**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**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»

на тему:

Основы объектно-ориентированного программирования

Выполнил: студент группы БФИ2101

Иванов Глеб

Проверил преподаватель:

Мосева Марина Сергеевна

Москва, 2022

Оглавление

[**Постановка задачи** 3](#_Toc122857659)

[**Скриншоты выполненного задания** 5](#_Toc122857660)

[**Код** 6](#_Toc122857661)

[**Вывод** 10](#_Toc122857662)

[**Список источников** 11](#_Toc122857663)

# 

**Постановка задачи**

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 .

2. 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.

3. Создайте другой класс под названием Lab1, который будет содержать статический метод main. Помните, что метод main должен быть общедоступным (public) с возвращаемым значением void, а в качестве аргумента должен принимать строку (String). 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.

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

**Скриншоты выполненного задания**

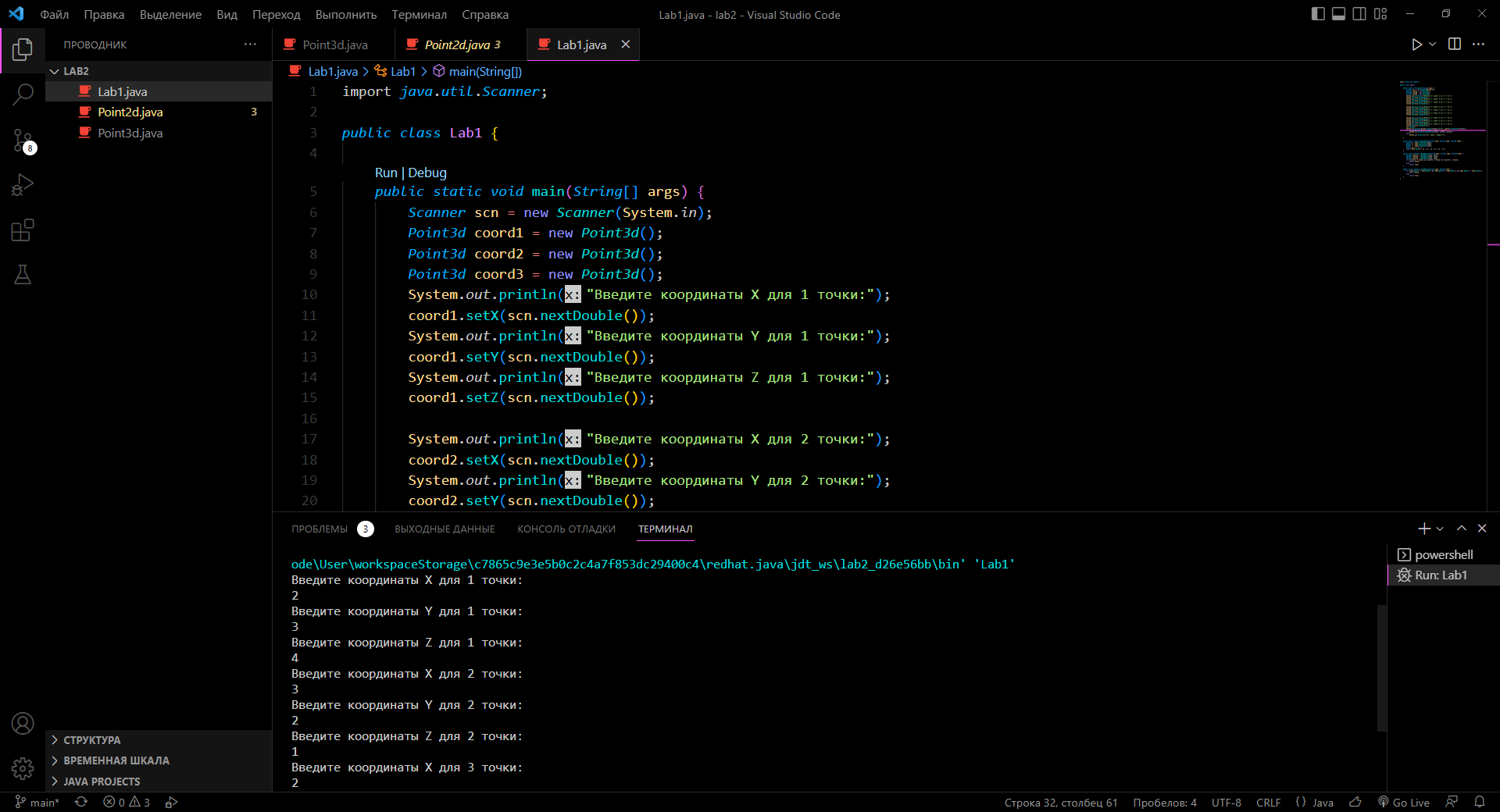
****

Рисунок 1 – Выполнение поставленной задачи

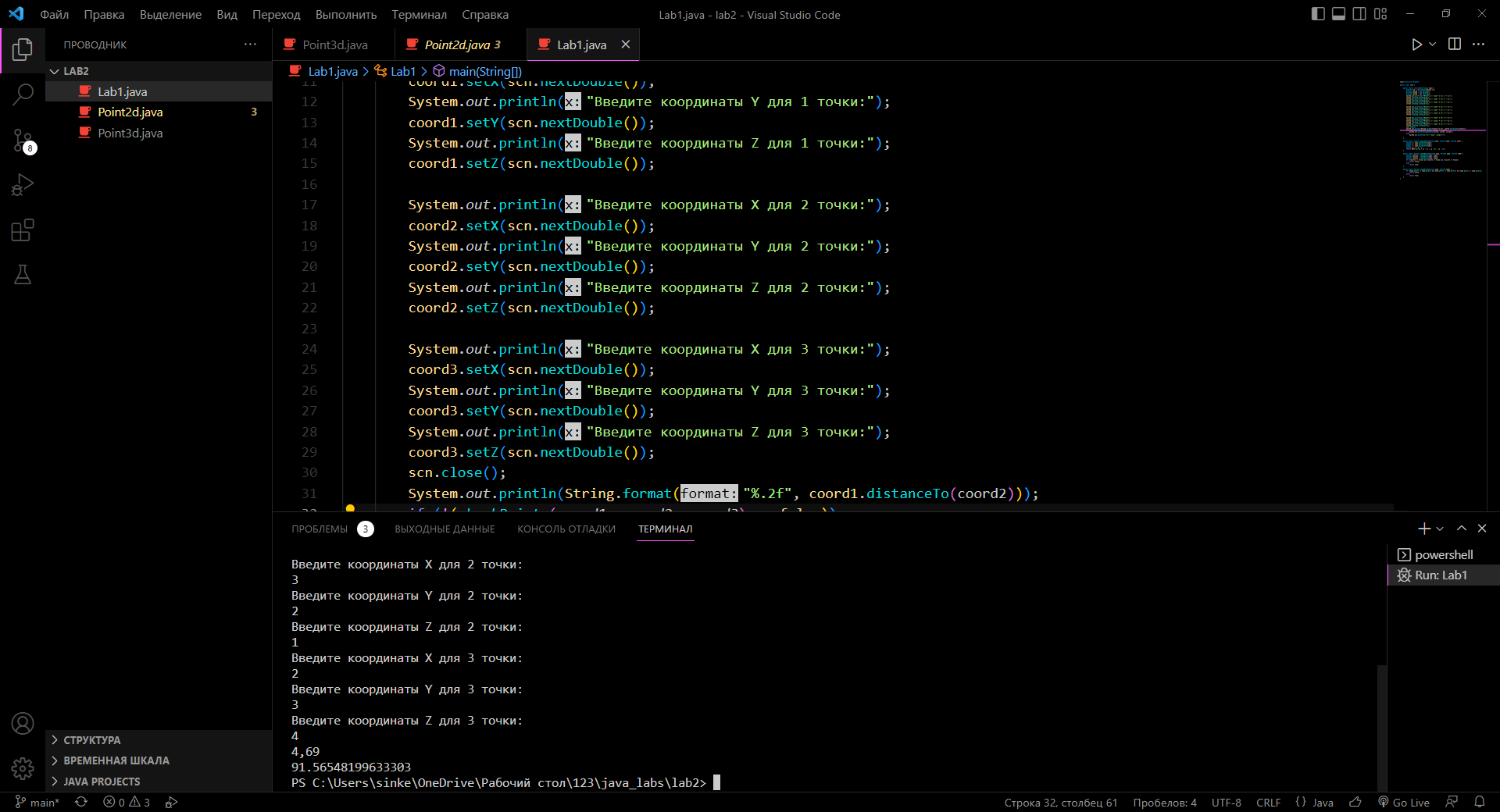


Рисунок 2 – Выполнение поставленной задачи

**Код**

import *java*.*util*.*Scanner*;

*public* *class* Lab1 {

*public* *static* *void* main(*String*[] args) {

*Scanner* scn = new *Scanner*(System.*in*);

*Point3d* coord1 = new *Point3d*();

*Point3d* coord2 = new *Point3d*();

*Point3d* coord3 = new *Point3d*();

        System.*out*.println("Введите координаты Х для 1 точки:");

        coord1.setX(scn.nextDouble());

        System.*out*.println("Введите координаты Y для 1 точки:");

        coord1.setY(scn.nextDouble());

        System.*out*.println("Введите координаты Z для 1 точки:");

        coord1.setZ(scn.nextDouble());

        System.*out*.println("Введите координаты Х для 2 точки:");

        coord2.setX(scn.nextDouble());

        System.*out*.println("Введите координаты Y для 2 точки:");

        coord2.setY(scn.nextDouble());

        System.*out*.println("Введите координаты Z для 2 точки:");

        coord2.setZ(scn.nextDouble());

        System.*out*.println("Введите координаты Х для 3 точки:");

        coord3.setX(scn.nextDouble());

        System.*out*.println("Введите координаты Y для 3 точки:");

        coord3.setY(scn.nextDouble());

        System.*out*.println("Введите координаты Z для 3 точки:");

        coord3.setZ(scn.nextDouble());

        scn.close();

        System.*out*.println(String.format("%.2f", coord1.distanceTo(coord2)));

        if (!(*checkPoints*(*coord1*, *coord2*, *coord3*) == false))

            System.*out*.println(*computeArea*(*coord1*, *coord2*, *coord3*));

        else

            System.*out*.println("Координаты одинаковы");

    }

*public* *static* *double* computeArea(*Point3d* num1, *Point3d* num2, *Point3d* num3) {

*double* a = num1.distanceTo(num2);

*double* b = num1.distanceTo(num3);

*double* c = num2.distanceTo(num3);

*double* p = a + b + c;

        return Math.sqrt(p \* (p - a) \* (p - b) \* (p - c));

    }

*public* *static* *boolean* checkPoints(*Point3d* num1, *Point3d* num2, *Point3d* num3) {

*boolean* result1 = *checkPoint*(*num1*, *num2*);

*boolean* result2 = *checkPoint*(*num1*, *num3*);

*boolean* result3 = *checkPoint*(*num2*, *num3*);

        if ((result1 == false) && (result2 == false) && (result3 == false))

            return false;

        else

            return true;

    }

*public* *static* *boolean* checkPoint(*Point3d* num1, *Point3d* num2) {

        if ((num1.getX() == num2.getX()) && (num1.getY() == num2.getY()) && (num1.getZ() == num2.getZ()))

            return false;

        else

            return true;

    }

}

Код файла Lab1.java

*public* *class* Point2d {

*public* *static* *void* main(*String*[] args) {

*Point2d* myPoint = new *Point2d*();

*Point2d* myOtherPoint = new *Point2d*(5,3);

*Point2d* aThirdPoint = new *Point2d*();

    }

*private* *double* xCoord;

*private* *double* yCoord;

*public* Point2d(*double* x, *double* y) {

        xCoord = x;

        yCoord = y;

    }

*public* Point2d() {

        this(0, 0);

    }

*public* *double* getX() {

        return xCoord;

    }

*public* *double* getY() {

        return yCoord;

    }

*public* *void* setX(*double* val) {

        xCoord = val;

    }

*public* *void* setY(*double* val) {

        yCoord = val;

    }

}

Код файла Point2d.java

*public* *class* Point3d *extends* Point2d {

*private* *double* xCoord;

*private* *double* yCoord;

*private* *double* zCoord;

*public* Point3d(*double* x, *double* y, *double* z) {

        super(x, y);

        zCoord = z;

    }

*public* Point3d() {

        this(0, 0, 0);

    }

*public* *double* getZ() {

        return zCoord;

    }

*public* *void* setZ(*double* val) {

        zCoord = val;

    }

*public* *double* distanceTo(*Point3d* num) {

        return (Math.sqrt(Math.pow((num.getX() - xCoord), 2) + Math.pow((num.getY() - yCoord), 2)

                + Math.pow((num.getZ() - zCoord), 2)));

    }

*public* *static* *String* checkPoints(*Point3d* num1, *Point3d* num2) {

*String* str = "";

        if (num1.getX() > num2.getX()) {

            str += "Coord x in first obj  is larger,";

        } else {

            str += "Coord x in first obj  is smaller,";

        }

        if (num1.getY() > num2.getY()) {

            str += " Coord y in first obj  is larger,";

        } else {

            str += " Coord y in first obj  is smaller,";

        }

        if (num1.getZ() > num2.getZ()) {

            str += " Coord z in first obj  is larger.";

        } else {

            str += " Coord z in first obj  is smaller.";

        }

        return str;

    }

}

Код файла Point3d.java

**Вывод**

В ходе выполненной работы я ознакомился с основами объектно-ориентированного программирования. Смог реализовать наследование класса с помощью ключевого слово expends, передать переменные в класс родителя с помощью метода super().

**Список источников**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006.

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