Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский экономический университет имени Г.В. Плеханова»   
**Московский приборостроительный техникум**

**Отчет**   
о выполнении практической работы №3.

Дисциплина ОП.04 «Основы алгоритмизации и программирования»

Тема: Создание программ линейной структуры

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Специалист по информационным системам

Выполнил: Dusky Fox.  
Студент группы:   
Вариант № 27  
Дата: «18» сентября 2021 г.

Проверил:   
Преподаватель: Клопов Д.А.  
Дата: «\_\_\_» \_\_\_\_\_\_\_\_\_ 202\_\_ г.   
Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

27. Треугольник задается координатами своих вершин на плоскости: A(x1, y1), В(х2,, у2), С(х3 ,y3). Найти длину и основание высоты, опущенной из вершины А на сторону ВС

#include <iostream>

#include <cmath>

using namespace std;

int main()

{

    setlocale(LC\_ALL, "Russian");

    char points[2];

    float x[2], y[2], lines[2], pol\_per, height, katets[1];

    points[0] = 'A';

    points[1] = 'B';

    points[2] = 'C';

    bool is\_real;

    is\_real = false;

    // Ввод данных о вершинах треугольника

    for (int i = 0; i < 3; i++) {

        cout << "Введите x точки " << points[i] << ' ';

        cin >> x[i];

        cout << "Введите y точки " << points[i] << ' ';

        cin >> y[i];

    }

    // Вычисление длин сторон

    // Вычисление стороны AB

    if (y[0] != y[1]) {

        if (x[0] != x[1]) {

            katets[0] = abs(x[0] - x[1]);

            katets[1] = abs(y[0] - y[1]);

            lines[0] = sqrt(pow(katets[0], 2) + pow(katets[1], 2));

        }

        else {

        }

    }

    else {

        lines[0] = abs(x[0] - x[1]);

    }

    // Вычисление стороны BC

    if (y[1] != y[2]) {

        if (x[1] != x[2]) {

            katets[0] = abs(x[1] - x[2]);

            katets[1] = abs(y[1] - y[2]);

            lines[1] = sqrt(pow(katets[0], 2) + pow(katets[1], 2));

        }

        else {

            lines[1] = abs(y[1] - y[2]);

        }

    }

    else {

        lines[1] = abs(x[1] - x[2]);

    }

    // Вычисление стороны BA

    if (y[0] != y[2]) {

        if (x[0] != x[2]) {

            katets[0] = abs(x[0] - x[2]);

            katets[1] = abs(y[0] - y[2]);

            lines[2] = sqrt(pow(katets[0], 2) + pow(katets[1], 2));

        }

        else {

            lines[2] = abs(y[0] - y[2]);

        }

    }

    else {

        lines[2] = abs(x[0] - x[2]);

    }

    // Проверка треугольника на существование

    if (lines[0] + lines[1] > lines[2]){

        if (lines[0] + lines[2] > lines[1]) {

            if (lines[2] + lines[1] > lines[0]) {

                is\_real = true;

            }

        }

    }

    // Расчёт высоты

    if (is\_real) {

        pol\_per = (lines[0] + lines[1] + lines[2]) / 2;

        height = (2 \* sqrt(pol\_per \* (pol\_per - lines[0]) \* (pol\_per - lines[1]) \* (pol\_per - lines[2]))/lines[1]);

        cout << "Длина высоты вашего треугольника - " << height;

    }

    else {

        cout << "Такого треугольника не существует.";

    }

    system("pause");

    return 0;

}

Начало

float x[2], y[2]

y[0] != y[1]

x[0] != x[1]

Float lines[2], pol\_per, height, katets[1]

Нет

lines[0] = abs(x[0] - x[1])

Да

Нет

lines[0] = abs(y[0] - y[1])

Да

katets[0] = abs(x[0] - x[1])

            katets[1] = abs(y[0] - y[1]);

            lines[0] = sqrt(pow(katets[0], 2) + pow(katets[1], 2))

y[2] != y[1]

x[2] != x[1]

katets[0] = abs(x[1] - x[2])

            katets[1] = abs(y[1] - y[2])

            lines[1] = sqrt(pow(katets[0], 2) + pow(katets[1], 2))

lines[1] = abs(x[1] - x[2])

lines[1] = abs(y[1] - y[2])

Нет

Да

Нет

Да

Нет

y[2] != y[0]

lines[2] = abs(x[0] - x[2])

Да

Нет

x[2] != x[0]

katets[0] = abs(x[0] - x[2])

            katets[1] = abs(y[0] - y[2])

            lines[2] = sqrt(pow(katets[0], 2) + pow(katets[1], 2))

lines[2] = abs(y[0] - y[2])

lines[0] + lines[1] > lines[2]

lines[0] + lines[2] > lines[1]

lines[2] + lines[1] > lines[0]

pol\_per = (lines[0] + lines[1] + lines[2]) / 2

        height = (2 \* sqrt(pol\_per \* (pol\_per - lines[0]) \* (pol\_per - lines[1]) \* (pol\_per - lines[2]))/lines[1])

Нет

"Такого треугольника не существует."

Да

"Длина высоты вашего треугольника - ", height

Конец

