

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

ПРЕЗИДЕНТСКИЙ ФИЗИКО-МАТЕМАТИЧЕСКИЙ ЛИЦЕЙ №
239

ОТЧЁТ ПО ГОДОВОМУ ПРОЕКТУ

Ученик:

Грек Полина

Преподаватель:

Клюнин Алексей Олегович

Класс:

10-3

Санкт-Петербург
2017

Содержание

1	Постановка задачи	3
2	Алгоритм решения задачи	3
2.1	Базовые структуры данных	3
2.2	Построение алгоритма	3

1 Постановка задачи

Из заданного множества точек на плоскости выбрать две различные точки так, чтобы окружности заданного радиуса с центрами в этих точках содержали внутри себя одинаковое количество заданных точек.

2 Алгоритм решения задачи

2.1 Базовые структуры данных

Класс Point (точка) с двумя полями: x и y типа `double`, обозначающих координаты точки.

Класс Set (множество) с двумя полями: число `quantity` типа `integer`, которое определяет количество элементов множества, и массив `arr` из элементов класса `Dot`. Длина массива равна `quantity`.

Класс Circle(окружность) с одним полем `radius`(радиус) типа `double`

2.2 Построение алгоритма

На вход подаётся целое число (n), определяющее количество точек в множестве A , затем несколько пар вещественных чисел, являющихся координатами точек множества A .

В самой программе сначала создаётся объект a класса `Set` с заданным числом элементов с помощью конструктора от поля `quantity`. В цикле `for` каждая пара чисел считывается, затем с помощью конструктора от x и y создаётся объект класса `Dot`, и этот объект сразу же записывается в массив `a.arr`. Когда количество точек достигает n , цикл `for` заканчивается.

Создаем переменную `cnt1` типа `double`, равную нулю.

Цикл `for` проходит от первого до второго элемента массива, до третьего и т.д. до последнего. Если расстояние меньше радиуса, то прибавляем 1 к `cnt`.

Создаем переменную `cnt2` типа `double`, равную нулю.

Аналогично действуем с любой другой точкой.

Если `cnt1 = cnt2`, то выводим координаты этих точек, если нет, пробуем тоже самое с другой точкой .