Лабораторна робота 5  
 з дисципліни «Об’єктно-орієнтоване програмування»

Тема роботи: «Класи і об’єкти. Конструктори. Основи перевантаження операцій»  
 Варіант 4

Виконав студент

групи КН-22

Пашковський П. В.

Викладач:

Красовська Г. В.

Клас Complex має поля:

* Private int **d1 -** довжина
* Private int **d2 -** ширина

та методи:

* Public double getInfo() - для зведення числа у квадрат.
* Public void getParemeter() – для зчитування значення з клавіатури
* Public void getSquare()



Код програми(java):

package com.fit.labs.lab05.entity;  
  
  
public abstract class Quadrangle {  
 private int d1;  
 private int d2;  
 private double angleSin;  
  
 public Quadrangle(int d1, int d2, double angleSin) {  
 this.d1 = d1;  
 this.d2 = d2;  
 this.angleSin = angleSin;  
 }  
  
 public int getD1() {  
 return d1;  
 }  
  
 public void setD1(int d1) {  
 this.d1 = d1;  
 }  
  
 public int getD2() {  
 return d2;  
 }  
  
 public void setD2(int d2) {  
 this.d2 = d2;  
 }  
  
 public double getAngle() {  
 return angleSin;  
 }  
  
 public void setAngle(double angleSin) {  
 this.angleSin = angleSin;  
 }  
  
 public abstract double getSquare();  
  
 public abstract double getPerimeter();  
  
 public String getInfo() {  
 return "Quadrangle = {" +  
 "d1=" + d1 +  
 ", d2=" + d2 +  
 ", angle in sin=" + angleSin +  
 ", square=" + getSquare() +  
 ", perimeter=" + getPerimeter() +  
 "}";  
 }  
}

package com.fit.labs.lab05.entity;  
  
public class Parallelogram extends Quadrangle{  
 public Parallelogram(int d1, int d2, double angleSin) {  
 super(d1, d2, angleSin);  
 }  
  
 @Override  
 public double getSquare() {  
 return (double) getD1() \* (double) getD2() / getAngle();  
 }  
  
 @Override  
 public double getPerimeter() {  
 return 2 \* getD1() + 2 \* getD2();  
 }  
}

package com.fit.labs.lab05.entity;  
  
public class Rectangle extends Quadrangle {  
 public Rectangle(int d) {  
 super(d, d, 1);  
 }  
  
 @Override  
 public double getSquare() {  
 return (double) getD1() \* (double) getD2() / 2;  
 }  
  
 @Override  
 public double getPerimeter() {  
 return 8 \* getD1() \* Math.*sqrt*(2);  
 }  
}