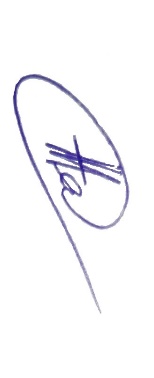
**МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ**

**Лабораторная работа №3**

«Классы»

Вариант 16

 Выполнил: Некрасов А. А.   
БСТ1601

Проверил: Городничев М.Г

Москва 2018

Оглавление

[Цель работы: 2](#_Toc511567644)

[Используемые инструменты: 2](#_Toc511567645)

[Тест программы и пояснения. 2](#_Toc511567646)

[Вывод: 4](#_Toc511567647)

Цель работы: Научиться работать с классами.

Используемые инструменты: для написания кода программы будем использовать редактор кода Visual Studio Code. Для создания, компилирования и дебага программ будем использовать jdk v9.0.4.

### Тест программы и пояснения.

Файл lab3.java

import java.io.\*;

import java.util.\*;

public class lab3 {

public static void main (String[] args) {

List<Point3d> list = new ArrayList<Point3d>();//массив для хранения точек

//проверка исключений на ошибку ввода вывода и запись в массив

try{

//инициализация буфера и входного потока

InputStreamReader inputReader = new InputStreamReader(System.in,"utf-8");

BufferedReader buffReader = new BufferedReader(inputReader);

//запись в массив

for (int i=0;i<3;i++){

System.out.print("Write the new point:\n x- ");

double x = Double.parseDouble(buffReader.readLine());

System.out.print(" y- ");

double y = Double.parseDouble(buffReader.readLine());

System.out.print(" z- ");

double z = Double.parseDouble(buffReader.readLine());

list.add(new Point3d(x,y,z));

}

} catch (IOException e) {

e.printStackTrace();

}

//вызов функции подсчета площади

try{

System.out.println("Triangle area: " + computeArea(list.get(0), list.get(1), list.get(2)));

}catch (IllegalArgumentException e) {//проверка на корректность передаваемого значения функции

e.printStackTrace();

}

}

// функция подсчета площади треугольника

public static double computeArea (Point3d point1, Point3d point2, Point3d point3) {

//проверка что 2 любые точки не эквивалентны иначе выдать сообщение

if (point1.equals(point2) || point2.equals(point3) || point3.equals(point1)) {

throw new IllegalArgumentException("Points shoud not be equal");

}

//подсчет площади

double a = point1.distanceTo(point2),b = point2.distanceTo(point3),c = point3.distanceTo(point1),p = (a + b + c)/2,S = Math.sqrt(p \* (p-a) \* (p-b) \* (p-c));;

//возврат площади

return S;

}

}

Файл point3d.java

//класс 3х мерной точки

public class Point3d {

private double \_x,\_y,\_z;//координаты точки

public Point3d(){//конструктор по умлчанию

this(0.,0.,0.);

}

public Point3d(double x,double y,double z) {// конструктор инициализации

\_x=x;

\_y=y;

\_z=z;

}

//функции для возврата значений

public double getX(){

return \_x;

}

public double getY(){

return \_y;

}

public double getZ(){

return \_z;

}

//функции для записи значений

public void setX(double val){

\_x=val;

}

public void setY(double val){

\_y=val;

}

public void setZ(double val){

\_z=val;

}

//функция проверки 2х точек на эквиваленность

public boolean equiv(Point3d point){

return (point.getX()==this.getX()&&point.getY()==this.getY()&&point.getZ()==this.getZ());

}

//рсчет растояния между двумя точками

public double distanceTo(Point3d point) {

return Math.sqrt(Math.pow(point.getX() - this.getX(), 2) + Math.pow(point.getY() - this.getY(), 2));

}

}

Вывод: В рамках этой работы я научился работать с классами.