

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

ПРЕЗИДЕНТСКИЙ ФИЗИКО-МАТЕМАТИЧЕСКИЙ ЛИЦЕЙ №  
239

ПОИСК МАКСИМАЛЬНОГО РАССТОЯНИЯ МЕЖДУ ТОЧКАМИ  
ВО МНОЖЕСТВЕ

ОТЧЁТ ПО ГОДОВОМУ ПРОЕКТУ

Ученик:	Никитин Михаил
Преподаватель:	Клюнин Алексей Олегович
Класс:	10-3

Санкт-Петербург  
2017

# Содержание

<b>1</b>	<b>Постановка задачи</b>	<b>3</b>
<b>2</b>	<b>Алгоритм решения задачи</b>	<b>3</b>
2.1	Базовые структуры данных . . . . .	3
2.2	Исходные данные . . . . .	3
2.3	Выходные данные . . . . .	3
<b>3</b>	<b>Листинг программы</b>	<b>3</b>
<b>4</b>	<b>Пример работы программы</b>	<b>4</b>
<b>5</b>	<b>Анализ правильности решения</b>	<b>4</b>

# 1 Постановка задачи

Во множестве точек на плоскости найти пару точек с максимальным расстоянием между ними.

## 2 Алгоритм решения задачи

1. Создаем класс точки и прямой. В классе точки создаем метод Distance, который используя теорему Пифагора считает расстояние между точками.

2. В классе main с помощью двух циклов for перебираем все пары точек, максимальное расстояние записывается.

3. Соединяем точки между которыми максимальное расстояние прямой для наглядности.

### 2.1 Базовые структуры данных

### 2.2 Исходные данные

Множество точек на плоскости.

То есть:

1.Количество точек

2.Их координаты

### 2.3 Выходные данные

Максимальное расстояние между точками.

## 3 Листинг программы

```
for (int i = 0; i < points.size(); i++)
for (int j = i + 1; j < points.size(); j++) //просчитываем всевозможные вариации с точками
double d = Point.Distance(points.get(i), points.get(j));
if (max < d)
max = d; //если нашли больше, то новый max
Max1 = i;
Max2 = j;
//System.out.println(max); //вывод max
com.bot.Line l = new com.bot.Line(points.get(Max1), points.get(Max2)); //создаем прямую,
которая будет соединять точки
lines.add(l);
l.setBounds(Max1, Max2, frame.getWidth(), frame.getHeight());
pointpane.add(l);
pointpane.revalidate();
pointpane.repaint();
answerL.setText("Ответ: " + max); //выводим ответ на панель
```

## 4 Пример работы программы

Исходные данные:

Первый пример: 3 1 1 4 1 1 5

Второй пример: 3 3 2 3 3 0 2

Выходные данные:

Первый пример: 5

Второй пример: 3

## 5 Анализ правильности решения

Теорема Пифагора позволяет легко посчитать расстояние. А сравнение расстояний позволяет легко найти наибольшее.