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: Нахождение расстояния между точками на координатной плоскости

Выполнила

студент: гр. 551003 Коржов В.В.

Проверила: Фадеева Е.П.

Минск 2016

Содержание

[1 Постановка задачи 3](#_Toc445028843)

[2 Описание алгоритмов 4](#_Toc445028844)

[3 Основные расчетные формулы 5](#_Toc445028845)

[4 Результаты расчетов и тестирование программы 6](#_Toc445028846)

[4.1 Тест 1 6](#_Toc445028847)

[4.2 Тест 2 6](#_Toc445028848)

[Приложение А 7](#_Toc445028849)

# 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.

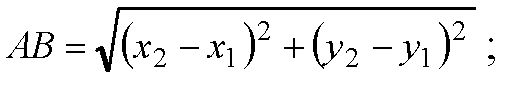
# Описание алгоритмов

Таблица – Описание алгоритмов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Наименование  алгоритма | Назначение  алгоритма | Формальные  параметры | Рекомендуемый  тип |
| 1 | Основной  алгоритм | Ввод исходных данных, вызов подпрограмм(FirstCoord (B, Result), SecondCoord (B, Result), MaxDist (X, B, Result), FindAnswer (X1, X2, Eps, B, Answer, Min)), вывод на экран координат искомой точки |  |  |
| 2 | FirstCoord (B, Result) | Определяет левую границу отрезка | B, Result.  Возвращаемый параметр: Result | Функция. Параметр Result – результат функции |
| 3 | SecondCoord (B, Result) | Определяет правую границу отрезка | B,Result.  Возвращаемый параметр: Result | Функция. Параметр Result – результат функции |
| 4 | MaxDist (X, B, Result) | Вычисляет наибольшее расстояние от X до заданных точек | X, B, Result.  Возвращаемый параметр: Result | Функция.  Параметр Result – результат функции |
| 5 | FindAnswer (X1, X2, Eps, B, Answer, Min) | Находит абсциссу точки, наибольшее расстояние от которой до заданных точек минимально | X1, X2, Eps, B, Answer, Min | Процедура |

# 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)

(33, 168)

(33, 162)

(122, 58)

(78, 98)

(126, 82)

(125, 93)

(114, -58)

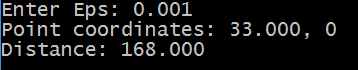
(126, -28)

(138, -62)

Eps = 0.001

Ожидаемый результат: Point cooResultdinates: 33.000, 0

Distance: 168.000

Полученный результат: 

## Тест 2

Исходные данные: (1, 1)

(2, 2)

(3, 3)

(4, 4)

(5, 5)

(6, 6)

(7, 7)

(8, 8)

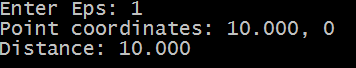
(9, 9)

(10, 10)

Eps = 1

Ожидаемый результат: Point cooResultdinates: 10.000, 0

Distance: 10.000

Полученный результат: 

Приложение А

(обязательное)

Исходный код программы

**program** Project2;

{$APPTYPE CONSOLE}

**uses**

SysUtils,

Windows;

**type**

TArray = **array**[1..2, 1..10] **of** Real;

**const**

A: TArray = ((1, 2, 3, 4, 5, 6, 7, 8, 9, 10),

(1, 2, 3, 4, 5, 6, 7, 8, 9, 10));

**var**

Eps, Answer, Min: Real;

**function** FirstCoord(B: TArray): Real;

**var**

j: Integer;

**begin**

// Нахождение левой границы отрезка

Result := B[1, 1];

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

**if** B[1, j] < Result **then**

Result := B[1, j];

**end**;

**function** SecondCoord(B: TArray): Real;

**var**

j: Integer;

**begin**

// Нахождение правой границы отрезка

Result := B[1, 1];

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

**if** B[1, j] > Result **then**

Result := B[1, j];

**end**;

**function** MaxDist(X: Real; B: TArray): Real;

**var**

j: Integer;

D: Real;

**begin**

Result := 0;

// Вычисление максимального расстояния от точки

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

**begin**

// Вычисление расстояния между двумя точками

D := Sqrt(Sqr(X - B[1, j]) + Sqr(B[2, j]));

**if** D > Result **then**

Result := D;

**end**;

**end**;

**procedure** FindAnswer(X1, X2, Eps: Real; B: TArray; **var** Answer,

Min: Real);

**var**

Max, j: Real;

**begin**

Min := MaxDist(X1, B);

Answer := X1;

j := X1 + Eps;

// Поиск минимального из максимальных расстояний с шагом Eps

**while** j < X2 + 0.00001 **do**

**begin**

Max := MaxDist(j, B);

**if** Max < Min **then**

**begin**

Min := Max;

Answer := j;

**end**;

j := j + Eps;

**end**;

**end**;

**begin**

// Ввод исходных данных

Write('Enter Eps: ');

Readln(Eps);

// Вызов процедуры поиска ответа

FindAnswer(FirstCoord(A), SecondCoord(A), Eps, A,

Answer, Min);

// Вывод ответа

Writeln('Point coordinates: ', Answer:0:3, ', 0');

Writeln('Distance: ', Min:0:3);

Readln;

**end**.