# Статик өгөгдөл, функц зарлах, тодорхойлох лабораторийн ажил (Лаборатори №8)

М.Солонго

ХШУИС, МКУТ-ийн оюутан, 20b1num2104@stud.num.edu.mn

# 1. ОРШИЛ

Эх класст статик хувьсагч зарлаж, статик хувьсагч боловсруулдаг статик функц тодорхойлон, ерөнхийлөл ашиглан объектыг зарлаж хэдэн объект үүсгэгдсэнийг тоолохын тулд жинхэнэ хийсвэр функц, хийсвэр классын онолын тодорхойлолтыг судалж ашиглан зорилгын дагуу код бичин хавсаргав.

# 2. ЗОРИЛГО

Статик өгөгдөл, функц тодорхойлон объектыг талбай болон периметрээр эрэмбэлэхийн тулд дараах зорилтуудыг тавьж ажилласан. Үүнд:

1. Эх класст статик хувьсагч зарлан гарааны утгыг оноох
2. Статик өгөгдөл боловсруулдаг статик функцийн тодорхойлолт хийх
3. 2DShape класст жинхэнэ хийсвэр функц зарлан хүүхэд класст функцийг дахин тодорхойлох
4. Объектыг талбай болон периметрээр эрэмбэлэхийн тулд ерөнхийлөл ашиглан 2DShape хаяган объект зарлан түүндээ square, triangle, circle санах ой үүсгэн статик өгөгдлийн тусламжтайгаар тоолон, эрэмбэлж дэлгэцэнд хэвлэж харуулах.

# 3. ОНОЛЫН СУДАЛГАА

## 3.1 this хувьсагч

This хувьсагч нь гишүүн функцийг хэрэглэж байгаа объектын хаягийг заадаг. Өөрөөр хэлбэл объектын хаяган хувьсагч.

## 3.2 Статик гишүүн өгөгдөл

Байгуулагч функц статик биш өгөгдлийг объект бүр дээр үүсгэдэг бол статик өгөгдөл нь бүх объект дундаа хэрэглэдэгээрээ онцлогтой. Тодорхойлохдоо класс тодорхойлолт дотор гишүүн өгөгдлийн өмнө бичиж өгдөг байна. Гарааны утга оноохдоо классын гадна бичиж өгнө.

## 3.3 Статик гишүүн функц

Статик функц нь зөвхөн статик гишүүн өгөгдлийг боловсруулдаг функц юм. Статик гишүүн өгөгдлийн адилаар класс дотор функцийн өмнө static түлхүүр үгийг бичиж өгнө.

# 4. ХЭРЭГЖҮҮЛЭЛТ

## 4.1. This хувьсагчийг комплиатор хэрхэн хэрэглэдэг талаар дэлгэрэнгүй судал

Объектоос гишүүн функцийг дуудах үед систем this хувьсагчид объектын хаягийг хийсний дараагаар дуудагдсан функц руу програмын удирдлагыг шилжүүлдэг. Ингэснээр цаашлаад функц дуудагдсан объектын хаягийг мэдэхээс гадна, this хувьсагчийг ашиглах боломжтой болдог.

## 4.2. Static гишүүн өгөгдөл хэрхэн зарладаг?

Классын тодорхойлолт дотор гарааны утга оноож болдоггүй учир классын гадна тодорхойлж өгдөг.

Class класс\_нэр{

Public:

Static төрөл гишүүн\_өгөгдөл\_нэр;

};

төрөл класс\_нэр::гишүүн\_өгөгдөл\_нэр=x;

## 4.3. Static гишүүн өгөгдөл мөчлөг ямар байдаг вэ?

Static өгөгдөл учир зарлах үед үүсэн програм ажиллаж дуусахад устана.

## 4.5. Static гишүүн өгөгдөлд хэрхэн хандах вэ? Объектоор эсвэл классаар дамжуулж хандах

Статик гишүүн өгөгдөлд заавал объектоор дамжуулан хандах шаардлагагүй шууд

Классын\_нэр::гишүүн өгөгдлийн\_нэр(статик) гэж дууддагаараа онцлог.

## 4.6. Static гишүүн өгөгдөл ямар ашигтай вэ?

Ямар ч класст харъяалагддаггүй бүх объектоос хандаж болдогоороо онцлогтой

## 4.7. Static гишүүн функцийг хэрхэн зарладаг, дууддаг вэ?

Зарлахдаа:

Class классын\_нэр{

Public:

Static төрөл гишүүн\_өгөгдлийн\_нэр;

Static төрөл функцийн\_нэр();

};

Төрөл класс\_нэр::функцийн\_нэр(){

...

}

## 4.8. Функц дахин тодорхойлох гэж юу вэ? Жишээгээр тайлбарла.

Эх классд байгаа функцийг хүүхэд класст дахин тодорхойлохыг хэлнэ.

Class 2dshape{

Public:

Float perimeter(){

Return a\*n;

}

};

Class square::public 2dshape{

Public:

Float perimeter(){

Return 4\*a

}

};

## 4.9. Жинхэнэ хийсвэр функц гэж юу вэ?

Жинхэнэ хийсвэр функц гэдэг нь их биеийн тодорхойлолтгүй буюу их бие нь NULL утгатай virtual төрөл функцийн\_нэр()=0; зарлалттай функцийг хэлнэ.

## 4.10. Хийсвэр класс гэж юу вэ? Хэрхэн ашигладаг вэ?

Жинхэнэ хийсвэр функцийг агуулсан классыг хийсвэр функц гэнэ. Хийсвэр классаас объект үүсгэж болдоггүй. Удамшиж байгаа класс функцийг дахин програмчлах шаардлага гарахаас гадна функцийг тодорхойлоогүй бол хүүхэд класс хийсвэр класс болдог.

Класс классын\_нэр(){

Virtual төрөл функцийн\_нэр=0;

}

# 5. ДҮГНЭЛТ

Статик хувьсагч, функцийн хэрэглээг танин мэдэж, давуу талыг код бичихдээ ашигтайгаар хэрэглэж сурсан.

## 6. АШИГЛАСАН МАТЕРИАЛ

1. Объект хандлагат технологийн С++ програмчлал, Ж.Пүрэв, 2008, Улаанбаатар.

2. [14.02 abstract class.mp4 (sharepoint.com)](https://nummta.sharepoint.com/sites/grp-oop-fall-2022461/_layouts/15/stream.aspx?id=%2Fsites%2Fgrp%2Doop%2Dfall%2D2022461%2FShared%20Documents%2FGeneral%2Fvideo%2Dlectures%2F14%2E02%20abstract%20class%2Emp4&referrer=Teams%2ETEAMS%2DELECTRON&referrerScenario=teamsSdk%2DopenFilePreview)

3. [10.03 function overriding.mp4 (sharepoint.com)](https://nummta.sharepoint.com/sites/grp-oop-fall-2022461/_layouts/15/stream.aspx?id=%2Fsites%2Fgrp%2Doop%2Dfall%2D2022461%2FShared%20Documents%2FGeneral%2Fvideo%2Dlectures%2F10%2E03%20function%20overriding%2Emp4&referrer=Teams%2ETEAMS%2DELECTRON&referrerScenario=teamsSdk%2DopenFilePreview)

# 7. ХАВСРАЛТ

Shape.h:

#ifndef \_\_shape\_\_

#define \_\_shape\_\_

class shape{                //shape үндсэн класс тодорхойлолт

    protected:              //private хандалтын түвшинтэй

        float a;            //талын урт, радиус хадгалах хувьсагч

        char \*name;         //нэр хадгалах хаяган хувьсагч

    public:                 //public хандалтын түвшинтэй

        shape(float a);     //дүрсийн нэрийг хадгалах хаяган хувьсагч

        ~shape();           //устгагч функц зарлалт

        static int number;  //статик хувьсагч зарлалт. нийт хэдэн объект үүсэж, устаж байгааг тоолно

        static void number\_ret();           //статик гишүүн өгөгдлийг хэвлэж харуулах функц

};

#endif

Shape.cpp

#include "shape.h"

#include<string.h>

#include<iostream>

using namespace std;

shape::shape(float a){      //байгуулагч функц тодорхойлолт

    this->a=a;              //a хувьсагчид параметрээр орж ирсэн утгыг онооно

    number++;               //статик хувьсагчийг нэмэгдүүлж байна.

}

int shape::number=0;        //статик хувьсагчид гарааны утга оноож байна.

shape::~shape(){

    cout<<"deleted";

    number--;               //статик хувьсагчийг хорогдуулж байна

}

void shape::number\_ret(){                            //статик функц тодорхойлолт

    cout<<"number: "<<number<<endl;                 //статик гишүүн өгөгдлийг хэвлэж харуулж байна

}

2d.h:

#include "shape.h"                          //shape.h header file холбож өгнө

#ifndef \_\_2d\_\_

#define \_\_2d\_\_

#include<string.h>

class twod: public shape{                   //shape ээс удамшсан 2Dshape класс тодорхойлолт

    public:                                 //public хандалтын түвшинтэй

        virtual float area()=0;             //ингэснээр удамшсан функцууд дахин тодорхойлолт хийж болно

        virtual float perimeter()=0;        //периметрийг удамшиж байгаа класс дахин тодорхойлно

        float \*x;                           //координатын х-ыг хадгалах хаяган хувьсагч

        float \*y;                           //координатын у-ыг хадгалах хувьсагч

        void show\_data(void);               //мэдээлэл дэлгэцэлж харуулах функц

        virtual void coordinate()=0;        //координатын цэгүүдийг олох функц. дахин тодорхойлох шаардлагатай

        twod(float a, int point);           //параметртэй байгуулагч функц

        ~twod();                            //устгагч функц

};

#endif

2d.cpp:

#include "2d.h"

#include <iostream>

#include<string.h>

using namespace std;

void twod::show\_data(void){                         //мэдээлэл харуулах функцийн тодорхойлолт

            cout<<name<<endl;

            cout<<"side: ";

            cout<<a<<endl;

            cout<<"area: ";

            cout<< area()<<endl;

            cout<<"perimeter: ";

            cout<< perimeter()<<endl;

        }

twod::twod(float a, int point):shape(a){            //параметртэй байгуулагч функцийн тодорхойлолт

    cout<<"created"<<endl;

    this->a = a;                                    //эх классын а хувьсагчид параметрээр дамжин орж ирсэн а хувьсагчийн утгыг дамжуулна

    this->x = new float[point];                     //2d shape-ын х хаяган хувьсагчид санах ой нөөцлөнө

    this->y = new float[point];                     //у хаяган хувьсагчид санах ой нөөцөлнө

    cout<<endl;

}

twod::~twod(){

    cout<<"deleted"<<endl;

    delete[] x;                                     //нөөцөлсөн санах ойг чөлөөлнө

    delete[] y;                                     //нөөцөлсөн санах ойг чөлөөлнө

}

Circle.h:

#include "2d.h"                  //2d.h header file холбож байна

#ifndef \_\_circle\_\_

#define \_\_circle\_\_

class circle:public twod{        //2dshape классаас удамшина

public:

   float area();                 //талбай олох функц

   float perimeter();            //периметр олох функц

   void coordinate();            //координатын цэгүүдийг олох функц

   circle(float a, float x1, float y1);//параметртэй байгуулагч функц зарлалт

   ~circle();

};

#endif

Circle.cpp:

#include "circle.h"             //circle.h header file холболт

#include <math.h>               //math сан холболт

#include<iostream>

using namespace std;

float circle::perimeter(){      //периметр функц тодорхойлолт

    return 2\*3.14\*a;            //периметрийг олон буцааж байна

}

float circle::area(){           //талбай олох функцийн тодорхойлолт

        return 3.14\*a\*a;        //талбайг олон буцааж байна

    }

void circle::coordinate(){

}

circle::circle(float a, float x1, float y1):twod(a,1){  //удамшиж байгаа класс руу а,1 дамжуулах байгуулагч функц тодорхойлолт

    name=new char[7];

    strcpy(name,"circle");

    cout << "circle area s: "<<this->area()<<endl;

    cout << "circle perimeter p: "<<this->perimeter()<<endl;

        this->x[0]=x1;           //x[0] массивт параметрээс х1 утгыг оруулна

        this->y[0]=y1;           //y[0] массивт параметрээс у1 утгыг оруулна

    cout << "x: "<<x[0]<<endl;

    cout << "y: "<<y[0]<<endl;

}

circle::~circle(){

    delete[] name;

}

Square.h:

#include "2d.h"                     //2d.h header file холбож байна

#ifndef \_\_square\_\_

#define \_\_square\_\_

class square:public twod{

   public:                          //2dshape классаас удамшина

      float area();                 //талбай олох функц

      float perimeter();            //периметр олох функц

      void coordinate();            //

      square(float a, float x1,float y1);           //координатын цэгүүдийг олох функц

      ~square();

};

#endif

Square.cpp:

#include <iostream>

using namespace std;

#include "square.h"                 //square.h header file холбож байна

    float square::perimeter(){      //периметр функц тодорхойлолт

        return 4\*a;                 //периметрийг олон буцааж байна

    }

    float square::area(){           //талбай олох функцийн тодорхойлолт

        return a\*a;                 //талбайг олон буцааж байна

    }

    void square::coordinate(){      //координатын цэгүүдийг олох функц тодорхойлолт

        x[1]=x[0]+a;                //х у цэгүүдийг олж байна

        y[1]=y[0];

        x[2]=x[0]+a;

        y[2]=y[0]-a;

        x[3]=x[0];

        y[3]=y[0]-a;

    }

    square::square(float a, float x1, float y1):twod(a, 4){//удамшсан байгуулагч функцийн тодорхойлолт, эх класс руу а,4 дамжуулна

        name=new char[7];                                   //нэр хаяган хувьсагчид 7 хэмжээтэй санах ой нөөцөлж байна

        strcpy(name,"square");                              //square гэсэн үгийг нөөцөлсөн санах ой руу хуулж өгч байна

        cout << "square constructor s: " << this->area() << endl;

        cout << "square perimeter p: " << this->perimeter() << endl;

        this->x[0]=x1;

        this->y[0]=y1;

        coordinate();

        for(int i=0;i<4;i++){

            cout << "x: "<<x[i]<<endl;

            cout << "y: "<<y[i]<<endl;

        }

    }

    square::~square(){

        delete[] name;

    }

Triangle.h:

#include "2d.h"                  //2d.h header file холбож байна

#ifndef \_\_triangle\_\_

#define \_\_triangle\_\_

class triangle:public twod{        //2dshape классаас удамшина

public:

   float area();                 //талбай олох функц

   float perimeter();            //периметр олох функц

   void coordinate();            //координатын цэгүүдийг олох функц

   triangle(float a, float x1, float y1);

   ~triangle();

};

#endif

Triangle.cpp:

#include "triangle.h"               //triangle.h header file холбож байна

#include <math.h>

#include<iostream>

using namespace std;

float triangle::perimeter(){        //периметр функц тодорхойлолт

    return 3\*a;                     //периметрийг олон буцааж байна

}

float triangle::area(){             //талбай олох функцийн тодорхойлолт

    float h=sqrt(a\*a-a/2\*a/2);      //гурвалжны өндөрийг олж байна

    return a\*h/2;                   //талбайг олон буцааж байна

}

void triangle::coordinate(){        //координатын цэгийг олох функц тодорхойлолт

    float h=sqrt(a\*a-a/2\*a/2);

    x[1]=x[0]+a/2;          //х у цэгүүдийг олж байна

    y[1]=y[0]-h;

    x[2]=x[0]-a/2;

    y[2]=y[0]-h;

}

triangle::triangle(float a, float x1, float y1):twod(a,3){

        name=new char[9];

        strcpy(name,"triangle");

        cout << "triangle area s: " << this->area() << endl;

        cout << "triangle perimeter p: " << this->perimeter() << endl;

        this->x[0]=x1;

        this->y[0]=y1;

        cout << "coordinate"<<endl;

        coordinate();

        for(int i=0;i<3;i++){

            cout << "x: "<<x[i]<<endl;

            cout << "y: "<<y[i]<<endl;

        }

}

triangle::~triangle(){

    delete[] name;

}

Main.cpp:

#include "circle.h"                         //circle.h file холбож байна

#include "triangle.h"                       //triangle.h file холбож байна

#include "square.h"                         //square.h file холбож байна

#include<iostream>                          //iostream сан холбож байна

#include<string.h>

using namespace std;

main(){

    cout<<"uusgeh dursnii too: ";

    int n;

    cin>> n;

    twod \*shapes[n];

    twod::number\_ret();

    for(int i=0;i<n;i++){

label1: char name1[20];

        cout<<"shape: ";

        cin>>name1;

        float a;

        cout<<"a: ";

        cin>>a;

        float x,y;

        cout<<"x: ";

        cin>>x;

        cout<<"y: ";

        cin>>y;

        if(strcmp(name1,"square")==0){

            shapes[i]=new square(a,x,y);

            twod::number\_ret();

            //square s1(a,x,y);

        }else if(strcmp(name1,"triangle")==0){

            shapes[i]=new triangle(a,x,y);

            twod::number\_ret();

            //triangle s2(a,x,y);

        }else if(strcmp(name1,"circle")==0){

            shapes[i]=new circle(a,x,y);

            twod::number\_ret();

            //circle s3(a,x,y);

        }else{

            cout<<"square, triangle, circle bicne uu "<<endl;

            goto label1;

        }

    }

    for(int i=0;i<n;i++){

        for(int j=0;j<n-1;j++){

            if(shapes[j]->area()>shapes[j+1]->area()){

                twod \*tmp;

                tmp=shapes[j];

                shapes[j]=shapes[j+1];

                shapes[j+1]=tmp;

            }

        }

    }

    cout<<"----------"<<endl<<"sorted area"<<endl<<"----------"<<endl;

    for(int i=0;i<n;i++){

        shapes[i]->show\_data();

    }

    for(int i=0;i<n;i++){

        for(int j=0;j<n-1;j++){

            if(shapes[j]->perimeter()>shapes[j+1]->perimeter()){

                twod \*tmp;

                tmp=shapes[j];

                shapes[j]=shapes[j+1];

                shapes[j+1]=tmp;

            }

        }

    }

    cout<<"----------"<<endl<<"sorted perimeter"<<endl<<"----------"<<endl;

    for(int i=0;i<n;i++){

        shapes[i]->show\_data();

    }

    shape::number\_ret();

}