**Задача 1. Точки**

Даны N точек на плоскости (для простоты можно считать, что у всех точек целочисленные координаты). Назовём расстояние от точки A до ближайшей к ней точки B "радиусом" точки A. "Соседями" точки A будем называть все точки, лежащие в пределах двойного радиуса от неё

включительно (кроме самой точки A). Для каждой точки из заданного набора определите её радиус и количество соседей.

**Описание решения**

Расстояние от точки A до точки B находится по формуле:

Для хранения вводимых данных и результатов вычислений используется двойной массив Nх4 элементов. Для каждой точки из множества N отведены четыре элемента массива в которые помещаются следующие данные:

a[ ][0] – Х a[ ][1] – Y a[ ][2] – радиус a[ ][3] – количество соседей

Нахождение радиуса. Если |АВ| меньше «радиуса» (изначально имеет максимально допустимое значение) для точки А или В, то соответствующему «радиусу» присваивается результат вычислений |АВ|.

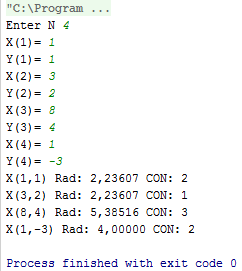
Нахождение количества соседей. Если |АВ| меньше или равно удвоенному «радиусу» точки А или В, то к значению соответствующего счетчика количества соседей (изначально он равен 0) прибавляется 1.

Так как вычисления используются сразу для двух точек и не производится вычислений расстояний от точки до неё же (Пример: |AA|) это позволяет сократить количество итераций циклов со значения до по сравнению с полным перебором.

**Листинг программы Java с комментариями**

**package** com.hh\_dots;  
**import** java.util.Scanner;  
**public class** Main{  
 **public static void** main(String[] args){  
 **int** n=0,i,j;  
 **double** distance;  
 Scanner in=**new** Scanner(System.***in***);  
 **while**(n<1){*// если значение N меньше 1 - вводим ещё раз* System.***out***.print(**"Enter N "**);  
 n=in.nextInt();*//ввод N* }  
 **double**[][]dotsData=**new double**[n][4]; *//инициализация массива* **for**(i=0;i<n;i++)  
 dotsData[i] = **new double**[4];  
 **for**(i=0;i<n;i++){ *//вод XY и форматирование массива* System.***out***.printf(**"X(%d)= "**,i+1);  
 dotsData[i][0] = in.nextDouble(); *//X* System.***out***.printf(**"Y(%d)= "**,i+1);  
 dotsData[i][1]=in.nextDouble(); *//Y* dotsData[i][2]=(n==1)?0:Double.***MAX\_VALUE***; *//«радиус»* dotsData[i][3]=0;*//счетчик «соседей»* }  
 **for**(i=0;i<n-1;i++) *//расчет «радиусов»* **for**(j=i+1;j<n;j++){  
 distance=Math.*sqrt*(Math.*pow*(dotsData[j][0]-dotsData[i][0],2)+ Math.*pow*(dotsData[j][1]-dotsData[i][1],2));  
 **if**(dotsData[i][2]>distance)dotsData[i][2]=distance;  
 **if**(dotsData[j][2]>distance)dotsData[j][2]=distance;  
 }  
 **for**(i=0;i<n-1;i++) *//расчет количества «соседей»* **for**(j=i+1;j<n;j++){  
 distance=Math.*sqrt*(Math.*pow*(dotsData[j][0]-dotsData[i][0],2)+ Math.*pow*(dotsData[j][1]-dotsData[i][1],2));  
 **if**(dotsData[i][2]\*2>=distance)dotsData[i][3]++;  
 **if**(dotsData[j][2]\*2>=distance)dotsData[j][3]++;  
 }  
 **for**(i=0;i<n;i++) *//вывод результатов* System.***out***.printf(**"X(%.0f,%.0f) Rad: %.5f CON: %.0f\n"**,dotsData[i][0], dotsData[i][1],dotsData[i][2],dotsData[i][3]);  
 }  
}

**Пример работы**

****

**Листинг программы С++ с комментариями**

#include<iostream>

#include<cmath>

#include<cfloat>

using namespace std;

int main()

{

int n=0,i,j;

while (n <1){//если N < 1 - вводим ещё раз

cout<<"Enter N ";

cin>>n; //ввод N

}

double a[n][4],distance; //объявление массива Nx4 элементов и переменной distance

for(i=0;i<n;i++){

cout<<"X"<<i+1<<"= ";

cin>>a[i][0]; //X

cout<<"Y"<<i+1<<"= ";

cin>>a[i][1]; //Y

if(n==1)a[i][2]=0; else a[i][2]= DBL\_MAX; // хц«радиус»

a[i][3]=0; //счетчик «соседей»

}

for (i=0;i<n-1;i++) //вычисление «радиуса»

for(j=i+1;j<n;j++){

distance=sqrt(pow(a[j][0]-a[i][0],2)+pow(a[j][1]-a[i][1],2)); //вычисление |AB|

if(a[i][2]>distance)a[i][2]=distance;

if(a[j][2]>distance)a[j][2]=distance;

}

for (i=0;i<n-1;i++) //вычисление количества «соседей»

for(j=i+1;j<n;j++){

distance=sqrt(pow(a[j][0]-a[i][0],2)+pow(a[j][1]-a[i][1],2));

if(a[i][2]\*2>=distance)a[i][3]++;

if(a[j][2]\*2>=distance)a[j][3]++;

}

for (i=0;i<n;i++) //вывод результата

cout<<"X("<<a[i][0]<<","<<a[i][1]<<") Rad: "<<a[i][2]<<" CON: "<<a[i][3]<<"\n";

getchar();

return 0;

}

**Пример работы**

