ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования

«МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ»

ЛАБОРАТОРНАЯ РАБОТА №2

«Сразу к делу»

Выполнил**:** Студент группы БСТ-1601

Пушкарев Павел Сергеевич



Преподаватель: Городничев М. Г.

Москва 2018

**Цель:** Разработать класс Point3d, который позволяет описать точку в трехмерном пространстве, а также разработать в нем метод distanceTo, который вычисляет расстояние между двумя точками.  
Разработать класс Lab1, в котором реализовать ввод трех точек и вычисление площади, ограниченной этими точками.

**Выбор инструментария:** Visual Studio Code для написания кода, jdk для компиляции кода, java для выполнения программы.

**Ход Выполнения**

1. Создадим класс Point3d в файле Point3d.java и создадим в нем конструкторы инициализации и по умолчанию:

/\*

|  |  |
| --- | --- |
|  | Класс, описывающий трехмерную точку |
|  | \*/ |
|  | public class Point3d { |
|  | private double x, y, z; |
|  |  |
|  | public Point3d(double \_x, double \_y, double \_z) { |
|  | x = \_x; |
|  | y = \_y; |
|  | z = \_z; |
|  | } |
|  |  |
|  | public Point3d() { |
|  | this(0., 0., 0.); |
|  | } |
|  |  |
|  |  |
|  | } |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

2. Теперь создадим методы setX, setY, setZ, которые позволят изменять значения каждой координаты, а также методы getX, getY, getZ, которые возвращают значения координат:

|  |
| --- |
| public void setX(double \_x) { |
| x = \_x; |
| } |
|  |
| public void setY(double \_y) { |
| y = \_y; |
| } |
|  |
| public void setZ(double \_z) { |
| z = \_z; |
| } |
|  |
| public double getX() { |
| return x; |
| } |
|  |
| public double getY() { |
| return y; |
| } |
|  |
| public double getZ() { |
| return z; |
| } |

3. Также необходимо создать метод equals, который сравнивает координаты двух точек и, если они равны, возвращает true, иначе false:

|  |  |
| --- | --- |
|  | //Проверяет равенство 2-х точек |
|  | public boolean equals(Point3d point) { |
|  | return this.getX() == point.getX() && |
|  | this.getY() == point.getY() && |
|  | this.getZ() == point.getZ(); |
|  | } |

4. Осталось только создать метод distanceTo(Point3d point), который вычисляет расстояние между точками при помощи формулы :

|  |  |
| --- | --- |
|  | //Вычисляет расстояние между точками  public double distanceTo(Point3d point) { |
|  | return Math.sqrt(Math.pow(point.getX() - this.getX(), 2) + |
|  | Math.pow(point.getY() - this.getY(), 2)); |
|  | }   |  |  | | --- | --- | | //Вычисляет площадь области, огранниченной точками  public static double computeArea (Point3d point1, Point3d point2, Point3d point3) { | | | if (point1.equals(point2) || point2.equals(point3) || point3.equals(point1)) { | | throw new IllegalArgumentException("Two points cannot be equals!"); | | } | |  |  | | | double a = point1.distanceTo(point2), | | b = point2.distanceTo(point3), | | c = point3.distanceTo(point1); | |  | | double p = (a + b + c)/2; | | double S = Math.sqrt(p \* (p-a) \* (p-b) \* (p-c)); | |  | | return S; | | } |   5. Теперь создадим класс Lab1 в файле Lab1.java, где реализуем метод computeArea(), который вычисляет площадь треугольника, ограниченного точками, используя формулу Герона:  6. В методе main сделаем ввод трех точек Point3d и выведем площадь треугольника, ограниченного ими:   |  | | --- | | public static void main (String[] args) { | | List<Point3d> points = new ArrayList<Point3d>(); | |  | | try { | | InputStreamReader inputReader = new InputStreamReader(System.in,"utf-8"); | | BufferedReader buffReader = new BufferedReader(inputReader); | |  | | for (int i = 0; i < 3; i++) { | | double x, y, z; | |  | | System.out.print("Write new point:\nx: "); | | x = Double.parseDouble(buffReader.readLine()); | | System.out.print("y: "); | | y = Double.parseDouble(buffReader.readLine()); | | System.out.print("z: "); | | z = Double.parseDouble(buffReader.readLine()); | |  | | points.add(new Point3d(x, y, z)); | | } | |  | | } catch (IOException e) { | | e.printStackTrace(); | | } | |  | | try { | | System.out.println("Triangle area: " + computeArea(points.get(0), points.get(1), points.get(2))); | | } catch (IllegalArgumentException e) { | | e.printStackTrace(); | | } | | }    Рисунок 1 – Результат работы программы | |

**Исходный код программы:**

Point3d.java:

|  |
| --- |
| /\* |
| Класс, описывающий трехмерную точку |
| \*/ |
| public class Point3d { |
| private double x, y, z; |
|  |
| public Point3d(double \_x, double \_y, double \_z) { |
| x = \_x; |
| y = \_y; |
| z = \_z; |
| } |
|  |
| public Point3d() { |
| this(0., 0., 0.); |
| } |
|  |
| public void setX(double \_x) { |
| x = \_x; |
| } |
|  |
| public void setY(double \_y) { |
| y = \_y; |
| } |
|  |
| public void setZ(double \_z) { |
| z = \_z; |
| } |
|  |
| public double getX() { |
| return x; |
| } |
|  |
| public double getY() { |
| return y; |
| } |
|  |
| public double getZ() { |
| return z; |
| } |
|  |
| //Проверяет равенство 2-х точек |
| public boolean equals(Point3d point) { |
| return this.getX() == point.getX() && |
| this.getY() == point.getY() && |
| this.getZ() == point.getZ(); |
| } |
|  |
| //Вычисляет расстояние между точками |
| public double distanceTo(Point3d point) { |
| return Math.sqrt(Math.pow(point.getX() - this.getX(), 2) + |
| Math.pow(point.getY() - this.getY(), 2)); |
| } |
| } |

Lab1.java:

|  |
| --- |
| import java.io.\*; |
| import java.util.\*; |
|  |
| /\* |
| Запрашивает ввод 3-х трехмерных точек |
| и выводит площадь треугольника, огранниченного ими |
| \*/ |
| public class Lab1 { |
| public static void main (String[] args) { |
| List<Point3d> points = new ArrayList<Point3d>(); |
|  |
| try { |
| InputStreamReader inputReader = new InputStreamReader(System.in,"utf-8"); |
| BufferedReader buffReader = new BufferedReader(inputReader); |
|  |
| for (int i = 0; i < 3; i++) { |
| double x, y, z; |
|  |
| System.out.print("Write new point:\nx: "); |
| x = Double.parseDouble(buffReader.readLine()); |
| System.out.print("y: "); |
| y = Double.parseDouble(buffReader.readLine()); |
| System.out.print("z: "); |
| z = Double.parseDouble(buffReader.readLine()); |
|  |
| points.add(new Point3d(x, y, z)); |
| } |
|  |
| } catch (IOException e) { |
| e.printStackTrace(); |
| } |
|  |
| try { |
| System.out.println("Triangle area: " + computeArea(points.get(0), points.get(1), points.get(2))); |
| } catch (IllegalArgumentException e) { |
| e.printStackTrace(); |
| } |
| } |
|  |
| //Вычисляет площадь области, огранниченной точками |
| public static double computeArea (Point3d point1, Point3d point2, Point3d point3) { |
| if (point1.equals(point2) || point2.equals(point3) || point3.equals(point1)) { |
| throw new IllegalArgumentException("Two points cannot be equals!"); |
| } |
|  |
| double a = point1.distanceTo(point2), |
| b = point2.distanceTo(point3), |
| c = point3.distanceTo(point1); |
|  |
| double p = (a + b + c)/2; |
| double S = Math.sqrt(p \* (p-a) \* (p-b) \* (p-c)); |
|  |
| return S; |
| } |
| } |

**Вывод:** в результате выполнения данной работы, я научился разрабатывать классы, а также создавать объекты данных классов.