

**O'ZBEKISTON ALOQA, AXBOROTLASHTIRISH VA
TELEKOMMUNIKATSIYA
TEXNOLOGIYALARI DAVLAT QO'MITASI
TOSHKENT AXBOROT TEXNOLOGIYALAR UNIVERSITETI
NUKUS FILIALI**

Kompyuter Injiniring fakulteti
3b Informationsion texnologiyalar yo'nalishi talabasi

Utepov Jasurbekning
C++ dasturlash tili
fanidan

Mustaqil ishi

Mavzu: C++ da mantiqiy o`yinlar dasturini yaratish

Bajargan:

Utepov J.

Qabul qilgan:

Tleuov K.

Nukus-2014

Mundarija:

I.Kirish.....	2
II.Asosiy qism:	
2.1. C++ Tili Tuzilishi:.....	4
2.2. C ++ tilida xizmachil so'zlar:.....	5
2.3. C++ tilida o`yinlar dasturini tuzish.....	7
2.4. General o`yini.....	8
2.5. Gonka o`yini	
III.Xulosa.....	26
IV.Foydalanilgan Adabiyotlar.....	27

KIRISH

C++ dasturlash tili C tiliga asoslangan. C esa o'z navbatida B va BCPL tillaridan kelib chiqqan. BCPL 1967 yilda Martin Richards tomonidan tuzilgan va operatsion sistemalarni yozish uchun mo'ljallangan edi. Ken Thompson o'zining B tilida BCPL ning ko'p hossalarni kiritgan va B da UNIX operatsion sistemasining birinchi versiyalarini yozgan.

BCPL ham, B ham tipsiz til bo'lgan. Yani o'garuvchilarning ma'lum bir tipi bo'lmagan - har bir o'zgaruvchi kompyuter hotirasida faqat bir bayt yer egallagan. O'zgaruvchini qanday sifatda ishlatish esa, yani butun sonmi, kasrli sonmi yoki harfdekmi, dasturchi vazifasi bo'lgan.

C tilini Dennis Ritchie B dan keltirib chiqardi va uni 1972 yili ilk bor Bell Laboratories da, DEC PDP-11 kompyuterida qo'lladi. C o'zidan oldingi B va BCPL tillarining juda ko'p muhim tomonlarini o'z ichiga olish bilan bir qatorda o'zgaruvchilarni tiplashtirdi va bir qator boshqa yangiliklarni kiritdi. Boshlanishda C asosan UNIX sistemalarida keng tarqaldi. Hozirda operatsion sistemalarning asosiy qismi C/C++ da yozilmoqda. C mashina arxitekturasiga bog'langan tildir. Lekin yahshi rejalashtirish orqali dasturlarni turli kompyuter platformalarida ishlaydigan qilsa bo'ladi.

1983 yilda, C tili keng tarqalganligi sababli, uni standartlash harakati boshlandi. Buning uchun Amerika Milliy Standartlar Komiteti (ANSI) qoshida X3J11 texnik komitet tuzildi. Va 1989 yilda ushbu standart qabul qilindi.

Standartni dunyo bo'yicha keng tarqatish maqsadida 1990 yilda ANSI va Dunyo Standartlar Tashkiloti (ISO) hamkorlikda C ning ANSI/ISO 9899:1990 standartini qabul qilishdi.

Shu sababli C da yozilgan dasturlar kam miqdordagi o'zgarishlar yoki umuman o'zgarishlarsiz juda ko'p kompyuter platformalarida ishlaydi.

C++ 1980 yillar boshida Bjarne Stroustrup tomonidan C ga asoslangan tarzda tuzildi. C++ juda ko'p qo'shimchalarni o'z ichiga olgan, lekin eng asosiysi u ob'ektlar bilan dasturlashga imkon beradi.

Dasturlarni tez va sifatli yozish hozirgi kunda katta ahamiyat kasb etmoda. Buni ta'minlash uchun ob'ektli dasturlash g'oyasi ilgari surildi. Huddi 70-chi yillar boshida strukturali dasturlash kabi, programmalarni hayotdagi jismlarni modellashtiruvchi ob'ektlar orqali tuzish dasturlash sohasida inqilob qildi.

C++ dan tashqari boshqa ko'p ob'ektli dasturlashga yo'naltirilgan tillar paydo bo'ldi. Shulardan eng ko'zga tashlanadigani Xerox ning Palo Altoda joylashgan ilmiy-qidiruv markazida (PARC) tuzilgan Smalltalk dasturlash tilidir. Smalltalk da hamma narsa ob'ektlarga asoslangan. C++ esa gibril tildir.

Unda C ga o'hshab strukturali dasturlash yoki yangicha, ob'ektlar bilan dasturlash mumkin. Yangicha deyishimiz ham nisbiydir. Ob'ektli dasturlash falsafasi paydo bo'lganiga ham yigirma yildan oshayapti.

C++ funksiya va ob'ektlarning juda boy kutubhonasiga ega. Yani C++ da dasturlashni o'rganish ikki qismga bo'linadi. Birinchisi bu C++ ni o'zini o'rganish, ikkinchisi esa C++ ning standart kutubhonasidagi tayyor ob'ekt/funksiyalarni qo'llashni o'rganishdir.

C++ TILI TUZILISHI.

1. Alfavit, identifikator, xizmatchi so'zlar.

Alfavit. C++ alfavitiga quyidagi simvollar kiradi.

- Katta va kichik lotin alfaviti xarflari (A,B,...,Z,a,b,...,z)
- Raqamlar: 0,1,2,3,4,5,6,7,8,9
- Maxsus simvollar: “ , { } | [] () + - / % \ ; ‘ . : ? < = > _ ! & * # ~ ^
- Ko'rinmaydigan simvollar (“umumlashgan bushliq simvollari”). Leksemalarni uzaro ajratish uchun ishlatiladigan simvollar (misol uchun bushlik, tabulyatsiya, yangi qatorga o'tish belgilari).

Izohlarda, satrlarda va simvolli konstantalarda boshqa literalalar, masalan rus xarflarini ishlatilishi mumkin.

C++ tilida olti khil turdagi leksemalar ishlatiladi: ehrkin tanlanadigan va ishlatiladigan identifikatorlar, khizmatchi suzlar, konstantalar(konstanta satrlar), amallar(amallar belgilari), azhratuvchi belgilar.

Identifikator.Identifikatorlar lotin xarflari,ostki chiziq belgisi va sonlar ketma ketligidan iborat buladi. Identifikator lotin xarfidan yoki ostki chizish belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

Katta va kichik xarflar farklanadi, shuning uchun ohirgi ikki identifikator bir biridan farq qiladi.

Borland kompilyatorlaridan foydalanilganda nomning birinchi 32 xarfi ,ba'zi kompilyatorlarda 8 ta xarfi inobatga olinadi. Bu holda NUMBER_OF_TEST va NUMBER_OF_ROOM identifikatorlari bir biridan farq qilmaydi.

Xizmatchi so'zlar. Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar xizmatchi so'zlar deyiladi.

C ++ tilida quyidagi xizmatchi so'zlar mavjud:

int	extern	else
char	register	for
float	typedef	do
double	static	while
struct	goto	switch
union	return	case
long	sizeof	default
short	break	entry
unsigned	continue	
auto	if	

O'zgaruvchilar. (VARIABLES)

O'zgaruvchilar ob'ekt sifatida. C++ tilining asosiy tushunchalaridan biri nomlangan hotira qismi – ob'ekt tushunchasidir. Ob'ektning xususiy holi bu o'zgaruvchidir. O'zgaruvchiga qiymat berilganda unga ajratilgan hotira qismiga shu qiymat kodi yoziladi. O'zgaruvchi qiymatiga nomi orqali murojaat qilish mumkin, hotira qismiga esa faqat adresi orqali murojaat qilinadi. O'zgaruvchi nomi bu erkin kiritiladigan identifikatordir. O'zgaruvchi nomi sifatida xizmatchi so'zlarni ishlatish mumkin emas.

O'zgaruvchilar tiplari. O'zgaruvchilarning qo'yidagi tiplari mavjuddir:

`char` – bitta simvol;

`long char` – uzun simvol;

`int` – butun son;

`short` yoki `short int` – qisqa butun son;

`long` yoki `long int` – uzun butun son;

`float` - haqiqiy son;

`long float` yoki `double` – ikkilangan haqiqiy son;

`long double` – uzun ikkilangan haqiqiy son;

Butun sonlar ta'riflanganda ko'rilgan tiplar oldiga `unsigned` (ishorasiz) ta'rifi kushilishi mumkin. Bu ta'rif qushilgan butun sonlar ustida amallar `mod 2n` arifmetikasiga asoslangandir. Bu erda n soni `int` tipi hotirada egallovchi razryadlar sonidir. Agar ishorasiz k soni uzunligi `int` soni razryadlar sonidan uzun bulsa, bu son qiyjmati $k \bmod 2^n$ ga teng bo'ladi. Ishorasiz k son uchun $-k$ amali $2^n - k$ formula asosida hisoblanadi. Ishorali ya'ni `signed` tipidagi sonlarning eng katta razryadi son ishorasini ko'rsatish uchun ishlatilsa `unsigned` (ishorasiz) tipdagi sonlarda bu razryad sonni tasvirlash uchun ishlatiladi.

O'zgaruvchilarni dasturning ixtiyoriy qismida ta'riflash yoki qayta ta'riflash mumkin.

Misol uchun:

`Int a, b1, ac; eki`

`Int a;`

`int b1;`

`int ac;`

O'zgaruvchilar ta'riflanganda ularning qiymatlari aniqlanmagan bo'ladi. Lekin

o'zgaruvchilarni ta'riflashda initsializatsiya ya'ni boshlang'ich qiyjmatlarini ko'rsatish mumkin.

Misol uchun:

```
Int I=0;
```

```
Char c='k';
```

Typedef ta'riflovchisi yangi tiplarni kiritishga imkon beradi.

Misol uchun yangi COD tipini kiritish:

```
Typedef unsigned char COD;
```

```
COD simbol;
```

Men bu o'yinni tuzguncha Visual Studio 2010 dan foydalandim. Dasturlash tili C/C++.

Ishni boshlashdan oldin www.libsdl.org/download-2.0.php ssilkadan Development Libraries :: SDL2-devel-2.0.1-VC.zip ni ko'chirib olamiz. Bu SDL kutubxonasi bo'lib dasturlash uchun zarur bo'lgan fayllar mavjud.

Endi VS2010 ni ishga tushirib, CTRL+SHIFT+N ni bosib Visual C++ bo'limidan Win32 ga tegishli Win32 Console Application ni tanlaymiz. Projekt nomini Gonka deb OK ni bosamiz. Keyin Next ni bosib, «Empty project» ni tanlab FINISH ni bosamiz. Keyin shu projektни ichiga main.cpp fayl yaratamiz.

Endi c:devlibssdl (bu manda siz boshqa joyda qilsangiz ham bo'ladi) ichiga SDL2-devel-2.0.1-VC.zip ni ochib tashlimiz, shunda c:devlibssdl ning ichida include, lib papkalari paydo bo'lishi lozim.

VS ga o'tib, ALT+F7 (Project => Properties) ga kiramiz va quyidagi sozlashlarni amalga oshiramiz:

General o`yini:

1. Configuration Properties -> C/C++ -> General -> Additional include directories = c:devlibssdlinclude
2. Configuration Properties -> Linker -> General -> Additional library directories = c:devlibssdllibx86
3. Configuration Properties -> Linker -> Input -> Additional Dependencies = opengl32.lib;SDL2.lib;SDL2main.lib;%(AdditionalDependencies)
4. Va oxirgisi: Configuration Properties -> Debugging -> Environment = PATH=%PATH%;c:devlibssdllibx86 bu SDL.DLL faylini bizni dastur qo'llash olishi uchun kerak. Agar siz oxirig EXE faylni bitta joyga ko'chirmoqchi bo'lsangiz SDL.DLL ni ham qo'shib ko'chiring. Aks holda SDL.DLL topilmadi degan xato beradi.

Agar sozlashlarni qilib bo'lgan bo'lsangiz main.cpp faylga quyidagini yozing va F5 ni bosing. Agar qizil oyna chiqsa demak hammasi OK, ishlashda davom etamiz.

```
#include <stdio.h>
```

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

```
//GL ni yuklash
```

```
#include "windows.h"
```

```
#include <gl/GL.h>
```

```
//SDL ni yuklash
```

```
#include <SDL.h>

#define PROGRAM_NAME "Mini gonka"

/* Xatolikni chiqaradigan funksiya */

void sldie(const char *msg)

{

    printf("%s: %sn", msg, SDL_GetError());

    SDL_Quit();

    exit(1);

}

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

{

    SDL_Window *mainwindow; /* Oynda o'zgaruvchisi */

    SDL_GLContext maincontext; /* OpenGL konteksti o'zgaruvchisi */

    if (SDL_Init(SDL_INIT_VIDEO) < 0) /* Initsalizatsiya SDL. Umuman
tekshirish. */

        sldie("Initsalizatsiya SDL da xatolik"); /* Agar xatolik bo'lsa */

    SDL_GL_SetAttribute(SDL_GL_DOUBLEBUFFER, 1); /* Double buffer
qo'shish */
```

```

//Oyna yaratish, o'lchami 300x450 bo'lgan

mainwindow=SDL_CreateWindow(PROGRAM_NAME,
SDL_WINDOWPOS_CENTERED, SDL_WINDOWPOS_CENTERED,
    300, 450, SDL_WINDOW_OPENGL | SDL_WINDOW_SHOWN);

if (!mainwindow) /* Agar yaratib bo'lmasa */

    sldie("Oynani yaratishda xatolik");


/* OpenGL oynasini yaratish */

maincontext = SDL_GL_CreateContext(mainwindow);


//oyna o'lchamini berish

glOrtho(0, 10, 15, 0, -1, 1);


/* Bufferni tozalash rangi */

glClearColor ( 1.0, 0.0, 0.0, 1.0 );

glClear ( GL_COLOR_BUFFER_BIT );


/* Qog'ozni almashtirish */

SDL_GL_SwapWindow(mainwindow);


//Main loop, ESC ni bosangiz chiqib ketadi

SDL_Event e;

```

```

while( SDL_WaitEvent(&e) )
{
    if (e.type == SDL_KEYDOWN && e.key.keysym.sym ==
SDLK_ESCAPE)
        break;
}

//tozalash va chiqish

SDL_GL_DeleteContext(maincontext);

SDL_DestroyWindow(mainwindow);

SDL_Quit();

return 0;
}

```

Biz hozir o'yin tuzish bilan ovoramiz, keling SDL_GL_DOUBLEBUFFER, SDL_GL_DEPTH_SIZE, glClearColor, GL_COLOR_BUFFER_BIT,... kabi terminlar haqida yana bitta maqola yozaman. Umuman OpenGL qanday ishlaydi mavzusida. Hozircha umumiy qilib tushintirib ketaman. Ya'ni hamma qismini bitta bitta funksiya qilaman va funksiyaning nima ish qilishini tushuntiraman.

Gonka o`yini:

Bizning gonkamiz quyidagi ko'rinishda bo'ladi:

Ha bu o'sha gonka, ko'pchilikni esidan ham chiqqan bo'lsa kerak. Bu NSF Most Wanted masu boshlovchi dasturchilar uchun undanda azizroq gonka bo'ladi. Keling endi o'zimizga oddiy maqsad qo'ysak. Bu o'yinni yasash uchun bizga asosiy narsa bu to'rt burcha chizishdir.

Bizning 4 burchak chizadigan funksiyamiz

```
void draw4burchak(int x, int y, int color)
{
    glBegin(GL_QUADS);

    if (color == 0)
        glColor3ub(0, 0, 0); //qora rang
    else
        glColor3ub(52, 111, 255); // ko'k rang

    glVertex2i(x, y);
    glVertex2i(x + 1, y);
    glVertex2i(x + 1, y + 1);
    glVertex2i(x, y + 1);
    glEnd();
}
```

```
}
```

ko'rinishida bo'ladi. x, y qaysi nuqtaligi $([0, 0]$ — chap yuqori nuqta, $color = 0$ — bu qora, aks holda ko'k rang)

Endiki masala mashinani chizishdir.

```
void drawcar(int x, int y, int color)
```

```
{
```

```
    draw4burchak(x + 1, y, color);
```

```
    draw4burchak(x, y + 1, color);
```

```
    draw4burchak(x + 2, y + 1, color);
```

```
    draw4burchak(x + 1, y + 2, color);
```

```
    draw4burchak(x, y + 3, color);
```

```
    draw4burchak(x + 2, y + 3, color);
```

```
}
```

Atrofidagi yo'llarni chizish

```
void drawyul(int pos)
```

```
{
```

```

pos = pos % 3;

for(int i = 0; i < 5; i++)
{
    //chap taraf

    draw4burchak(0, 3 * i + pos, 0);

    draw4burchak(0, 3 * i + pos + 1, 0);


    //o'ng taraf

    draw4burchak(9, 3 * i + pos, 0);

    draw4burchak(9, 3 * i + pos + 1, 0);

}
}

```

pos — yo'l harakatini hosil qilish uchun kerak.

Endi o'yinni yuragini yaratamiz ya'ni GAME LOOP ni, buning uchun main loop ni sal o'zgartiramiz. Qolgani o'yinni logikasi.

O'yin kodi:

```
#include <stdio.h>
```

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

```
#include <ctime>
```

```
//GL ni yuklash
```

```
#include "windows.h"
```

```
#include <gl/GL.h>
```

```
//SDL ni yuklash
```

```
#include <SDL.h>
```

```
#define PROGRAM_NAME "Mini gonka"
```

```
/* Xatolikni chiqaradigan funksiya */
```

```
void sldie(const char *msg)
```

```
{
```

```
    printf("%s: %sn", msg, SDL_GetError());
```

```
    SDL_Quit();
```

```
    exit(1);
```

```
}
```

```
void draw4burchak(int x, int y, int color)
```

```
{
```

```
    glBegin(GL_QUADS);
```



```
if (color == 0)
```

```
    glColor3ub(0, 0, 0); //qora rang
```

```
else
```

```
    glColor3ub(52, 111, 255); // ko'k rang
```

```
glVertex2i(x, y);
```

```
glVertex2i(x + 1, y);
```

```
glVertex2i(x + 1, y + 1);
```

```
glVertex2i(x, y + 1);
```

```
glEnd();
```

```
}
```

```
void drawcar(int x, int y, int color)
```

```
{
```

```
    draw4burchak(x + 1, y, color);
```

```
    draw4burchak(x, y + 1, color);
```

```
    draw4burchak(x + 2, y + 1, color);
```

```
    draw4burchak(x + 1, y + 2, color);
```

```

        draw4burchak(x, y + 3, color);

        draw4burchak(x + 2, y + 3, color);

    }


void drawyul(int pos)

{
    pos = pos % 3;

    for(int i = -1; i < 5; i++)
    {

        //chap taraf

        draw4burchak(0, 3 * i + pos, 0);

        draw4burchak(0, 3 * i + pos + 1, 0);


        //o'ng taraf

        draw4burchak(9, 3 * i + pos, 0);

        draw4burchak(9, 3 * i + pos + 1, 0);

    }

}


int game_time = 0;

```

```
int game_time_interval = 50; //ms
```

```
float last_time = 0;
```

```
int car_position = 0;
```

```
int car1_position = 0, car2_position = 1;
```

```
int car1_position_y = 0, car2_position_y = -10;
```

```
int counter = 0;
```

```
//o'zgaruvchilarni joyiga qaytarish
```

```
void reset()
```

```
{
```

```
    MessageBoxA(NULL, "Yutqazdingiz", "Xabar", MB_OK);
```

```
    game_time = 0;
```

```
    last_time = 0;
```

```
    car_position = 0;
```

```
    car1_position = 0;
```

```
    car2_position = 1;
```

```
    car1_position_y = 0;
```

```

    car2_position_y = -10;

    counter = 0;

}

void drawscene()

{

    //vaqtni olish ms da

    SYSTEMTIME st;

    GetSystemTime(&st);

    float cur_time = st.wMinute * 60 * 1000 + st.wSecond * 1000 +
st.wMilliseconds;

    //ekranni tozalash

    glClear(GL_COLOR_BUFFER_BIT);

    //obyektlarni chizish

    drawyul(game_time);

    drawcar(3 * car_position + 2, 11, 1);

    drawcar(3 * car1_position + 2, car1_position_y, 0);

    drawcar(3 * car2_position + 2, car2_position_y, 0);

    //ekranga chiqarish

```

```
printf("rOchko: %d      ", counter);
```

```
//agar game interval ga to'g'ri keldi
```

```
if (cur_time - last_time >= game_time_interval)
```

```
{
```

```
    //o'yinni vaqtini oshirhs
```

```
    game_time++;
```

```
    //2 ta mashina harakatini hosil qilish
```

```
    car2_position_y++;
```

```
    car1_position_y++;
```

```
    //ekranga ko'rinmiy qolsa, tepaga olib chiqish
```

```
    if (car1_position_y >= 15)
```

```
    {
```

```
        car1_position_y = -5;
```

```
        counter++;
```

```
    }
```

```
    if (car2_position_y >= 15)
```

```
    {
```

```
car2_position_y = -5;
```

```
counter++;
```

```
}
```

```
//oxirgi vaqtni saqlash
```

```
last_time = cur_time;
```

```
}
```

```
//o'yinni vaqtini 0 ga tushirish
```

```
if (game_time >= 100)
```

```
    game_time = 0;
```

```
//mashina o'rilganligini tekshirish
```

```
if (car1_position == car_position)
```

```
{
```

```
    if (car1_position_y >= 9 && car1_position_y <= 14)
```

```
    {
```

```
        reset();
```

```
    }
```

```
}
```

```

else if (car2_position == car_position)
{
    if (car2_position_y >= 9 && car2_position_y <= 14)
    {
        reset();
    }
}
}

```

```

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

```

```

{

```

```

    /* O'yinni to'xtatish o'zgaruvchisi */

```

```

    int isGameRunning = 1;

```

```

    SDL_Window *mainwindow; /* Oynda o'zgaruvchisi */

```

```

    SDL_GLContext maincontext; /* OpenGL konteksi o'zgaruvchisi */

```

```

    SDL_Thread *thread; /* Game LOOP paroki */

```

```

    if (SDL_Init(SDL_INIT_VIDEO) < 0) /* Initsalizatsiya SDL. Umuman
tekshirish. */

```

```

        sddie("Initsalizatsiya SDL da xatolik"); /* Agar xatolik bo'lsa */

```

```
    SDL_GL_SetAttribute(SDL_GL_DOUBLEBUFFER, 1); /* Double buffer
qo'shish */
```

```
//Oyna yaratish, o'lchami 300x450 bo'lgan
```

```
    mainwindow = SDL_CreateWindow(PROGRAM_NAME,
SDL_WINDOWPOS_CENTERED, SDL_WINDOWPOS_CENTERED,
```

```
    300, 450, SDL_WINDOW_OPENGL | SDL_WINDOW_SHOWN);
```

```
    if (!mainwindow) /* Agar yaratib bo'lmasa */
```

```
        sldie("Oynani yaratishda xatolik");
```

```
/* OpenGL oynasini yaratish */
```

```
maincontext = SDL_GL_CreateContext(mainwindow);
```

```
//oyna o'lchamini berish
```

```
glOrtho(0, 10, 15, 0, -1, 1);
```

```
/* Bufferni tozalash rangi */
```

```
glClearColor ( 0.93, 0.93, 0.93, 1.0 );
```

```
SDL_Event e;
```

```
//game loop
```

```
while( isGameRunning )
```

```
{
```

```
    //klaviatura bosilganligini tekshiradi
```

```
    if (SDL_PollEvent(&e))
```



```
{  
  
    if (e.type == SDL_KEYDOWN)  
    {  
  
        switch(e.key.keysym.sym)  
        {  
  
            case SDLK_ESCAPE:  
  
                isGameRunning = false;  
  
                break;  
  
  
            case SDLK_LEFT:  
  
                car_position = 0;  
  
                break;  
  
  
            case SDLK_RIGHT:  
  
                car_position = 1;  
  
                break;  
  
        }  
    }  
  
}
```

```
drawscene();
```

```
SDL_GL_SwapWindow(mainwindow);
```

```
}
```

```
//o'yin tugasa, ya'ni oyna yopilsa
```

```
isGameRunning = 0;
```

```
//tozalash va chiqish
```

```
SDL_GL_DeleteContext(maincontext);
```

```
SDL_DestroyWindow(mainwindow);
```

```
SDL_Quit();
```

```
return 0;
```

```
}
```

Xulosa

Men ushbu kurs loyixasini bajarish mobayanida C++ tili dasturining juda ko`p imkoniyatlari bilan tanishdim. C++ tili yordamida professional darajadagi modullar, shuningdek C++ tili yordamida har xil formalarga asoslangan turli xil darajadagi dasturlar yaratish usullari bilan tanishdim. Xullas, men ushbu kurs loyixasini yaratish mobayanida C++ tilining juda ko`p professional darajadagi dasturlar yaratish ko`nikmalarini egalladim.

Foydalanilgan Adabiyotlar.

1. Гради Буч. Объектно –ориентированной анализ и проектирование с примерами приложений на C++. Невский диалект, 560 стр, 2001 г.
2. Шмидский Я.К. Программирование на языке C++: Самоучитель. Учебное пособие. Диалектика. 361 стр, 2004 г.
3. Николаенко Д.В. Самоучитель по Visual C++.Спб, 2001.
4. www.arxiv.uz.
5. www.referat.uz