**Государственное бюджетное общеобразовательное учреждение "Президентский физико-математический лицей № 239"**

**Нахождение окружности минимального радиуса, содержащей все точки множества**

**Отчёт о проекте по информатике**

Работу выполнил  
Ученик 10-7 класса  
Захаров Владимир

Санкт-Петербург

2021

**1. Постановка задачи**

На плоскости задано множество точек. Найти окружность наименьшей площади, внутри которой находятся все точки множества. Если таких окружностей несколько, найти любую. В качестве ответа нарисовать найденную окружность.

**2. Уточнение исходных и выходных данных и ограничений на них**

**2.1.Исходные данные**Во входном файле points.txt содержится множество точек на плоскости (точка задаётся с помощью двух координат типа double). Можно добавлять как по одной точке с конкретными координатами, так и добавлять требуемое количество случайных точек.

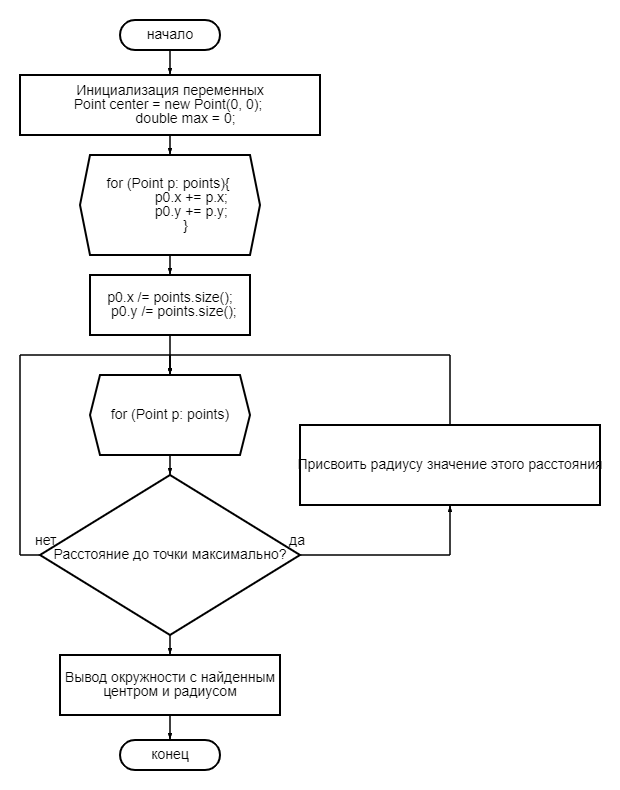
**2.2. Выходные данные**   
На выходе программа рисует искомую окружность, которая задаётся по центру и радиусу. Для наглядности её центр выделяется другим цветом

**3. Выбор метода решения**

**3.1. Анализ исходных данных и выбор используемой структуры данных**  
 Так как точка задается двумя координатами типа double рационально использовать класс Point, хранящий координаты абсциссы и ординаты точки. Для дальнейшей удобной работы точки будут заноситься в массив Arraylist <Point> points. Так будет удобно делать перебор точек в цикле for (Point p: points), а больше в решении задачи практически ничего не требуется

**3.2. Математическая модель**Нахождение окружности условно делится на две части:  
1) Поиск геометрического центра точек. Это будет центр искомой окружности. Его координаты:  
center.x = )/N; center.y = )/N, где N — число точек множества  
2) Поиск наиболее удаленной от этого центра точки. Расстояние между этими двумя точками будет искомым, а ищется оно по формуле   
d =, где d — расстояние между произвольной точкой и центром окружности

**3.3 Описание алгоритма**

****

**4. Листинг программы***Инициализация переменных*  
Point center = new Point(0,0);  
double max = 0;  
*Поиск геометрического центра точек*for (Point p: points)  
{  
 center.x += p.x;  
 center.y += p.y;  
}  
center.x /= points.size();  
center.y /= points.size();  
*Поиск радиуса окружности:*for (Point p: points)  
{  
 if (dist(center, p) > max) max = dist(center, p);  
}  
*Запись окружности*if (max != 0) resultCircle = new Circle(new Point(p0.x, p0.y), max + 0);

**5. Пример работы программы**

**5.1. Исходные данные**

1.1 1

1.2 1.2

1.3 -1

1.1 -1

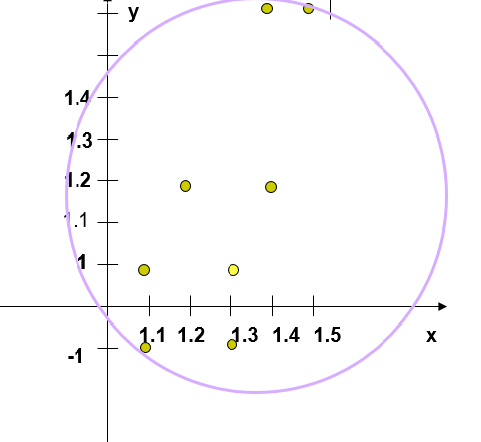
1.3 1

1.4 1.2

1.4 1.6

1.5 1.6

**5.2. Выходные данные**

****

**6. Анализ правильности решения**Программа работает правильно, причём из-за математической простоты решения ошибку допустить было практически невозможно.