

Цэрэнбат Номин-Эрдэнэ
(20B1NUM0426)

ICSI201 - Объект хандлагат програмчлал

МУИС-ХШУИС
Мэдээллийн технологи

Лаборатори №9

Цэрэнбат Номин-Эрдэнэ

1. ОРШИЛ

Энэ лабораторийн ажлийн хүрээнд хийсвэр функц гэж юу болох, жинхэнэ хийсвэр функц, Функц дахин программчлах талаар тайлбарлаж бичих ба энэхүү мэдлэгээрээ бодлогуудаа бодно.

2. ЗОРИЛГО

- Лабораторийн ажлыг алдаагүй зөв хийж гүйцэтгэх.
- Хийсвэр эх классыг тодорхойлох
- Хийсвэр эх класаас TwoDimentional хийсвэр классыг удамшуулах
- Харилцан адилгүй талбайтай олон ширхэг тойрог, гурвалжин, тэгш өнцөгтүүд үүсгэх
- Удамшил ба устгагч функц ямар хамааралтай талаар судлах

3. ОНОЛЫН СУДАЛГАА

3.1 Хийсвэр функц гэж юу вэ? Тодорхойлолт, зарлалт, жишээ, давуу талыг тус тус бич.

Эх классыг зарлахдаа virtual түлхүүр үгийг Virtual void show(void); гэж хэрэглэнэ. Виртуал функц нь эх класс дотор зарлагдсан, үүссэн классаар дахин тодорхойлогдсон (overridden) гишүүн функцийг хэлнэ. Заагч эсвэл эх классынг ашиглан үүссэн классын объект руу хандахдаа тухайн объектын виртуал функцийг дуудаж, тухайн функцийг үүссэн классын хувилбарыг гүйцэтгэж болно.

Хийсвэр функцийг зөвхөн эх класс дотор зарлана.Эх ба удамших классууд ижил нэртэй функцтэй байвал эх классынхыг нь virtual гэж зарласнаар эх ба удамших классын аль нь ч зөв функцээ дуудах боломжтой болдог.

-Хийсвэр функц нь эх классын кодыг сольж програмчлахыг удамших класст зөвшөөрдөг.

-Эх классын байгуулагч функцээс бусад нь хийсвэр байж болно.

-Энэ тусгай үгийг ашигласнаар C++ олон удамших классын аль нь дуудагдаж байгааг таньж зөв функцийг дуудаж чадна.Ингэснээр өөр өөр объектын хувьд ижил нэртэй функцээр өөр өөр үр дүнд хүрэх боломжтой болно. Үүнийг polymorphism гэнэ.

C++ дээр хийсвэр функцийг зарлах:

```
Public virtual float piremetr() = 0;
```

3.2 Жинхэнэ хийсвэр функц гэж юу вэ? Тодорхойлолт, зарлалт, жишээ, давуу талыг тус тус бич.

Классын тодорхойлолт доторх эх загвартаа `=0` бүхий функцыг `pure abstract function` буюу жинхэнэ хийсвэр функц гэдэг. Ийм функцыг програмд бүрэн програмчлаж өгөөгүй байдаг ба классаас нь удамшиж үүсэх класс дотор дахин програмчлах ёстой. Жинхэнэ хийсвэр функц нь классын объект байж болохгүй.

`Virtual void perimetrvirtual(); = 0;` гэж зарлаж болно.

3.3 Функц дахин программчлах гэж юу вэ? Эх классын дахин программчилсан функцыг хүүхэд классын функц дотроос хэрхэн дууддаг вэ?

Эх класс дахь гишүүн функцыг нэр нь яг адил байхаар дахин тодорхойлохыг функц дахин програмчлах гэнэ. Дахин програмчлах удамших классын арга нь бичдэсийн хувьд эх класстай адилхан байх ёстой. Удамших класс нь эх классын зарим аргыг байгаагаар нь, заримыг нь дахин програмчлах замаар хэрэглэж болно. Эх класст дахин тодорхойлсон функцийг дахин програмчлах бол функц тус бүрийг дахин програмчлах эсэхээ сайн нягтлах хэрэгтэй.

3.4 Хийсвэр класс гэж юу вэ? Хэрхэн объект байгуулдаг вэ?

Жинхэнэ хийсвэр функцыг агуулсан классыг хийсвэр класс/`abstract class`/ гэнэ. Үүнээс объект байгуулж болохгүй. Харин удамшсан классынх нь объектыг үүсгэж болох ба ингэхийн тулд жинхэнэ хийсвэр функцыг заавал дахин програмчлах хэрэгтэй. Дотроо хийсвэр жинхэнэ функц агуулсан байдаг.

3.5 Удамшилд байгуулагч функц хэрхэн ашиглагддаг вэ? Удамшил ба байгуулагч функцийг хүрээнд үзсэн зүйлүүдийг бич.

Удамшлын үед байгуулагч функц дуудагдах дараалал: Хүүхэд классын объект байгуулагдах үед эхлээд эх классын байгуулагч функц дуудагдаж дууссаны дараа нь удамшсан классын байгуулагч функц дуудагдан ажиллана.

3.6 Удамшил ба устгагч функц хоёр ямар хамааралтай вэ?

Удамших классын объект устахад эхлээд удамших классын устгагч, дараагаар нь эх классын устгагч тус тус дуудагдан ажилладаг. Дарааллын хувьд удамшил нь эхлээд дараагаар нь эх класс нь дуудагдана.

4. ХЭРЭГЖҮҮЛЭЛТ

4.1 Lab07-д хийсэн дүрсүүдэд жинхэнэ хийсвэр функц нэмж

a. Shape класст примитив олох функцийг жинхэнэ хийсвэрээр зарлана,

b. TwoDimensionalShape класст талбай олох функцийг жинхэнэ хийсвэрээр зарлана.

4.2 Хийсвэр эх классын параметертэй, параметергүй байгуулагч функцийг хүүхэд класс бүрийн байгуулагч функцтэй давхар дууддаг болгоно.

4.3 Харилцан адилгүй талбайтай олон ширхэг тойрог, гурвалжин, тэгш өнцөгтүүд үүсгэж хооронд нь талбайгаар нь эрэмбэл.

```
1  #include<iostream>
2  #include<string.h>
3  #include<math.h>
4  #define pi 3.14
5  using namespace std;
6
7  class shape{
8
9      protected:
10         char *name;
11
12     public:
13         shape();
14         ~shape();
15         // Pure Virtual Function
16         virtual float primetr() = 0;
17
18 };
19
```

```
28
29 //_____ 2DSHAPE _____//
30
31 class TwoDShape : public shape{
32
33     protected:
34         int *x;
35         int *y;
36
37     public:
38         // Pure Virtual Function
39         virtual float area() = 0;
40
```

```
#include<iostream>
#include<string.h>
#include<math.h>
#define pi 3.14
using namespace std;

class shape{
```

```

protected:
    char *name;

public:
    shape();
    ~shape();
    // Pure Virtual Function
    virtual float primetr() = 0;
};

shape::shape(){
    name = new char[2];
    strcpy(name, "");
}

shape::~~shape(){
    delete name;
}

//_____ 2DSHAPE _____//

class TwoDShape : public shape{

protected:
    int *x;
    int *y;

public:
    // Pure Virtual Function
    virtual float area() = 0;
    virtual void read() = 0;
    virtual void print() = 0;
};

//_____ TOIROG _____//

class toirog: public TwoDShape{
private:
    int r, tx,ty;
public:
    float area();
    float primetr();

    toirog(){

```

```

        name = new char[7];
        strcpy(name, "toirog");
        x = new int[1];
        y = new int[1];
        r = 0;
    }

    ~toirog(){
        delete name;
        delete x;
        delete y;
    }

    void set_x(int _x){
        x=&_x;
        cout<<"x ni "<<*x<<" bolloo.\n";
    }

    void set_y(int _y){
        y=&_y;
        cout<<"y ni "<<*y<<" bolloo.\n";
    }

    void set_r(int _r){
        r=_r;
    }

    void read(){

        cout<<"Toirog coordinat x: ";
        cin>>tx;
        set_x(tx);
        cout<<"Toirog coordinat y: ";
        cin>>ty;
        set_y(ty);
        cout<<"Toirognii radius:";
        cin>>r;
        set_r(r);
    }

    void print(){
        cout<<"Toirognii area:"<<area()<<endl;
        cout<<"Toirgnii urt:"<<primetr()<<endl;
    }

```

```

};

```

```
float toirog::area(){
    return pi * (r * r);
}
```

```
float toirog::primetr(){
    return 2*pi*r;
}
```

```
//_____ KVADRAT _____//
```

```
class kvadrat: public TwoDShape{
private:
    float a;
    int r,ka,kx,ky;
public:
    float area();
    float primetr();

    kvadrat(){
        name = new char[7];
        strcpy(name,"kvadrat");
        x = new int[4];
        y = new int[4];
        a=0;
    }

    ~kvadrat(){
        delete name;
        delete x;
        delete y;
    }

    void set_a(float _a){
        a=_a;
    }
    void set_x(int _x){
        x[0] = _x;
        cout<<"x ni "<<x[0]<<" bolloo.\n";
        x[1] = _x + a;
        cout<<"x1 ni "<<x[1]<<" bolloo.\n";
        x[2] = _x + a;
        cout<<"x2 ni "<<x[2]<<" bolloo.\n";
        x[3] = _x;
```

```

        cout<<"x3 ni "<<x[3]<<" bolloo.\n";
    }

    void set_y(int _y){
        y[0] = _y;
        cout<<"y ni "<<y[0]<<" bolloo.\n";
        y[1] = _y;
        cout<<"y1 ni "<<y[1]<<" bolloo.\n";
        y[2] = y[3] = _y - a;
        cout<<"y2 ni "<<y[2]<<" bolloo.\n";
        cout<<"y3 ni "<<y[3]<<" bolloo.\n";
    }
    void read(){

        cout<<"\nKvadrat a:";
        cin>>ka;
        set_a(ka);
        cout<<"Kvadratnii koordinat x:";
        cin>>kx;
        set_x(kx);
        cout<<"Kvadratnii koordinat y:";
        cin>>ky;
        set_y(ky);
    }

    void print(){
        cout<<"\nKvadratnii area:"<<area()<<endl;
        cout<<"Kvadratnii primetr:"<<primetr()<<endl;
    }

};

float kvadrat::area(){
    return a * a;
}

float kvadrat::primetr(){
    return a*4;
}

//_____ GURVALJIN _____//

class Zgurvaljin: public TwoDShape{

```



```

private:
    float a;
    int ga,gx,gy;
public:
    float area();
    float primetr();

    Zgurvaljin(){
        name = new char[9];
        strcpy(name, "ZuwGurvaljin");
        x = new int[3];
        y = new int[3];
    }

    ~Zgurvaljin(){
        delete name;
        delete x;
        delete y;
    }

    void set_x(int _x){
        x[0] = _x;
        cout<<"x ni "<<x[0]<<" bolloo.\n";
        x[1] = _x + a;
        cout<<"x1 ni "<<x[1]<<" bolloo.\n";
        x[2] = _x;
        cout<<"x2 ni "<<x[2]<<" bolloo.\n";
    }
    void set_y(int _y){
        x[0] = _y;
        cout<<"y0 ni "<<y[0]<<" bolloo.\n";
        x[1] = _y - a;
        cout<<"y1 ni "<<y[1]<<" bolloo.\n";
        x[2] = _y - a;
        cout<<"y2 ni "<<y[2]<<" bolloo.\n";
    }
    void set_a(float _a){
        a=_a;
        cout<<"Gurvaljnii tal ni"<<a<<"bolloo.\n";
    }

    void read(){

        cout<<"\nGurvaljnii Tal:";
        cin>>ga;

```

```

        set_a(ga);
        cout<<"Zuv Gurvaljnii koordinat x:";
        cin>>gx;
        set_x(gx);
        cout<<"Zuv Gurvaljnii koordinat y:";
        cin>>gy;
        set_y(gy);

    }
    void print(){

        cout<<"\nZuv Gurvaljnii area:"<<area()<<endl;
        cout<<"Zuv Gurvaljnii primetr:"<<primetr()<<endl;

    }

};

float Zgurvaljin::area(){
    float h = sqrt(a*a-a/2*a/2);
    return h*a/2;
}

float Zgurvaljin::primetr(){
    return a*3;
}

int main(){

    int n, c, s, g;

    TwoDShape * sh[n];

    cout << "Heden toirog oruulah ve: ";
    cin >> c;

    cout << "Heden kvadrat oruulah ve: ";
    cin >> s;

    cout << "Heden gurvajin oruulah ve: ";
    cin >> g;

```

```

    for (int i = 0; i < c; i++) {
        sh[i] = new toirog;
        cout << "\n" << i + 1 << " -r Toirgiin medeelel\n";
        sh[i] -> read();

    }

    for (int j = 0; j < s; j++) {
        sh[j] = new kvadrat;
        cout << "\n" << j + 1 << " -r kvadratiin medeelel\n";
        sh[j] -> read();

    }

    for (int k = 0; k < g; k++) {
        sh[k] = new Zgurvaljin;
        cout << "\n" << k + 1 << " -r gurvaljingiin medeelel\n";
        sh[k] -> read();

    }

    cout << "\ntalbaigar n eremblem: \n";
    for (int z = 0; z < n; z++){
        cout << "Durs ";
        cout << "Dursiin S = " << sh[z] -> area() << endl;

    }
    return 0;
}

```