Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по разминочной работе №1

Тема работы: Поиск минимального и максимального элементов

Выполнил

студент: гр. 551003 Белых В.В.

Проверила: Фадеева Е.П.

Минск 2016

Содержание

[1 Постановка задачи 3](#_Toc445646286)

[2 Описание алгоритмов 4](#_Toc445646287)

[3 Основные расчетные формулы 5](#_Toc445646288)

[4 Результаты расчетов и тестирование программы 6](#_Toc445646289)

[4.1 Тест 1 6](#_Toc445646290)

[4.2 Тест 2 6](#_Toc445646291)

[4.3 Тест 3 7](#_Toc445646292)

[Приложение А 8](#_Toc445646293)

# Постановка задачи

На плоскости расположены 10 точек, которые заданы своими координатами. Найти на оси абсцисс точку, наибольшее расстояние от которой до выбранных точек было бы минимальным.

Ввод:

1. Координаты 10 точек.
2. Степень точности, т.е. если введено 6, то это 10^-6

Вывод:

1. Расстояние
2. Координата точки на оси абсцисс

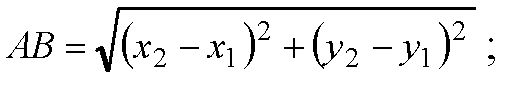
# Описание алгоритмов

Таблица – Описание алгоритмов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование  алгоритма | Назначение  алгоритма | Формальные  параметры | Рекомендуемый  тип |
| 1 | Основной  алгоритм | Вызов алгоритмов ввода, поиска границ, поиска точки и конца программы |  |  |
| 2 | Vvod(a,Eps,  EpsNum) | Алгоритм ввода | a,Eps,EpsNum | Процедура |
| 3 | SelectBr  (a,Right,Left) | Определяет границы прохода по оси абсцисс | a,Right,Left | Процедура |
| 4 | FindPoint(a,  Right,Left,eps,Abs(EpsNum),Minmax,  MinPoint) | Алгоритм для поиска точки на оси абсцисс с минимальным расстоянием до самой дальней точки. Вызывает алгоритм Max(a,x) | a,Right,Left,eps,  counter. Возвращаемые параметры:  Minmax, MinPoint | Процедура |
| 5 | EndOfProg(minPoint,minMax) | Алгоритм вывода значений. | MinPoint,MinMax | Процедура |
| 6 | Max(a, x) | Алгоритм вычисляющий максимальное расстояние от точки на оси абсцисс | a,x Возвращаемый параметр:  Result | Функция. Параметр Result – результат функции |

# Основные расчетные формулы

Расстояние между двумя точками A(x1, y1) и B(x2, y2) в прямоугольной системе координат выражается формулой:



# Результаты расчетов и тестирование программы

## Тест 1

Исходные данные: (145, 112)

(133, 168)

(33, 162)

(122, 58)

(78, 98)

(126, 82)

(125, 93)

(114, -58)

(126, -28)

(138, -62)

Accuracy: 3

Ожидаемый результат: Point coordinates: 92.900, 0

Distance: 172.719

Полученный результат:



## Тест 2

Исходные данные: (1, 1)

(2, 2)

(3, 3)

(4, 4)

(5, 5)

(6, 6)

(7, 7)

(8, 8)

(9, 9)

(10, 13)

Accuracy = 3

Ожидаемый результат: Point coordinates: 10.000, 0

Distance: 13.000

Полученный результат:



## Тест 3

Исходные данные: (145, 112)

(33, 168)

(33, 162)

(122, 58)

(78, 98)

(126, 82)

(125, 93)

(114, -58)

(126, -28)

(138, -62)

Accuracy: 3

Ожидаемый результат: Point coordinates: 33.000

Distance: 168.000

Полученный результат:



Приложение А

(обязательное)

Исходный код программы

**Program** Project1;

{$APPTYPE CONSOLE}

**Uses**

SysUtils;

**Const**

n=10;

**Type**

ArrayOFPoint = **array**[1..N,1..2] **of** real;

**Var**

a:ArrayOfPoint;

Eps,MinPoint,Minmax,right,left:real;

EpsNum:integer;

**Procedure** Vvod(**var** a:ArrayOfPoint; **var** Eps:real; **var** EpsNum:integer);

**var**

i:integer;

f:textfile;

**begin**

//Выбор ввода

writeln('Tests or your values?(1 or 2)');

readln(i);

**if** i=1 **then**

**begin**

//Выбор Теста

writeln('1 or 2 or 3?');

readln(i);

assign(f,'Test'+Inttostr(i)+'.txt');

reset(f);

**for** i:=1 **to** 10 **do**

**begin**

Write(i,'. Write x: ');

readln(f,a[i,1]);

Writeln(a[i,1]:0:3);

Write(' Write y: ');

readln(f,a[i,2]);

Writeln(a[i,2]:0:3);

**end**;

Write('Accuracy:');

readln(f,EpsNum);

Writeln(EpsNum);

Eps:=1;

close(f);

**end**

**else**

**begin**

**for** i:=1 **to** 10 **do**

**begin**

Write(i,'.Write x: ');

readln(a[i,1]);

Write('Write y: ');

readln(a[i,2]);

**end**;

Writeln('Accuracy:');

readln(EpsNum);

**end**;

Eps:=1;

**end**;

**Procedure** SelectBr(**var** a:ArrayOfPoint; **out** Right:real;**out** Left:real);

**var**

i:integer;

**begin**

Right:=a[1,1];

Left:=a[1,1];

**for** i:= 2 **to** 10 **do**

**begin**

**if** Right>=a[i,1] **then**

Right:=a[i,1]

**else**

**if** left<=a[i,1] **then**

left:=a[i,1];

**end**;

**end**;

**Function** max(**var** a:arrayOfPoint; x:real):real;

**var**

lengthnow:real;

i:integer;

**begin**

result:=Sqrt(sqr(a[1,1]-x)+sqr(a[1,2]));

**for** i:=2 **to** 10 **do**

**begin**

Lengthnow:=Sqrt(sqr(a[i,1]-x)+sqr(a[i,2]));

**if** result<lengthnow **then**

result:=lengthnow;

**end**;

**end**;

**Procedure** FindPoint(**var** a:ArrayOfPoint; Right,Left:real; eps:real;

Counter:integer; **out** Minmax, MinPoint:real);

**var**

maxnow:real;

**begin**

Minmax:=max(a,right);

MinPoint:=right;

right:=right+eps;

//Цикл поиска минимального при заданной точности

**while** right<=left **do**

**begin**

MaxNow:=max(a,right);

**if** Minmax>maxnow **then**

**begin**

Minmax:=maxnow;

MinPoint:=right;

**end**;

right:=right+eps;

**end**;

**if** counter<>0 **then**

**begin**

dec(counter);

//Вызов процедуры для поиска с еще меньшей точностью

findpoint(a,MinPoint-Eps,MinPoint+eps,eps/10,counter,Minmax,Minpoint);

**end**;

**end**;

**Procedure** EndofProg(MinPoint,Minmax:real);

**begin**

Write('Point coordinates: ');

Writeln(MinPoint:0:epsnum);

Write('Distance: ');

Writeln(MinMax:0:epsnum);

readln;

**end**;

**Begin**

Vvod(a,Eps,EpsNum);

SelectBr(a,Right,Left);

FindPoint(a,Right,Left,eps,Abs(EpsNum),Minmax,MinPoint);

EndOfProg(minPoint,minMax);

**End**.