Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра «Математическая кибернетика и информационные технологии»

Лабораторная работа №1

«Основы объектно-ориентированного программирования»

по дисциплине «Кроссплатформенные технологии»

Выполнила: студентка

группы БСТ2004

Ус О.Н.

Проверила:

Полянцева К.А.

Москва 2022

**Содержание**

[**1. Цель работы:** 3](#_Toc96546823)

[**2. Задание:** 3](#_Toc96546824)

[**3. Ход выполнения лабораторной работы** 4](#_Toc96546825)

[3.1. Создание нового класса Point3d для представления точек в трехмерном Евклидовом пространстве 4](#_Toc96546826)

[3.2. Создание класса Lab2. 9](#_Toc96546827)

[**4. Ссылка на репозиторий гитхаба** 12](#_Toc96546828)

[**5. Вывод** 12](#_Toc96546829)

[**Список использованной литературы** 13](#_Toc96546830)

**1. Цель работы:** изучить объектно-ориентированного программирования с помощью нескольких задач программирования.

**2. Задание:**

**2.1** Создать новый класс Point3d для представления точек в трехмерном Евклидовом пространстве.

Задачи:

1. создать новый объект Point3d с тремя значениями с плавающей точкой (double)
2. создать новый объект Point3d со значениями (0.0, 0.0, 0.0) по умолчанию
3. организовать возможность получения и изменения всех трех значений переменных координат по отдельности
4. создать метод comparison для сравнения значений двух объектов Point3d
5. добавить метод distanceTo, который вычисляет расстояние между двумя точками

**2.2** Создать класс под названием Lab2.

Задачи:

1. организовать ввод координат трех точек, находящихся в трехмерном пространстве. Создать три объекта типа Point3d на основании полученных данных
2. создать статический метод computeArea, который вычисляет площадь треугольника, образованного введенными точками.
3. посчитать площадь и вывести полученное значение пользователю

**3. Ход выполнения лабораторной работы**

## 3.1. Создание нового класса Point3d для представления точек в трехмерном Евклидовом пространстве

Создание нового объекта Point3d с тремя значениями с плавающей точкой (double), объекта Point3d со значениями (0.0, 0.0, 0.0) по умолчанию

Создаю файл Point3d.java в папке для работы над лабораторной работой и добавляю в нее метод main, в котором будет находиться основной код программы. В нем объявляю новые объекты класса, используя конструкторы инициализации. Это представлено на рисунках 1 и 2.

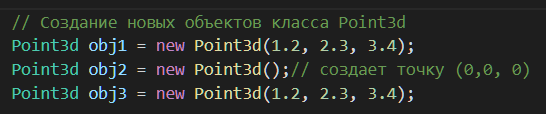


Рисунок 1 – Создание файла и класса Point3d, объявление трех точек

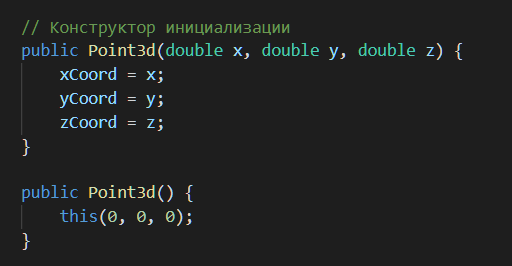


Рисунок 2 – Конструкторы класса Point3d

Организация возможности получения и изменения всех трех значений переменных координат по отдельности.

Получаю координаты первой точки для вывода их на печать с помощью методов get\*, меняю координаты методом set\* и снова вывожу координаты точки на печать, чтобы проверять корректность работы программы. Код методов get\*, set\* представлен на рисунке 3. Основной код программы представлен на рисунке 4. Результат работы программы представлен на рисунке 5.

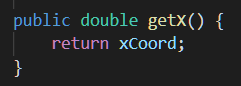
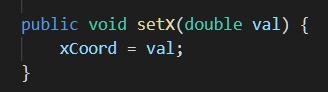
 

Рисунок 3 – Код методов get\*, set\*

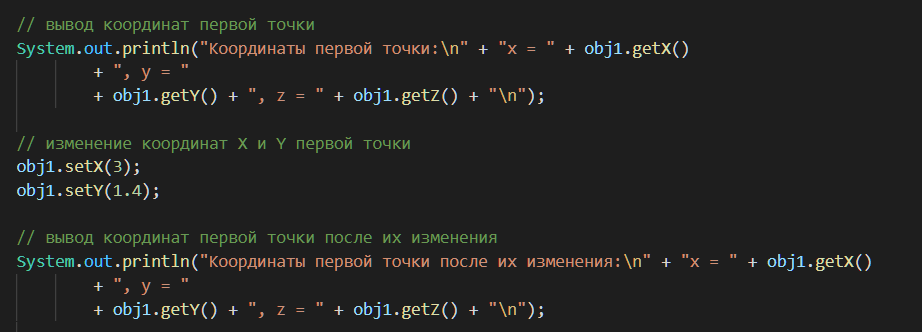


Рисунок 4 – Код программы

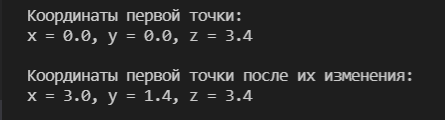


Рисунок 5 – Результат работы программы

Создать метода comparison для сравнения значений двух объектов Point3d

Создаю метод comparison для сравнения значений двух объектов Point3d. На вход подается два объекта. Они сравниваются с помощью метода equals, который нужно было переопределить для корректной работы. Код метода equals представлен на рисунке 6. Код метода comparison представлен на рисунке 7.

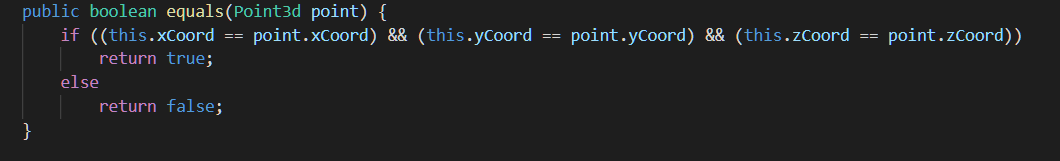


Рисунок 6 – Код метода equals

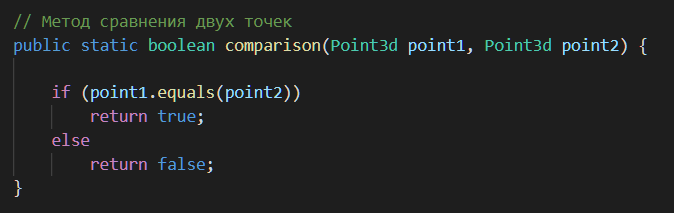


Рисунок 7 – Код метода comparison

Реализация обоих методов в основном коде программы представлена на рисунке 8. На рисунке 9 изображен результат работы программы.

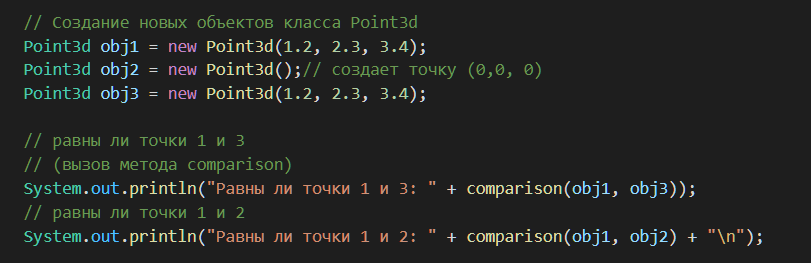


Рисунок 8 – Реализация методов equals, comparison в основном коде программы



Рисунок 9 – Результат работы программы

Создание метода distanceTo, который вычисляет расстояние между двумя точками

Создаю метод distanceTo, который вычисляет расстояние между двумя точками. Код метода представлен на рисунке 10. Реализация метода в основном коде программы представлена на рисунке 11. Результат работы показан на рисунке 12.

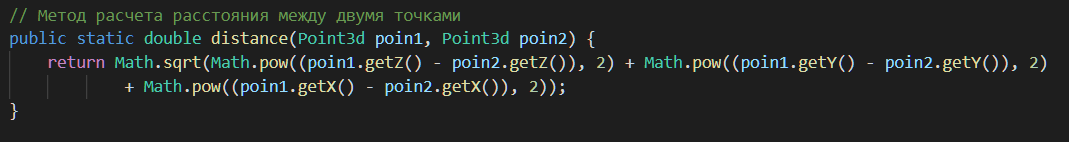


Рисунок 10 – Код метода comparison

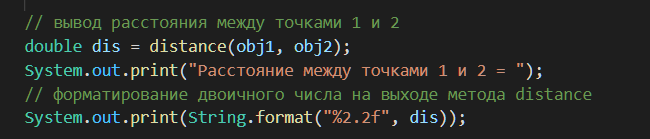


Рисунок 11 – Реализация метода comparison в основном коде программы



Рисунок 12 – Результат работы программы

## 3.2. Создание класса Lab2.

Организация ввода координат трех точек, находящихся в трехмерном пространстве. Создание трех объектов типа Point3d на основании полученных данных

Создаю файл Lab2.java в папке для работы над лабораторной работой и добавляю в нее метод main, в котором будет находиться основной код программы. Объявляю объект класса Scanner для получения возможности ввода чисел с клавиатуры. Объявляю три пустых объекта класса Point3d и ввожу координаты каждой, используя метод pointInput. Код метода pointInput представлен на рисунке 13. Реализация задачи представлена на рисунке 14.

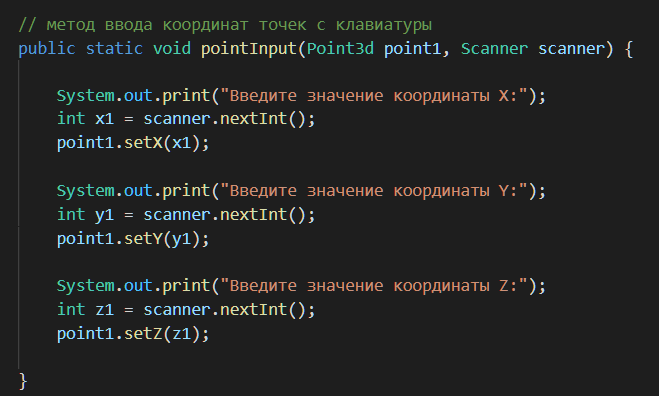


Рисунок 13 – Код метода pointInput

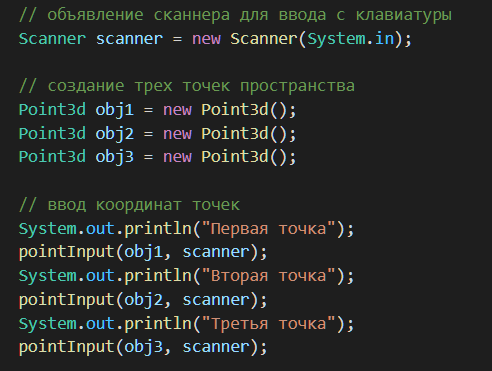


Рисунок 14 – Реализация метода pointInput в основном коде программы

Создание статического метода computeArea, вычисляющего площадь треугольника, образованного введенными точками

Создаю метод computeArea, вычисляющего площадь треугольника, образованного введенными точками, по формуле Герона. Код метода представлен на рисунке 15.

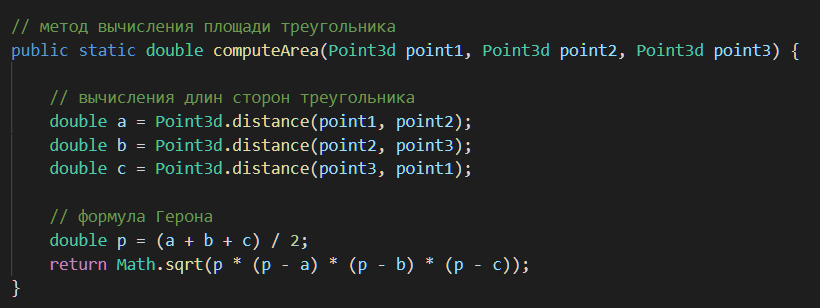


Рисунок 15 – Код метода computeArea

При реализации метода ввожу условие – если вычисленная площадь равна нулю, значит, такой треугольник не существует. Это представлено на рисунке 16

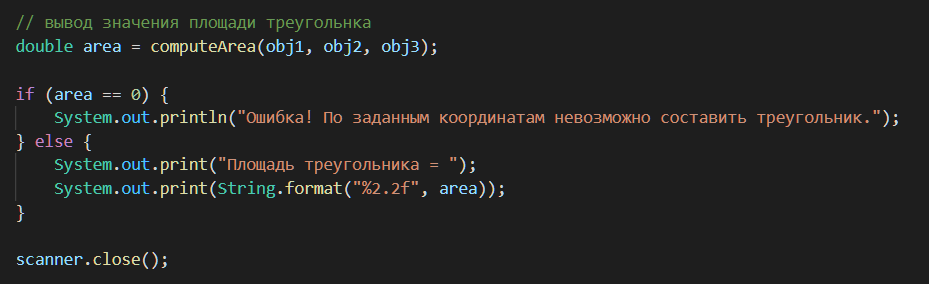


Рисунок 16 – Реализация метода computeArea в основном коде программы

Вычисление площади и вывод полученного значения пользователю

Результат работы метода main на примере разных точек представлен на рисунках 17-19.

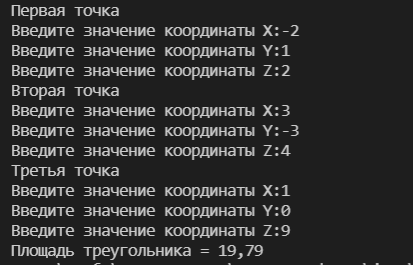


Рисунок 17 – Результат работы программы при корректно введенных координатах точек

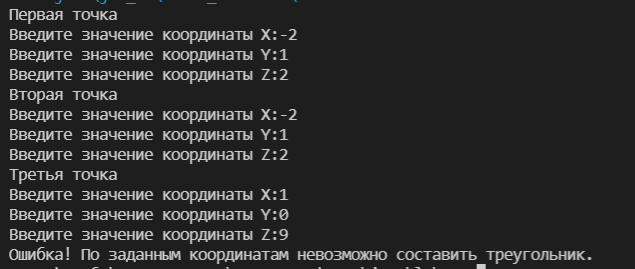


Рисунок 18 – Результат работы программы при двух совпадающих точках

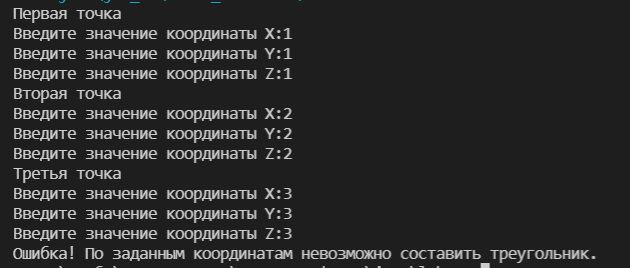


Рисунок 19 – Результат работы программы при введении точек, из которых невозможно составить треугольник

**4. Ссылка на репозиторий гитхаба**

[https://github.com/TerraficMint/ssss](https://github.com/TerraficMint/ssss%20)

**5. Вывод**

Я изучила объектно-ориентированного программирования с помощью нескольких задач программирования.

**Список использованной литературы**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006.
2. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.
3. ГОСТ 7.32-2017 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления. - URL: <https://files.stroyinf.ru/Index/655/65555.htm>

Git:

1. Scott Chacon, Ben Straub «Pro Git»
2. [git-scm.com](https://git-scm.com/)