Для решения предложенной задачи необходимо создать четыре функции :

1 ) Input - ввод координат вершин треугольника и проверка на правильность ввода . Очевидно , что если пользователь случайно или намеренно введет координаты таких точек , из которых нельзя построить треугольник (например , точки будут лежать на одной прямой) , то дальнейшее решение задачи невозможно. Поэтому сделаем выходным параметром этой функции тоже булеву переменную , котюорая будет фиксировать , правильно введены координаты точки ( true ) или нет ( false ) . Значение этой переменной мы присваиваем , учитывая такое известное всем ученикам правило: треугольник существует тогда и только тогда , когда длина любой его стороны будет меньше суммы двух других сторон .

Далее в основной программе мы будем контролировать значения, возвращает данная функция и заставлять пользователя ввести правильные данные ( принудительное введение безошибочных данных не обязательно , но в данном случае мы поставили цель научить учащихся оформлять алгоритмы таким образом , чтобы программа " не пропускала " неправильное введение ) .

2 ) Len - функция, вычисляет длину отрезка по заданным координатам его концов. Для выполнения этого вычисления вспомним , что длину отрезка можно найти по теореме Пифагора : Image005

Где z - искомая длина отрезка ;

( x1 , y1 ) , ( x2 , y2) - координаты концов отрезка .

3 ) S - функция, определяющая площадь треугольника. Так как нам известны только координаты вершин треугольника , то площадь удобно вычислять по формуле Герона : Image006

Где p - полупериметр треугольника ;

a , b , c - стороны треугольника.

4 ) P - функция, определяющая периметр треугольника ( эта формула очевидна , поэтому здесь мы ее приводить не будем )

Причем для конкретно заданной задачи функция поиска периметру треугольника может быть вложенной в функцию поиска площади треугольника.