

*O'zbekistan aloqa va axboratlashtirish agentligi
Toshkent axborat texnologiyalar universiteti
Nukus filiali Kompyuter injiniringi
fakulteti Kompyuter injiniringi
yo'nalishi 2 A kurs talabasi
Embergenov Baburning
C++ dasturlash tili fanidan
tayyorlagan*

Kurs ishi

*Mavzu: C++ tilida funktsiya. Funktsiyaning
asosiy parametrlari*



Topshirgan:

Embergenov.B

Qabul qilgan:

No'kis 2015-yil

Dasturlash tillari va ularning xossalari

Reja

Kirish

Asosiy bolim

- a)** Dasturlash tillari turlari.
- b)** Borland C++ Builder tilining dastur yaratish muhiti.
- c)** Borland C++ Builder tilining boshqa dasturlash tillaridan farqi.
- d)** Delphi dasturlash tilining asosida Pascal tili.

Xulosa

Amaliy ish

Foydalanilgan adabiyotlar

Kirish

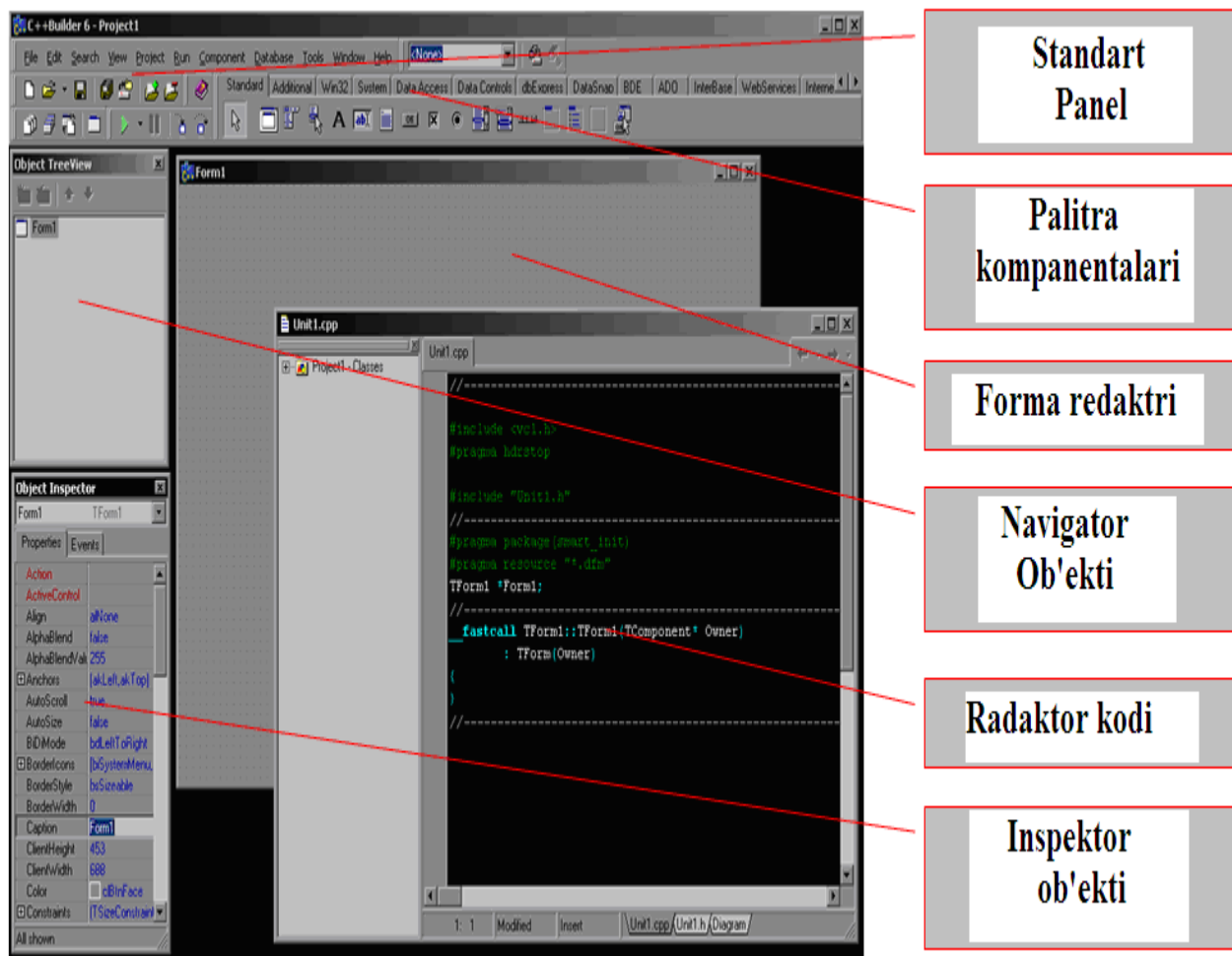
Biz bu kurs ishimizda barsha yuqori darajali dasturlash tillari haqqida va ularning hossalari haqqida toliq malumot berip otamiz. Barsha dasturlash tillari harxil kompyuterlar, telefonlar va hattoki kundalikli hayotimiz ushun kerakli dasturlar tuzush ushun kerak. Hozirgi kunda dasturlash tillarining turlari kop. Bu dasturlash tillari qanaqa dastur ishlap shiqishga qarab tanlashimiz mumkun. Hozirgi kunda yuqori darajadagi dasturlash tillari **C++, Delphi, Java, Pascal** tillari mavjud. Biz bu kurs ishimizda C++, Delphi, Pascal dasturlash tillari haqqida va ularning qanday va qay tarizda yuzaga kelganligi haqqida keng bayon etishga harakat etamiz. Yana biz bu dasturlash tillarining hossalari va ularning bir-biridan farqlari haqqida soz etamiz. Bu dasturlash tillarida har xil hisoplash ishlarini ham amalga oshirish mumkun. Bu ishlarini ham biz kurs ishimizda korsatishga harakat qilganmiz.

Dasturlash – kompyuterlar va boshqa mikroprosessorli elektron mashinalar uchun dasturlar tuzish, sinash va o‘zgartirish jarayonidan iborat. Odatda dasturlash yuqori saviyali dasturlash tillari (Delphi, C++ Builder, Pascal) vositasida amalga oshiriladi. Bu dasturlash tillarining semantikasi odam tiliga yaqinligi tufayli dastur tuzish jarayoni ancha oson kechadi.

C++ Builder dasturlash tili. Xozirgi vaqtga kelib komp`yuter olamida ko'plab dasturlash tillari mavjud. Paskal, C++, Delphi va boshqa dasturlash tillaridir. C++ dasturlash tili universal tildir. U UNIX sistemasi bilan bog'langan bo'lib, bu sistemada ishlatiladigan bir qancha dasturlar C++ tilida yozilgan. C++ Denis Ritchi tomonidan 1972 yili UNIX tipidagi operasion sistemalarini yaratish uchun loyihalashtirilgan. Borland C++ dasturlash tili Windows uchun mo'ljallangan bo'lib, uning birinchi versiyasi Windows operatsion sistema qobig'ida ishlagan. Borland C++ dasturlash tili – bu dasturlarni qayta ishlash muxiti bo'lib, Windows operatsion sistemasida ishlaydi. Unda ob'ektli dasturlash tillari bo'lgan Object mujassamlashgan. Borland C++ dasturlash tili turli xolat protseduralarini qayta ishlash va dasturlarni qayta ishlashda vaqtdan yutish va boshqalarni o'z ichiga oladi.

Dastur yaratish muhiti

Dastur yaratish umumlashgan muhiti Redaktor form – Shakllar muharriri, Inspektor ob'ektov – Ob'ektlar inspektori, Palitra komponentov – Komponentlar palitrasi, Administrator proekta – Proekt administratori va to'la umumlashgan Redaktor koda – Kodlar muharriri hamda kodlar va resurslar ustidan to'liq nazoratni ta'minlaydigan , dastur ilovalarini tezkor yaratadigan Otladchik - instrumentov - Sozlash-instrumentlari kabilarni birlashtiradi.



Komponentlar

Komponentlarni shaklga o'rnatish uchun komponentlar palitrasidagi kerakli piktogramma tanlanadi, so'ngra shaklning komponenta joylanishi kerak bo'lgan joyi tanlanadi. Shundan so'ng komponentalar xossalarini ob'ektlar inspektori yordamida tahrirlash mumkin. Properties bandida komponentalar xossalarining ro'yxati (chapda) va bu xossalarning qiymatlar ro'yxati (o'ngda) joylashgan.

Komponentalar ko'rinadigan (vizual) va ko'rinmaydigan (vizual bo'lmagan) larga bo'linadi. Vizual komponentalar bajarilish paytida proektlash paytidagidek paydo bo'ladi. Bunga knopkalar va tahrirlanuvchi maydonlar misol bo'la oladi. Vizual bo'lmagan komponentalar proektlash vaqtida shakldagi piktogramma ko'rinishida paydo bo'ladi. Ular bajarilish paytida hech qachon ko'rinmaydi, ammo ma'lum funktsionallikka ega bo'ladi (masalan, berilganlarga murojatni ta'minlaydi, Windowsning standart muloqatlarini chaqiradi).

Xossalar

Xossalar komponentalarning tashqi ko'rinishi va tabiatini aniqlovchi atributlar hisoblanadi. Xossalar ustunidagi ko'p xossalar komponentalari oldindan o'rnatilgan (po umolchaniyu) qiymatlarga ega bo'ladi (masalan, knopkplar balandligi). Komponentalar xossalari xossalar varag'i (Properties) da aks ettiriladi. Ob'ektlar inspektori komponentalarning nashr etilgan (published) xossalarini aks ettiriladi. published-xossalardan tashqari komponentalar umumiy (public), faqat ilovalarning bajarilish paytidagina murojat qilish mumkin bo'lgan nashr qilingan xossalarga ega bo'ladi. Xossalar ro'yxati ob'ektlar inspektori xossalar varag'ida joylahadi. Xossalarni proektlash paytida aniqlash mumkin yoki ilovalarning bajarilish paytida ko'rinishini o'zgartirish uchun kod yozish mumkin. Komponenta xossalarini proektlash paytida aniqlash uchun shakldagi komponenta tanlanadi, ob'ektlar inspektori xossalari varag'i ochiladi, aniqlanadigan xossa tanlanadi va zurrur bo'lsa xossalar muharriri yordamida o'zgartiriladi (bu kiritish uchun oddiy maydon yoki son, osilib tushuvchi ro'yxat, ochiluvchi ro'yxat, muloqat paneli va boshqalar bo'lishi mumkin).Biror komponentaning xossalarini dasturning bajarilish paytida o'zgartirish uchun «Imya Komponenta» → «Nazvanie svoystva» tavsifiga o'zaruvchidek murojat qilish kerak, ya'ni qiymatlarni o'zimiz hohlagandek o'qishimiz yoki almashtirishimiz mumkin.

Xodisalar

Ob'ektlar inspektorining xodisalar varag'i (Events) komponentalar tomonidan taniladigan xodisalar ro'yxatini ko'rsatadi. Har bir komponenta o'zining shaxsiy xodisalarni qayta ishlovchi naborga ega bo'ladi. C++ Builder da xodisalarni qayta ishlovchi funksiyalarni yozish va xodisalarni bu funksiya bilan bog'lashga to'g'ri keladi. Biror bir xodisaga qayta ishlovchi yozib, siz dasturga bu xodisa ro'y berganda yozilgan funksiyaning bajarilishini topshirasiz.

Xodisani qayta ishlovchini qo'shish uchun shaklda xodisani qayta ishlovchi komponenta tanlanadi. So'ngra xodisalar varag'ida ob'ektlar inspektori ochilib (Event bandi) xodisaning qatoridagi qiymatlar ustunida sichqonning chap tugmasi

ikki marta bosiladi. Bu bilan C++ Builder ni xodisalarni qayta ishlash prototipini generatsiya qilishga va uni kodlar muharririda ko'rinishiga majbur qiladi. Bu holda bo'sh funksiya nomi generatsiya qilinadi va muharrir kod kiritilishi zarur bo'lgan joyda ochiladi. Kursor buyruqlar qavslari ichiga joylashadi { ... }. So'ngra xodisa sodir bo'lganda bajarilishi kerak bo'lgan kod kiritiladi. Xodisalarni qayta ishlovchi funksiya nomidan keyin ko'rsatiladigan parametrlarga ega bo'lishi mumkin.

Quyida xodisalarni qayta ishlovchi protseduraning shunday bo'sh karkasi ko'rsatilgan:

```
void __fastcall TForm1::Button2Click(TObject *Sender){ }
```

Turlar va C++ da o'zgaruvchilarni tavsiflash

Har bir nom va har bir o'zgaruvchi ular ustida bajariluvchi amallar aniqlovchi turlarga ega bo'ladi. Masalan, **int i**; tavsiflash i o'zgaruvchi int turiga tegishli, ya'ni i butun o'zgaruvchi deb aniqlaydi. Tavsiflash - dasturga nom kirituvchi buyruqdir. Tavsiflash o'zgaruvchining turini aniqlaydi. Tur nom va ifodalardan to'g'ri foydalanishni aniqlaydi. Butun tur uchun quyidagi amallar aniqlangan: +, -, * va /.

Asosiy turlar

Bevosita apparat ta'minotiga javob beradigan asosiy turlar quyidagilar: char; short; int; long; float; double. Birinchi to'rtta tur butun kattaliklarni, oxirgi ikkitasi suzuvchi nuqtali, ya'ni kasr sonlarni tasvirlash uchun ishlatiladi. Char turidagi o'zgaruvchi mazkur kompyuterda belgilarni (odatda bayt) saqlash o'lchoviga ega, int turidagi o'zgaruvchi esa mazkur kompyuterdagi butun arifmetikaga mos o'lchovga ega (odatda so'z). Turlar bilan tasvirlangan butun sonlar diapazoni uning o'lchoviga bog'liq bo'ladi (uni sizeof buyrug'i yordamida hisoblash mumkin). C++ da o'lchovlar char turidagi kattaliklar o'lchovi birligida o'lchanadi. Asosiy turlar o'rtasidagi munosabatlarni quyidagicha yozish mumkin:

```
1 = sizeof(char) <= sizeof(short) <= sizeof(int) <= sizeof(long) = sizeof(float) <= sizeof(double).
```

Umuman, asosiy turlar xususida yana boshqa narsalarni faraz qilish ma'nosiz. Xususan, ko'rsatgichlarni saqlash uchun butun tur etarli, degan xulosa barcha kompyuterlar uchun to'g'ri emas. Asosiy turlarga const so'zini qo'shib tavsiflash mumkin. Bu boshlang'ich turga shu turning o'zini beradi, faqat bu holatda const turidagi o'zgaruvchilarning qiymatlari initsializatsiyadan so'ng o'zgarishi mumkin emas.

const float pi = 3.14;

const char plus = '+';

Bittalik qo'shtirnoqqa olingan belgilar belgi o'zgarmaslar hisoblanadi. Shunga e'tibor berish lozimki, bu usulda tavsiflangan o'zgarmaslar xotirada joy egallamaydi. uning qiymati talab qilingan joyda bevosita ishlatiladi. O'zgarmaslar initsializatsiya paytida tavsiflanishi shart. O'zgaruvchilar uchun initsializatsiya shartemas, ammo albatta tavsiya qilinadi. Lokal o'zgaruvchilarni initsializatsiyasiz kiritish asoslari juda ko'p. Bu turlarning ixtiyoriy kombinatsiyasiga quyidagi arifmetik amallar qo'llanilishi mumkin:

+ (plyus, unar va binar);

- (minus, unar va binar);

* (ko'paytirish);

/ (bo'lish).

Hamda taqqoslash amallari:

== (teng);

!= (teng emas);

< (kichik);

> (katta);

<= (kichik yoki teng);

>= (katta yoki teng).

Agar operandlar qo'yilgan shartni qanoatlantirsa , u holda taqqoslash amallari natijada 1 qiymatni beradi, aks holda esa 0 qiymatni beradi. Butunga bo'lish amali butun natijani beradi: $7/2 = 3$. Butun kattaliklar ustida % - qoldiqni hisoblash amali

bajariladi: $7\%2 = 1$. O'zlashtirishda va arifmetik amallarda C++ ularni guruhlash uchun asosiy turlar o'rtasida barcha ma'noli almashtirishlarni bajaradi:

```
double d = 1;  
int i = 1;  
d = d + i;  
i = d + i;
```

Qo'shimcha turlar

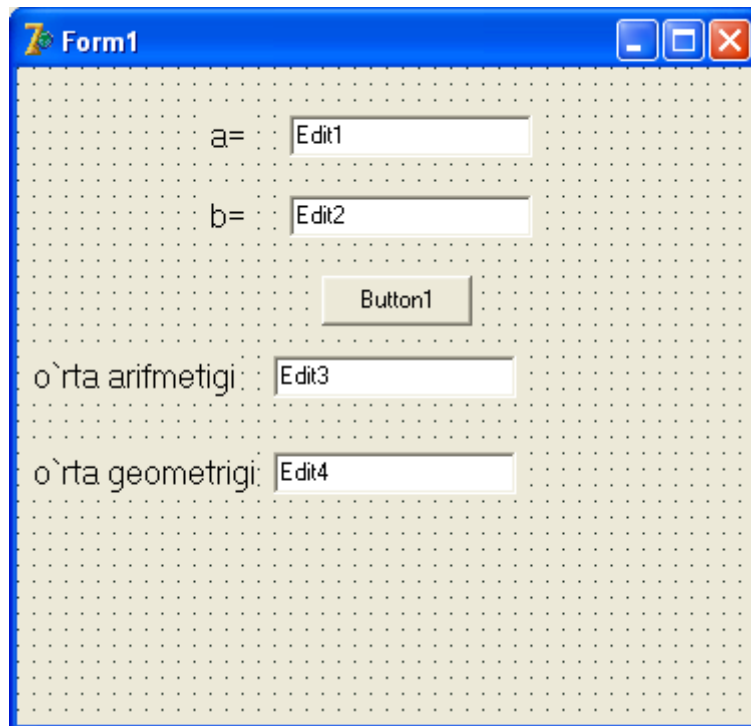
Borland C++ da butun qiymatli o'zgaruvchilarning turlarini qo'shimcha ajratish imkoni mavjud. Bu holda o'zgaruvchilarning barcha tur nomlari quyidagicha yoziladi - int X, bu erda X o'zgaruvchiining bitlardagi maydon o'lchami. X quyidagi qiymatlardan birini qabul qilishi mumkin: 8, 16, 32 va 64. Bu turdagi o'zgaruvchilardan foydalanish standart turda aniqlangan o'zgaruvchilardan foydalanishdan farq qilmaydi. Quyidagi jadvalda bunday turlar bilan ishlash yaqqol ko'rsatilgan.

r nomi	zgaruvchini tavsiflashga misol	lcham
int8	int8 c = 128;	it
int16	int16 s = 32767;	bit
int32	int32 i = 123456789;	bit
int64	int64 big = 12345654321;	bit
signed __int64	signed __int64 huge = 1234567887654321;	bit

Endi Borland C++ Builder dasturlash tilida bir masala korip shiqaylik.

Masala: Ikkita musbat son berilgan, bu sonlarning o'rta arifmetik va o'rta geometrik qiymatlarini aniqlang.

Yechish. A va b sonlarning o'rta arifmetik qiymatini c, o'rta geometrik qiymatini B bilan belgilasak, $c = \frac{a+b}{2}$; $d = \sqrt{a \cdot b}$; formulalar o'rinli.



Borland C++ Builder da dasturu:

```
#include <math.h>
```

```
void __fastcall TForm1::Button1Click(TObject *Sender)
```

```
{
```

```
float a,b,s,p;
```

```
a=StrToFloat(Edit1->Text);
```

```
b=StrToFloat(Edit2->Text);
```

```
s=(a+b)/2;
```

```
p=sqrt(a*b);
```

```
Edit3->Text=FloatToStr(s);
```

```
Edit4->Text=FloatToStr(p);
```

```
}
```

Turbo Pascal dasturlash tili: XVII asrda yashagan va dunyoda birinchi xisoblash mashinasini yaratgan frantsuz olimi Blez Paskal sharafiga shunday deb nomlangan PASKAL tili Shvetsariyaning Syurix shahridagi oliy texnika maktabining professori Nikols Virt tomonidan 70-yillarda yaratilgan bulib 1979 yilda standart Paskal deb tasdiklangan. Uzining soddaligi, mantiykiyligi va samaraligi tufayli bu til butun dunyoga tez tarkaldi. Xozirgi paytda barcha xisoblash mashinalari, xususan, mikro EXM lar xam shu tilda ishlash imkoniyatiga ega. Dasturlar matnining tugriligini osonlik bilan tekshirish mumkinligini, ularning ma`nosi yakkol kuzga tashlanishi va oddiyligi bilan ajralib turadi. Paskal tili ancha murakkab va kup vakt oladigan xisob ishlari bajarishga muljallangan tartiblashtirilgan dasturlar tuzishga imkon beradi. Yana bir afzalligi shundan iboratki, foydalanuvchi xatolikka yul kuymasligi uchun yoki xato yozib kuygan bulsa, tez tuzatib olishi uchun dasturda ishlatiladigan uzgaruvchilar oldindan kaysi turga (toifa) mansubligi dasturning barcha elementlari xakida ma`lumot tavsiflash bulimida mujassamlashgan buladi. Operator soni esa minimal darajada kamaytirilgandir. EXMda xisoblash uchun programma asosan ikki kismdan iborat buladi:

berilganlarni tavsiflash (bular ustida amallar bajariladi); amallarni tavsiflash (bu amallar berilganlar ustida bajarilishi kerak). Amallar operatorlar yordamida beriladi, berilganlar esa ta`riflar va tavsiflar yordamida beriladi.

Standart funktsiyalar

Standart funktsiyalar juda kup xollarda masala va misollarni echish uchun dastur tuzayotganda tez-tez uchrab turadi. Masalan: trigonometrik, logarifmik, eksponenta, ildiz chikarish va Xokazolar. Xar bir EXM uchun bunday funktsiyalarni xisoblash dasturi tuzilgan va EXM xotirasiga kiritilib, ular standart funktsiyalar deyiladi. Bu funktsiyalarga murojaat kilinganda avval funktsiyalarni nomlari keyin dumalok kavs ichiga ularning argumentlari yoziladi. Kup ishlatiladigan standart funktsiyalarning ba`zilarini keltiramiz:

$(x) - \text{ABS}(x)$ $\text{ex-EXP}(x)$

x^2 - SQR (x)	$\ln x$ - LN(x)	TRUNC(x)- sonni butun kismga ajratish
		ROUND(x)-sonni yaxlitlash,
x^{100} - EXP(100*LN(X))		PRED(x) -oldingi elementni topish
$\sin x$ - SIN(x)	$\sqrt{x}$ - SQRT(x)	SUCC(x)- keyingi elementini topish
$\cos x$ - COS(x)	$\arctg x$ - ARCTAN(x)	

Ifodalar

Ifoda qiymatlarni hisoblash ketma-ketligini va bajariladigan amallarni aniklaydi. Ular kavs va amal belgilari bilan ajratilgan uzgarmaslardan, uzgaruvchilardan, standart funktsiyalardan iboratdir. Programma strukturasi esa quyidagi oltita bulimdan iborat buladi. Bulimlar albatta quyidagi tartibda joylashgan bulishi kerak: belgilar (metkalar); uzgarmaslar; turlar; uzgaruvchilar; protsedura va funktsiyalar; operatorlar. Paskal tilida suz deb bir nechta belgilar ketma-ketligi tushuniladi. Xizmatchi suzlar deb Paskal tilidagi standart nom tushuniladi. Bu nom maxsus ma`noni anglatadi va uni ma`lumotlarga berib bulmaydi. Maslan, PROGRAM, BEGIN, END va xokazo. Paskal tilidagi ma`lumotlarning elementlari bulib uzgaruvchilar, uzgarmaslar, izoxlar xizmat kiladi. Uzgaruvchilar deb hisoblash jaraenida uz qiymatini uzgartiradigan katta- liklarga aytiladi. Uzgaruvchilarning nomlari (identifikatorlar) - xarflardan yoki xarf va rakmlardan iborat buladi, belgilar soni 8 tadan oshmasligi kerak. Uzgarmaslar (sonst) - deb hisoblash jarayonida uz qiymatini uzgartirmaydigan kattaliklarga aytiladi. Bularga xam uzgaruvchilar kabi nom beradi. Izoxlar - dasturning ma`lum kismini tavsiflash uchun ishlatiladi va bu katorda xech kanday amal bajarilmaydi, ya`ni dasturning biror blokini yaxshirok tushunishga xizmat kiladi. Izox (*,*) simvollar orasida beriladi. Katta *, * va [*,*] lar xam ishlatilishi mumkin. Paskal tilida ma`lumotlarning toifa-lari ikki xil buladi. Oddiy skalyar va murakkab. Skalyar toifa uz navbatida uzgaruvchi va standart toifalarga bulinadi. Uzgaruvchi toifaga kayd kilingan va chegaralangan toifalar kirsas, standart toifaga butun - INTEGER, xakikiy - REAL, mantiqiy - BOOLEAN, belgili - CHAR toifalar

kiradi. Murakkab toifaga esa tup-lamlar, massivlar, yozuvlar, fayllar, belgilar katori kabi toifalar kiradi. INTEGER - BUTUN tipidagi ma'lumotlar fakat butun sonlarni kabul kiladi. Ular Paskal tilida quyidagicha ifodalanadi : 7,10,89 va x.k. REAL - XAKIKIY turdagi ma'lumotlar xakikiy sonlarni kabul kiladi va ikki xil kurinishda buladi:

a) Kuzgalmas nuktali xakikiy sonlar (3.56, 0.88, 150.45)

b) Kuzgaluvchi nuktali xakikiy sonlar (0.546 q 546E - 3
96.78 *10² q96.78E2 va x.k.)

CHAR - BELGI turidagi ma'lumotlar kiymati sifatida barcha belgilarni kabul kiladi. Bu ma'lumotlar apostrof ichiga olib eziladi. Masalan ('F','A','R','5','8' va x.k.). BOOLEAN - MANTIKEY turdagi ma'lumotlar. Bu nom ingliz matematigi Jorj Bul sharafiga shunday deb atalgan va shu turdagi uzgaruvchilar fakat ikkita kiymat kabul kiladi TRUE eki FALSE.

Turbo Pascal dasturlash tilining ish stoli quyidagi rasimda korsatilgan.



Delphi Dasturlash til: 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.

Standart va nostandart matematik funkstiyalar

Funksiya nomi	Yozilishi	Ma'nosi
Sinx	SIN(x)	sinus
Cosx	COS(x)	kosinus
Ln x	LN(x)	natural logarifm
e^x	EXP(x)	eksponenta
$\sqrt{x}$	SQR(x)	kvadrat ildiz
Arctgx	ATAN(x)	arktangensi
x	ABS(x)	moduli
x^2	SQR(x)	kvadrati
a^b	EXP(b*LN(a))	a ning b chi darajasi

	(x)	qiqiy son butun qismi
	und(x)	klitlangan butun son
	(x)	nni matnga o'tkazish
	(x)	tnli sonni raqamga o'tkazish
	r	hvolni kodi bilan chiqarish
	adkey(kod)	os tugmacha belgisini aniqlash

Nostandart matematik funkstiyalar.

$$\begin{aligned}
 &1. \operatorname{Sec} x = \frac{1}{\operatorname{Sin} x}; \quad 2. \operatorname{Cosec} x = \frac{1}{\operatorname{Cos} x}; \quad 3. \operatorname{Tgx} = \frac{\operatorname{Sin} x}{\operatorname{Cos} x}; \quad 4. \operatorname{Arcctgx} = \operatorname{Arctg} \frac{1}{x}; \\
 &5. \operatorname{Arcsin} x = \operatorname{Arctg} \frac{x}{\sqrt{1-x^2}}; \quad 6. \operatorname{Arc} \cos x = \operatorname{Arctg} \frac{\sqrt{1-x^2}}{x}; \quad 7. \operatorname{Arcsec} x = \operatorname{Arctg} \frac{1}{\sqrt{1-x^2}}; \\
 &8. \operatorname{Arccosec} x = \operatorname{Arctg} \sqrt{1-x^2}; \quad 9. \operatorname{Log}_a b = \frac{\operatorname{Lnb}}{\operatorname{Lna}}; \quad 10. \operatorname{Padian} = \frac{\operatorname{Gradius} \cdot \pi}{180}
 \end{aligned}$$

Dasturda arifmetik va mantiqiy ifodalar o'zgaruvchi, o'zgarmas, standart funkstiyalar, 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'zgaruchining 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 operastiyani 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), prosteduralar va funkstiyalarni 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>;

<Prostedura va funkstiyalarni aniqlash>

Begin

<Bajariladigan operatorlar ketma ketligi>

End.

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 klassifikastiya qilish mumkin.

- ❖ Oddiy
- Tartibli
- Butun
- Belgi

- Mantiqiy
- Sanoqli (Perechislyаемы)
- Chegaralangan
- Haqiqiy
- ❖ Qator
- ❖ Struktura
- To'plam
- Massiv
- Yozuv
- Fayl
- Klass
- Interfeys
- ❖ Ko'rsatgichli
- ❖ Prostedurali
- ❖ 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'zgprish diapazoni	O'lcham (baytda)
eger	-2147483648..2147483647	4
rdinal	0..4294967295	4
rtint	-128..127	1
allint	-32768..32767	2
hgint	-2147483648..2147483647	4

64	$-2^{63}..2^{63}-1$	8
te	0..255	1
ord	0..65535	2
ngWord	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)
al	$5.0*10^{-324}..1.7*10^{308}$	8
al48	$2.9*10^{-39}..1.7*10^{38}$	6
gle	$1.5*10^{-45}..3.4*10^{38}$	4
uble	$5.0*10^{-324}..1.7*10^{308}$	8
ended	$3.6*10^{-4951}..1.1*10^{4932}$	10
mp	$-2^{63}+1..2^{63}-1$	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)
ar	1
SChar	1

deChar	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	D'lcham (baytda)
olean	1
teBool	1
ordBool	2
ngBool	4

Ma'lumotlarni kiritish va chiqarish operatorlari

Biror bir masalani echishning 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 erda 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 erda 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 erda 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 echimi 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 erda 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 echim 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.

Delphida konsol ilovasini yaratish

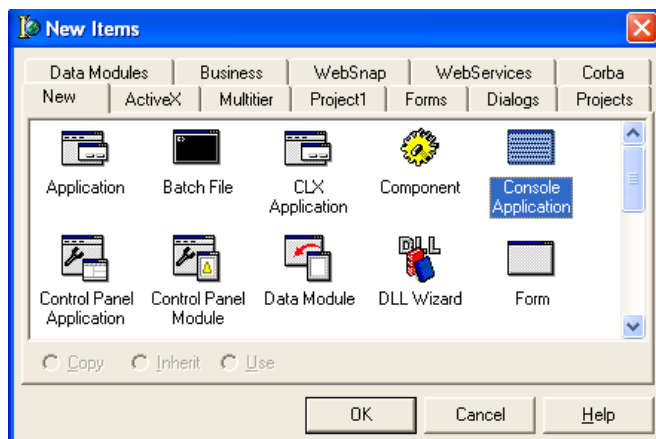
Delphida konsol ilovalarini har xil usullarda yaratish mumkin. Ulardan eng oson usuli quyidagicha:

1.Delphi muhiti ishga tushiriladi.

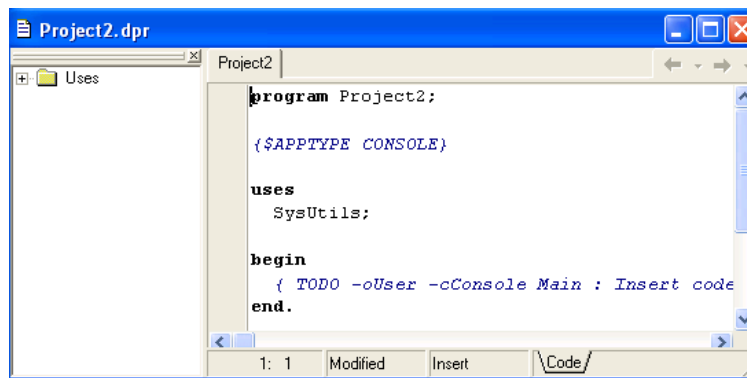
Pusk=>Программы=>Borland Delphi

2.Bosh menyudan File punktini ochib u erdan 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.

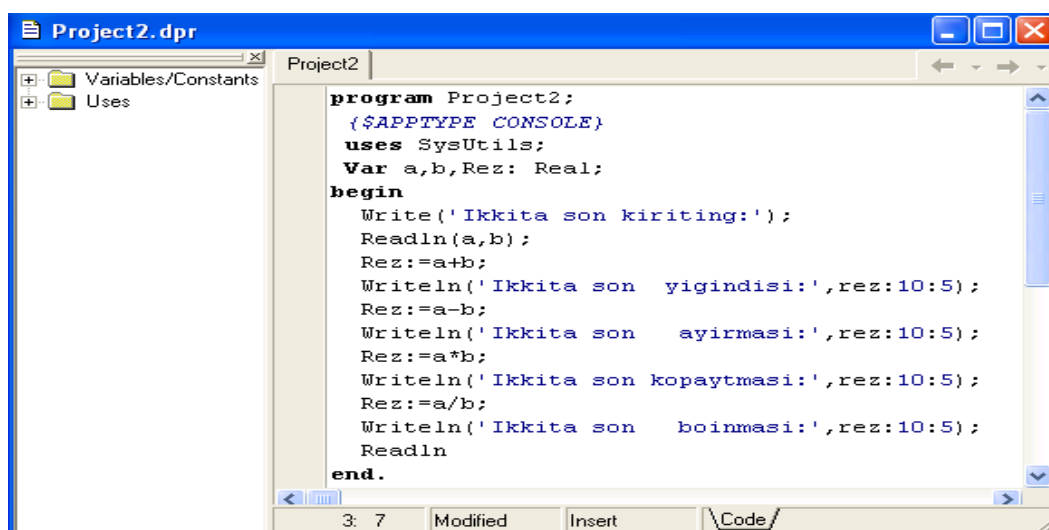


4.Natijada ekranda loyiha oynasi ochiladi (.dpr kungaytmali nom bilan).



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 erda 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 funkstional tugmachasini bosish kerak bo'ladi. Dastur normal ishga tushgandan so'ng ekranda DOSning standart dastur oynasi namayon bo'ladi.



Xulosa

Bugungi kunga kelib axborot texnologiyalari juda shiddad bilan rivojlanib ketmoqda. Yuqori darajali dasturlash tillarini kunnan kunga har xil versialari oylap topilmoqda. Bundan maxsad insonlarning dastur tuzish paytida eng birinshi orinda vaqtini tejash undan keyin esa qulaylilik yaratish hisoplanadi. Bugungi kunga kelip **C++**, **Delphi**, **Pascal** dasturlash tillari mavjud ekan. Bu dasturlash tillari orqali axborot texnologiyalari boyisha harqanday maumualarni hal etish mumkun.

Delphi dasturlash tilining asosini esa Pascal dasturlash tili tashkil etar ekan. Buning sababi esa Pascal tilini takomillashtirish bolip hisoplanadi. Bu takomillashtirish natijasida esa yangi bir Delphi dasturlash tili juvudga keldi. Bu yuqori darajali dasturlash tillari ishida C++ dasturlash tili juda kata ahamiyatga ega ekan. Saba bu dasturlash tilida ishlash ansha qulay eng ahamiyatliysi C++tili mashina kodiga juda yaqin hisoplanadi.

Amaliy ish

Borland C++ Builder tilida bir misol korip shiqaylik.

```
//-----  
#include <vcl.h>  
#pragma hdrstop  
#include "Unit1.h"  
#include <math.h>  
//-----  
#pragma package(smart_init)  
#pragma resource "*.dfm"
```



```
TForm1 *Form1;
```

```
//-----
```

```
__fastcall TForm1::TForm1(TComponent* Owner)
```

```
    : TForm(Owner)
```

```
{
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
```

```
{
```

```
    double x, y;
```

```
    const pi=3.14;
```

```
    x=StrToFloat(Edit1->Text);
```

```
    if(x<0.5)
```

```
    {
```

```
        y=0.5*pow(x,2)/pow((x+0.5),2);
```

```
    }
```

```
    else
```

```
    {
```

```
        y=sqrt(pow(x,2)+3.5*pow(x,3));
```

```
    }
```

```
    Label2->Caption="y="+FloatToStr(y);
```

```
}
```



```

TForm1 *Form1;

//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
}

//-----

void __fastcall TForm1::BitBtn1Click(TObject *Sender)
{
    double k, n, s=0, s1=1;
    for(int k=1; k<=4; k++)
    {
        for(int n=2; n<=5; n++)
        {
            s1=s1+pow((-k),n)*log(pow(n,k))/(2*k+n);
        }
        s=s+s1;
    }

    Memo1->Lines->Add("2-Ėàáîðàòîðèÿÿüê æóîûñ");
    Memo1->Lines->Add("s="+FloatToStr(s));
}

```

Foydalanilgan adabiyotlar

1. Бьярн Страустрап. Введение в язык C++
2. Крис Паппас, Уильям Мюррей. Программирование на C и C++
“Ирина”, BHV, Киев 2000г
3. Фридман Александр Львович, Язык программирования C++
4. Aripov M.M., Imomov T., Irmuhamedov Z.M. va boshqalar. Informatika. 5. Axborot texnologiyalari. Toshkent, 1-qism. 2002, 2-qism. 2003

Foydalanilgan elektron saytlar:

<http://www.infocity.kiev.ua/>

<http://www.intuit.ru/>

www.google.uz

www.ziyonet.uz

www.elivrary.tuit.uz