**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена трудового Красного Знамени федеральное государственное бюджетное**

**образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра Математическая кибернетика и информационные технологии

Отчет по лабораторной работе № 2

по дисциплине «Информационные технологии и программирование»

«Основы объектно-ориентированного программирования»

Выполнил: студент группы БФИ2102

Голубева Елена Александровна

Проверила:

Мосева Марина Сергеевна

Москва, 2022

Цель работы

Изучение синтаксиса Java при создании новых классов и их использовании.

Задание

1. Создайте новый класс Point3d для представления точек в трехмерном Евклидовом пространстве. Необходимо реализовать:

• создание нового объекта Point3d с тремя значениями с плавающей точкой (double);

• создание нового объекта Point3d со значениями (0.0, 0.0, 0.0) по умолчанию,

• возможность получения и изменения всех трех значений по отдельности;

• метод для сравнения значений двух объектов Point3d. Нельзя предоставлять непосредственный доступ к внутренним элементам объекта класса Point3d .

1. Добавьте новый метод distanceTo, который в качестве параметра принимает другой объект Point3d, вычисляет расстояние между двумя точками с точность двух знаков после запятой и возвращает полученное значение.
2. Создайте другой класс под названием Lab1, который будет содержать статический метод main. Этот класс должен иметь следующую функциональность:

• Ввод координат трех точек, находящихся в трехмерном пространстве. Создание трех объектов типа Point3d на основании полученных данных. (Предполагается, что пользователь вводит корректные данные.)

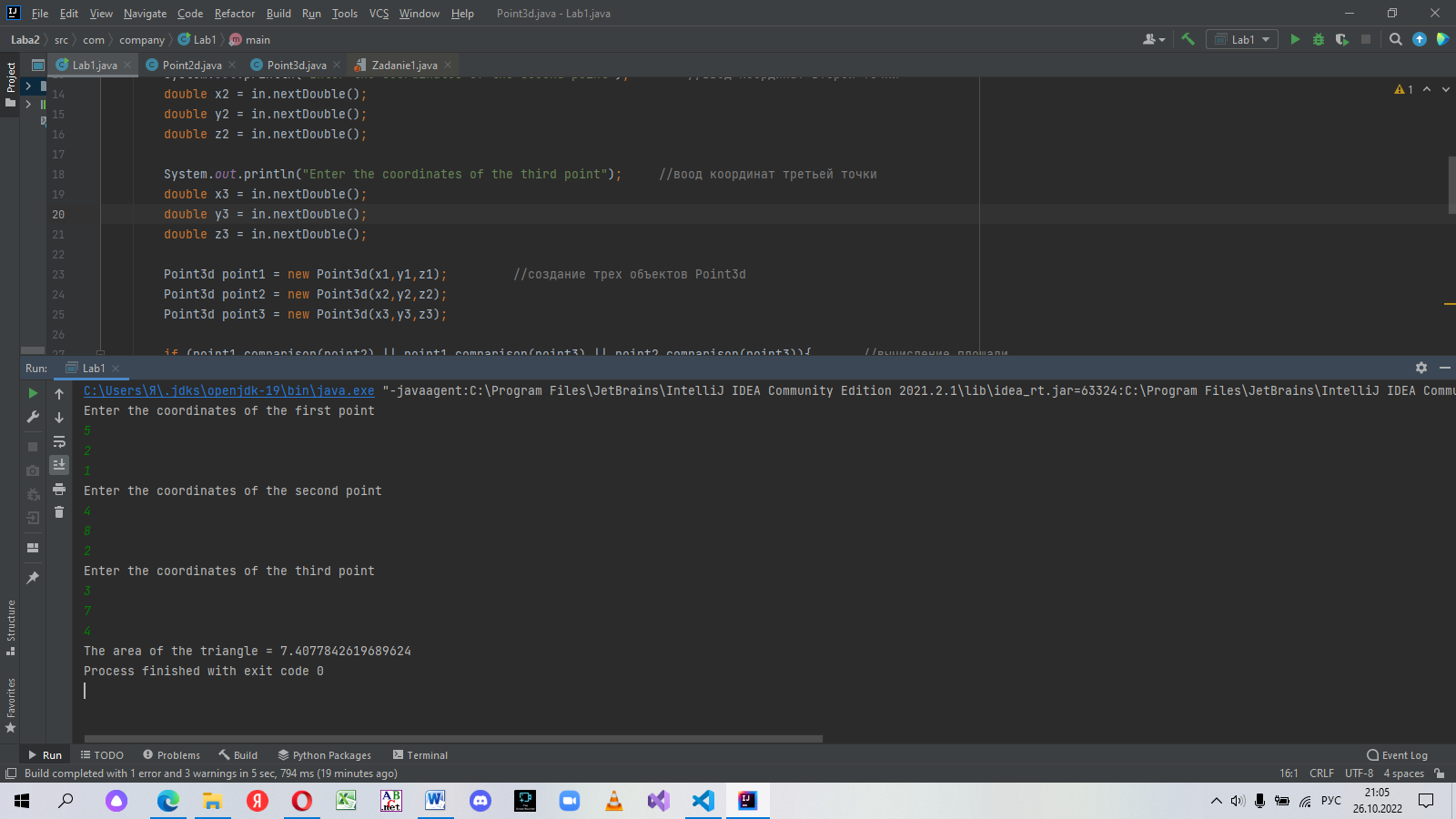
• Создайте второй статический метод computeArea, который принимает три объекта типа Point3d и вычисляет площадь треугольника, образованного этими точками. (Вы можете использовать формулу Герона.) Верните получившееся значение площади в формате типа double.

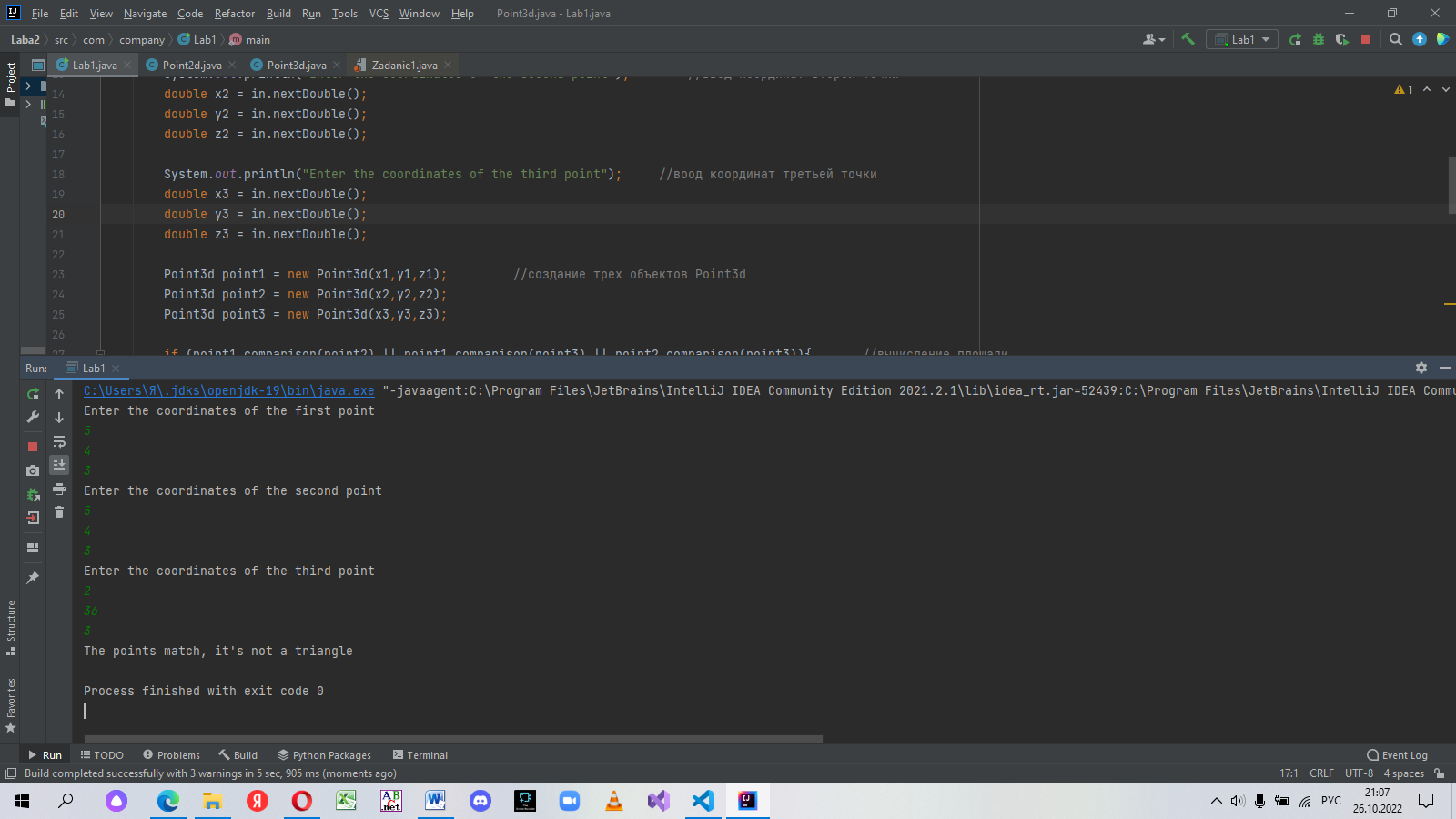
• На основе полученных данных и с использованием реализованного алгоритма посчитайте площадь и выведите полученное значение пользователю. Перед вызовом метода computeArea проверьте на равенство значений всех трех объектов Point3d. Если одна из точек равна другой, то выведите соответствующее сообщение пользователю и не вычисляйте площадь.

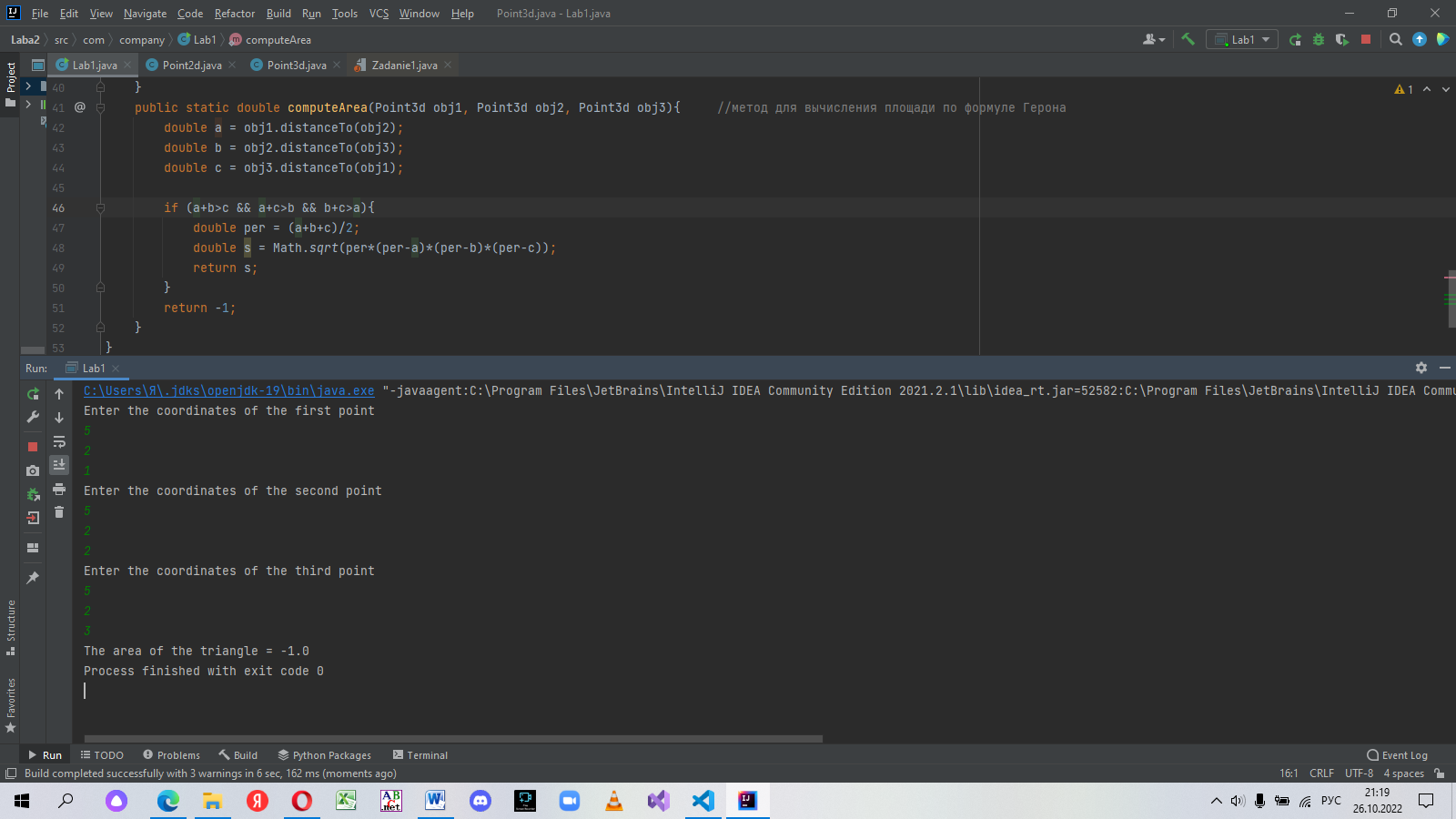
1. Скомпилируйте оба исходных файла вместе: javac Point3d.java Lab1.java и затем запустите программу Lab1, тестируя ее с несколькими образцами треугольников.

Ход работы

1. Результат работы кода файла Lab1 при разных входных данных.







Код файла Point3d

package com.company;  
  
public class Point3d extends Point2d {  
  
 private double zCoord; //координата z  
  
 public Point3d (double x, double y, double z){ //конструктор инициализации  
  
 super(x, y);  
 zCoord=z;  
 }  
 public Point3d(){  
 this(0, 0, 0);  
 } //начальная координата  
  
 public double getZ(){  
 return zCoord;  
 } //возвращение координаты z  
  
 public void setZ(double val){  
 zCoord=val;  
 } //установка значения координаты z  
  
 //@Override  
  
 public boolean comparison(Object x){ //сравнение двух объектов  
 if (x== this){  
 return true;  
 }  
 if (x==null || getClass()!=x.getClass()){  
 return false;  
 }  
 Point3d object1 = (Point3d) x;  
 if (getX()!=object1.getX()|| getY()!=object1.getY()|| getZ()!=object1.getZ()){  
 return false;  
  
 }  
 return true;  
 }  
  
 public double distanceTo(Point3d x){ //расстояние между двумя точками  
 double result = Math.*sqrt*( Math.*pow*(this.getX()-x.getX(), 2) + Math.*pow*(this.getY()-x.getY(), 2)+ Math.*pow*(this.getZ()-x.getZ(), 2) );  
 double forRounding = Math.*pow*(10,2);  
 result = Math.*ceil*(result\*forRounding)/forRounding;  
  
 return result;  
 }  
  
}

Код файла Lab1

package com.company;  
import java.util.Scanner;  
  
  
public class Lab1 {  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("Enter the coordinates of the first point"); //ввод координат первой точки  
 double x1 = in.nextDouble();  
 double y1 = in.nextDouble();  
 double z1 = in.nextDouble();  
  
 System.*out*.println("Enter the coordinates of the second point"); //ввод координат второй точки  
 double x2 = in.nextDouble();  
 double y2 = in.nextDouble();  
 double z2 = in.nextDouble();  
  
 System.*out*.println("Enter the coordinates of the third point"); //воод координат третьей точки  
 double x3 = in.nextDouble();  
 double y3 = in.nextDouble();  
 double z3 = in.nextDouble();  
  
 Point3d point1 = new Point3d(x1,y1,z1); //создание трех объектов Point3d  
 Point3d point2 = new Point3d(x2,y2,z2);  
 Point3d point3 = new Point3d(x3,y3,z3);  
  
 if (point1.comparison(point2) || point1.comparison(point3) || point2.comparison(point3)){ //вычисление площади  
 System.*out*.println("The points match, it's not a triangle");  
 }else {  
 System.*out*.print("The area of the triangle = ");  
 System.*out*.print(*computeArea*(point1,point2,point3));  
 }  
  
  
  
 in.close();  
  
  
 // write your code here  
 }  
 public static double computeArea(Point3d obj1, Point3d obj2, Point3d obj3){ //метод для вычисления площади по формуле Герона  
 double a = obj1.distanceTo(obj2);  
 double b = obj2.distanceTo(obj3);  
 double c = obj3.distanceTo(obj1);  
  
 if (a+b>c && a+c>b && b+c>a){  
 double per = (a+b+c)/2;  
 double s = Math.*sqrt*(per\*(per-a)\*(per-b)\*(per-c));  
 return s;  
 }  
 return -1;  
 }  
}

Вывод

В процессе выполнения данной лабораторной работы мы изучили синтаксис Java при создании новых классов и их использовании.

Ссылка на Github: https://github.com/ElenaGolubeva/Laboratory-2-Java

Список использованной литературы

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа,

2006.

1. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.