

O'ZBEKISTON RESPUBLIKASI ALOQA, AXBOROTLASHTIRISH
VA TELEKOMUNIKATSIYA DAVLAT QO'MITASI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI
FARG'ONA FILIALI

”Axborot texnologiyalari” kafedrası

”C++ da dasturlash”
fanidan

KURS ISHI

Bajardi:

611-12 Guruh talabasi

Shuxratov Dostonbek

Qabul qildi:

Bazarbayev M.

Hashimov A

Farg'ona-2015

C++ dasturlash tili va massivlar haqida
tushuncha.

REJA:

1. Kirish.

2. Nazariy qism:

a. C++ tarixi va asosiy tushunchalari.

b. Massivlar haqida tushuncha.

c. Dasturiy qism.

3. Xulosa.

4. Foydalanilgan adabiyotlar.

5. Ilova.

Kirish

Insoniyat tarixining ko'p asrlik tajribasi ezgu go'yalardan va sog'lom mafkura xamda zamonaviy bilimlardan maxrum xar qanday jamiyat uzoqqa bora olmasligini ko'rsatdi. SHuning uchun, mustakillikka erishgan mamlakatimiz o'z oldiga ozod va obod Vatan, demokratik jamiyat barpo qilish, erkin va farovon xayot kurish, rivojlangan mamlakatlar katoridan o'rin olish kabi muxim vazifalarni qo'ydi. Bu vazifalarni xal qilish asosan yosh avlod zimmasiga tushadi.

Yoshlarni kelajak jamiyatning faol quruvchilari bo'lishi uchun ularni fan va texnikaning eng ilg'or yutuqlari xamda kuchli bilimlar bilan qurollantirish, olingan bilimlarni amaliyotda qo'llay bilish ana shu yo'ldagi eng muhim talablardan biri xisoblanadi. Bu narsa ayniqsa EXM bilan aloqador kundalik masalalarni yechishda yaqqol ko'rinadi. Demak, yoshlardan zamonaviy EXM lar bilan ishlashni o'rganish, xalk xo'jaliginining turli masalalarini yechishga mo'ljallangan dasturiy ta'minot bilan tanishish xamda dasturlash vositalari yordamida xali EXM da yechilmagan masalalar uchun yangi dasturlar yaratishni talab kilinadi.

Zamonaviy EHMlar (kompyuterlar) o'zlarining avvalgi avlodlari kabi ishlash chog'ida eng oddiy arifmetik va mantiqiy amallarni bajara olar edi xolos. Mashina murakkab masalani echishi uchun uning xotirasiga mos dastur kiritilgan bo'lishi kerak. Dastur buyruqlar ketma-ketligidan, ko'rsatmalar – biror oddiy amalni bajarishga oid mashinaga beriladigan ko'rsatmalardan iborat bo'lib, ancha murakkab amalni ifodalaydi.

EHM larga dastur-topshiriq yozish uchun istalgan dasturlash tilini tanlab olish mumkin.

Nazariy qism.

C++ tarixi va asosiy tushunchalari

Si dasturlash tili Unix operatsion tizimi bilan bog'langan bo'lib, shu tizimda ishlatiladigan bir qancha dasturlar Si tilida yaratilgan. Si dasturlash tili mashinaning barcha imkoniyatlaridan to'laligicha foydalanishga imkon beradi.

Si tili mo'ljallanishiga ko'ra umumiy bo'lib, u yuqori darajali dasturlash tillari bilan Assembler orasida oraliq vaziyatni egallaydi.

Dasturni boshqaruvchi operatorlar strukturali dasturlash talablariga javob beradi. Unda kiritish chiqarish, xotirani dinamik taqsimlash, multidasturlash, parallel hisoblash vositalari yo'q. Bu vositalar tashqi funksiyalar orqali amalga oshiriladi.

Si tilida dastur tuzish hisoblash mashinalari ishini va uning operatsion tizimini tushunish imkoniyatini yaratadi, shuning uchun ham u kasbiy dasturchilarga mo'ljallangan.

Lekin, Si dasturlash tili biron-bir tizim yoki mashina bilan qattiq bog'lanib qolmagan.

Si tilida dasturlar ma'lumotlar ustida natija olish maqsadida qilinadigan harakatni bildiradi. Dasturda xarakatlar operatorlar orqali beriladi, malumotlar esa ob'ektlarni aniqlash va tavsiflash orqali amalga oshiriladi.

Dasturda foydalaniladigan har bir ob'ekt tavsiflanishi zarur. Tavsiflash ob'ekt bilan bir necha xarakteristikalarini bog'laydi. Bu xarakteristikalar: tip, belgilash, xotira sinfi, harakat sohasi, boshlang'ich qiymatlar kabilardir.

Tilda asosiy tiplar: belgi (liter) bilan birgalikda butun va suzuvchi vergulli sonlar ishlatiladi. Bundan tashqari ko'rsatkichlar, massivlar, operandalardan chiqariluvchi ma'lumotlarning to'la versiyasini hosil qilish mumkin.

Ifoda operandalar va ko'rsatkichlardan tashkil topadi. Qiymat uzatish funksiyasini chaqarishni qo'shib hisoblaganda, har bir ifoda ko'rsatma bo'lishi mumkin.

Ko'rsatkichlar bog'liqmas adresli mashina arifmetikasini belgilaydi. Si da ko'rsatma tuzuvchi ({}), shart bo'yicha tarmoqlanish (if), ko'pdan biron bir muqobillarini (switch) tanlash, yuqoriga qarab takrorlash (for, while) quyiga qarab takrorlash (do), shuningdek takrorlashni to'xtatish (break) kabi boshqaruvchi tuzilmalar mavjud.

Mazkur tilning muayyat kamchiligi ham mavjud bo'lib, til tuzilmalari sintaksisi noqulay.

C dasturlash tili birinchi marta Denis Ritchi («BELL LABORATORIES» AQSH) tomonidan 1972 yilda loyihalashtirilgan. Bu tilda skalyar qiymatlar bilan ham ish olib borib, ular o'rtasida operatsiyalar bajarish mumkin. Taqribiy qiymatlar esa elementlarga yaqinlashishni tartiblash uchun ishlatiladi.

Bir o'lchovli massivlar.

Massivlarni ta'riflash. Bir o'lchovli massivlar.

Massiv bu bir tipli nomerlangan ma'lumotlar jamlanmasidir. Massiv indeksli o'zgaruvchi tushunchasiga mos keladi. Massiv ta'riflanganda tipi, nomi va indekslar chegarasi ko'rsatiladi. Misol uchun: long int a[5]; char w[200]; double f[4][5][7]; char[7][200]. Massiv indekslar xar doim 0 dan boshlanadi. Si tili standarti bo'yicha indekslar soni 31 tagacha bo'lishi mumkin, lekin amalda bir o'lchovli va ikki ulchovli massivlar qo'llaniladi. Bir ulchovli massivlarga matematikada vektor tushunchasi mos keladi. Massivning int z[3]

shakldagi ta'rifi, int tipiga tegishli z[0],z[1],z[2] elementlardan iborat massivni aniklaydi.

Massivlar ta'riflanganda initsializatsiya qilinishi, ya'ni boshlang'ich qiymatlarlari ko'rsatilishi mumkin. Misol uchun:

```
float C[]={1,-1,2,10,-12.5};
```

Bu misolda massiv chegarasi avtomatik aniqlanadi. Agar massiv initsializatsiya qilinganda elementlar chegarasi ko'rsatilgan bulsa , ro'yxatdagi elementlar soni bu chegaradan kam bo'lishi mumkin, lekin ortik bo'lishi mumkin emas.

Misol uchun :

int A[5]={2,-2}. Bu xolda a[0] va a[1] qiymatlari aniqlangan bo'lib, mos xolda 2 va -2 ga teng.

Massivda musbat elemenlar soni va summasini xisoblash.

```
# include <iostream.h>;
```

```
# include <conio.h>;
```

```
Main()
```

```
{
```

```
Int x[]={-1;2;5;-4;8;9};
```

```
Clrscr();
```

```
For (int s=0,int k=0, int I=0; I<6; I++)
```

```
{
```

```
    If (x[I]<=0) continue;
```

```
    k++;s++; }
```

```
    Cout<<k;
```

```
    Cout<<s;
```

```
    getch( );
```

```
}
```

```
};
```

Massivning eng katta, eng kichik elementi va o'rta qiymatini aniqlash:

```
#include <iostream.h>

Void main( )
{
    Int I,j,n;
    Float a,b,d,x[100];
    While(1)
    {
        Cout<<("\n n="); Cin>>n;
        If ( n>0 && n <= 100 ) break;
        Cout<<("\n Xato 0<n<101 bulishi kerak");
    }
    Cout<<("\n elementlar kiymatlarini kiriting:\n");
    For (i=0;i<n;i++)
    {
        Cout<<"x["<<i<<"]=""; Cin>>x[i];}
        max=x[0];min=x[0];
        For (s=0,i=0;i<n;i++)
        { s++;
            If (max<x[i]) max=x[i];
            If (min>x[i]) min=x[i]; }
        s/=n;
        Cout<<"\n max="<<max;
        Cout<<"\n min="<<min;
        Cout<<"\n urta kiymat="<<s;
```

}

Ikki o'lchovli massivlar.

Jadvallar bilan ishlash

Ikki o'lchovli massivlar matematikada matritsa yoki jadval tushunchasiga mos keladi. Jadvallarning insializatsiya qilish qoidasi, ikki o'lchovli massivning elementlari massivlardan iborat bo'lgan bir o'lchovli massiv ta'rifiga asoslangandir.

Misol uchun ikki qator va uch ustundan iborat bo'lgan xaqiqiy tipga tegishli d massiv boshlang'ich qiymatlari quyidagicha ko'rsatilishi mumkin:

```
float d[2][3]={(1,-2.5,10),(-5.3,2,14)};
```

Bu yozuv quyidagi qiymat berish operatorlariga mosdir:

```
d[0][0]=1;d[0][1]=-2.5;d[0][2]=10;d[1][0]=-5.3;d[1][1]=2;d[1][2]=14;
```

Bu qiymatlarni bitta ro'yxat bilan xosil qilish mumkin:

```
float d[2][3]={1,-2.5,10,-5.3,2,14};
```

Initsializatsiya yordamida boshlang'ich qiymatlar aniqlanganda massivning xamma elementlariga qiymat berish shart emas.

Misol uchun: `int x[3][3]={(1,-2,3),(1,2),(-4)}.`

Bu yozuv quyidagi qiymat berish operatorlariga mosdir:

```
x[0][0]=1;x[0][1]=-2;x[0][2]=3;x[1][0]=-1;x[1][1]=2;x[2][0]=-4;
```

Initsializatsiya yordamida boshlang'ich qiymatlar aniqlanganda massivning birinchi indeksi chegarasi qo'rsatilishi shart emas, lekin qolgan indekslar chegaralari ko'rsatilishi shart.

Misol uchun:

Double x[][2]={(1.1,1.5),(-1.6,2.5),(3,-4)}

Bu misolda avtomatik ravishda qatorlar soni uchga teng deb olinadi. Kuyidagi ko'radigan misolimizda jadval kiritilib xar bir katorning maksimal elementi aniqlanadi va bu elementlar orasida eng kichigi aniklanadi:

```
#include <iostream.h>
```

```
void main()
```

```

{
    double a[4,3]; double s,max=0.0,min=0.0;
    int i,j;
    for(i=0;i<4;i++)
    {
        for(j=0;j<3;j++)
        {
            Cout<<"a["<<i<<"][<<j<<"]=";Cin>>s;a[i,j]=s;
            if (max<s) max=s;
        };
        Cout<<"\n";
        if (max<min) min=max;
    }
    Cout<<"\n min=",min;
}

```

Massiv elementlariga son qiymat berishda kompyuter xotirasidagi tasodifiy butun sonlardan foydalanish ham mumkin. Buning uchun standart kutubxonaning `rand ( )` funksiyasini ishga tushirish kerak. `rand ( )` funksiyasi yordamida `0 † 32767` oralikdagi ixtiyoriy sonlarni olish mumkin. Bu qiymatlar umuman tasodifiydir. (psevdo – tasodifiy degani). Agar dastur qayta-qayta ishlatilsa, aytish tasodifiy qiymatlar takrorlanaveradi. Ularni yangi tasodifiy qiymatlar qilish uchun `srand ( )` funksiyasini dasturda bir marta e’lon qilish kerak. Dastur ishlashi jarayonida ehtiyojga qarab `rand ( )` funksiyasi chaqirilaveradi. Tasodifiy qiymatlar bilan ishlash uchun `<stdlib.h>` faylini e’lon qilish zarur. `srand ( )` funksiyasidagi qiymatni avtomatik ravishda o’zgaradigan xolatga keltirish uchun `srand ( time (NULL))` yozish ma’kul, shunda kompyuter ichidagi soatning qiymati `time ( )` funksiyasi yordamida o’rnatiladi va `srand` ga parametr sifatida beriladi. `NULL` yoki `0` deb yozilsa, qiymat sekundlar ko’rinishida beriladi. Vaqt bilan ishlash uchun `<time.h>` ni e’lon qilish kerak.

```

# include <iostream.h>

```

```

# include <conio.h>

# include <stdlib.h>

# include <time.h>

void main ( )
{
    srand ( time (0));
    int a[5], b[5], i;
    for (i = 0; i < 5; i++)    a[i] = rand ( );
    for (i = 0; i < 5; i++)
    {
        b[i] = a[i] + 64;
        cout << "b="<<b[i]<<endl;
    }
        getch ( );
    }

```

Izox: tasodifiy sonlar ichida manfiy sonlarning ham qatnashishini ixtiyoriy aytsak, $a[i] = 1050 - \text{rand}()$; èki $a[i] = \text{rand}() - 1000$; deb yozish ham mumkin.

2-misol. 2ta matritsa berilgan. Ularni o‘zaro ko‘paytirib yangi matritsa hosil qiling. Bu yerda 1-matritsaning ustunlar soni 2-matritsaning satrlar soniga teng bo‘lishi kerak.

Boshlanish

Kiritish

A, V

$i = 1, n$

$J = 1, m$

$$C_{ij} = 0$$

$$K = 1, n$$

$$C_{ij} = C_{ij} + A_{ik} * B_{kj}$$

$$C_{ij}$$

Tamom

```
# include <iostream.h>
```

```
# include <conio.h>
```

```
# include <stdlib.h>
```

```
# include <time.h>
```

```
void main ( )
```

```
{
```

```
    srand ( time (0));
```

```
    int a[3][3], b[3][3],c[3][3], i, j, k;
```

```
    for (i=0; i<3; i++)
```

```
    for (j=0; j<3; j++)
```

```
    a[i][j] = rand ( );
```

```
    for (i=0; i<3; i++)
```

```
    for (j=0; j<3; j++)
```

```
    b[i][j] = rand ( );
```

```
    for (i=0; i<3; i++)
```

```
{
```

```
    for (j=0; j<3; j++)
```

```
{
```

```
    c[i][j] = 0;
```

```
    for (k=0; k<3; k++)
```

```
    c[i][j] = c[i][j] + a[i][k]*b[k][j];
```

```
    cout <<"c="<<c[i][j]<<"\t";
```

```

}
cout << endl;
}
getch ();
}

```

Simvolli massivlar.

Si tilida satrlar simvolli massivlar sifatida ta'riflanadi.

Simvolli massivlar quyidagicha tasvirlanishi mumkin: Char pas[10];

Simvolli massivlar quyidagicha initsializatsiya qilinadi:

Char capital[]="TASHKENT"; Bu xolda avtomatik ravishda massiv elementlari soni aniklanadi va massiv oxiriga satr ko'chirish „\n“ simvoli qo'shiladi.

Yuqoridagi initsializatsiyani quyidagicha amalga oshirish mumkin:

```
Char capital[]={„T“,„A“,„S“,„H“,„K“,„E“,„N“,„T“,„\n“};
```

Bu xolda suz oxirida „\n“ simvoli anik kursatilishi shart.

Misol uchun : palindrom masalasini ko'rib chiqamiz. Palindrom deb oldidan ham oxiridan ham bir xil o'qiladigan so'zlarga aytiladi. Misol uchun non. Dasturda kiritilgan suz palindrom ekanligi aniqlanadi:

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
    gets(a);
```

```
    for( int j=0, a[j]!='\0';j++);
```

```
    I=0;
```

```
    while(I<j)
```

```

        if (a[I++]!=a[j--]) break;
        if ((j-I)>1)
Cout<<("Palindrom emas") else Cout<<("Palindrom");
}

```

Keyingi misolimizda kiritilgan so'zdan berilgan xarf olib tashlash dasturi berilgan:

```

#include <iostream.h>
void main()
{
    char s[100];
    int c;
    cin>>s;
    int i, j;
    for ( i = j = 0; s[i] != '\0'; i++)
if ( s[i] != c )
    s[j++] = s[i];
    s[j] = '\0';
cout<<s;
}

```

Xar gal 's' dan farqli simvol uchraganda , u J pozitsiyaga yoziladi va faqat shundan so'ng J qiymati 1 ga oshadi. Bu quyidagi yozuvga ekvivalent:

```

if ( s[i] != c )
    s[j] = s[i];
    j++;

```

So'zlar massivlari.

Si tilida so'zlar massivlari ikki o'lchovli simvolli massivlar sifatida ta'riflanadi. Misol uchun:

```
Char Name[4][5].
```

Bu ta'rif yordamida xar biri 5 ta xarfdan iborat bo'lgan 4 ta so'zli massiv kiritiladi. So'zlar massivlari quyidagicha initsializatsiya qilinishi mumkin:

```
Char Name[3][8]={“Anvar”,”Mirkomil”,”YUsuf”}.
```

Bu ta'rifda xar bir so'z uchun xotiradan 8 bayt joy ajratiladi va xar bir so'z oxiriga „\0“ belgisi qo'yiladi.

So'zlar massivlari initsializatsiya qilinganda so'zlar soni ko'rsatilmasligi mumkin. Bu xolda so'zlar soni avtomatik aniqlanadi:

```
Char comp[][9]={“kompyuter”,”printer”,”kartridj”}.
```

Quyidagi dasturda berilgan xarf bilan boshlanuvchi so'zlar ro'yxati bosib chiqariladi:

```
#include <iostream.h>
```

```
void main()
```

```
{
```

```
char a[10][10];
```

```
char c;
```

```
for (int i=0;i<10;i++)
```

```
cin>>a[i];
```

```
cin>>c;
```

```
for (i=0;i<10;i++)
```

```
if (a[i][0]==c) cin>>a[i];
```

```
}
```

Quyidagi dasturda fan nomi, talabalar ro'yxati va ularning baxolari kiritiladi. Dastur bajarilganda ikki olgan talabalar ruyxati bosib chiqariladi:

```
#include <iostream.h>

void main()
{
    char a[10][10];
    char s[10]; int k[10];
    cin>>s;
    for (int i=0;i<10;i++)
        cin>>a[i];
    for (i=0;i<10;i++)
        {Cin>>k[i]};
    for (int i=0;i<10;i++) if (k[i]==2) cout<<a[i];
}
```

Ko'rsatkichlar massivlari.

Ko'rsatkichlar massivlari quyidagicha ta'riflanadi

<tip> *<nom>[<son>]

Misol uchun int *pt[6] ta'rif int tipidagi ob'ektlarga olti elementli massivni kiritadi.

Ko'rsatkichlar massivlari satrlar masssivlarini tasvirlash uchun qulaydir.

Misol uchun familiyalar ruyxatini kiritish uchun ikki o'lchovli massivdan foydalani kerak. char

fam[][20]={“Olimov”,”Raximov”,”Ergashev”}

Xotirada 60 elementdan iborat bo'ladi, chunki har bir familiya gacha 0 lar bilan to'ldiriladi.

Ko'rsatkichlar massivi yordamida bu massivni quyidagicha ta'riflash mumkin.

Char *pf[]= {"Olimov"},"Raximov"},"Ergashev"}.

Bu xolda ro'yxat xotirada 23 elementdan iborat bo'ladi, chunki xar bir familiya oxiriga 0 belgisi qo'yiladi

Ko'rsatkichlar massivlari murakkab elemenlarni sodda usulda tartiblashga imkon beradi.

Quyidagi misolda matritsa satrlari birinchi elementlari o'sishi tartibida chiqariladi. Bu misolda yordamchi kursatkichlar massivi yaratilib shu massiv tartiblanadi va massiv asosida matritsa elementlari chiqariladi.

```
#include <iostream.h>

void main()
{
    int n=2; int m=3;
    array[][3]={(1,3,5),(3,1,4),(5,7,1)};
    int *pa[n];
    for (i=0;i<n;i++) pa[i]=(int *)&a[i];

    for (i=0;i<n-1;i++)
    {
        for (int k=i+1;k<n;k++)
            if a[i][1]>a[k][1]
            {
                int *pp=pa[i];
                pa[i]=pa[k];pa[k]=pp;};

    for (i=0;i<n;i++)
    {
        Cout<<'n'<<i+1;
        for (int j=0;j<magistr;j++)
            Cout<<pa[i][j];}
    };
};
```

Ko'rsatkichlar massivlari funksiyalarda matritsalar qiymatlarini o'zgartirish uchun mumkin. Quyidagi misolda matritsani transponirlash funksiyasi ishlatiladi.

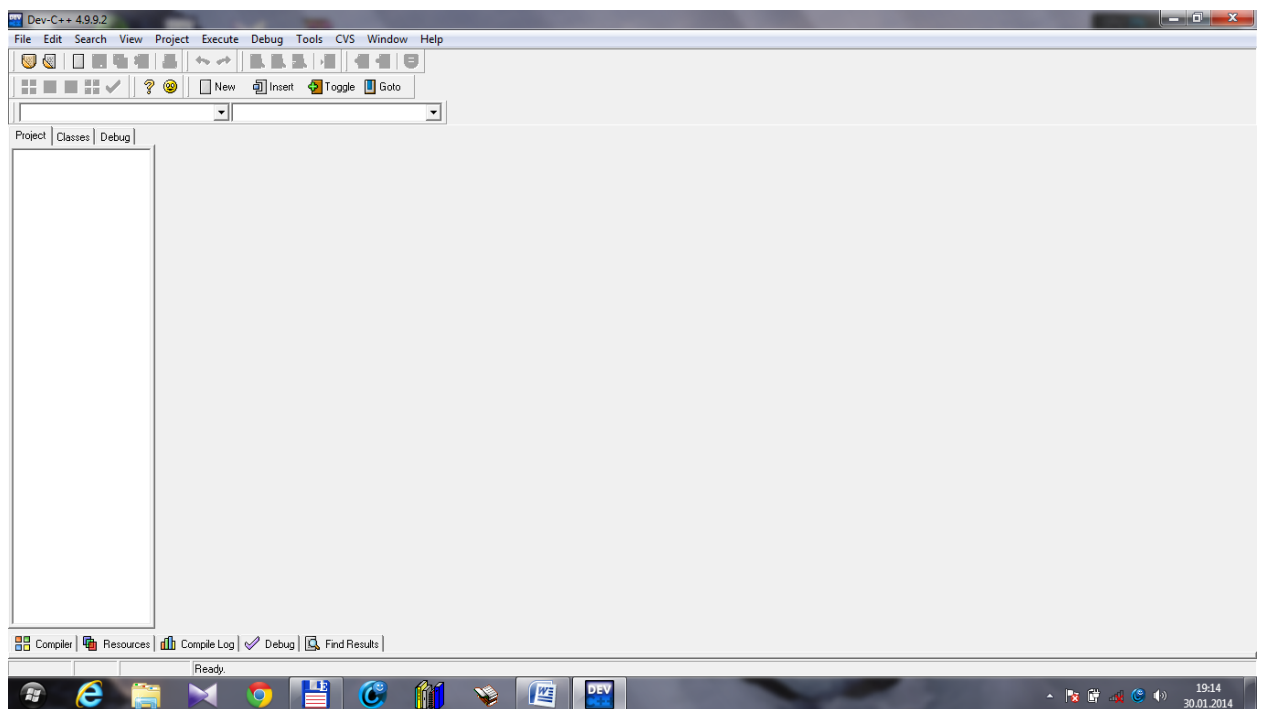
```
Void trans(int n,double *p[]);
{
    double x;
    for (int i=0;i<n-1;i++)
        for (int j=i+1;j<n;j++)
        {
            x=p[i][j];
            p[i][j]=p[j][i];
            p[j][i]=x;
        }
    };
void main()
{
    double a[3,3]={11,12,13,21,22,23,31,32,33};
    double ptr={{(double*)&a[0], (double*)&a[1], (double*)&a[2]};
    int n=3;
    trans(n,ptr);
    for (int i=0;i<n;i++)
    {
        Cout<<'n'<<,i+1;
        for (int j=0;j<n;j++)
            Cout<<'n'<<a[i][j]);
    } };
```

Dasturiy qism.

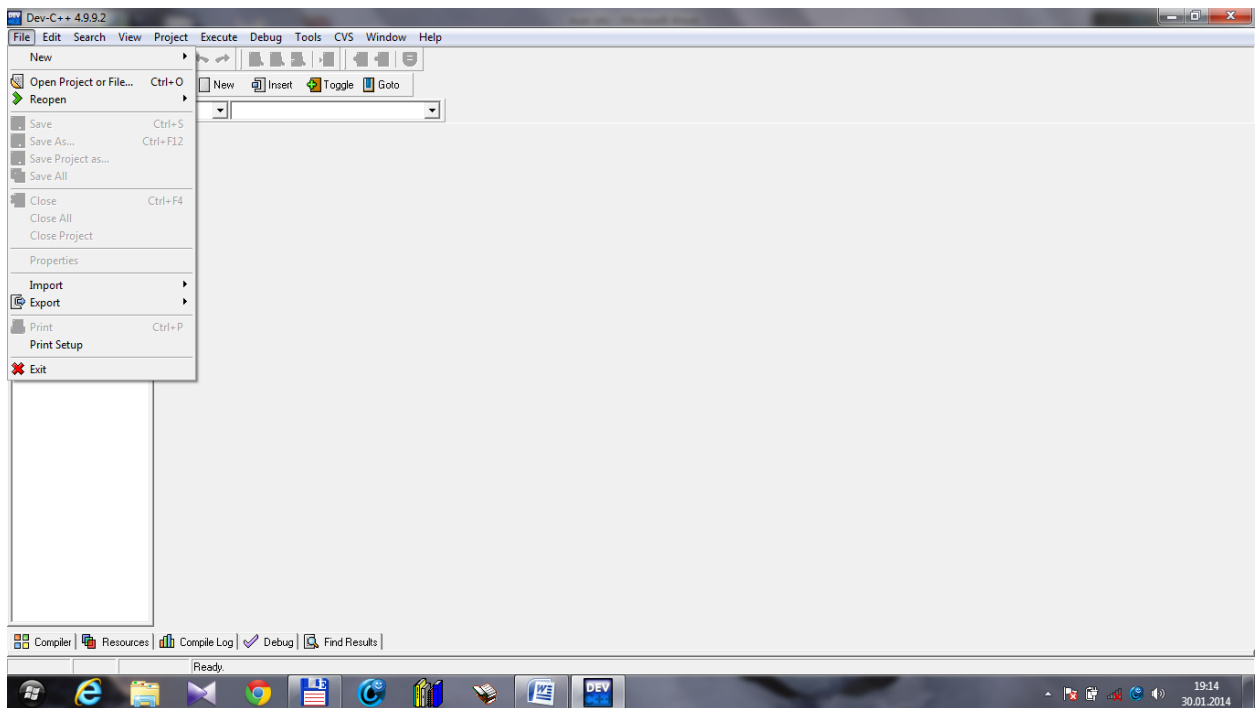
Masala: M(9) massivda qandaydir natural sonning I-sanoq sistemasida raqamlar yozilgan. (M(1)-birlar xonasi va hakazo). Birlar xonasidan boshlab turib , J-sanoq sistemasida bu son raqamlarini chop eting .I,J sonlar 10 dan oshmaydi.

Bu masalani yechish uchun bizga DEV C++ dasturi kerak bo'ladi.DEV C++

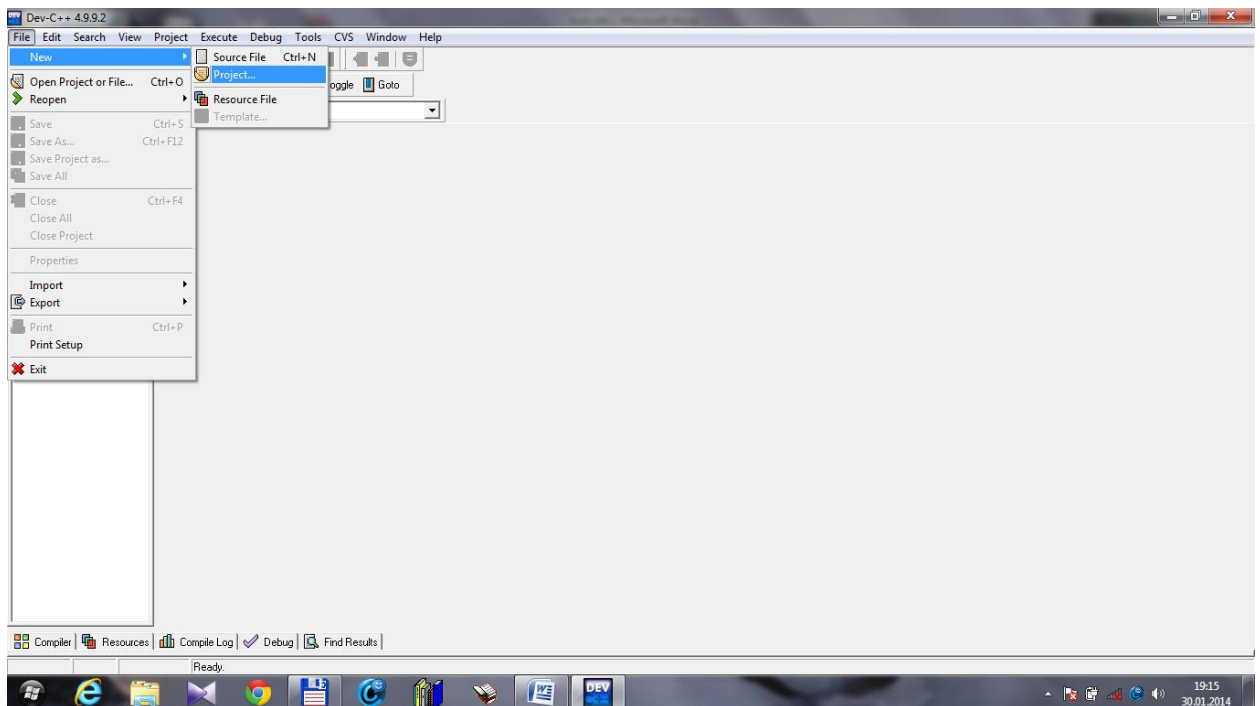
dasturini kompyuterimizga o'rnatib,  manashu yarliqni oxtarib topamiz.Bu yarliq shu DEV C++ dasturini belgisini bildiradi. Biz bu yarliqni ustiga borib, ya'ni sichqoncha orqali strelka ni olib borib ikki marta sichqonchani chap tugmasini tez-tez bosamiz. Shunda dastur ishga tushadi. Dasturni ishga tushirganimizda quyidagi holatda bo'ladi.



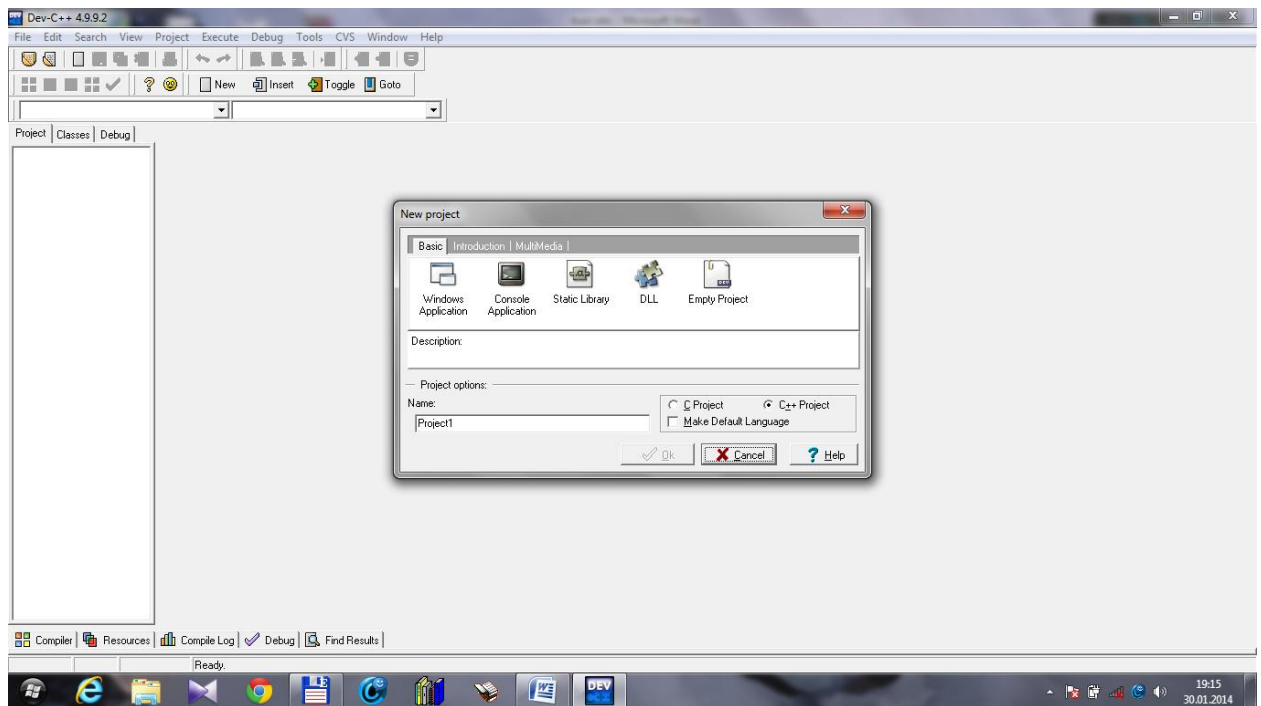
DEV C++ dasturini ishga tushirgach sarlovha satri, menyular satri va asosiysi instrumentlar paneli joylashgan bo'ladi.Biz bu dasturdan menyular qatoridagi "file" menyusini tanlaymiz.



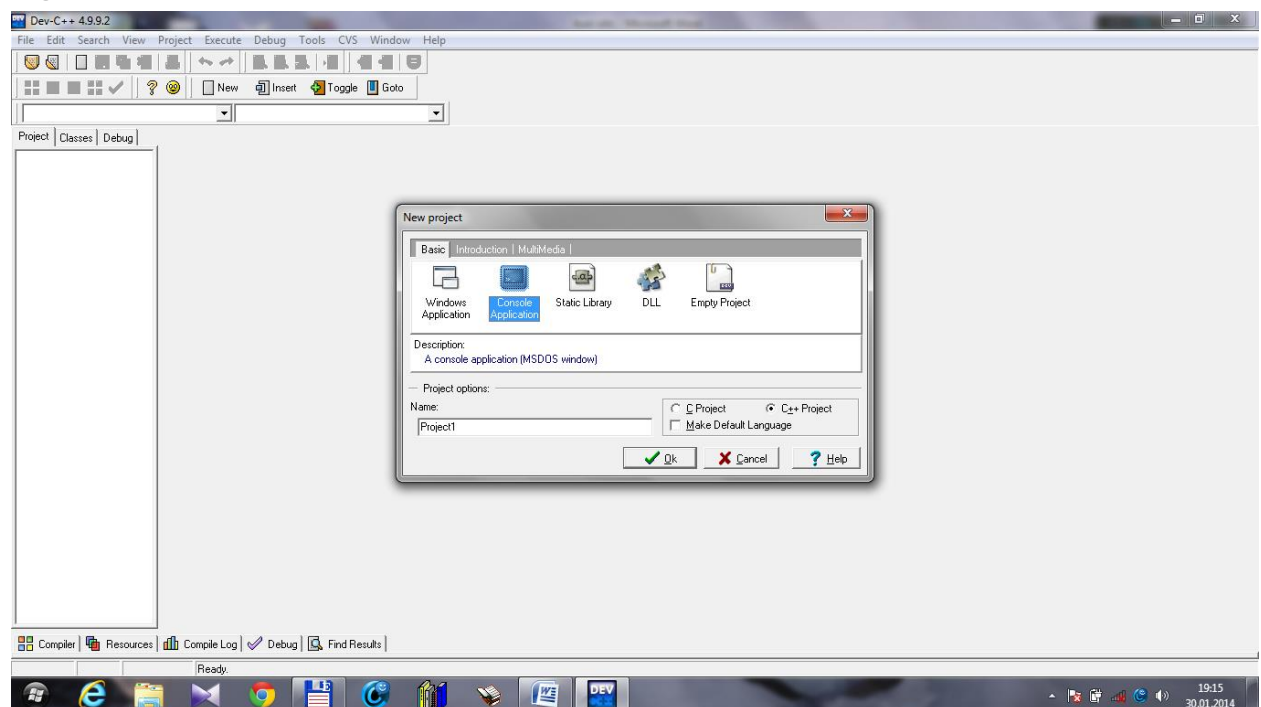
“file” menyusida bir qancha yozuvlar chiqadi. Biz yozuvlardan “new”(yangi) qatoriga o’tamiz, bizga yana yozuvlar taklif qilinadi. Biz “project” bo’limini tanlaymiz.



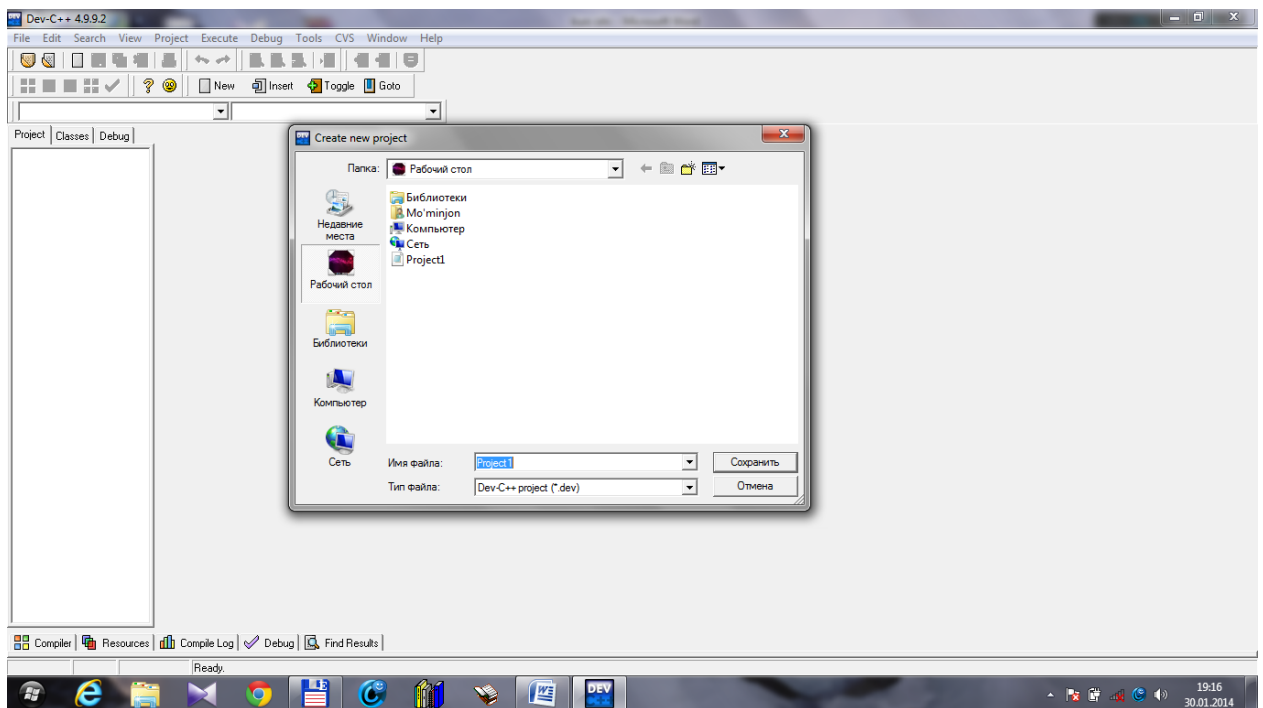
Project bo’limini tanlaganimizda “new project” oynasi ochiladi. Bu oynada qaysi programma tuzish kerak bo’lsa shuni tanlaymiz.



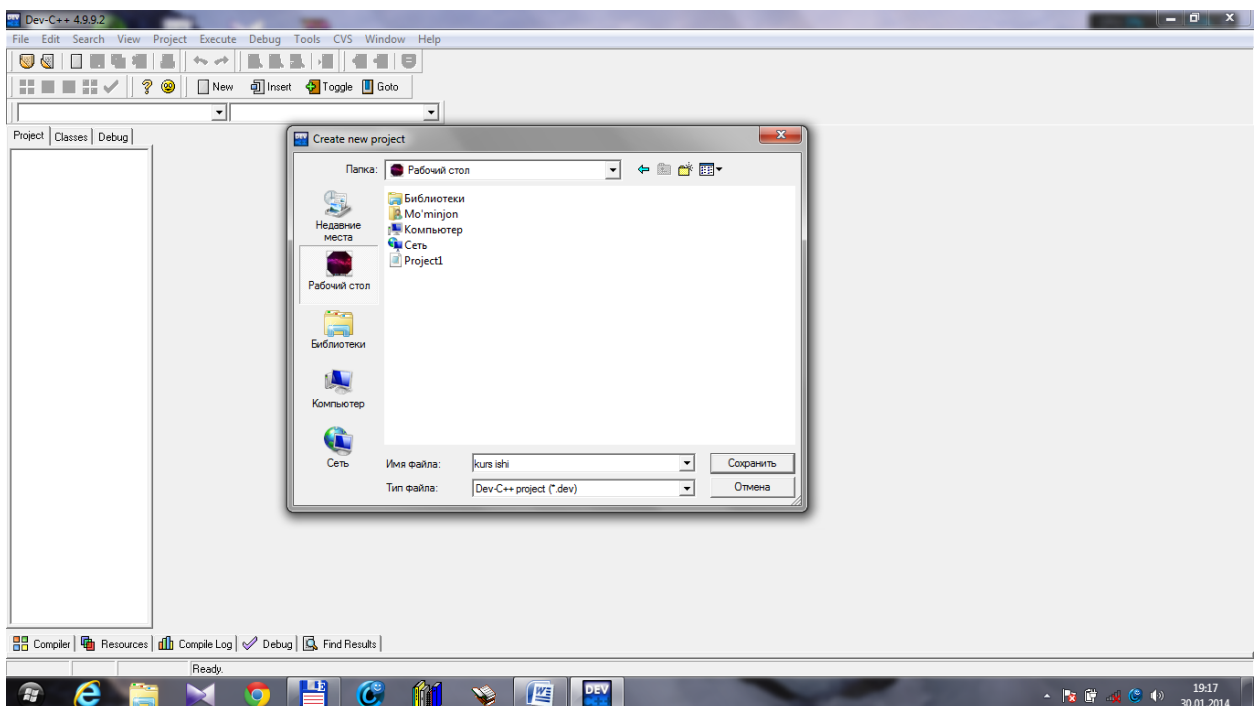
Endi biz masalamizga yondashgan holda “Concle Applicasion” bo’limini tanlaymiz. Shu holatda C++ project bo’limi, <name>da project ham tanlaniq turishi kerak. Va ok tugmasiga kursor orqali olib borib sichqonchani chap tugmasi bosiladi.



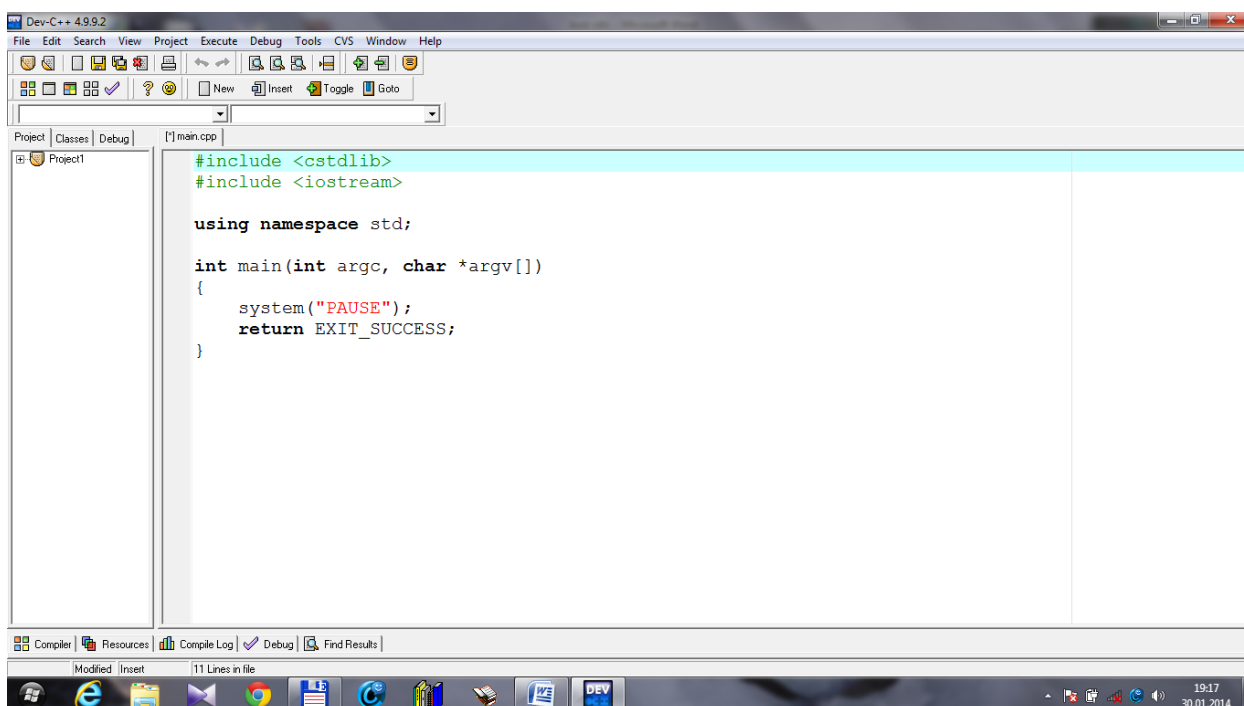
Biz “ok” tugmasini bosganimizdan so’ng bizga manashu ishlatadigan dasturimizdagi , masalani programmasini tuzishimiz qayerga saqlanishi ko’rsatib beradi. Hohlagan joyimizga saqlashimiz mumkin. Faqatgina esdan chiqarmagan holda. Saqlagan joyimiz esda turishi lozim.



”imya fayla” buyrug’i orqali dasturimizga nom beramiz. Faqat nom berishda lotin harflari bilan yozilishi kerak. Kursorni “saxranit” bo’limiga olib borib, “saxranit” bo’limini tanlaymiz.



Shunda biz DEV C++ dasturini ishga tushirgan bo'lamiz. Bizga kerakli programma ochiladi. Ya'ni quyidagi ko'rinishda bo'ladi.



Bizga programmamizda tayyor kodlari bilan dastur ochib beradi.

Oldin bizga taklif qilingan kodlar bilan tanishib chiqamiz.

Bu yerda *#include*—qo'shib olishlikni bildiradi. *ctdlib*—(standartleyb)

#include<stdlib>

ya'ni standart kutubxonalar sarlovxa faylini asosiy dasturga qo'shib oladi.

<iostream>—(input oput strim)

#include<iostream>

Ya'ni dasturga kiritish—chiqarishni boshqarish oqimini qo'shib oladi.

Using namespace std:

Bu esa tarjimada nomlanishi “standart nomlar fazosi” deb ataladi. “standart nomlar fazosi” e'lon qiladi.

*Int main(int argc, char*argv[])*

Main –asosiy funksiyalar nomi.

System (“pause”);

Bu natijani ushlab (pauza qilib) turish uchun xizmat qiladi. Agar slesh orqali uni oldiga 2 ta slesh qo'yib, komentariya qilsak ahu qatorni komplyator tomonidan tashlab ketadi. Shunda birorbir dasturni ishlatadigan bo'lsak bizga qora oyna yani "compile" ni ushlab turmaydi. Va ishlatadigan dasturimiz tez ko'rinib "compile" o'chib qolaveradi. Shuning uchun *System* ("pause"); C++ dasturida ishlaganimizda turishi lozim. PAUSE windows amaliyotida tizimida ishlaydi. SYSTEM buyrug'i orqali biz window tizimida bo'lganligimiz uchun ixtiyoriy komandani berishimiz mumkin. agar UNIX da bo'lganda PAUSE buyrug'ini qo'yishimiz shart emas edi.

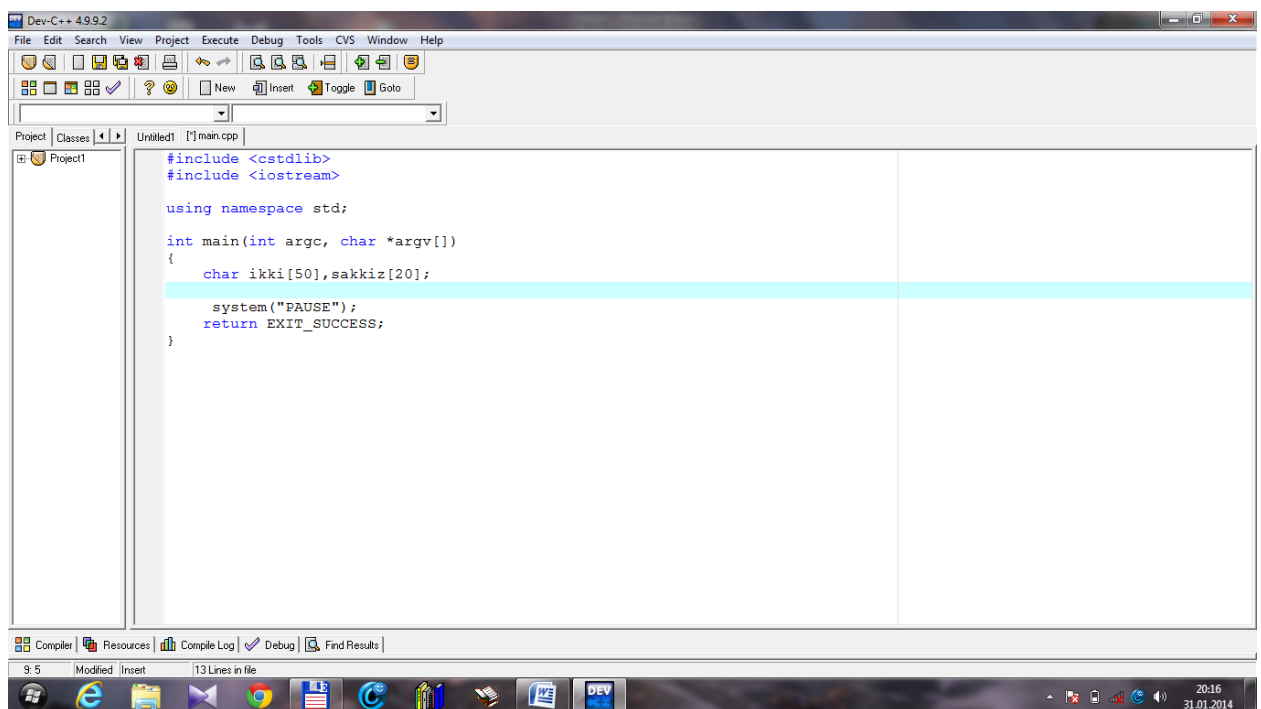
" ; " — bu belgi operator tugallanganligini bildiradi.

Return—programmani tugallanganligini bildiradi.

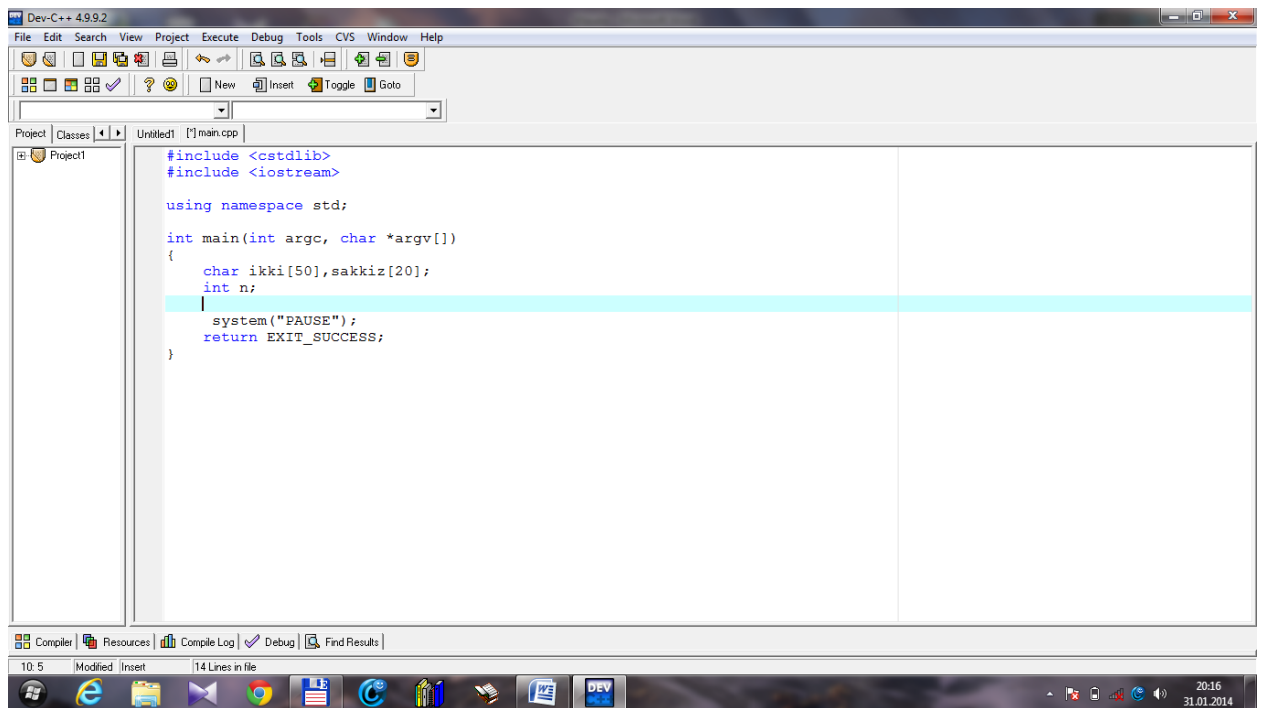
Biz qisqa bo'lsada programma haqida ma'lumot oldik.

Endi biz masalamizga kora ish yuritamiz.

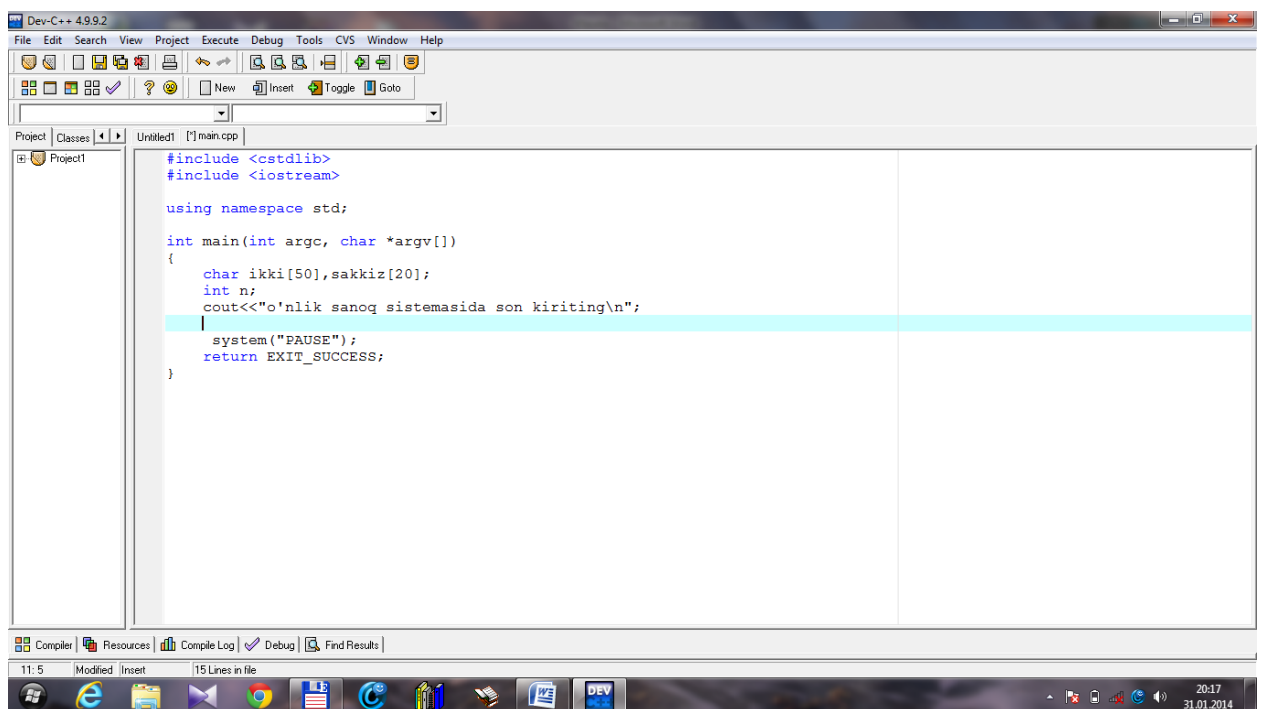
Figurali qavs ichiga masalamiz kodini kiritishni boshlaymiz. Biz avval o'zgaruvchilarni e'lon qilamiz buning uchun bizga char toifasi kerak bo'ladi. Char toifasi biga belgilar bilan ishlashda kerak bo'ladi. Va quyidagicha yozamiz.



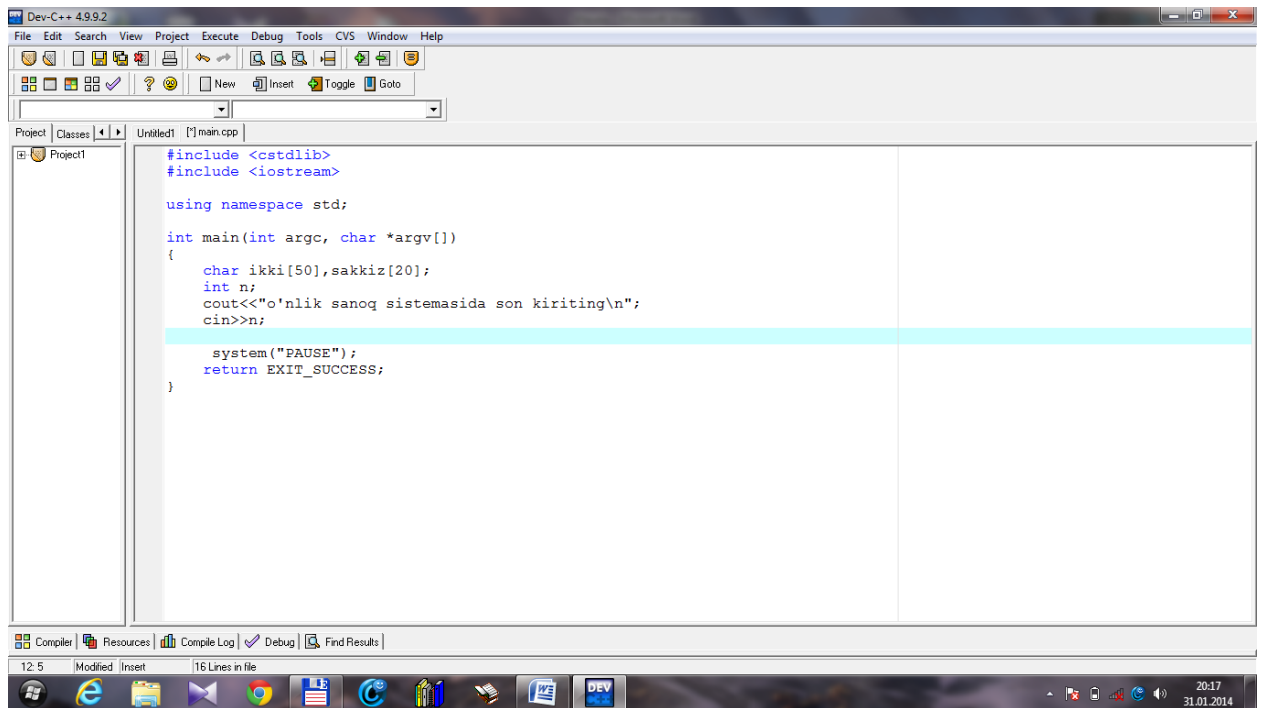
So'ngra int butun qiymatni qabul qiladigan n o'zgaruvchini kiritamiz.



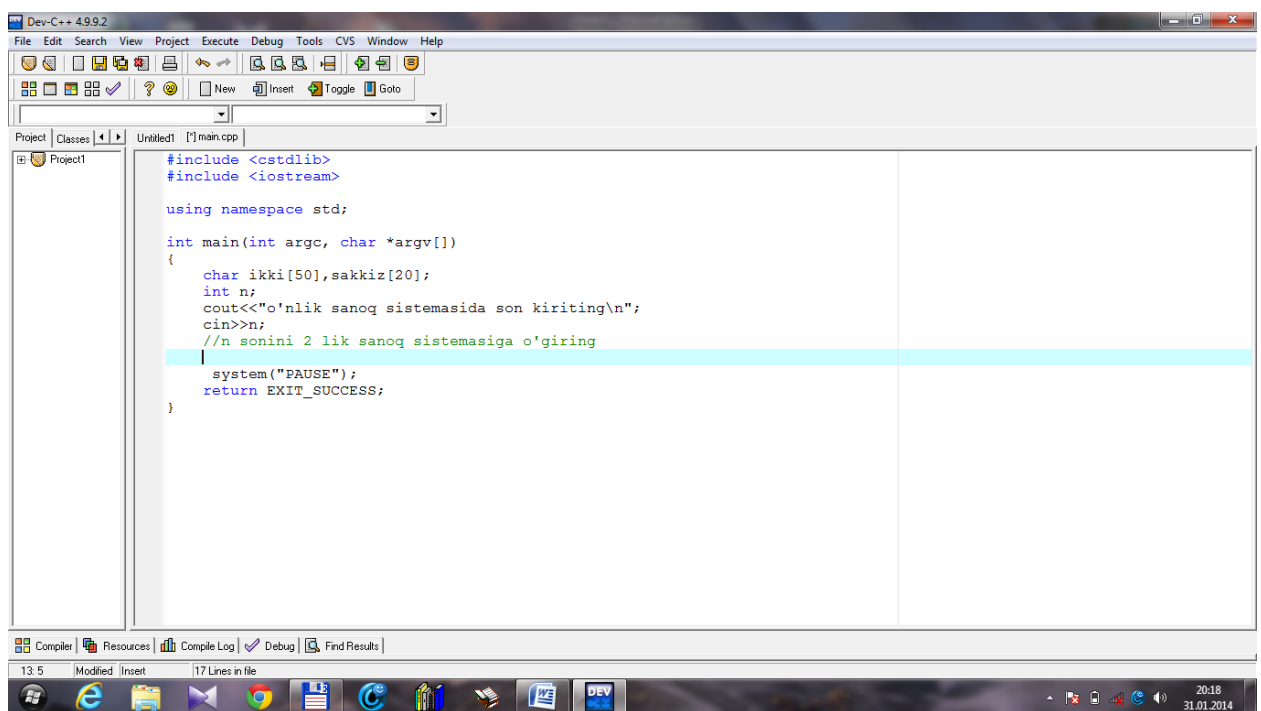
Davom etamiz : “ cout ” chiqarish operatoridan foydalanib yozgan ma’lumotimizni bizga chiqarib beruvchi “ compile ”(qora oyna) da ko’rishimiz mumkin. Buning uchun biz quyidagi rasm bo’yicha yozamiz. Ya’ni “ compile ” da chiqaradigan “ o’nlik sanoq sistemasida son kiriting” ni kodini yozamiz.



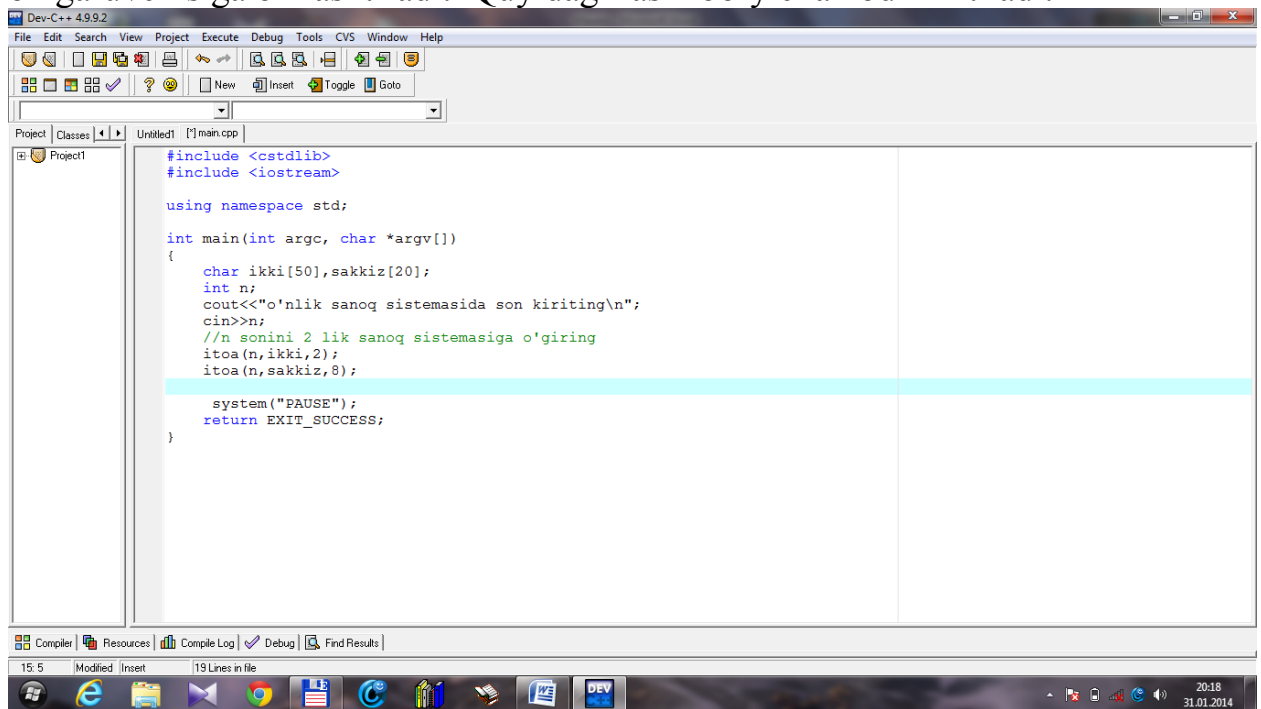
Endi “cin” kiritish operatoridan foydalanib butun son kiritish kodini yozamiz.



Komentariya orqali keying yozuvni kiritamiz. Bu qatorni “ compile ”da ish bajarmay tashlab ketadi.



“Itoa” funksiya tipidan foydalanib n sonini radix sanoq sistemasida satr o’zgaruvchisiga o’zlashtiradi. Quyidagi rasm bo’yicha kodi kiritiladi.



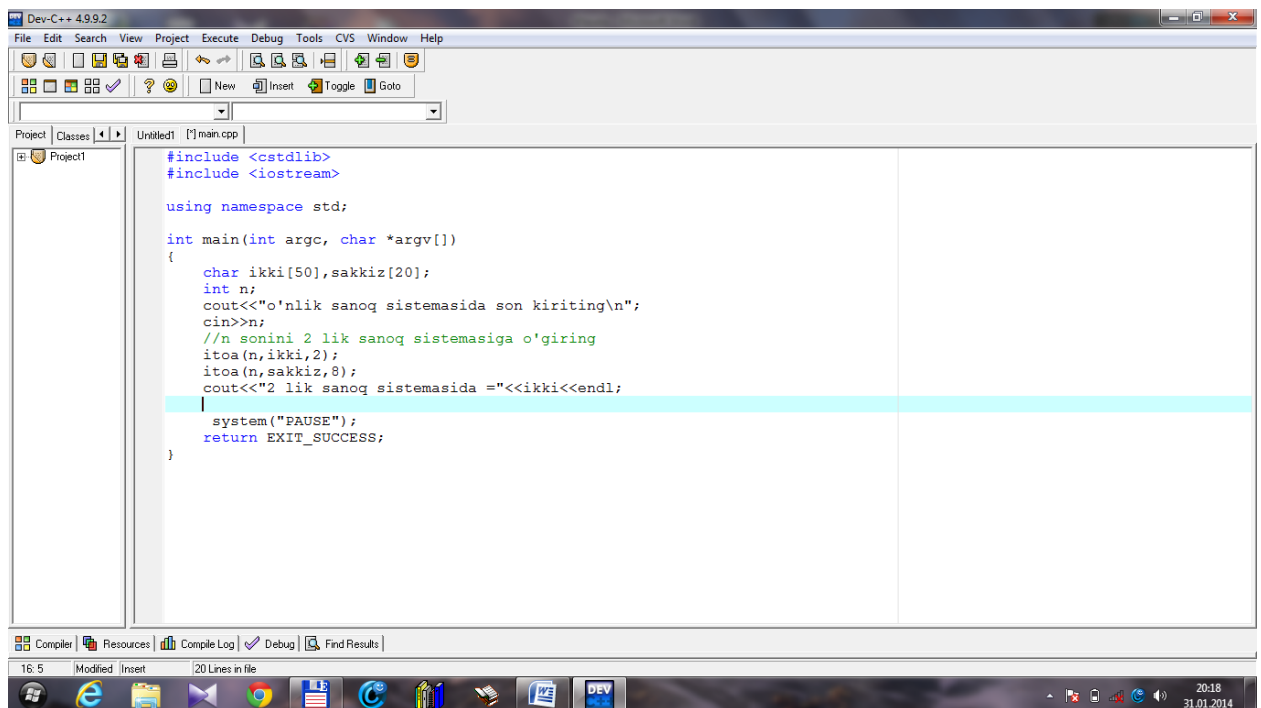
```
#include <cstdlib>
#include <iostream>

using namespace std;

int main(int argc, char *argv[])
{
    char ikki[50], sakkiz[20];
    int n;
    cout<<"o'nlik sanoq sistemasida son kiriting\n";
    cin>>n;
    //n sonini 2 lik sanoq sistemasiga o'giring
    itoa(n, ikki, 2);
    itoa(n, sakkiz, 8);

    system("PAUSE");
    return EXIT_SUCCESS;
}
```

Oxiri bizga chiqarib beruvchi javobni kodini kiritamiz. Buning uchun biz “cout” operatoridan foydalanamiz, rasmdagicha kodni kiritamiz.



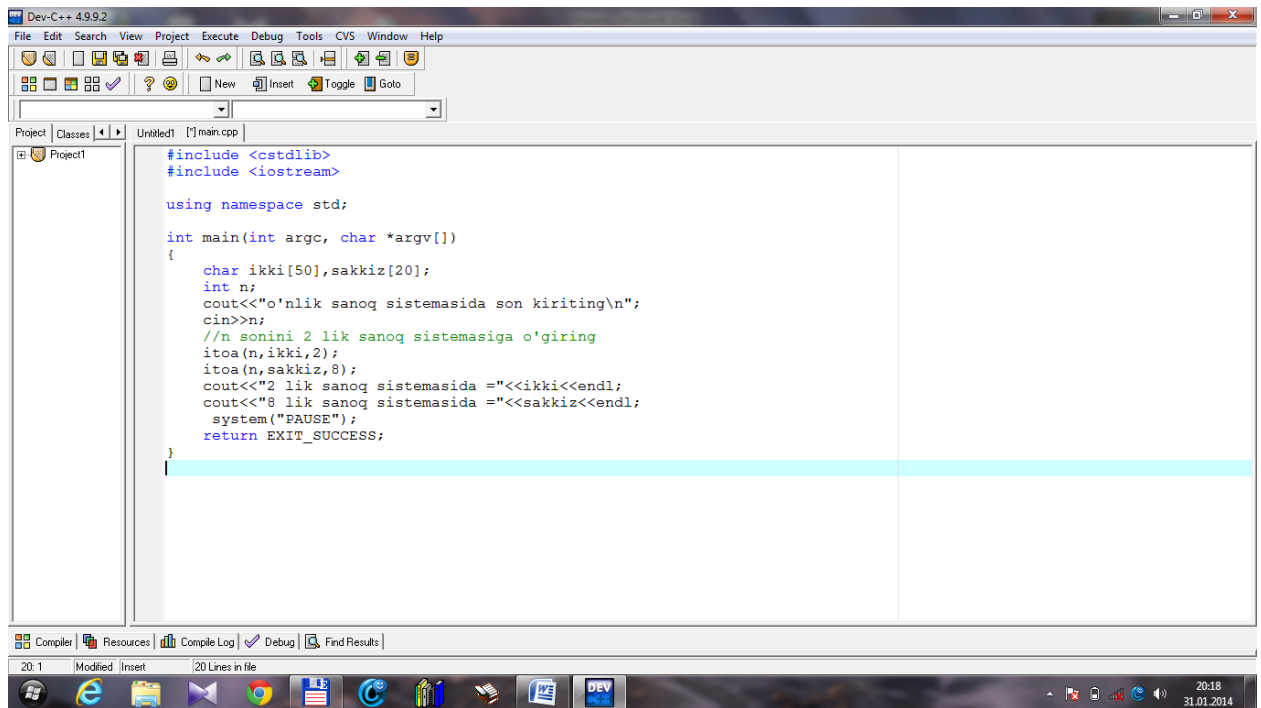
```
#include <cstdlib>
#include <iostream>

using namespace std;

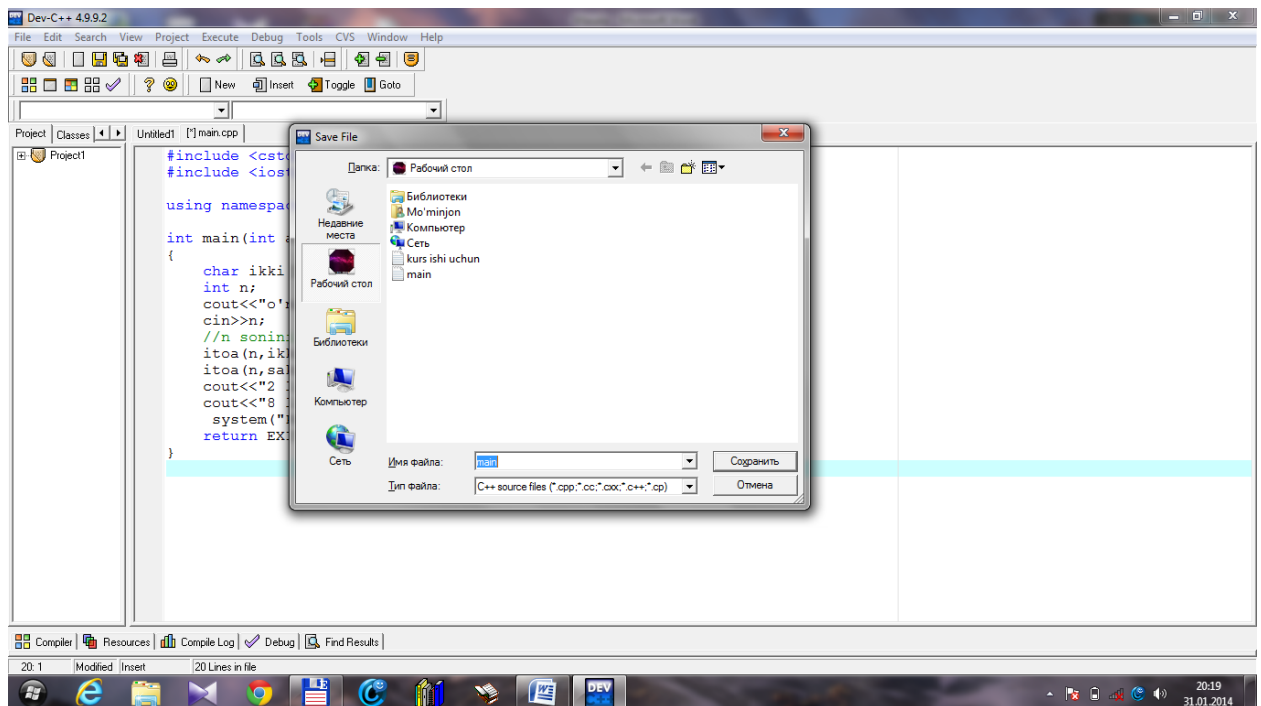
int main(int argc, char *argv[])
{
    char ikki[50], sakkiz[20];
    int n;
    cout<<"o'nlik sanoq sistemasida son kiriting\n";
    cin>>n;
    //n sonini 2 lik sanoq sistemasiga o'giring
    itoa(n, ikki, 2);
    itoa(n, sakkiz, 8);
    cout<<"2 lik sanoq sistemasida ="<<ikki<<endl;

    system("PAUSE");
    return EXIT_SUCCESS;
}
```

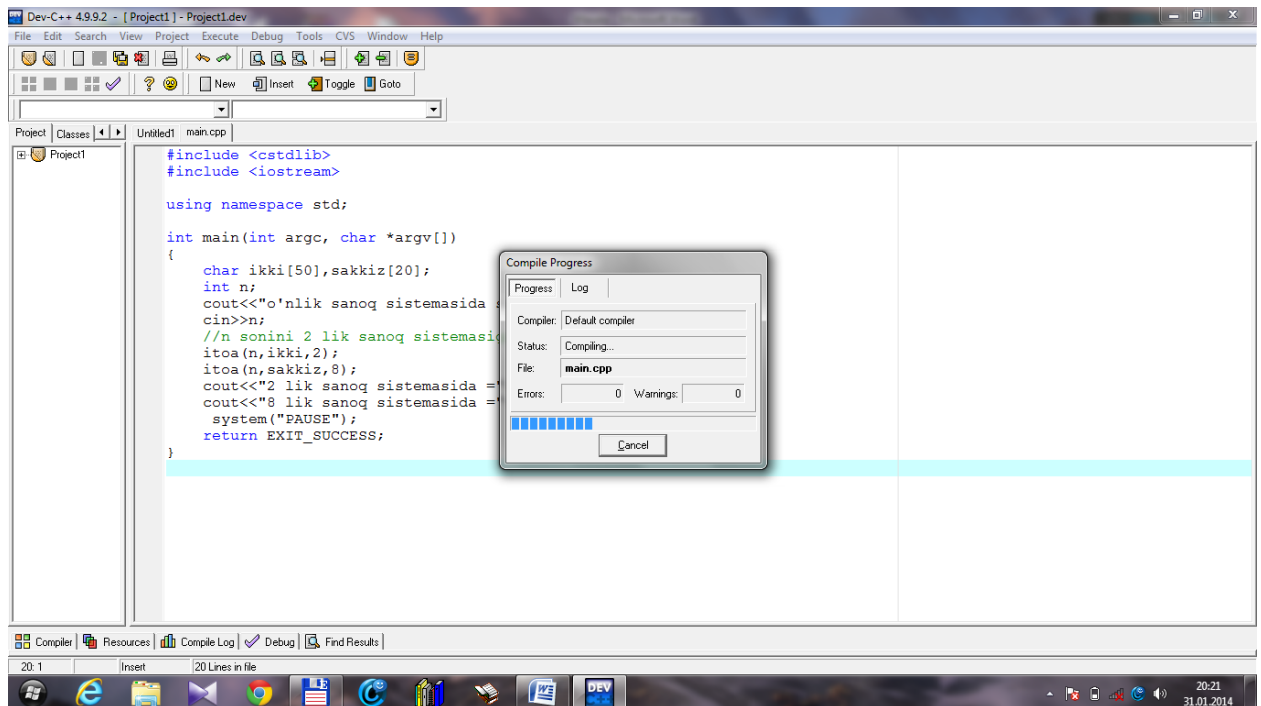
Ikkilik va sakkizlik sanoq sistemasidagi sonni chiqarib beruvchi kodni kiritamiz.



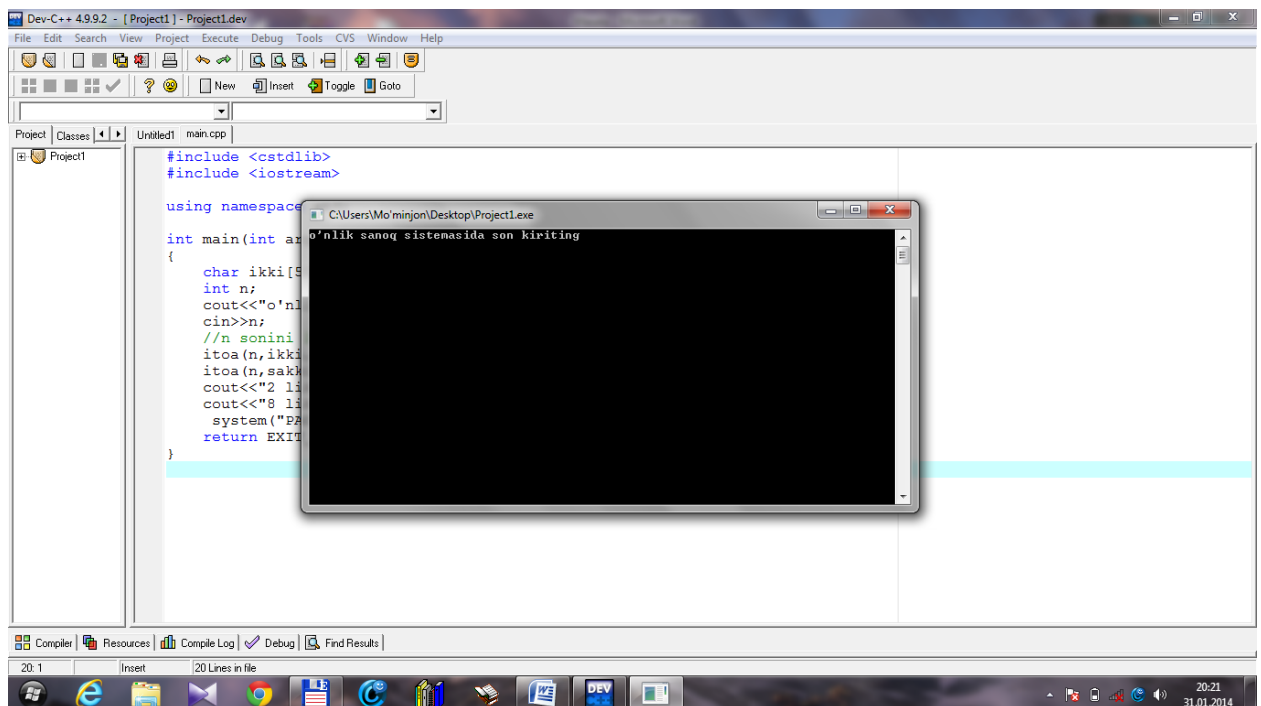
Eng so'ngi qiladigan ishimiz biz yaratgan masala kodini yozib bo'lganimizdan keyin dastur ishlashi exe fayliga kiramiz. Buning uchun klaviatura orqali "F9" tugmasini bosamiz. Bizdan ishlayotgan dasturimizni saqlash so'raladi. Nom berib saqlanishni bosamiz.



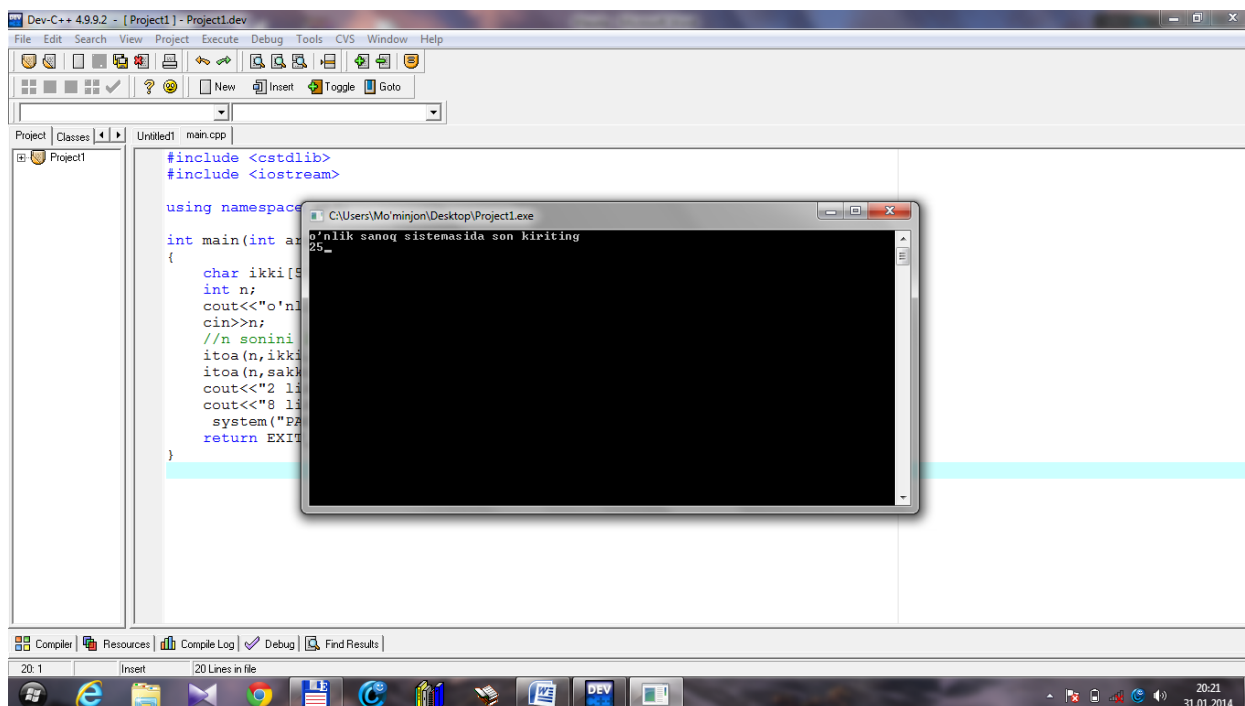
Ozgina kutamiz!



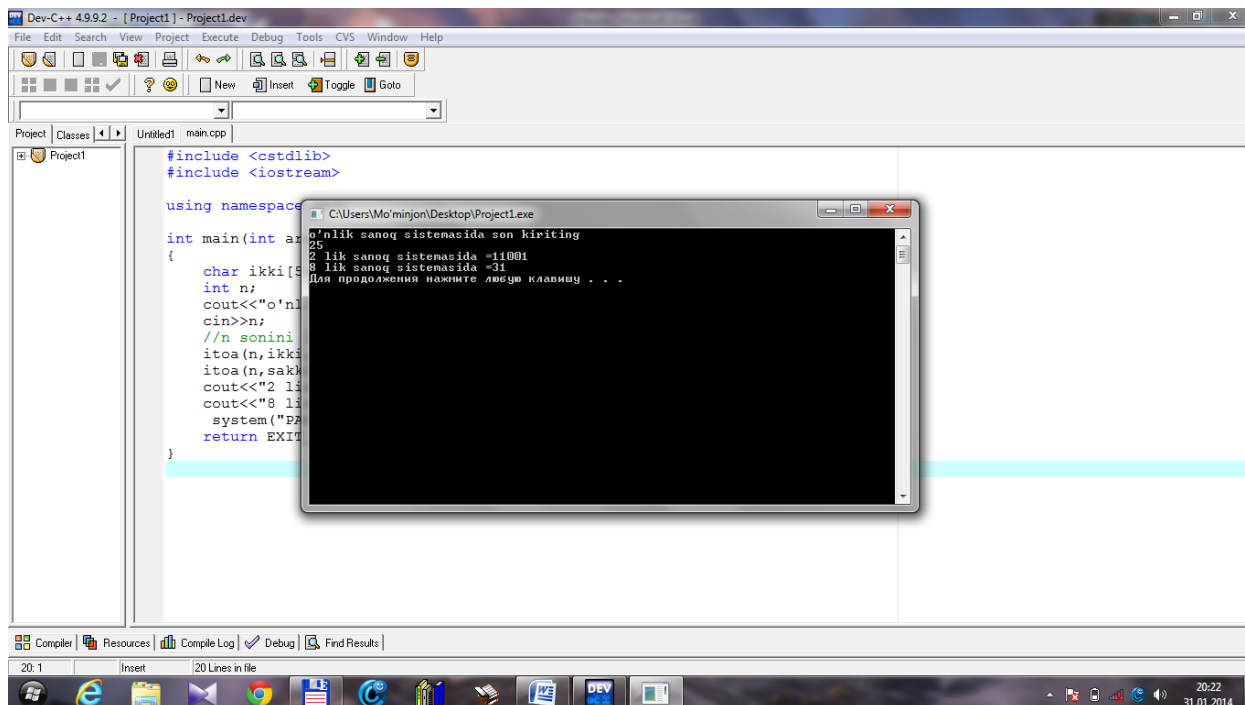
O'nlik sanoq sistemasida son kiritamiz



Misol uchun 25 sonini kiritamiz! Va enter tugmasini bosamiz.



Bizga 2 lik va 8 lik sanoq sistemasiga o'girib chiqarib beradi.



Xulosa

Men bu kurs ishi mavzusini olgan paytdan boshlab, o'z ustimda ko'prok ishlashni boshladim. massivlar xaqida ko'prok malumotlar to'plab, barchasini ko'rib chikdim. Massiv nima va uning kaday imkoniyatlari mavjud, satirli massivlar, ko'rsatkichli massivlar xaqida ko'plab malumotlar o'rganib chiqdim.

Xulosa qilib shuni aytamanki massivlar dasturchi uchun muammo bo'lgan bazi jarayonlarni osonlik bilan xal etishi mumkin. Men o'z kursdoshlarimga shuni maslaxat bergan bo'lardimki massivlarni chuqurrok o'rganishsin va kelajakdagi faoliyatiga omad tilayman.

Foydalanilgan adabiyotlar

1. S. Bobrovskiy “Самоучител программирования на языке C++”; Moskva – 2001.
2. *The C++ Programming Language Third Edition*; Bjarne Stroustrup.
3. «Informatika» fani bo'yicha ma'ruzalar matni. Toshkent -2011 .
4. N.G. Mardanova “ Programmalash bo'yicha chuqurlashtirilgan kurs 1 qism” toshkent 2006;
5. WWW.Google.com
6. www.dasturchi.uz
7. www.Dev-cpp.com
8. www.wikipediya.org

Ilova

Masala : M(9)massivda qandaydir natural sonning I-sanoq sistemasida raqamlari yozilgan. (m(1)-birlar xonasi va h. z).Birlar xonasidan boshlab turib, J-sanoq sistemasida bu son raqamlarini chop eting. I,Jsonlar 10 dan oshmaydi.

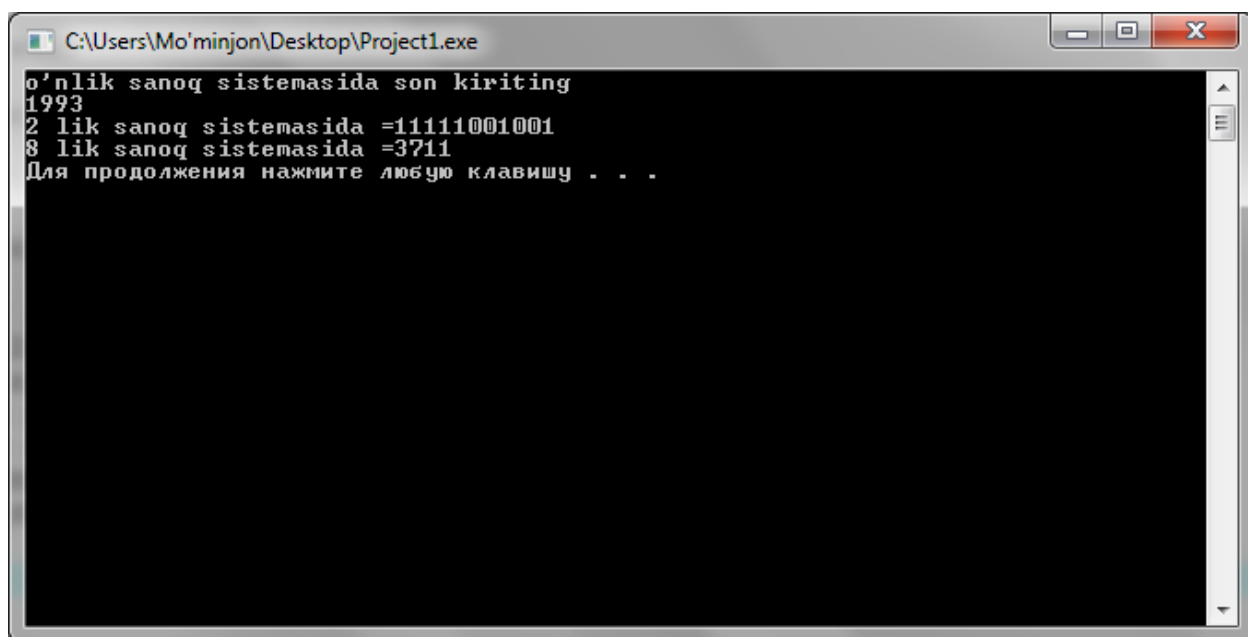
Dastur kodi:

```
#include <cstdlib>
#include <iostream>

using namespace std;

int main(int argc, char *argv[])
{
    char ikki[50],sakkiz[20];
    int n;
    cout<<"o'nlik sanoq sistemasida son kiriting\n";
    cin>>n;
    //n sonini 2 lik sanoq sistemasiga o'giring
    itoa(n,ikki,2);
    itoa(n,sakkiz,8);
    cout<<"2 lik sanoq sistemasida ="<<ikki<<endl;
    cout<<"8 lik sanoq sistemasida ="<<sakkiz<<endl;
    system("PAUSE");
    return EXIT_SUCCESS;
}
```

NATIJA



```
C:\Users\Mo'minjon\Desktop\Project1.exe
o'nlik sanoq sistemasida son kiriting
1993
2 lik sanoq sistemasida =11111001001
8 lik sanoq sistemasida =3711
Для продолжения нажмите любую клавишу . . .
```