# Хийсвэр Класс, Функц Тодорхойлох Лабораторын Ажил (Лаборатори №8)

С.Төгөлдөрсайхан

ХШУИС, МКУТ, Компьютерийн ухаан, 16B1SEAS0935@stud.num.edu.mn

# **1. ОРШИЛ.**

Обьект хандлагат технологийн үндсэн суурь ойлголт болох удамшил, удамшлын модыг ашиглан хоёр хэмжээст геомерт дүрсүүдийн классыг байгуулна.

# **2. ЗОРИЛГО**

Дүрс эх класс болон түүний дэд классууд болох хоёр хэмжээст класс, тойрог, квадрат, ижил талт гурвалжин классуудыг, мөн хийсвэр класс, функцуудыг шаардлагын дагуу JAVA прорамчлалын хэл дээр тодорхойлно. Үүний тулд дараах зорилтуудыг тавьж ажилласан:

1. Эх класс тодорхойлох,
2. Хоёр хэмжээст дэд класс тодорхойлох,
3. Тойрог класс тодорхойлох,
4. Квадрат класс тодорхойлох,
5. Гурвалжин класс тодорхойлох
6. Хийсвэр класс, функц тодорхойлох

# **3. ОНОЛЫН СУДАЛГАА**

## 3.1 Хийсвэр функц

Функцийн хаяг нь 0 буюу тодорхойгүй зөвхөн зарласан функцийг хийсвэр функц гэнэ. Хийсвэр арга (abstract method) нь их биегүй аргыг хэлнэ. Эх классд тодорхойлогдсон хийсвэр аргын хэрэгжилтийг хүү классууд нь өөр өөрийнхөөрөө хэрэгжүүлдэг. Тэгэхлээр нэг буюу түүнээс олон хийсвэр арга агуулах классыг abstract class гэнэ.

public abstract double findPerimeter();

public abstract findArea();

## 3.2 Функц дахин програмчлах

Эх классын функцийг хүүхэд классд бичдэс нь ижилээр дахин бичихийг/тодорхойлохыг функц дахин програмчлах гэнэ. (OverRiding) Эх классд дахин програмчлах гэж функцийн өмнө JAVA програмчлалын хэлэнд abstract, C++ програмчлалын хэлэнд virtual гэсэн түлхүүр үгсийг ашиглана.

class Animal {  
 public void move() {  
 System.out.println("Animals can move");  
 }  
}  
  
class Dog extends Animal {  
 public void move() {  
 super.move(); // invokes the super class method  
 System.out.println("Dogs can walk and run");  
 }  
}  
  
public class TestDog {  
  
 public static void main(String args[]) {  
 Animal b = new Dog(); // Animal reference but Dog object  
 b.move(); // runs the method in Dog class  
 }  
}

## 3.3 Хийсвэр класс

Жинхэнэ хийсвэр функцийг агуулсан классыг хийсвэр класс буюу abstract class гэнэ. Хийсвэр классаас обьект үүдсэггүй. Харин түүнээс удамшсан классд хийсвэр функцийг нь дахин програмчилсны дараа удамшсан классын обьектыг байгуулж болно.

Хийсвэр класс нь дараах онцлогтой. Үүнд:

* Бүрэн бус класс бөгөөд түүнээс заавал дэд класс удамшина.
* Дор хаяж нэг буюу түүнээс олон хийсвэр арга агуулсан байх.
* Түүний обьектыг үүсгэх боложгүй.
* Түүний хүү класс нь түүний хийсвэр аргуудыг биелүүлнэ эсвэл түүний хүү класс нь өөрөө хийсвэрээр зарлагдах гэсэн онцлог шинжүүдтэй.

Хийсвэр классын энгийн жишээ тодорхойлбол:

abstract class Shape {

public double getArea(){ return 0.0; }

public double getVolume(){ return 0.0; }

public abstract String getName();

}

## 3.4 Удамшил ба байгуулагч функц

Хүүхэд классын обьектыг байгуулах үед эх классын анхдагч байгуулагч эхлээд дуудагддаг ба үүний дараа хүүхэд классын байгуулагч дуудагдана. Хүүхэд классын параметртэй байгуулагчийг тодорхойлохдоо аргуументуудыг эх класс руу дамжуулна. Эх классаас удамшиж ирсэн гишүүн өгөгдлийг эх классын гишүүн функцээр нь утга оноолгуулах хэрэгтэй.

## 3.5 Удамшилд устгагч функц

Удамших классын обьект устахад эхлээд удамших классын устгагч, дараагаар нь эх классын устгагч тус тус дуудагдана. Нэмэлт ажил хийлгэхгүй бол устгагч функцийг програмд тодорхойлох шаардлага байдаггүй ба ийм тохиолдолд анхдагч устгагчийг систем өөрөө үүсгэдэг.

# **4. ХЭРЭГЖҮҮЛЭЛТ**

1. Өгөгдсөн бодлогуудыг бодоход ашигласан алгоритмууд:

//Дүрсийн нэр

protected String name;

//Дүрсийн х координат

protected double x;

//Дүрсийн y координат

protected double y;

//Дүрс классын классын байгуулагч

public Shape(String name, double x, double y){

System.out.println("Constructing a Shape");

//Дүрсийн нэрэнд утга олгох

setName(name);

//Дүрсийн координатад утга олгох

setCoordinates(x, y);

}

Дээрх хэрэгжүүлэлт нь байгуулах гэж буй гурвалжин, дөрвөлжин, квадрат дүрсүүдийн бүгдэнд байх шинк болох нэр, координат юм. Супер классын байгуулагч нь дээрх нэр, координатуудад утга оноох үүрэгтэй.

//Дүрсийн хүрээний урт олох функц

public abstract double findPerimeter(){

double perimeter; //perimeter value

perimeter = this.edges \* this.length;

return perimeter;

}

//Дүрсийн талбай олох хийсвэр функц

public abstract findArea();

Дээрх хэрэгжүүлэлтээр супер классд тодорхойлогдсон хийсвэр функц болох дүрсийн хүрээний уртыг олох функцийн тодорхойлолтыг хийж өгсөн байна. Үүнд ирмэгийн уртыг дүрсийн талаар үржүүлсэнээр приметрийг олсон. Мөн талбай олох хийсвэр функцийг хоосон орхиж дүрс бүрийн хувьд онцлог шинжээс хамааруулан дахин програмчилж тухайн дүрсийн талбайг олох болно.

* return this.length\*this.length\*0.6; //Гурвалжин
* return this.length \* this.length; //Квадрат
* return Math.PI \* this.length \* this.length; //Тойрог

Дээрх хэрэгжүүлэлтээр дүрс бүрийн хувьд талбайг өөрсдийн онцлогт тохируулан дахин програмчилсаныг харууллаа.

Өгөгдсөн бодлогыг бодоход дээрх алгоритмыг ашиглан бодсон ба Дүрс нэртэй классыг тодорхойлж, түүндээ гишүүн өгөгдөл болон гишүүн функцуудыг өгөгдсөн шаардлагын дагуу загварчлан оруулсан. Хэрэгжүүлэлтийн нарийн төвөгтэй асуудлыг кодонд коммент хэлбэрээр оруулсан тул дэлгэрэнгүйг хавсралтаас харна уу.

# **5. ДҮГНЭЛТ**

Дээрх ажлын хүрээнд обьект хандлагат технологийн үндсэн ойлголтуудын нэг болох удамшлын шинж чанарыг ашиглан тойрог, гурвалжин, квадрат гэх мэт дүрсүүдийн классуусыг хийсвэрээр тодорхойлж өгөгдсөн шаардлагын дагуу үүгэсэн дүрсүүдийнхаа обьектуудыг талбайгаар нь эрэмбэлэх ажил хийж гүйцэтгэлээ. Классуудыг байгуулж байх явцад эх классаас өөр онцлог шинжээ хадгалж үлдэхийн тулд функц болон классыг хийсвэрээр зарлаж өгөх нь зөв шийдэл болж өгсөн мөн удамшил ашигласан учир мөр бичлэг хэмнэх, алдаа гарахаас сэргийлэх, цаг хугацаа хэмнэх гэх мэт давуу талууд гарлаа.

# **6. АШИГЛАСАН МАТЕРИАЛ**

1. JAVA2 Програмчлалын Хэл, Ш.Нямбаа, 2001, Улаанбаатар

2. Pointer, <https://www.tutorialspoint.com/cplusplus/cpp_pointers.htm>

# **7. ХАВСРАЛТ**

|  |
| --- |
| public abstract class Shape {  //Дүрсийн нэр   protected String name;  //Дүрсийн х координат   protected double x;  //Дүрсийн y координат   protected double y;   //Дүрс классын классын байгуулагч   public Shape(String name, double x, double y){  System.out.println("Constructing a Shape");  //Дүрсийн нэрэнд утга олгох   setName(name);  //Дүрсийн координатад утга олгох  setCoordinates(x, y);  }  //Дүрсийн нэрэнд утга олгох функц   public void setName(String name){  this.name = name;  }  //Дүрсийн координатад утга олгох функц  public void setCoordinates(double x, double y){  this.x = x;  this.y = y;  }  //Дүрсийн нэрийг буцаах функц   public String getName(){  return name;  }  //Дүрсийн координатыг хэвлэх функц  public void getCoordinates(){  System.out.println("X: " + this.x + " Y: " + this.y);  }  //Приметр олох хийсвэр функц  public abstract double findPerimeter(); }  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  public abstract class TwoDimensionalShape extends Shape {  //Дүрсийн талын урт  protected double length;  //Дүрсийн ирмэгийн тоо  protected int edges;  //Дүрсийн бусад координатууд  protected double sX[];  protected double sY[];  protected double tX[];  protected double tY[];  //Хоёр хэмжээст классын байгуулагч функц  public TwoDimensionalShape(String name, double x, double y, double . len, int edges){  //Эх классынхаа байгуулагчид нэр, координатаа дамжуулах  super(name, x, y);  //Координатуудад утга олгох  setCoordinates(x, y, len, edges);  }  //Координатуудад утга олгох функц (гурвалжин, тойрог, квадрат)  setCoordinates(double x, double y, double len, int edges){  //Хэрэв ирмэг нь 4 бол энэ дүрс Квадрат  if(edges == 4){  //Доорход квадратын 4 оройн координатад утга олгож байна.  this.sX = new double[4];  this.xY = new double[4];  this.sX[0] = x;  this.sX[1] = x + len;  this.sX[2] = x;  this.sX[3] = x + len;  this.sY[0] = y;  this.sY[1] = y;  this.sY[2] = y - len;  this.sY[3] = y - len;  }//Хэрэв ирмэг нь 3 бол энэ дүрс Гурвалжин  else if(edges == 3){  //Доорход гурвалжны 3 оройн координатад утга олгож байна.  this.tX = new double[3];  this.tY = new double[3];  // sqrt(3)/2 = 1.2 | h = a\*(1.2)  this.tX[0] = x;  this.tX[1] = x - len/2;  this.tX[2] = x + len/2;  this.tY[0] = y;  this.tY[1] = y - len\*1.2;  this.tY[2] = y - len\*1.2;  }//Бусад тохиолдолд энэ дүрс Тойрог  else{  this.x = x;  this.y = y;  }  }  //Координатууудыг хэвлэх функц  public void getCoordinates(int edges){  //Ирмэг 4 бол энэ дүрс Квадрат болно  if(this.edge == 4){  for(int i = 0; i < 4; i++)  System.out.println("X:"+this.sX[i]+"Y:"+this.sY[i]);  }//Ирмэг 3 бол энэ дүрс гурвалжин болно  else if(this.edge == 3){  for(int i = 0; i < 3; i++)  System.out.println("X:"+this.tX[i]+"Y:"+this.tY[i]);  }//Бусад үед энэ дүрс Тойрог болно  else{  System.out.println("X:"+this.x+"Y:"+this.y);  }  }  //Дүрсийн хүрээний урт олох функц  public abstract double findPerimeter(){  double perimeter; //perimeter value  perimeter = this.edges \* this.length;  return perimeter;  }  //Дүрсийн талбай олох хийсвэр функц  public abstract findArea();  }  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  public class Triangle extends TwoDimensionalShape {  //Гурвалжин дүрсийн байгуулагч  public Triangle(double x, double y, double len){  //Эх классруугаа нэр, координат, урт, ирмэгийн тоог дамжуулна.  super("Triangle", x, y, len, 3);  }  //Гурвалжны хүрээний урт олох функц  public double findPerimeter(){  return this.length\*3;  }  //Гурвалжны талбай олох функц  @Override  public double findArea(){  return this.length\*this.length\*0.6;  }  }  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  public class Square extends TwoDimensionalShape {  //Квадрат дүрсийн байгуулагч  public Square(double x, double y, double len){  //Эх классруугаа нэр, координат, урт, ирмэгийн тоог дамжуулна.  super("Square", x, y, len, 4);  }  //Квадратын хүрээний урт олох функц  public double findPerimeter(){  return this.length\*4;  }  //Квадратын талбай олох функц  @Override  public double findArea(){  return this.length \* this.length;  }  }  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  public class Circle extends TwoDimensionalShape {  //Тойрог дүрсийн байгуулагч  public Circle(double x, double y, double len){  //Эх классруугаа нэр, координат, урт, ирмэгийн тоог дамжуулна.  super("Circle", x, y, len, 1);  }  //Тойргийн хүрээний урт олох функц  public double findPerimeter(){  //Хүрээний урт = 2\*PI\*r  return 2 \* Math.PI \* this.length;  }  //Тойргийн талбай олох функц  @Override  public double findArea(){  //Талбай = PI\*r^2  return Math.PI \* this.length \* this.length;  }  }  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  public class MainShapeTest {  public static void main(String[] args){  //9 ширхэг дүрсийн хаягийг Shape төрлөөр үүсгэж байна.  Shape[] testShapes = new TwoDimensionalShape[9];  //Дүрсүүдийг эрэмлэх үед түр зуурын хаягийг хадгалах  Shape[] tempObject = new TwoDimensionalShape[1];  //Доор нийт 9 ширхэг обьект үүсгэж байна.  testShapes[0] = new Triangle(1, 1, 1);  testShapes[1] = new Triangle(4, 3, 4);  testShapes[2] = new Triangle(6, 2, 3);  testShapes[3] = new Square(6, 1, 3);  testShapes[4] = new Square(2, 6, 7);  testShapes[5] = new Square(1, 5, 3);  testShapes[6] = new Circle(8, 2, 3);  testShapes[7] = new Circle(2, 3, 5);  testShapes[8] = new Circle(9, 2, 6);  //Дүрсүүдийн талбайг хүснэгтэнд хуулж авч байна.  double[] tempArray = new double[9];  for(int i = 0; i < 9; i++)  tempArray[i] = testShapes[i].findArea();  //Хуулж авсан хүснэгтээ эрэмбэлж байна.  for(int i = 0; i < 9; i++){  for(int j = 0; j < 9; j++){  if(tempArray[j] > tempArray[j + 1]){  //Обьектын хаягуудыг солих үйлдэл  tempObject[0] = testShapes[j+1];  testShapes[j+1] = testShapes[j];  testShapes[j] = tempObject[0];  }  } |

} } }