

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Вычислительной Техники**

**ОТЧЕТ**  
**по лабораторной работе №5**  
**по дисциплине «Информатика»**  
**Тема: Классы в C++**

Студенты гр.6493

Преподаватель

\_\_\_\_\_

\_\_\_\_\_

Мурашко А.С  
Огурецкий Д.В.

Гречухин М. Н

Санкт-Петербург

2017

## Цель работы.

**Цель работы:** научиться описывать классы, познакомиться с методами и свойствами классов языка C++.

**Результат выполнения работы:**

Задача: Описать класс "Отрезок", представляющий собой описание отрезка прямой на плоскости.

Условие: Класс должен содержать следующие свойства:

- координаты концов отрезка;
- длина отрезка

Класс должен содержать следующие методы:

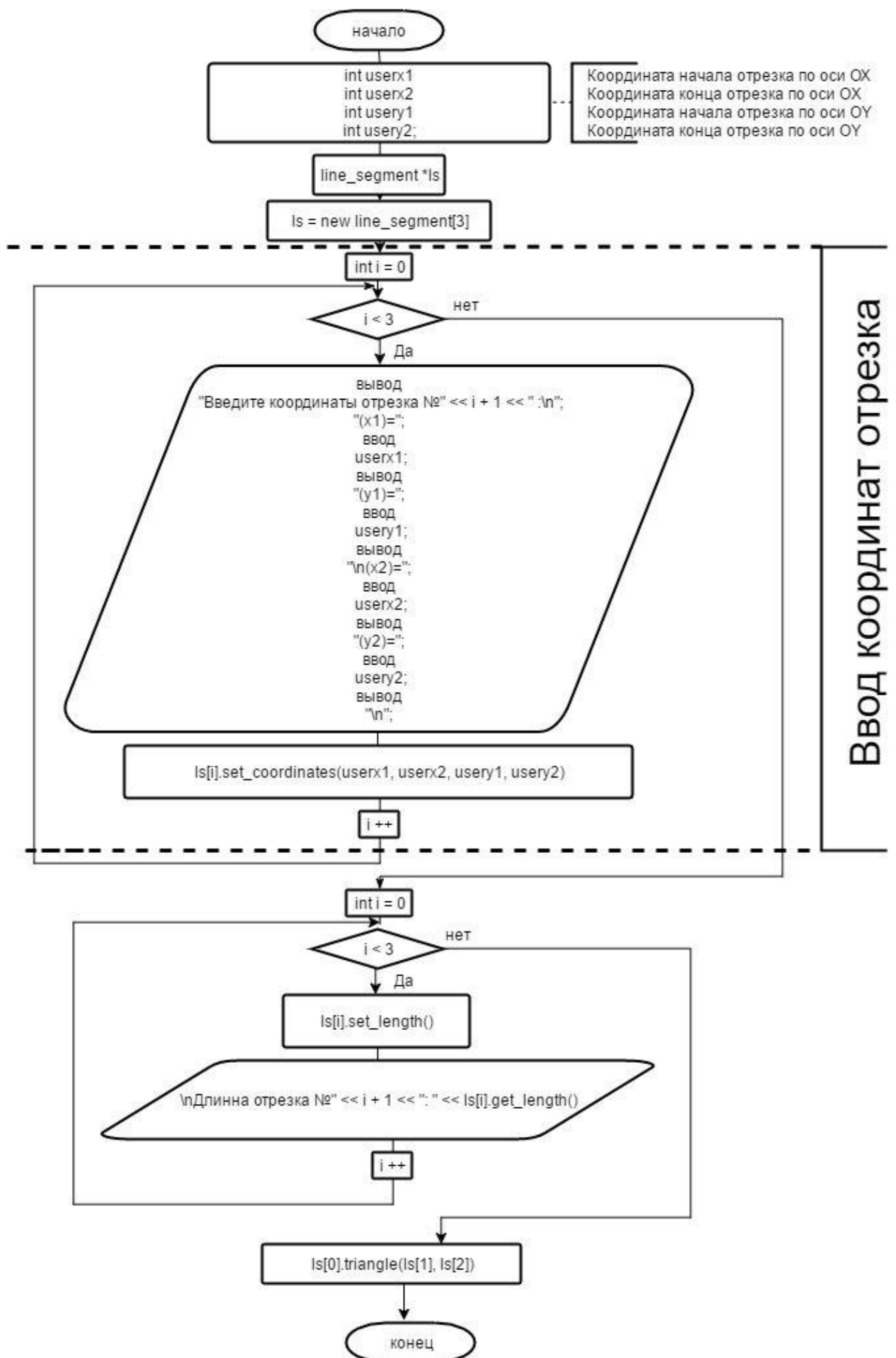
- геттеры и сеттеры для координат;
- геттер для длины отрезка;
- метод, определяющий длину отрезка (результат заносится в соответствующее свойство);
- метод, определяющий возможность построения треугольника из данного отрезка и двух других отрезков (в качестве параметра метод принимает два объекта класса "Отрезок").

Запишем решение этой задачи:

В главной функции программы создаем три произвольных отрезка, заполняем с клавиатуры координаты их концов. Определяем и выводим на экран длины трёх отрезков.

Для этого создаем класс `line_segment`, в котором указываем его свойства и методы. Чтобы эти три отрезка создавали треугольник воспользуемся неравенством треугольника  $|A-B| < C < |A+B|$ .

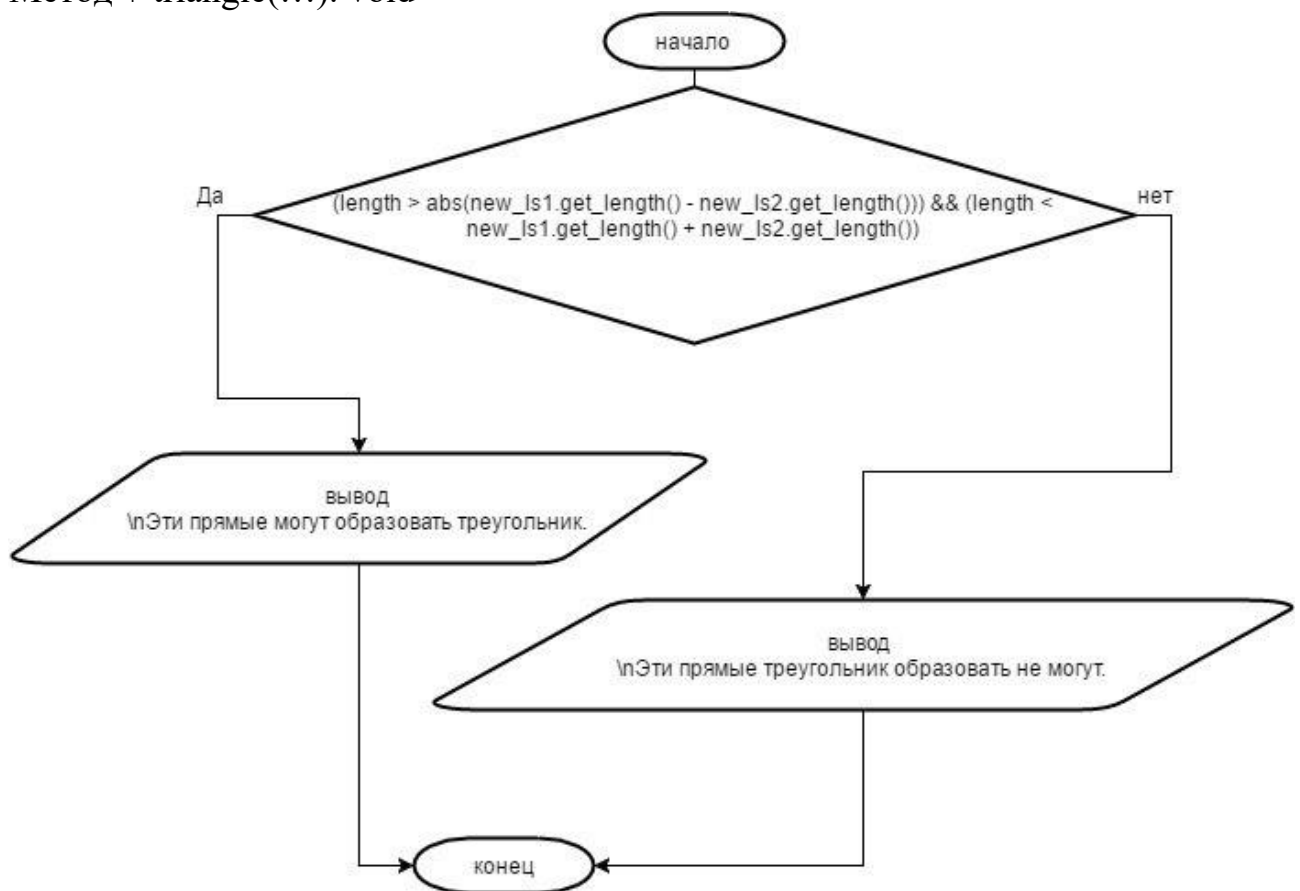
Оформим решение в виде алгоритма: Главная функция `main`



## Наш класс

line_segment
- x1: int - x2: int - y1: int - y2: int - length: float
+ set_length(): void + get_length(): float + set_coordinates(int new_x1, int new_x2, int new_y1, int new_y2): void + get_x1(): int + get_x2(): int + get_y1(): int + get_y2(): int + triangle(line_segment new_ls1, line_segment new_ls2): void

Метод + triangle(...): void



## Ход работы

```
#include<iostream>
#include<conio.h>
using namespace std;

class line_segment
{
private:
    int x1; //Координата начала отрезка по оси OX
    int x2; //Координата конца отрезка по оси OX
    int y1; //Координата начала отрезка по оси OY
    int y2; //Координата конца отрезка по оси OY
    float length; // Длина отрезка
public:
    void set_length()
    {
        length = sqrt((float)((x2 - x1)*(x2 - x1) + (y2 - y1)*(y2 - y1)));
    }

    float get_length()
    {
        return length;
    }

    void set_coordinates(int new_x1, int new_x2, int new_y1, int new_y2)
    {
        x1 = new_x1;
        x2 = new_x2;
        y1 = new_y1;
        y2 = new_y2;
    }

    int get_x1()
    {
        return x1;
    }
    int get_x2()
    {
        return x2;
    }
    int get_y1()
    {
        return y1;
    }
    int get_y2()
    {
        return y2;
    }

    void triangle(line_segment new_ls1, line_segment new_ls2)
    {
        if ((length > abs(new_ls1.get_length() - new_ls2.get_length())) &&
            (length < new_ls1.get_length() + new_ls2.get_length()))
            cout << "\nЭти прямые могут образовать треугольник.";
        else
            cout << "\nЭти прямые треугольник образовать не могут.";
    }
};

int main(void)
```

```

{
    setlocale(LC_ALL, "Russian");
    int userx1; //Координата начала отрезка по оси OX
    int userx2; //Координата конца отрезка по оси OX
    int usery1; //Координата начала отрезка по оси OY
    int usery2; //Координата конца отрезка по оси OY

    line_segment *ls;
    ls = new line_segment[3];

    //ввод координат отрезка и запись их в свойства класса
    for (int i = 0; i < 3; i++)
    {
        cout << "Введите координаты отрезка №" << i + 1 << " : \n";
        cout << "(x1)=";
        cin >> userx1;
        cout << "(y1)=";
        cin >> usery1;
        cout << "\n(x2)=";
        cin >> userx2;
        cout << "(y2)=";
        cin >> usery2;
        cout << "\n";
        ls[i].set_coordinates(userx1, userx2, usery1, usery2);
    }

    for (int i = 0; i < 3; i++)
    {
        ls[i].set_length();
        cout << "\nДлина отрезка №" << i + 1 << ": " << ls[i].get_length() << " ";
    }

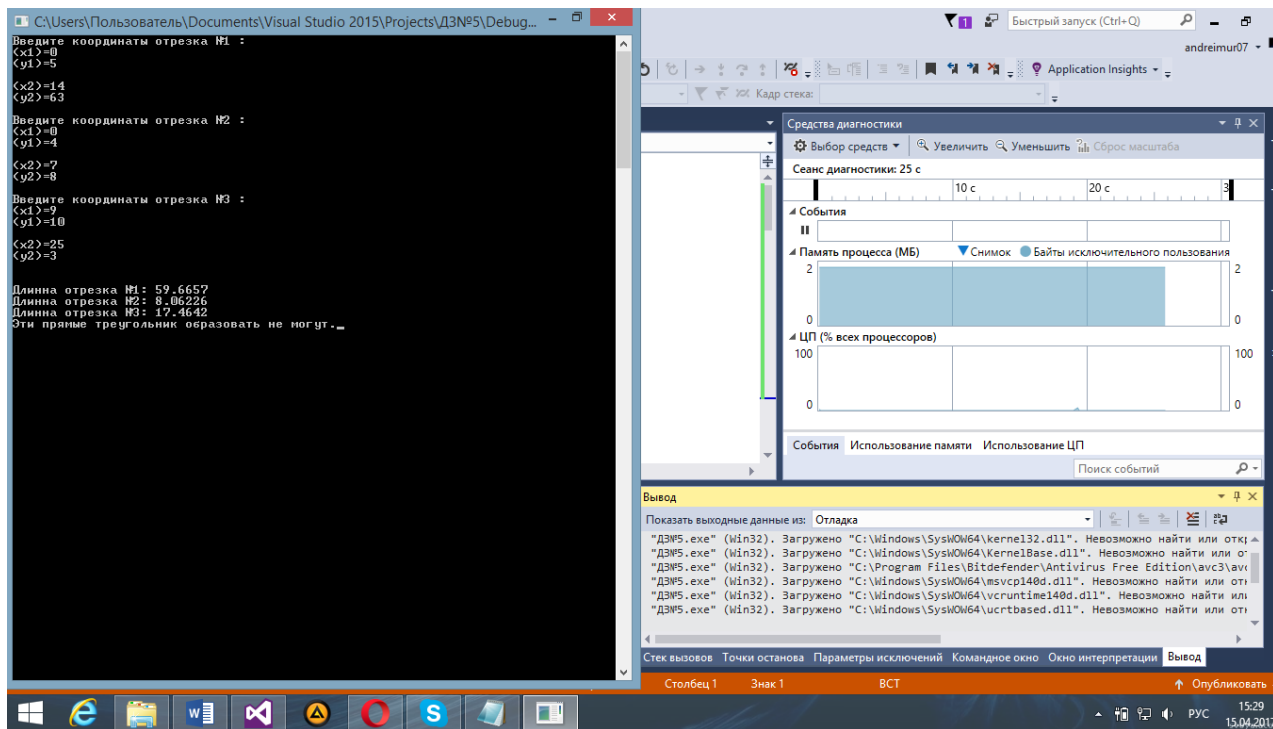
    ls[0].triangle(ls[1], ls[2]);

    _getch();
    return 0;
}

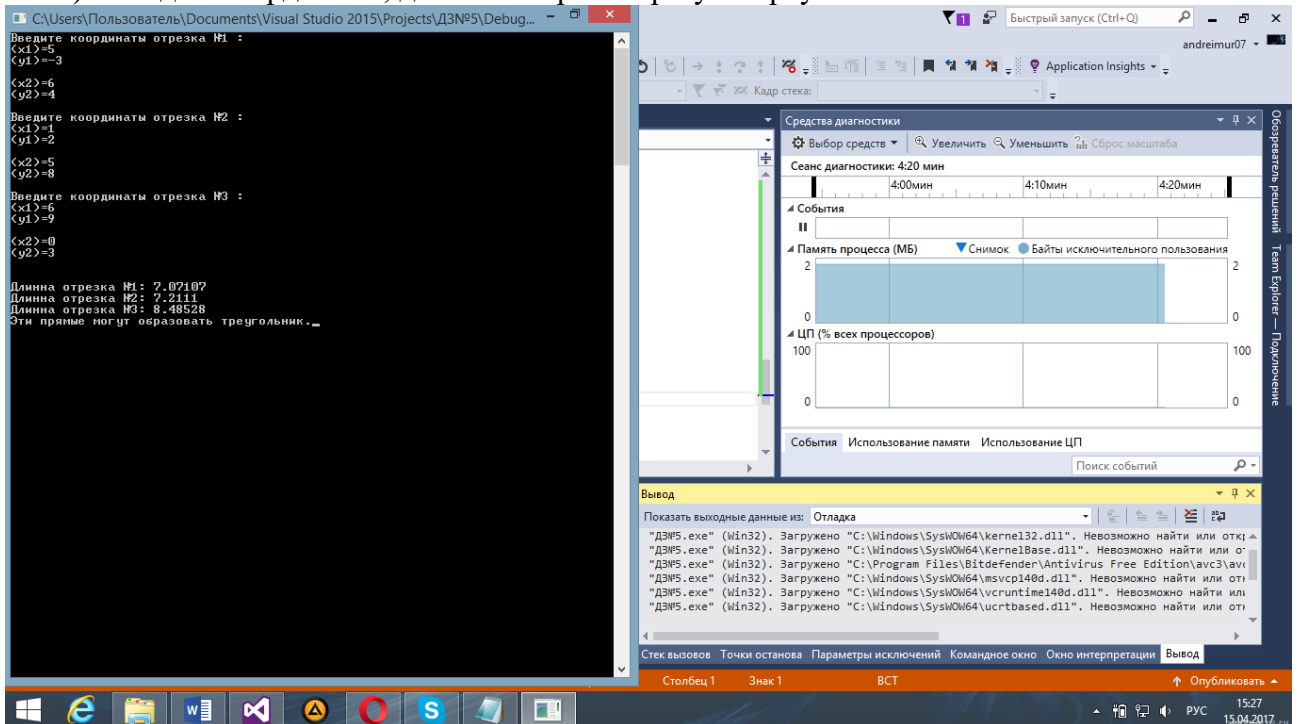
```

# Проверка

## 1) Вводим координаты, длины которых не образуют треугольники



## 2) Введем координаты, длины которых образуют треугольники.



Наша программа работает успешно!

