Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №1

по дисциплине «Платформонезависимое

программирование»

Тема: «Классы и объекты. Пакеты.»

Выполнил:

студент гр. ПРИ-123

А.Ц. Нямаа

Приняла:

Ст. пр. кафедры ИСПИ

О.Н. Шамышева

Владимир, 2025 г.

ЦЕЛЬ РАБОТЫ:

Освоить базовые принципы ООП. Получить практические навыки по работе с классами и объектами на платформе java. Научиться проводить декомпозицию приложений с использованием пакетов.

ЗАДАНИЕ 1 (вариант 12).

Выполнить пример создания приложения с пакетами согласно методическому указанию.

ВЫПОЛНЕНИЕ РАБОТЫ:

В начале выполнения данного задания я создал необходимую физическую структуру, которую взял из примера из методического указания.

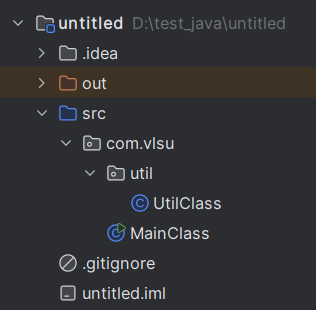


Рисунок 1. Физическая структура проекта.

Далее использовал в своей программе прикрепленный к примеру код и, при помощи командной строки, скомпилировал его.

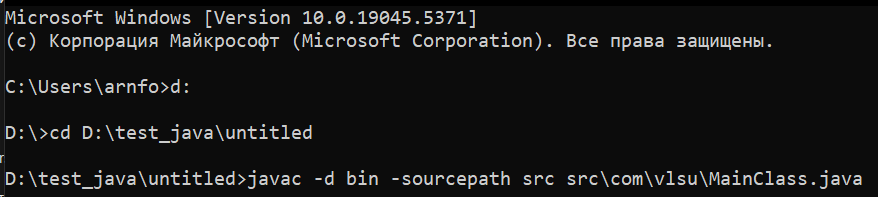


Рисунок 2. Компилирование программы через командную строку.

Выполнив скомпилированный код также при помощи командной строки, получил результат.

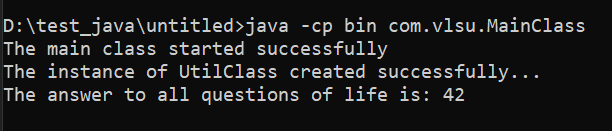


Рисунок 3. Результат выполнения программы.

ЗАДАНИЕ 2 (вариант 12).

Необходимо реализовать приложение, работающее с кругами. Первый вспомогательный класс – точка – определяется параметрами *x* и *y* – координатами на плоскости. Второй вспомогательный класс – круг – определяется параметрами *p* и *R* – точкой центра круга и радиусом. Помимо этого, должен быть метод inside, который в качестве аргумента принимает точку. Он должен возвращать *true*, если точка находится внутри круга, и *false* в противном случае. Основной класс должен запросить у пользователя координаты центра круга и его радиус, после чего создать объект круга. После этого он должен запросить у пользователя координаты точки и вывести ответ – находится ли данная точка внутри круга.

ВЫПОЛНЕНИЕ РАБОТЫ:

В ходе выполнения задания выше я создал исполняющий класс MainClass с двумя методами: main (основной метод, являющийся точкой входа в программу) и inside (согласно заданию, проверяет, находится ли точка в круге). Также написал два вспомогательных класса Circle и pointCircle, которые запрашивают у пользователя необходимые данные (координаты центра круга и произвольной точки, а также радиус). Физическая структура выглядит следующим образом:

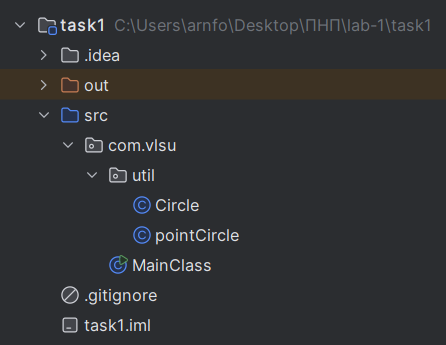


Рисунок 4. Физическая структура программы.

Листинг основного класса MainClass.java:

package com.vlsu;  
  
import com.vlsu.util.Circle;  
import com.vlsu.util.pointCircle;  
  
public class MainClass {  
 public static void main(String [] args){  
 Circle circle = new Circle();  
  
 float[] centerPoint = circle.getCenterPoint();  
 float radius = circle.getRadius();  
  
 System.out.println("Круг: координаты центральной точки - (" + centerPoint[0] + ", " + centerPoint[1] + "), " +  
 "радиус - " + radius);  
  
 float[] point = pointCircle.readPoint();  
 System.out.println("Точка: координаты точки - (" + point[0] + ", " + point[1] + ")");  
  
 System.out.print("\nОтвет: ");  
 boolean pointIn = inside(centerPoint, radius, point);  
 if (pointIn) {  
 System.out.println("точка находится внутри круга");  
 } else {  
 System.out.println("точка не находится внутри круга");  
 }  
 }  
  
 private static boolean inside(float[] centerPoint, float radius, float[] point){  
 if ((Math.pow(point[0] - centerPoint[0], 2) + Math.pow(point[1] - centerPoint[1], 2)) <= Math.pow(radius, 2)){  
 return true;  
 }  
 return false;  
 }  
}

Листинг вспомогательного класса Circle.java:

package com.vlsu.util;  
  
import java.util.Scanner;  
  
public class Circle {  
  
 private float[] centerPoint = new float[2];  
 private float Radius;  
  
 public Circle(){  
 Scanner in = new Scanner(System.in);  
  
 System.out.println("Круг: введите координаты центра круга (x, y): ");  
 centerPoint[0] = in.nextFloat();  
 centerPoint[1] = in.nextFloat();  
  
 System.out.println("Круг: введите радиус: ");  
 Radius = in.nextFloat();  
 }  
  
 public float[] getCenterPoint() {return centerPoint;}  
 public float getRadius() {return Radius;}  
}

Листинг вспомогательного класса pointCircle.java:

package com.vlsu.util;  
  
import java.util.Scanner;  
  
public class pointCircle {  
 public static float[] readPoint(){  
 Scanner in = new Scanner(System.in);  
  
 System.out.println("\nВведите координаты точки (x, y): ");  
 return new float[] {in.nextFloat(), in.nextFloat()};  
 }  
}

Тестирование программы на java и сравнение результатов в графическом калькуляторе desmos:

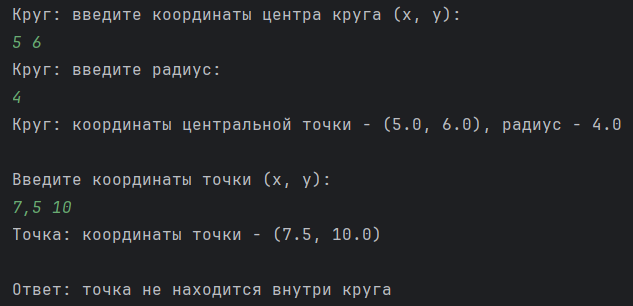


Рисунок 5. Выполнение программы. Тест 1.

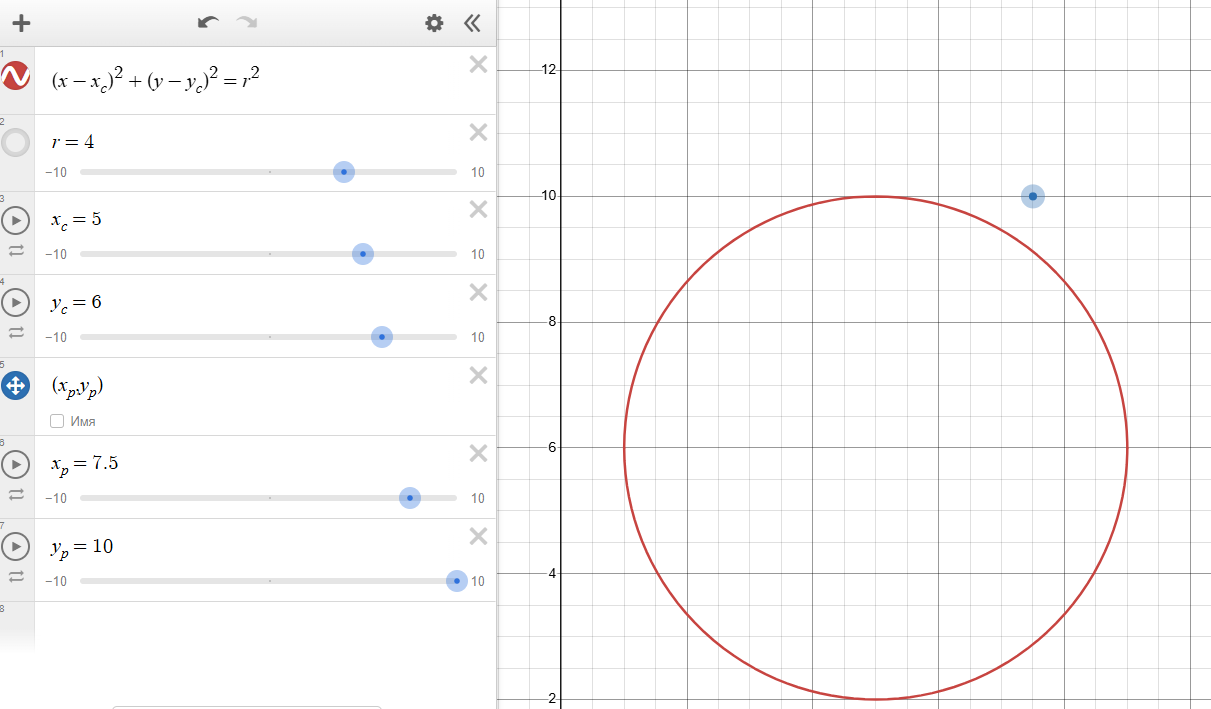


Рисунок 6. Проверка результатов. Тест 1.

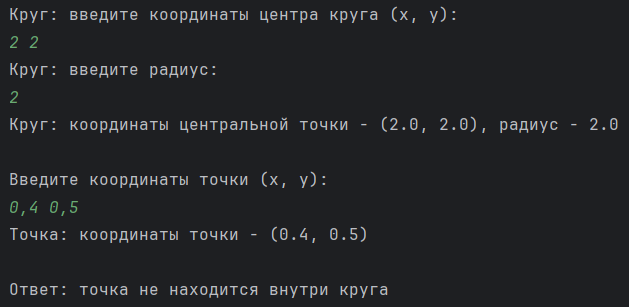


Рисунок 7. Выполнение программы. Тест 2.

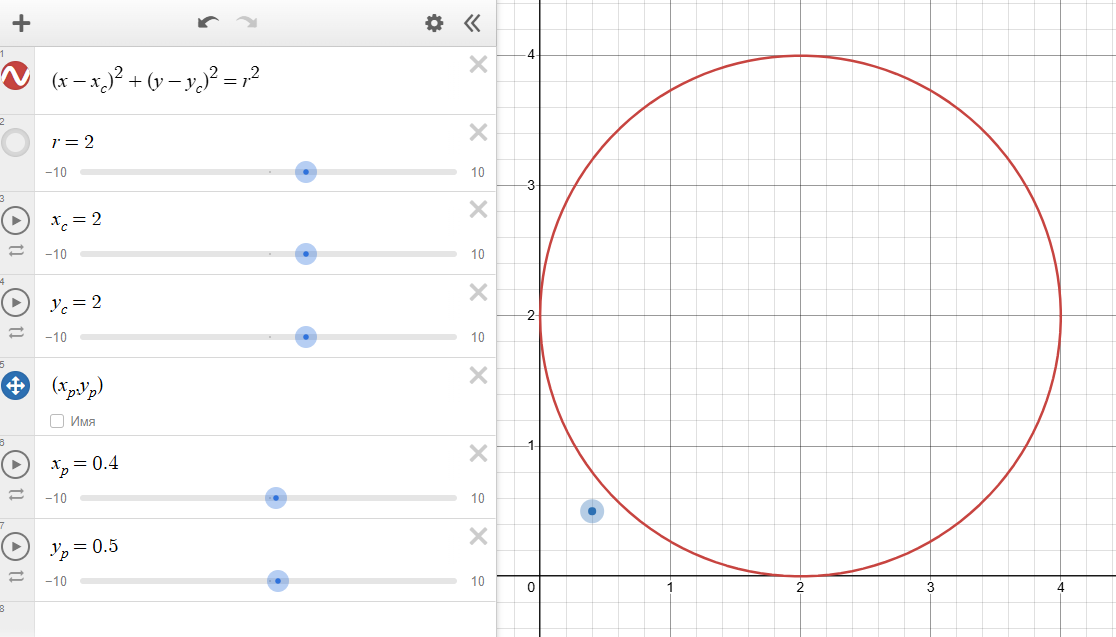


Рисунок 8. Проверка результатов. Тест 2.

ЗАДАНИЕ 3 (вариант 12).

Реализовать классы Rectangle и Point. Класс Point определяется двумя координатами точки, класс Rectangle определяется точками левого верхнего угла и правого нижнего угла.

ВЫПОЛНЕНИЕ РАБОТЫ:

Для выполнения данного задания я написал основной класс Main, где создаются объекты и выводится информация о них, а также два дополнительных класса Rectangle и Point, в которых создал конструкторы. Физическая структура выглядит следующим образом:

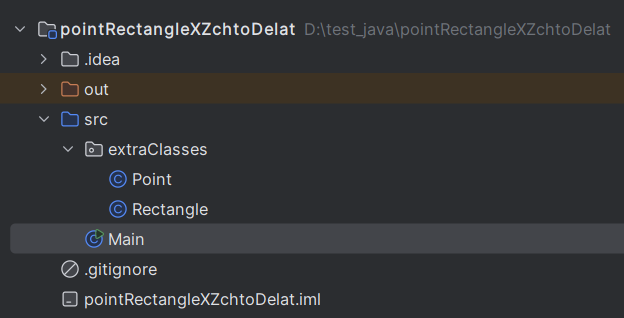


Рисунок 9. Физическая структура проекта.

Листинг исполняющего класса Main.java

import extraClasses.\*;  
import java.util.Arrays;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.in);  
  
 System.out.println("Point: введите координаты: ");  
 Point point = new Point(  
 new float[]{in.nextFloat(), in.nextFloat()}  
 );  
  
 System.out.println("Rectangle: введите координаты левого верхнего угла: ");  
 float[] leftTop = new float[]{in.nextFloat(), in.nextFloat()};  
  
 System.out.println("Rectangle: введите координаты правого нижнего угла: ");  
 float[] rightBottom = new float[]{in.nextFloat(), in.nextFloat()};  
  
 Rectangle rectangle = new Rectangle(leftTop, rightBottom);  
  
 System.out.println("Point: координаты точки - (" + point.xCoor + ", " + point.yCoor + ")");  
 System.out.println("Rectangle: координаты левого верхнего угла - " + Arrays.toString(rectangle.leftTop) +  
 ", координаты правого нижнего угла - " + Arrays.toString(rectangle.rightBottom) + ")");  
 }  
}

Листинг дополнительного класса Rectangle.java

package extraClasses;  
  
public class Rectangle {  
 public float[] leftTop = new float[2];  
 public float[] rightBottom = new float[2];  
  
 public Rectangle(float[] leftTop, float[] rightBottom){  
 this.leftTop = leftTop;  
 this.rightBottom = rightBottom;  
 }  
}

Листинг дополнительного класса Point.java

package extraClasses;  
  
public class Point {  
 public float xCoor;  
 public float yCoor;  
  
 public Point(float[] points) {  
 this.xCoor = points[0];  
 this.yCoor = points[1];  
 }  
}

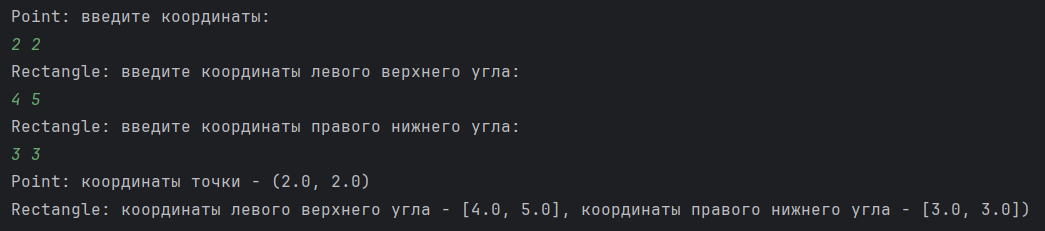


Рисунок 10. Выполнение программы.

ВЫВОД К РАБОТЕ:

Я освоил базовые принципы ООП. Получил практические навыки по работе с классами и объектами на платформе java. Научился проводить декомпозицию приложений с использованием пакетов.