

O‘ZBEKISTON RESPUBLIKASI
OLIY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI

QARSHI MUHANDISLIK – IQTISODIYOT INSTITUTI
“AVTOMATIKA VA AXBOROT TEXNOLOGIYALARI”
KAFEDRASI

«Informatika va Axborot texnologiyalari» fanidan

KURS ISHI

Mavzu: Delphi tilida asosiy operatorlar tavsifi

Bajardi:

TJA-173 -guruh talabasi

S.Teshayev

Qabul qildi:

A.Xurramov

Qarshi-2014 y.

Mavzu : Delphi tilida asosiy operatorlar tavsifi

REJA

Kirish

I. Delphi tilida asosiy operatorlar tavsifi

- 1.1. Dasturlash til elementlari**
- 1.2. O'zgarmlar, o'zgaruvchilar va standart funksiyalar**
- 1.3. Ma'lumotlar turlari**
- 1.4. Ma'lumotlarni kiritish va chiqarish operatorlari**
- 1.5. Delphida konsol ilovasini yaratish**
- 1.6. Shartli o'tish operatori**
- 1.7. Shartsiz o'tish va tanlash operatorlari**
- 1.8. Sikl operatorlari.**

II. Amaliy masala

- 1.1 Amaliy masalaning berilishi va uning algoritmi**
- 1.2 Amaliy masalaning echimini topish dasturi va natijalari**

Xulosa

Adabiyotlar

K I R I SH

Kompyuter dunyosida ko‘plab dasturlash tillari mavjud. Bir xil turdagi ishni bajaradigan dasturlarni Basic, Paskal, C++, kabi tillarda yozish mumkin. Lekin, qaysi dasturlash tili yaxshi? Bu savolga javob berish oddiy emas. Biroq shuni ishonch bilan aytish mumkinki, Pascal tili boshqa tillarga qaraganda dastur tuzishni o‘rganish uchun ancha qulay til bo‘lib hisoblanadi.

Paskal tili Shvesariyalik olim N.Virt tamonidan yaratilib, keyinchalik Borland korporasiyasi tamonidan rivjlantirildi. Bu til rivojlantirilib Turbo Pascal, Borland Pascal va keyinchalik esa Object Pascal nomini oldi. Hozirgi kunda Object Pascal tili asosi bo‘lgan Windows muhitida ishlovchi Delphi dasturiy vositasida murakkab professional dasturlar ishlab chiqilmoqda.

Kompyuterda dasturlash oxirgi yillarda juda tez rivojlanib dastur tuzushga qiziquvchilar soni oshib bormoqda. 10-15 yil oldin o‘z dasturlarini Windows muhitida yaratish ko‘pgina dasturchilarning orzusi edi. Delphi dasturlash vositasining yaratilishi esa nafaqat professional dasturchilar, balki oddiy dastur tuzuvchilar uchun ham keng yo‘l ochib berdi.

1.1. Dasturlash til elementlari

Hozirgi kunda juda ko'p algoritmik tillar mavjud. Bu tillar ichida Paskal tili universal tillardan biri bo'lib, boshqa tillarga qaraganda imkoniyatlari kengroq tildir. So'ngi yillarda Paskal tili juda takomillashib, tobora ommalashib bormoqda. Paskal tilida programa tuzish uchun Turbo Paskal va Delfi dasturlash vositalari mavjud. Bu dasturlash vositalari zamonaviy kompyuter texnologiyasining hamma talablarini o'z ichiga olgan va unda dastur tuzuvchi uchun hamma qulayliklar yaratilgan.

Delphi dasturlash vositaci Turbo Pascal tilining rivoji bo'lgan Object Pascal tilini ishlatadi. Hozirgi kunda bu tilga juda ko'plab yangiliklar kiritilgan uning imkoniyatlari yanada kengaytirilgan, shu sabab bu tilni Delphi tili deb ham atash mumkin.

Delphi tili ham boshqa dasturlash tillari kabi o'z alfavitiga va belgilariga ega. U 26 bosh lotin harflarini, 0 dan 9 gacha bo'lgan arab raqamlarini va quyidagi belgilarni ishlatadi: bo'shliq belgisi; 4 ta arifmetik amallar + , - , * , / ; mantiqiy amallarni bajarish uchun < , > , <= , >= , <> , = belgilarini ishlatadi. Bulardan tashqari vergul, nuqta, ikki nuqta, kichik qavs, katta va o'rta qavslar. Dasturda izohlar istalgan joyda berilishi mumkin. Ular katta qavs ichida yoziladi.

Masalan. Program ad; { Bu dastur nomi }

1.2 O'zgarmlar, o'zgaruvchilar va standart funksiyalar

Haqiqiy turdagi sonlar umumiy holda quyidagi ko'rinishda bo'ladi:

$s a_1 a_2 \dots a_n . b_1 b_2 \dots b_k$

Bu yerda s ishora (+ yoki -) yoki bush joy; $a_1 a_2 \dots a_n$ butun qism; $b_1 b_2 \dots b_k$ kasr qism. Masalan: +3,147 soni +3.147 yoki 3.147

-143,03 soni -143.03

57,0 soni 57.0

0,493 soni 0.493 yoki .493

Haqiqiy sonlarning o'zgarish diapazoni kompyuterning turiga karab turlicha bo'ladi. $10^{-38} < x < 10^{+38}$ x-ixtiyoriy son. Ular eksponensial (darajali)

ko'rinishda ifodalanishi ham mumkin, ya'ni $\pm m 10^{\pm n}$. Bunday sonlar quyidagicha yoziladi $\pm mE \pm n$. Masalan:

$0,43 \cdot 10^{-6}$.43E-6

0,0003 3E-4

Butun sonlar umumiy holda quyidagicha yoziladi $s a_1 a_2 \dots a_n$.

Masalan: +345 soni +345 yoki 345

-106 soni -106

Butun sonlar o'zgarish diapozoni -32768 dan +32767 gacha. Agar butun son qiymati bu dipazondan chiqsa, u haqiqiy son shaklida ifodalanadi yoki kompyuter turiga qarab, u o'noltilik sanoq sistemasida ifodalanishi ham mumkin. Belgililar qo'shtirnoq ichida yoziladi. Yozilish diapazoni 0 dan 255 tagachadir. Misol. "Paskal", "405.5"

Paskal tilida identifikator so'zi ishlatilib dasturda obyektlarni nomlashda ishlatiladi. O'zgarmlarni, o'zgaruvchilarni, belgi(metka), protsedura va

funksiyalarni belgilashda ishlatilgan nom **identifikatorlar** deyiladi. Identifikatorlar lotin alfaviti harflaridan boshlanib qolganlari belgi yoki raqam ketma-ketligidan tashkil topgan bo'lishi mumkin. Masalan: xx, xx1, alfa&.

Delphi tilida dastur ishlashi mobaynida qiymati o'zgarmaydigan identifikatorlar **o'zgarmaslar** deyiladi va ular dasturning bosh qismida **Const** so'zi bilan e'lon qilinib unga aniq qiymat tenglashtiriladi.

Misol. Const aa1=2.27;
 Pi=3.14;
 radius=14;

Dastur ishlashi mobaynida qiymatlari o'zgarishi mumkin bo'lgan identifikatorga **o'zgaruvchilar** deyiladi va ular dastur bosh qismida **Var** so'zi bilan e'lon qilinadi. O'zgaruvchilar nomi keltirilib, ularning turlari beriladi. O'zgaruvchilarning eng ko'p ishlatiladigan turlari **butun, haqiqiy, belgili, qator** va **mantiqiy**dir. Ular mos ravishda butun - **Integer**, haqiqiy - **Real**, belgili - **Char**, qator (matn) - **String** va mantiqiy - **Boolean** deb yoziladi.

Masalan: Var a, d1, alfa : Integer;
 c121, df : Real;
 Etx, xx : Char;
 St,Sw: String;
 fl : Boolean;

Mantiqiy o'zgaruvchilar faqat ikkita qiymat qabul qiladi: "True" (chin) va "False" (yolg'on).

Standart va nostandart matematik funksiyalar

Funksiya nomi	Tilda yozilishi	Ma'nosi
Sinx	SIN(x)	x ning sinusi
Cosx	COS(x)	x ning kosinusi
Ln x	Ln(X)	x ning natural logarifmi
e ^x	EXP(x)	Yeksponenta
$\sqrt{x}$	SQRT(x)	Kvadrat ildiz
Arctgx	ARCTAN(x)	x ning arktangensi
x	ABS(x)	x ning moduli
x ²	SQR(x)	x ning kvadrati
a ^b	EXP(b*LN(a))	a ning b chi darajasi
	Int(x)	haqiqiy son butun qismi
	Round(x)	Yaxlitlangan butun son
	Str(x)	Sonni matnga o'tkazish
	Val(x)	Matnli sonni raqamga o'tkazish
	Chr	Simvolni kodi bilan chiqarish
	Readkey(kod)	Mos tugmacha belgisini aniqlash

Nostandart matematik funksiyalar.

$$\begin{aligned}
&1. \sec x = \frac{1}{\sin x}; \quad 2. \operatorname{cosec} x = \frac{1}{\cos x}; \quad 3. \operatorname{Tgx} = \frac{\sin x}{\cos x}; \quad 4. \operatorname{Arcctg} x = \operatorname{Arctg} \frac{1}{x}; \\
&5. \operatorname{Arc} \sin x = \operatorname{Arctg} \frac{x}{\sqrt{1-x^2}}; \quad 6. \operatorname{Arc} \cos x = \operatorname{Arctg} \frac{\sqrt{1-x^2}}{x}; \quad 7. \operatorname{Arc} \sec x = \operatorname{Arctg} \frac{1}{\sqrt{1-x^2}}; \\
&8. \operatorname{Arc} \operatorname{cosec} x = \operatorname{Arctg} \sqrt{1-x^2}; \quad 9. \operatorname{Log}_a b = \frac{\operatorname{Lnb}}{\operatorname{Lna}}; \quad 10. \operatorname{Padian} = \frac{\operatorname{Gradius} \cdot \pi}{180}
\end{aligned}$$

Dasturda arifmetik va mantiqiy ifodalar o'zgaruvchi, o'zgarmas, standart funksiyalar, qavslar va amal belgilari orqali tashkil qilinadi.

Ifodalarda hisoblashlar tartibi qavslar ichidagi ifodalar bajarilgandan keyin quyidagi tartibda bajariladi:

1. NOT amali;
2. *, /, DIV, MOD, AND;
3. +, -, OR;
4. taqqoslash belgilari: <, >, <=, >=, <>, =, IN.

Ifodadagi amal natijasi qanday turda bo'lishi amallarda qatnashayotgan o'zgaruvchilarning turlariga bog'liq. Agar ikkita o'zgaruvchining turi Integer yoki Real bo'lsa, amal natijasi ham Integer yoki Real bo'ladi. Agar biri Integer ikkinchisi Real bo'lsa natija Real bo'ladi. NOT, OR, AND va taqqoslash amallarining natijalari esa Boolean turida bo'ladi.

Kompyuter foydalanuvchi tomonidan qo'yilgan masalani aniq va tushunarli ko'rsatmalar berilgandagina bajara oladi. Bu ko'rsatmalar ma'lum bir ma'noni anglatuvchi so'zlardan iborat bo'lib, kompyuterga qanday operatsiyani bajarish lozimligini bildiradi va bu ko'rsatmalarga **operatorlar** deyiladi. Operatorlar dastur ishlaganda ketma-ket ravishda bajariladi. Delphi tilida bir satrga bir necha operatorlarni yozish umkin.

Delphi tilida dastur matni bosh va asosiy bo'limdan tashkil topadi. Bosh bo'lim dastur nomi va o'zgaruvchilar, o'zgarmaslar, massivlar, belgilar(metkalar), protseduralar va funksiyalarni tavsiflashdan iborat bo'ladi. Asosiy bo'lim dastur tanasi deyilib, unda dasturda bajariladigan hamma operatorlar ketma-ketligi beriladi va u Begin (boshlamoq) so'zi bilan boshlanib End (tugash) so'zi bilan tugaydi. Umumiy holda dastur strukturasi quyidagi ko'rinishga ega:

```

Program <dastur nomi>;
Uses <Foydalanadigan bibliotekalar (modullar) ro'yxati>;
Label <Ishlatiladigan belgilar(metkalar) ro'yxati>;
Const <Ishlatiladigan o'zgarmaslarni aniqlash>;
Type <Yangi turlarni aniqlash>;
Var <O'zgaruvchilarni e'lon qilish>;
<Protsedura va funksiyalarni aniqlash>
Begin
<Bajariladigan operatorlar ketma ketligi>
End.

```

1.3 Ma'lumotlar turlari

Ma'lumotlar turlarini Delphi tilida umumiy holda ikkiga ajratish mumkin:

- standart turlar. Bu turlar oldindan Delphi tili tomonidan aniqlangan bo'ladi;
- dasturchi tomonidan kiritiladigan (aniqlanadigan) turlar.

Standart turlar tarkibiga quyidagilar kiradi: butun, haqiqiy, belgili (simvol), qator (strok), mantiqiy, ko'rsatgichli va variant.

Dasturchi turlarni dasturning **Var** bo'limida o'zgaruvchilarni tavsiflashda aniqlaydi yoki maxsus turlarni aniqlash uchun bo'lim bo'lgan -turlarni tavsiflash **Type** bo'limida aniqlaydi.

Bu bo'lim umumiy holda quyidagicha bo'ladi.

Type

<tur nomi>=<turning tavsifi>;

Misol:

Type

TColor=(Red,Blue,Black);

Var Color1,Color2,Color3: TColor;

Type bo'limida dasturchi tomonidan yangi Tcolor nomli tur kiritilayapdi va u Red,Blue,Black mumkin bo'lgan qiymatlarni qabul qilishi mumkin.

Var bo'limida dasturchi tomonidan turi aniqlangan uchta Color1,Color2,Color3 o'zgaruvchilar tavsiflanmoqda.

Bu o'zgaruvchilarni to'g'ridan to'g'ri quyidagicha ham tavsiflash mumkin.

Var Color1,Color2,Color3: (Red,Blue,Black);

Standart turlarni Type bo'limida tavsiflash shart emas, ularni to'g'ridan to'g'ri Var bo'limida tavsiflash mumkin.

Delphida standart turlarni quyidagicha klassifikatsiya qilish mumkin.

❖ Oddiy

➤ Tartibli

- Butun
- Belgi
- Mantiqiy
- Sanoqli (Perechislyayemiy)
- Chegaralangan

➤ Haqiqiy

❖ Qator

❖ Struktura

➤ To'plam

➤ Massiv

➤ Yozuv

➤ Fayl

➤ Klass

➤ Interfeys

❖ Ko'rsatgichli

❖ Protsedurali

❖ Variant

Oddiy turlarga tartiblashgan va haqiqiy turlar kiradi. Tartiblashgan turlar shu bilan xarakterlanadiki uning har bir qiymati o'zining tartiblangan nomeriga ega. Haqiqiy tur qiymatlari kasr qismidan iborat bo'lgan sonlardan iboratdir.

Tartiblashgan turlarga butun, belgili, mantiqiy, sanoqli va chegaralangan turlar kiradi.

Butun turlar. Butun turlar butun sonlarni tasvirlash uchun ishlatiladi. Jadvalda Delphi 7 da ishlatiladigan butun turlar ro'yxati keltirilgan.

Tur	O'zgrish diapazoni	O'lcham (baytda)
Integer	-2147483648..2147483647	4
Cardinal	0..4294967295	4
Shjrtint	-128..127	1
Smallint	-32768..32767	2
Longint	-2147483648..2147483647	4
Int64	$-2^{63}..2^{63}-1$	8
Byte	0..255	1
Word	0..65535	2
LongWord	0..4294967295	4

Haqiqiy turlar. Haqiqiy turlar haqiqiy sonlarni tasvirlash uchun ishlatiladi. Jadvalda Delphi 7 da ishlatiladigan haqiqiy turlar ro'yxati keltirilgan.

Tur	O'zgrish diapazoni	O'lcham (baytda)
Real	$5.0*10^{-324}..1.7*10^{308}$	8
Real48	$2.9*10^{-39}..1.7*10^{38}$	6
Single	$1.5*10^{-45}..3.4*10^{38}$	4
Double	$5.0*10^{-324}..1.7*10^{308}$	8
Extended	$3.6*10^{-4951}..1.1*10^{4932}$	10
Comp	$-2^{63}+1..2^{63}-1$	8

Belgili turlar. Ma'lumotlarning belgili turlari faqat bitta belgini saqlash uchun xizmat qiladi. Jadvalda Delphi 7 da ishlatiladigan belgili turlar ro'yxati keltirilgan.

Tur	O'lcham (baytda)
Char	1
ANSChar	1
WideChar	2

Mantiqiy turlar. Mantiqiy turlar chin (True) yoki yolg'on (False) qiymatning birini qabul qiladi. Jadvalda Delphi 7 da ishlatiladigan mantiqiy turlar ro'yxati keltirilgan.

Tur	O'lcham (baytda)
Boolean	1
ByteBool	1
WordBool	2
LongBool	4

1.4. Ma'lumotlarni kiritish va chiqarish operatorlari

Biror bir masalani yechishning chiziqli bo'lgan algoritmiga dastur tuzishda algoritmdagi keltirilgan ketma-ketliklar asosida operatorlar yoziladi. Bunday dasturlarni tuzushda asosan o'zgaruvchilar qiymatini kiritish, natijalarni chiqarish va shu bilan birga o'zlashtirish operatorlari ishlatiladi.

Dasturdagi o'zgaruvchilar qiymatlarini dastur ichida o'zlashtirish operatori yordamida ham berish mumkin. Lekin dasturda o'zgaruvchi qiymatini tashqaridan kiritish qulaylik tug'diradi va umumiylikni ta'minlaydi.

Read operatori o'zgaruvchilar qiymatlarini ekrandan kompyuter xotirasiga kiritish uchun ishlatiladi. U quyidagi ko'rinishlarga ega.

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

bu yerda c1,c2,...,cn - o'zgaruvchilar nomi; ln - qo'shimchasi qiymatni kiritib keyingi qatorga o'tishni bildiradi.

Misollar: Read(Sm1,Sm2);
Readln(x1,x2,x3);
Readln;

Bu yerda birinchi operator Sm1 va Sm2 o'zgaruvchilar qiymatini ekrandan kiritadi. Ikkinchi operator esa x1,x2,x3 o'zgaruvchilar qiymatini ekrandan kiritadi va kiritishni keyingi qatorga o'tkazadi. Oxirgi operator esa kiritishni kutadi va qator o'tkazadi.

Write operatori oddiy ma'lumotlarni va o'zgaruvchilar qiymatlarini kompyuter ekraniga chiqarish uchun ishlatiladi. U quyidagi ko'rinishlarga ega.

```
Write(c1,c2,...,cn);
Writeln(c1,c2,...,cn);
Writeln;
```

bu yerda c1,c2,...,cn - oddiy matnlar yoki o'zgaruvchilar nomi; ln - qo'shimchasi chiqarishni keyingi qatorga o'tishni bildiradi.

Misollar: Write(Summa);
Write('Natija yuk');
Write('Tenglama yechimi x1=', x1, 'x2=', x2);

Oddiy ma'lumotlarni chiqarishda ular matn deb qaraladi va u qo'shtirnoq ichida yoziladi. Chiqarish operatori yordamida o'zgaruvchilar qiymatini format ko'rinishda ham berish mumkin:

```
Write(c:m:n);
```

bu yerda s-o'zgaruvchi; m-shu o'zgaruvchi qiymati uzunligi; n-qiymatning kasr qismi va unda $n-1 < m$ bo'lishi kerak.

Misol. Write(x:8:4);

Agar x=155.01021 bo'lsa, quyidagi yozuv chiqadi 115.0102.

Write('Maxsulot soni:', kol:5);

Agar kol=15 bo'lsa, quyidagi yozuv ekranga chiqadi,

Maxsulot soni: 15

Dastur matnini tushuntirish maqsadida ko'pincha dasturda izohlar keltiriladi. Dasturda izohlar istalgan joyda berilishi mumkin. Izoh katta qavs ichida yoziladi.

Masalan: { Bu matn dasturga izoh beradi }

{ Bu joyda yechim aniqlanmoqda }

Dasturda ma'lum hisoblashlarning natijalarini biror bir o'zgaruvchida saqlash uchun o'zlashtirish (yuborish) operatori ishlatilib, u «:=» belgisi yordamida qiymat yuborilishi kerak bo'lgan o'zgaruvchidan keyin qo'yiladi.

Masalan: i:=0; i-qiymati nolga tenglashadi, ya'ni i o'zgaruvchiga nol yuboriladi deb tushuniladi. Bunda mashina i o'zgaruvchi uchun ajratilgan xotirasiga nol yozib saqlaydi.

Misol: B:=5; C:=4; A:=(B+C)/2;

Bu yerda, agar A butun identifikator bo'lsa, uning qiymati 4 ga, aks holda esa 4.5 qiymatga ega bo'ladi.

Chiziqli strukturali algoritmlarni dastur shaklida yozish uchun oldin ishlatiladigan o'zgaruvchilar ruyxati keltirilib, keyin algoritmdagi bajarilishlar ketma-ket ravishda amalga oshirilishi kerak.

Misol: Tekislikdagi ikki nuqta orasidagi masofani topish dasturi.

Program XY;

Var

x1,y1,x2,y2,d: Real;

Begin

Write('Nukta koordinatalarini kiriting:');

Read(x1,y1,x2,y2);

d:=Sqrt(Sqr(x1-x2)+Sqr(y1-y2));

Writeln;

Writeln('Nukta koordinatalari:',x1,y1,x2,y2);

Writeln('Masofa=',d);

Readln;

End.

1.5.Delphida konsol ilovasini yaratish

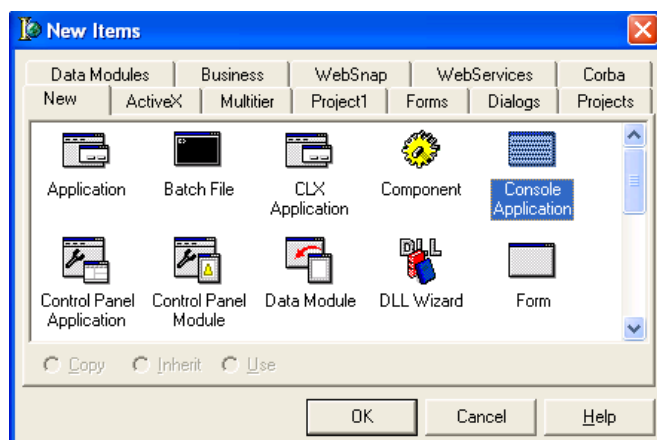
Delphida konsol ilovalarini har xil usullarda yaratish mumkin. Ulardan eng oson usuli quyidagicha:

1.Delphi muhiti ishga tushiriladi.

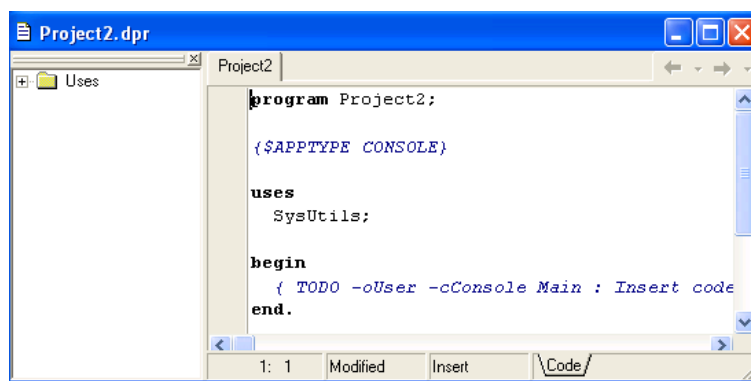
Pusk=>Programmi=>Borland Delphi

2.Bosh menyudan File punktini ochib u yerdan New, keyin esa Other buyruqlari beriladi. **File=> New=> Other**

3. Forma va loyihalarni saqlash uchun ochilgan maxsus oynadan (bu oynaga Delphi arxiv oynasi deyiladi) “Console Application” piktogrammasi tanlanadi va Ok tugmasi bosiladi.



4. Natijada ekranda loyiha oynasi ochiladi (.dpr kungaytmali nom bilan).



Begin – end ichiga olingan

{ TODO -oUser -cConsole Main : Insert code here }

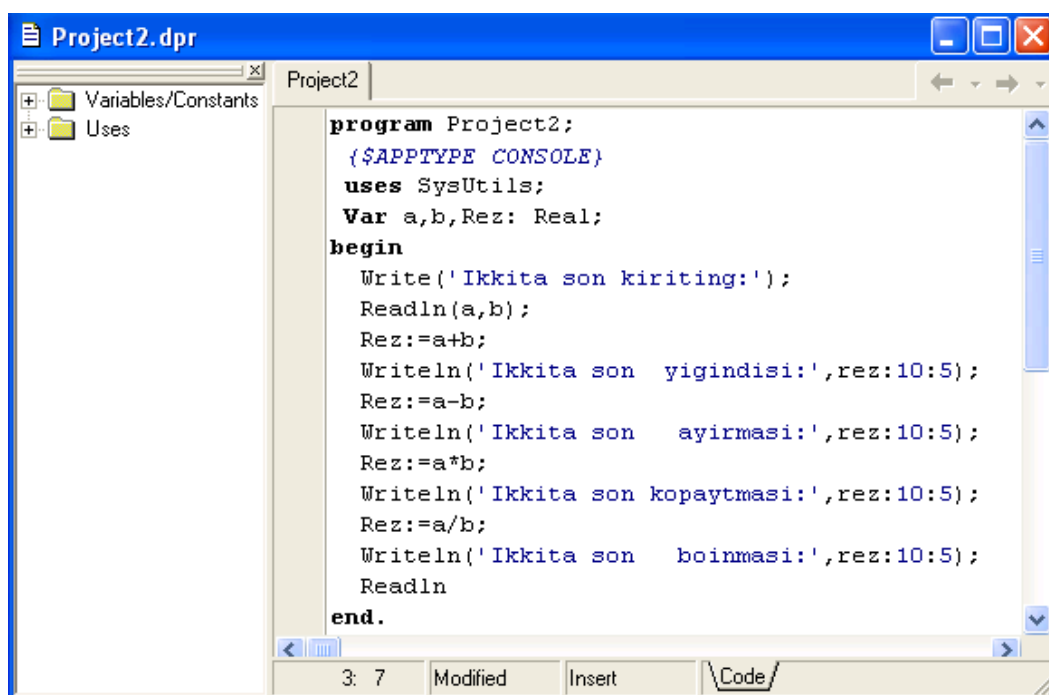
izoh o‘rniga loyiha faylining dastur matni kiritiladi.

Tuzilgan dasturni ishga tushirishdan oldin uni saqlash kerak bo‘ladi. Uni saqlash file=>Save All buyrug‘ini berish lozim. Har bir loyiha alohida yangi papkaga saqlanishni tavsiya beradi. Loyiha faylini saqlashda alohida kursatilmagan holatida ProjectN.dpr nomli fayl nomini tavsiya qiladi. Bu yerda N har bir ketma ket nomlanadigan loyiha nomeri (son, masalan 1,2,3,..). Lekin biz loyiha faylini istalgan nom bilan saqlashimiz mumkin. Masalan MyProgram.dpr. Bu nom avtomatik ravishda chiqadi.

Loyihani saqlab bo‘lgandan so‘ng, uni bajarishga beramiz. Buning uchun bosh menyudan quyidagi buyruqni berish lozim: Run=>Run yoki F9 funksional tugmachasini bosish kerak bo‘ladi. Dastur narmal ishga tushgandan so‘ng ekranda DOSning standart dastur oynasi namayon bo‘ladi.

Misol. Ikkita sonning yig‘indisi, ayirmasi, ko‘paytmasi va bo‘linmasini hisoblash dasturini yarating.

Bu misolni yechish uchun yuqorida keltirilgan to‘rtta ketma ketlikni bajaramiz va dastur kodini kiritamiz.



Dastur kodi kiritilgandan so‘ng uni saqlab keyin ishga tushiramiz. Natijada ekranda Dos oynasi ochilib unda “Ikkita son kiriting:” so‘zi chiqadi. Keyin ikkita son kiritib Enter tugmasini bosish kerak bo‘ladi.

1.6. Shartli o‘tish operatori

Pascal tilida shart - bu mantiqiy turdagi ifoda bo‘lib, u faqat «chin»(True) yoki «yolg‘on»(False) qiymatni qabul qiladi.

Quyidagi mantiqiy belgilar ishlatiladi: >, <,<=,>=,<>,. Bularga munosabat amallari deyiladi.

Quyidagi mantiqiy amallar ishlatiladi:

- NOT-«inkor»;
- AND-«mantiqiy va»;
- OR-«mantiqiy yoki».

Bu mantiqiy amallarning bajarilish natijalari quyidagicha:

Op1	Op2	Op1 AND Op2	Op1 OR Op2	NOT Op1
False	False	False	False	True
False	True	False	True	True
True	False	False	True	False
True	True	True	True	False

Masalan: (5<6) AND (6<50) -mantiqiy ifoda rost (True),
 (20>0) OR (20<0.5) -mantiqiy ifoda rost (True),
 (10<8) AND (10<15) -mantiqiy ifoda yolg‘on (False),

NOT(100>3) -mantiqiy ifoda yo'lQon (False).

Mantiqiy ifodalarni biror bir mantiqiy o'zgaruvchiga yuborish ham mumkin.

Masalan:

F:=(A<B) AND (A<C);

Bu yerda, agar ikkala shart bajarilgandagina F mantiqiy o'zgaruvchi "chin" (True) qiymatni qabul qiladi. Aks holda "yolg'on" (False) qiymatni qabul qiladi.

Pascal tilida shartli o'tish operatorining ikki xil ko'rinishi mavjud: to'liq va qisqa.

To'liq ko'rinish:

If <shart> then Begin

<shart rost bo'lganda bajariladigan operatorlar>

End

Else

Begin

<shart yo'lQon bo'lganda bajariladigan operatorlar>

End;

Qisqa ko'rinish:

If <shart> then Begin

<shart rost bo'lganda bajariladigan operatorlar>

End;

Bu yerda IF -agar; then -u holda; else -aks holda ma'nosini bildiruvchi xizmatchi (kalit) so'zlar.

Birinchi ko'rinishdagi shartli operatorlarda, agar shart bajarilsa birinchi Begin va end ichidagi operatorlar ketma-ket bajariladi, aks holda ikkinchi Begin va end ichidagi operatorlar ketma-ket bajariladi.

Ikkinchi ko'rinishdagi shartli operator quyidagicha ishlaydi. Agar berilgan shart bajarilsa Begin va end ichidagi operatorlar ketma-ket bajariladi, aks holda ular bajarilmaydi.

Agar bajariluvchi operatorlar soni bitta bo'lsa Begin va End so'zlarini yozish shart emas.

Misollar:

1) If A>0 Then Begin C:=1; B:=C+1; End

Else Begin C:=0; B:=4; End;

2) If D=A Then D:=A Else A:=D;

O'ar bir shartli o'tish operatori ichida boshqa ichki shartli operatorlar joylashishi ham mumkin, masalan.

If b1 then a1 else If b2 then a2 Else a3;

Misollar.

A:=0.5; B:=-1.7; IF A<B THEN A:=B ELSE B:=A;

Javob: 0.5<-1.7 yolg'on bo'lganligi sababli B:=A operator bajariladi, va bunda A=0,5 va B=0,5 ekenligi kelib chiqadi.

A:=0.1; B:=0.1; C:=0.5; D:=0;

IF (A<B) OR (A>C) THEN D:=B+C ELSE

IF B=A THEN BEGIN D:=C; C:=A; END;

Javob: $(0.1 < 0.1)$ yoki $(0.1 > 0.5)$ bu mantiqiy ifoda yolg'on bo'lganligi sababli $B=A$ shart tekshiriladi. Bu shart chin bo'lganligi sabab $D=0,5$ ga, $S=0,1$ qiymatlarga teng ekanligi kelib chiqadi.

1.7. Shartsiz o'tish va tanlash operatorlari

Dasturda shunday holatlar bo'ladiki operatorlarning bajarilish shartiga qarab dasturning u yoki bu qismiga to'g'ridan-to'g'ri o'tishga to'g'ri keladi. Bunday holatlarda shartsiz o'tish operatoridan foydalanish mumkin.

Shartsiz o'tish operatorining ko'rinishi quyidagicha:

Goto n;

Bu yerda n -belgi(metka) bo'lib identifikator yoki butun son bo'lishi mumkin. Goto - o'tish ma'nosini bildiradi.

n -belgi dasturning bosh qismida Label so'zi yordamida e'lon qilingan bo'lishi shart. n boshqarilish uzatiladigan joyga n : shaklida qo'yiladi.

Misol:

```
.....  
Goto L2;  
.....  
L2: C:=x*y;  
.....
```

Ko'p hollarda baror bir parametrlarning qiymatiga qarab kerakli operatorlarni bajarishga to'g'ri keladi. Bunday hollarda tanlash operatorini ishlatgan qulay. Tanlash operatori ko'rinishi quyidagicha bo'ladi:

```
Case s of  
  1: A1;  
  2: A2;  
  .....  
  n: An;  
Else Begin  
  <B1,B2,..Bn>  
End;  
End;
```

Bu yerda Case -xizmatchi so'z bo'lib tanlash ma'nosini beradi; of -«dan» ma'nosini beradi; s-operator selektori; 1,2,..n-operator belgilari; $A1, A2, \dots, A_n$ va $B1, B2, \dots, B_n$ -operatorlar.

Case operatori tarmoqlanish jarayonida berilgan bir necha operatoridan birini tanlash yo'li bilan amalga oshiradi. Operatorlar ketma-ketligini tanlash operator selektorining qiymatiga qarab aniqlanadi. Operator selektori haqiqiy bo'lmagan o'zgaruvchi yoki ifoda bo'lishi mumkin. Agar operator selektori qiymati operator belgilari o'zgarmas qiymatiga teng bo'lmasa $B1, B2, \dots, B_n$ -operatorlari ketma-ket bajariladi.

Shartli o'tish operatorining quyidagi ko'rinishi

If B Then A1 Else A2;

tanlash operatorining quyidagi operatoriga ekvivalentdir.

Case B of

```

      True: A1;
      False: A2;
End;

```

Misol

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

PROGRAM Corni;

Label 20;

Var A, B, C, D, E, F, X, X1, X2, Z:real;

BEGIN

Writeln ('a, b, c koeffitsiyentlar qiymatini kiriting:');

Read (A,B,C);

if A=0 THEN

BEGIN

x:=-B/C; writeln (x);

goto 20

end

else

BEGIN

D:=B*B-4.0*A*C; Z:=2.0*A; E:=-B/Z;

F:=sqrt(ABS(d)/Z;

End;

if D>=0 THEN

BEGIN

x1:=E+F; x2:=E-F; writeln (x1,x2);

End

else if D=0 then

begin X:=E; writeln(x); end else writeln ('yechim yo'q');

20: end.

1.8. Sikl operatorlari.

Ayrim masalalarda bir yoki bir necha parametrlarning o'zgarishiga qarab ma'lum hisoblashlar bir necha marta takrorlanib bajarilishi mumkin. Masalan, $y=ax+b$ funksiyani x ning bir necha qiymatida uning mos qiymatlarini hisoblash kerak deylik. Bunday hisoblashlarni kompyuterda dastur tuzib bajarish uchun siklik strukturali dasturlar tuzish kerak bo'ladi. Bu kabi dasturlarni shartli o'tish operatori yordamida ham tuzish mumkin. Lekin Paskal tilida siklik strukturali dastur tuzish uchun bir necha maxsus operatorlar mavjud.

For operatori takrorlanishlar soni aniq bo'lgan siklik jarayonlar tashkil etishda ishlatiladi. Uning umumiy ko'rinishi quyidagicha: **For i:=m1 to m2 Do S;**

Bu yerda i-sikl parametri; m1,m2 -i parametrining boshlanQich va oxirgi qiymati bo'lib, ular o'zgarmas son yoki ifoda bo'lishi mumkin; S-sikl tanasi bo'lib, bir necha operatorlardan tashkil topishi mumkin.

Agar sikl tanasi bir necha operatorlardan iborat bo'lsa ular **Begin** va **End** ichiga olinadi.

Misol. 1,2,...10 conlar yiQindisini hisoblash dastursini tuzing.

Program S10;

```
Const kn=10;
Var i: Integer; S: Real;
Begin
  S:=0;
  For i:=1 to kn do S:=S+i;
  Write ('S=',S); Readln;
End.
```

Agar to so'zni **DoWnto** so'ziga almashtirilsa sikl parametri teskari bo'yicha o'zgaradi, ya'ni -1 qadam bilan. U holda sikl ko'rinishi quyidagicha bo'ladi.

For i:=m1 DoWnto m2 Do S;

Misol. 10 dan 1 gacha conlarni ekranga chiqarish dastursini tuzing.

Program SP;

```
Var i: Integer;
Begin
  For i:=10 DoWnto 1 do Write (i); Readln;
End.
```

While sikl operatori takrorlanishlar soni oldindan aniq bo'lmagan hollarda takrorlanishni biror bir shart asosida bajaradi. Berilgan shart oldin tekshiriladi va keyin shartning bajarilishiga qarab kerakli operatorlar ketma-ketligi bajariladi. Bu operatorning umumiy ko'rinishi quyidagicha: **While B Do S;**

Bu yerda B - mantiqiy ifoda; S -sikl tanasi bo'lib, bir yoki bir necha operatorlar ketma-ketligidan iborat bo'lishi mumkin. Mantiqiy ifoda 'True' yoki 'False' qiymat qabul qiladi.

Agar mantiqiy ifoda 'True' qiymat qabul qilsa S operatorlari bajariladi, aks holda bajarilmaydi, ya'ni sikl ishlashdan to'xtaydi.

Misol. 1,2,...,10 conlar yiQindisini hisoblash dastursini tuzing.

Program S10;

```
Const kn=10;
Var i: Integer; S: Real;
Begin
  S:=0; i:=0;
  While i<=kn do Begin i:=i+1; S:=S+i; End;
  Write ('S=',S);
  Readln;
End.
```

Repeat sikl operatori ham takrorlanishlar soni oldindan aniq bo'lmagan hollarda takrorlanishni biror bir shart asosida bajaradi. Oldin sikl tanasidagi operatorlar ketma-ketligi bajariladi. Berilgan shart keyin tekshiriladi. Agar

berilgan shart rost (True) bo'lsa, boshqaruv sikldan keyingi operatorni bajarishga o'tadi, aks holda sikl takrorlanadi. Bu operatorning umumiy ko'rinishi quyidagicha:

Repeat
S

Until B

Bu yerda B -mantiqiy ifoda, 'True' yoki 'False' qiymat qabul qiladi; S -sikl tanasi bo'lib, bir yoki bir necha operatorlar ketma-ketligidan iborat bo'lishi mumkin. Agar mantiqiy ifoda 'False' qiymat qabul qilsa siklda takrorlanish davom etadi, aks holda to'xtaydi.

Misol. 1,2,...,10 conlar yig'indisini hisoblash dastursini tuzing.

Program S10;

Const kn=10;

Var i: Integer; S: Real;

Begin

S:=0; i:=0;

Repeat

i:=i+1; S:=S+i;;

Until I>kn;

Write ('S=',S);

Readln;

End.

Odatda WHILE operatori REPEAT operatoriga nisbatan ko'p ishlatiladi. Bunga sabab ko'pchilik masalalarda sikl tugallanish sharti sikl boshlanmasdan tekshirish maqsadga muvofiqdar. Zarur bo'lsa siklni umuman bajarmasdan o'tish mumkin.

Ko'pchilik masalalarni yechishda tuzilgan dasturda ichma-ich joylashgan sikllar tashkil etishga to'g'ri keladi. Bunday sikllarga murakkab sikllar deyiladi. Murakkab sikllar tashkil etilganda quyidagi talablar bajarilishi zarur.

-ichki sikl tashqi sikl ichida to'liq yotishi kerak;

-sikllar bir-biri bilan kesishmasligi kerak;

-sikl ichiga tashqaridan to'g'ridan-to'g'ri kirish mumkin emas;

-sikl parametrlari boshqa-boshqa identifikatorlar bilan belgila-nishi kerak;

Misol. $S = \sum_{i=1}^{10} \prod_{j=1}^5 \frac{i+j}{\sqrt{i \cdot j}}$ ifodani hisoblash dastursini tuzing.

Bu formulada agar yig'indini ochsak u quyidagi ko'rinishga keladi.

$$S = \sum_{i=1}^{10} \prod_{j=1}^5 \frac{i+j}{\sqrt{i \cdot j}} = \prod_{j=1}^5 \frac{1+j}{\sqrt{1 \cdot j}} + \prod_{j=1}^5 \frac{2+j}{\sqrt{2 \cdot j}} + \dots + \prod_{j=1}^5 \frac{10+j}{\sqrt{10 \cdot j}}$$

```
Program SP;  
  Var  
    i,j: Integer; S: Real;  
  Begin  
    S:=0;  
    For i:=1 to 10 do  
      Begin  
        P:=1;  
        For j:=1 to 5 do P:=P*(i+j)/Sqrt(i*j);  
        S:=S+P;  
      End; Write ('S=',S);  
    End.
```

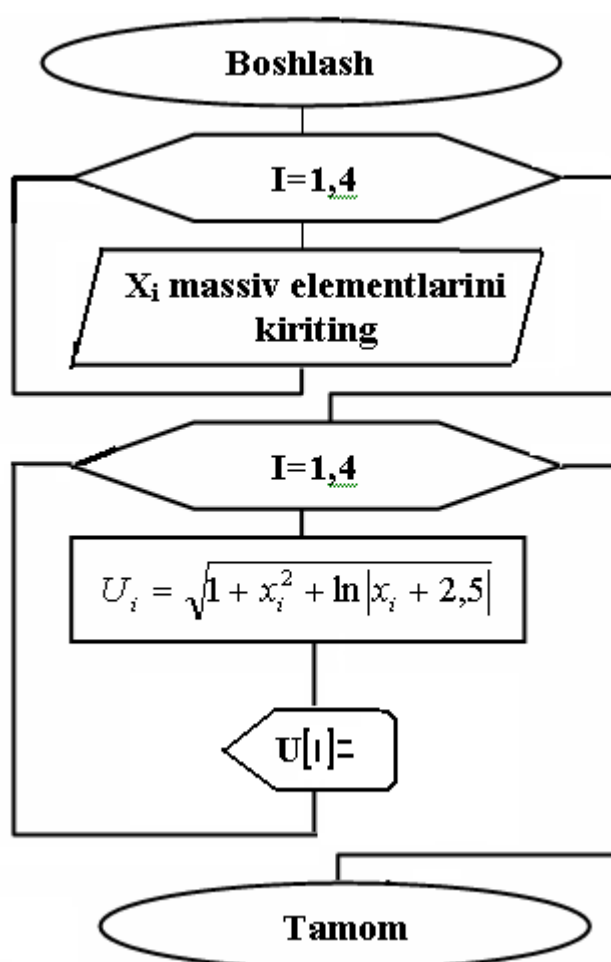
1.1. Amaliy masalaning berilishi va uning algoritmi

Agar $\vec{x}=(4,2; 3; 6; 5,3;)$ ekanligi malum bo'lsa

$$U_i = \sqrt{1 + x_i^2 + \ln|x_i + 2,5|}$$

ni hisoblang.

Ushbu masalani echish algoritmini blok-sxemasi quyidagicha bo'ladi.



Amaliy masalaning echimini topish dasturi va natijalari

Ushbu $U[i]$ vektorni hisoblash dasturi quyidagicha bo'ladi.

```
Program massiv;  
Var  
U, X:array [1..4] of real;  
i:integer;  
b:real;  
Begin  
  for i:=1 to 4 do  
    begin  
      Write('y['I,']='); Readln(b); X[i]:=b;  
    end;  
    for i:=1 to 4 do  
      begin  
        U[i]:=sqrt(1+x[i]*x[i])+ln(abs(x[i]+2,5));  
        Writeln('u['I,']=',u[i]:3:3); readln;  
      end;  
    end.
```

Masala shartiga berilgan x - vector parametrlarini kiritib

$U(1)=5.093$

$U(2)=5.995$

$U(3)=7.114$

$U(4)=8.554$

natijalarini xosil qilamiz.

Adabiyotlar

1. SH.A.Nazirov va boshqalar “Delphi Tilida Dasturlash Asoslari”
2. Файсман А. Профессиональное программирование на Turbo Paskale. 1992.
3. Культин М.Б. Программирование в Turbo Pascal и Delphi, Санкт-Петербург, 2002г.
4. Кондзюба С.П., Громов В.Н. Delphi 6/7. База данных и приложения. М-Санкт-Петербург - Киев, 2002 г.
5. WWW.Intuit.ru. Интернет-Университет информационных технологии. Москва.
6. WWW.ziyonet.uz
7. WWW.REF.uz Интернет сайтлари.