Turbo Paskal muhitini ishga 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.

2-masala. a,b,c tomonlari bilan berilgan uchburchakni yuzasini hisoblash dasturini tuzing.

Bu masalani yechish uchun Geron formulasidan foydalanamiz. Uning blok-sxema ko'rinishidagi dasturi quyidagicha bo'ladi:

<i>dasturi</i>
<pre>Program uchyuza; var a,b,c,p,S:real; begin writeln('Uchburchak tomonlarini kiritng;); readln(a,b,c); writeln('Uchburchak perimetri'); P:=(a+b+c)/2; writeln('p=',2*p); writeln('uchburchak yuzasi'); S:=sqrt(p*(p-a)*(p-b)*(p-c)); writeln('S=',S); end.</pre>

3-masala. Arifmetik progressiyaning birinchi hadi va ayirmasi berilgan bo'lsa uning n-hadi va n ta hadlari yig'indisini hisoblash dasturini tuzing.

Bu yerda arifmetik progressiyaning birinchi hadi a_1 , ayirmasi d va hadlar soni n berilgan bo'lsin. U holda uni yechish dasturi quyidagicha bo'ladi.

<i>dasturi</i>
<pre>Program arifprogressiya; var a1, d, Sn : real; n:integer; begin write('Arifmetik progressiyaning birinchi hadi a1='); readln(a1); write('ayirmasi d=');readln(d); write('hadlar soni n-');readln(n); write('Arif.progressiyning ', n, '- hadi='); an:=a1+(n-1)*d; writeln('an=',an); write('Arif.progressiyning ', n, ' ta hadi yig'indisi='); Sn:=(a1+an)/2*n; writeln('Sn=',Sn); end.</pre>

Tarmoqlanuvchi algoritm va dastur tuzish

1-masala. Berilgan x sonini $[2,5]$ oraliqqa tegishli yoki tegishli emasligini aniqlash dasturini tuzing.

<i>dasturi</i>
<pre>program masala; var x: integer; a: string; begin writeln('x ning qiymatini kiriting'); readln(x); a:= 'x soni 2 dan kichik'; if x>= 2 then if x<= 5 then a:= 'x soni [2, 5] oraliqqa tegishli' else a:= 'x soni 5 dan katta'; writeln(a) end.</pre>

Bu dastur uchun kompyuter quyidagi natijalarni beradi:

x ning qiymatini kiriting 4

x soni [2, 5] oraliqqa tegishli

x ning qiymatini kiriting 7

x soni 5 dan katta

x ning qiymatini kiriting 1

x soni 2 dan kichik

2-Masala. Sportchi musobaqada M – o‘rinni egalladi. U qanday medal olganini aniqlash dasturini tanlash operatori orqali tuzing.

Dasturi
<pre>Program sport; label 1,2,3,4; var m: integer; ms : string[16]; begin write('sportchi nechanchi urini oldi?'); readln(m); s:= 'sportchi medal olmagan' case m of 1: s:= 'oltin'; 2: s:= 'kumush'; 3: s:= 'bronza'; end; writeln(s) end.</pre>

Kompyuter bu dastur uchun quyidagi natijani beradi:

sportchi nechanchi urini oldi? 5

sportchi medal olmagan

sportchi nechanchi urini oldi?1

oltin

3-masala. O‘quvchilarni olgan baholarini ifodalash dasturini tuzing.

Dasturi
<pre>Program baho; label 0,1,2,3,4,5; var a: integer; b: string[9]; begin write('bahoni kiriting'); readln(a); case a of 5: b:= 'alo'; 4: b:= 'yaxshi' 3: b:= 'qoniqarli' ; 0,1,2: b:= 'yomon' ; end; writeln(b) end.</pre>

Kompyuter bu dastur uchun quyidagi natijalarni beradi:

Bahoni kiritng 4 yahshi

Bahoni kiritng 1 yomon

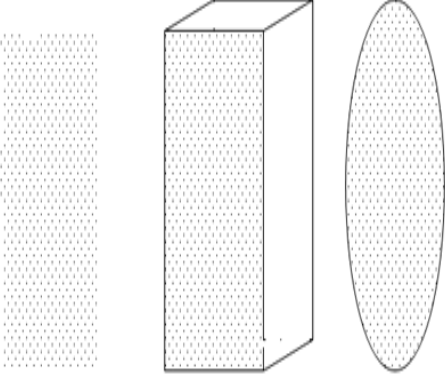
4-masala. $ax^2 + bx + c = 0$ ($a \neq 0, b \neq 0, c \neq 0$) kvadrat tenglamani yechish dasturini tuzing.

<i>Dasturi</i>
<pre>Program kvt; label 4,9; var a,b,c,D,x1,x2:real; begin write ('a, b, c - koeffisiyentlarni kiriting'); readln(a,b,c); d:=b*b-4*a*c; Writeln('diskriminant-', d); if d>=0 then begin x1:=(-b+sqrt(d))/(2*a); x2:=(-b-sqrt(d))/(2*a); goto 4; end else begin writeln('haqiqiy yechim yo'q'); goto 9; end; 4: writeln('x1=',x1,',':3,'x2=',x2); 9:end.</pre>

5- masala. Ifodaning qiymatini hisoblash algoritmi (blok-sxema) va dasturini tuzing.

$$y = \begin{cases} x^2 + 2x + 3, & \text{agar } \delta < 0 \\ 2x - 1, & \text{agar } \delta > \frac{\pi}{2} \\ \sin^2 x + 3, & \text{agar } 0 \leq x \leq \frac{\pi}{2} \end{cases}$$

<i>Dasturi</i>
<pre>Program ifoda; const pi=3.1415; var x,y: real; begin write(' x ga qiymat bering'); readln (x); if x<0 then y:=x*x+2*x+3; if x>(pi/2) then y:=2*x-1 else y:=SQR (sin(x))+3; writeln ('Y=',Y); end.</pre>

<p>4. Geometrik figuralarni hosil qilish dasturi</p>	<p>Dastur natijasi</p>
<pre> Program tasvir ; uses crt, graph ; var x, u, gd, gm : integer ; a, t : real ; begin gd:= detect; initgraph(gd,gm, ' '); setfillstyle(11,15) ; bar(20,100,100,300) ; bar3d(200,100,300,300,40,true) ; fillellipse(500,200,20,100); readln ; end. </pre>	
<p>5. Grafik holatida matn chop etish</p>	<p>Dastur natijasi</p>
<pre> Program shriftlar; uses crt, graph ; var x, u, gd, gm : integer ; begin gd:=detect; initgraph (gd,gm, " "); settextstyle(0,0,8) ; outtextxy(20,100, 'KOMPYUTER') ; settextstyle(3,0,8) ; outtextxy(20,300,'SINFI'); readln ;end. </pre>	<p style="text-align: center;">KOMPYUTER SINFI</p>
<p>6. Ekranning o'rtasida gorizontaal yo'nalishda xarakat qilayotgan nuqta tasvirini hosil qiling.</p>	<p>7. Aylana bo'ylab xarakat qilayotgan nuqta tasvirini hosil qilish.</p>
<pre> program nuqta; uses graph, crt; var gd,gm,x,k: integer; begin gd:= detect; initgraph (gd,gm, ' '); k:= 1; x:= 1; repeat if x = 1 then k:= 1 else if x= 620 then k:= 1; x:= x + k; putpixel(x,150,15); delay(20); putpixel(x, 150,0) ; until keypressed; end. </pre>	<pre> program aylana; uses graph, crt; var gd,gm,x,y,k: integer; fi: real; begin gd:= detect; initgraph (gd,gm, ' '); fi:= 0; k:= 50; repeat fi:= fi + 0.01; if fi = 6.28 then fi:= 0; x:= trunc(k*cos(fi)); y:= trunc(k*sin(fi)); putpixel(x + 320,240-y,15); delay(20); putpixel(x+ 320,240-y,1); until keypressed; </pre>

	<i>end.</i>
<i>8. O'z o'qi atrofida aylanayotgan kesma tasvirini yasash.</i>	<i>9. O'z o'qi atrofida aylanayotgan uchburchak tasvirini hosil qilish.</i>
<pre> Program kesma; uses graph, crt; var gd,gm,x1 ,y1 ,x,y,k: integer; fi: real; begin gd:= detect; initgraph (gd,gm, ""); fi:= 0; k:= 50; repeat fi:= fi+ 0.01; if fi = 6.28 then fi:= 0; x:= trunc(k*cos(fi)); y:= trunc(k*sin(fi)); x1:= trunc(k*cos(fi + 3.14)); y1:= trunc(k*sin(fi + 3.14)); setcolor(15); line(x+320, 240-y, x1+320, 240- y1); delay(20); setcolor(0); line(x+320, 240-y, x1+320, 240- y1); until keypressed; end. </pre>	<pre> program uchhurchak; uses graph, crt; var gd,gm,x,y,x1,y1,x2,y2,k; integer; fi: real; begin gd:= detect; initgraph (gd,gm,""); fi:= 0; k:= 50; repeat fi:= fi + 0.01; if fi= 6.28 then fi:= 0; x:=trunc(k*cos(fi)); y:=trunc(k*sin(fi)); setcolor(15); Line(x+300, y+200, x + 400, y + 100); Line(x +300, y +200, x +200, y +180); Line(x+400, y +100, x +200, y + 180); delay(20); setcolor(0); Line(x +300, y +200, x +400, y + 100); Line(x +300, y +200, x +200, y +180); Line(x+400, y +100, x +200, y + 180); until keypressed; end. </pre>

Foydalanish uchun adabiyotlar

1. A.Abduqodirov va boshqalar. Informatika va hisoblash texnikasi asoslari. 9- sinf uchun darslik- Toshkent, 2006.
2. Зуев Е.А. Программирование на языке Turbo Pascal 6.0, 7.0. - М., Радио и связь, 1993. -380с.
3. Фаронов В.В. Основы Турбо-Паскаля. - М., 1992.
4. Абрамов С.А., Гнездилова Г.Г., Капустина Е.Н., Селюн М.И. Задачи по программированию. - М., "Наука", 1988.
5. Попов В.Б. Turbo Pascal для школьников. Версия 7.0: Учебное пособие. - М., "Финансы и статистика", 1996.
6. Лукин С.Н. TURBO PASCAL 7.0. Самоучитель для начинающих. - М., "Диалог-МИФИ", 1999. -400 с.
7. Вольвачев А.К., Криевич В. С. Программирование на языке Паскаль для персональных ЕС ЭВМ- Минск, Высшая школа, 1989 г.
8. Васюкова Н.Д., Тюляева В.В. Практикум по основам программирования. Язык Паскаль – М.: Высшая школа, 1991 г.
9. Зуев Е.А. Язык программирования Турбо- Паскал 6.0, 7.0- М; Радио и связь, 1993.
10. Файсман А. Профессиональное программирование на языке Паскаль- М; Наука, 1989 г.
11. Aminov I. Paskal dasturlash tili. O'quv qullanma- SamDU, Samarqand, 1996 y.

MUNDARIJA

KIRISH	3
Pascal dasturlash tili to'g'risida qichqacha ma'lumotlar	4
Turbo – paskal tizimi va u bilan ishlash qoidalari.....	4
Turbo Paskal tizimining bosh “menyu” si.....	4
Turbo-Paskal matn muhariri buyruqlari	7
Paskal tilining asosiy operatorlari	12
Paskal tilida qism dastur.....	15
Paskal tilida satri ma'lumotlar bilan ishlash	16
Algoritm va dastur tuzishga doir amaliy mashg'ulotlar.	
Chiziqli algoritm va dastur tuzish	26
XULOSA	33
FOYDALANILGAN ADABIYOTLAR	34

AMINOV I.B., TUGALOV R.G'

PASKAL DASTURLASH TILIDA MASALALAR TO'PLAMI

Texnik muharrir *Abdullayev F.*

Terishga berildi: 10.01.2021 y.

Bosishga ruxsat berildi: 13.01.2021 y

Ofset bosma qog'ozi. Qog'oz bichimi 60x84 ^{1/16}.

« Cambria» garniturasini. Ofset bosma usuli.

2,25 bosma taboq Adadi: 50 nusxa. Buyurtma №36/20

Samarqand viloyati Samarqand viloyat xalq ta'limi xodimlarini qayta tayyorlash va ularning malakasini oshirish hududiy markazi bosmaxonasida chop etildi.

Samarqand shahar, Obidinov ko'chasi 7-uy.

