**Федеральное агентство связи**

**Ордена трудового Красного Знамени федеральное государственное бюджетное**

**образовательное учреждение высшего образования**

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

Кафедра Математическая кибернетика и информационные технологии

Лабораторная работа № 2

Выполнил: студент группы БФИ2002

Ковачев В.Е.

Проверила:

Москва, 2021

**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.

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.

**Ход работы:**

**Задание №1,2**

**Для выполнения первого и второго задания был написан класс Point3d и метод distanceTo в этом классе.**

import static java.lang.Math.\*;

public class Point3d {

private double x;

private double y;

private double z;

public Point3d() {

x = y = z = 0.0;

}

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

this.x = x;

this.y = y;

this.z = z;

}

**// Проверка координат точек на идентичность**

public boolean isEqual(Point3d p) {

if (this.x == p.getX()) {

if (this.y == p.getY()) {

if (this.z == p.getZ()) {

return true;

}

}

}

return false;

}

**// Возвращает расстояние между двумя точками**

public double distanceTo(Point3d p) {

return sqrt(pow(p.getX() - this.x, 2) + pow(p.getY() - this.y, 2) + pow(p.getZ() - this.z, 2));

}

public void setX(double x) {

this.x = x;

}

public void setY(double y) {

this.y = y;

}

public void setZ(double z) {

this.z = z;

}

public double getX() {

return x;

}

public double getY() {

return y;

}

public double getZ() {

return z;

}

}

**Задание №3**

**Для выполнения третьего задания был написан класс Lab1.**

public class Lab1 {

public static void main(String[] args) {

Point3d p1 = new Point3d(Double.parseDouble(args[0]), Double.parseDouble(args[1]), Double.parseDouble(args[2]));

Point3d p2 = new Point3d(Double.parseDouble(args[3]), Double.parseDouble(args[4]), Double.parseDouble(args[5]));

Point3d p3 = new Point3d();

System.out.println("Площадь треугольника: " + computeArea(p1, p2, p3));

}

public static double computeArea(Point3d p1, Point3d p2, Point3d p3) {

**// Если одна из точек равна другой, то площадь не вычисляется**

if (p1.isEqual(p2) || p2.isEqual(p3) || p3.isEqual(p1)) {

return -1;

} else {

double a = p1.distanceTo(p2);

double b = p2.distanceTo(p3);

double c = p3.distanceTo(p1);

double p = (a+b+c)/2;

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

}

}

}

**Задание №4**

**1.Скомпилированы оба исходных файла вместе: javac Point3d.java Lab1.java .**

**2.Проведены тесты с несколькими образцами треугольников.**

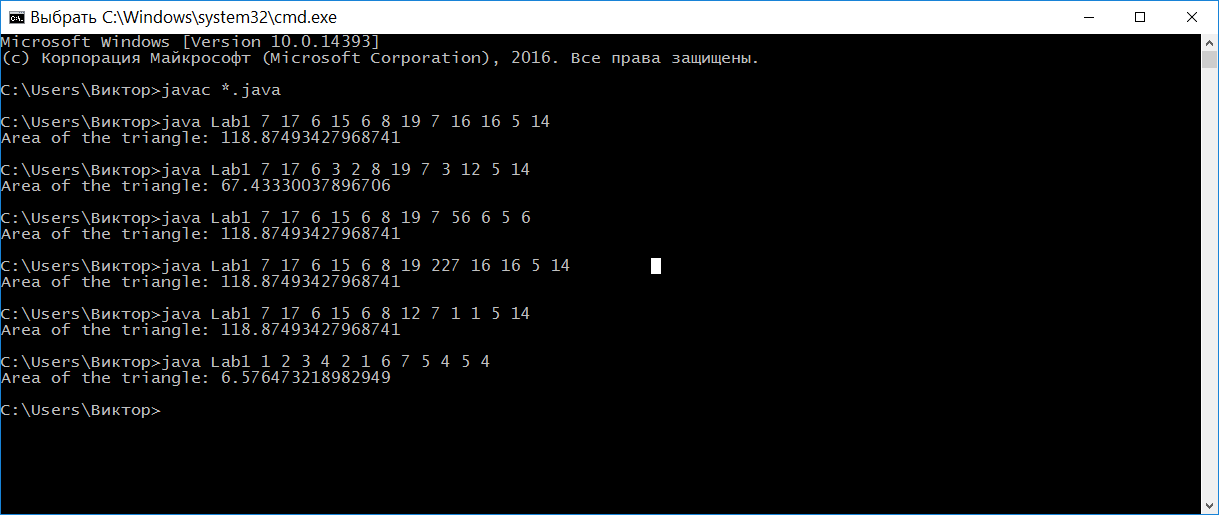


Рисунок №1- Результат работы при вводе треугольника №1

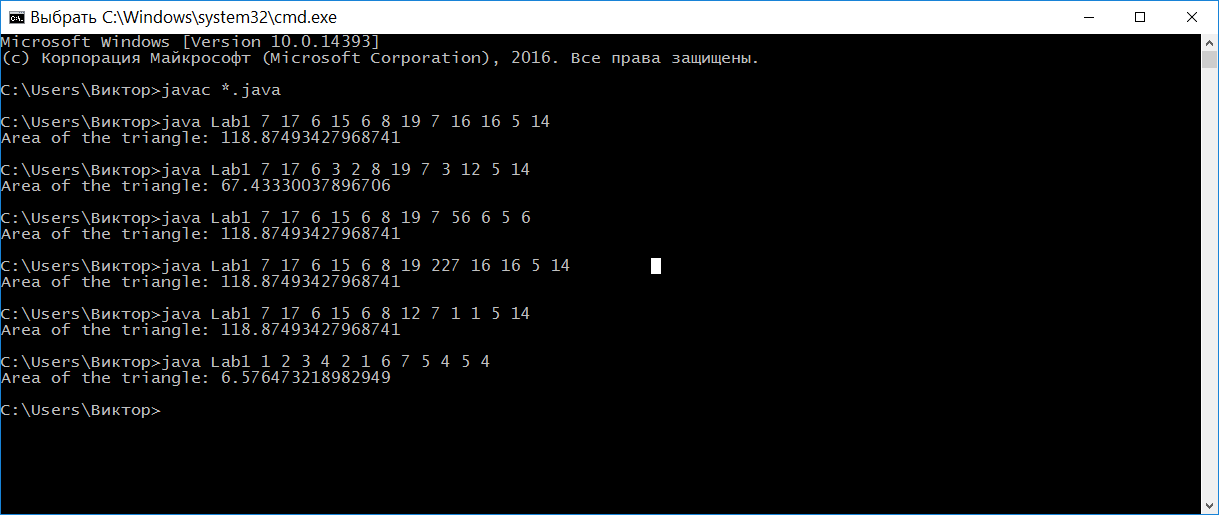


Рисунок №2- результат работы при вводе треугольника №2

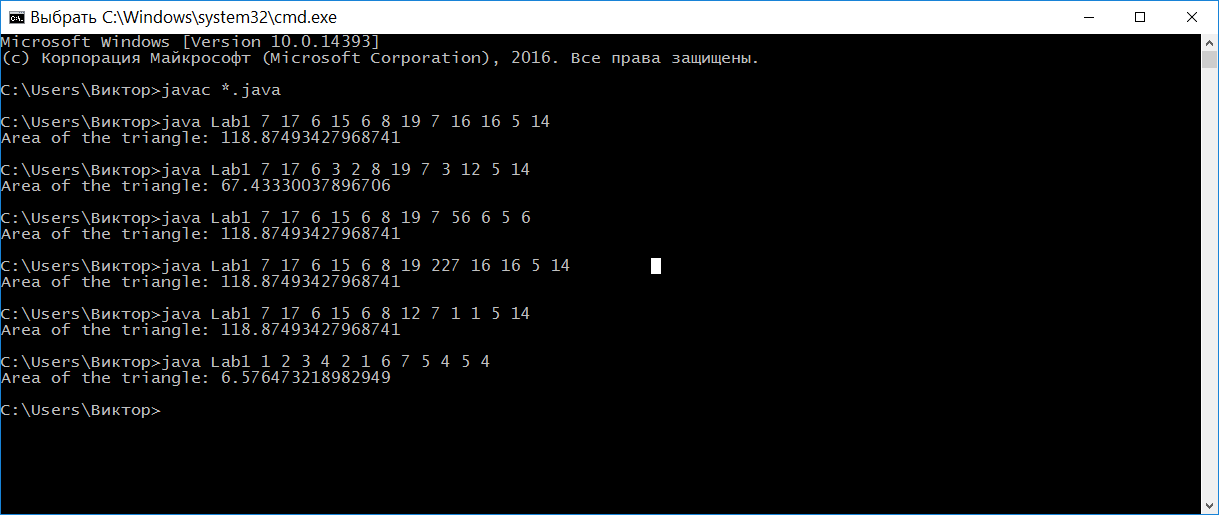


Рисунок №3- результат работы при вводе треугольника №3

**Вывод**

Мы научились использовать объекты и классы. Мы использовали классы по одному на

файл, чтобы описать, как эти объекты работают.

**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.