

Яндекс. Тренировки по алгоритмам 2.0, занятие 1 (В)

Е. Точка и треугольник

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

На координатной плоскости расположены равнобедренный прямоугольный треугольник ABC с длиной катета d и точка X. Катеты треугольника лежат на осях координат, а вершины расположены в точках: A (0,0), B (d,0), C (0,d).
Напишите программу, которая определяет взаимное расположение точки X и треугольника. Если точка X расположена внутри или на сторонах треугольника, выведите 0. Если же точка находится вне треугольника, выведите номер ближайшей к ней вершины.

Формат ввода

Сначала вводится натуральное число d (не превосходящее 1000), а затем координаты точки X – два целых числа из диапазона от –1000 до 1000.

Формат вывода

Если точка лежит внутри, на стороне треугольника или совпадает с одной из вершин, то выведите число 0. Если точка лежит вне треугольника, то выведите номер вершины треугольника, к которой она расположена ближе всего (1 – к вершине A, 2 – к B, 3 – к C). Если точка расположена на одинаковом расстоянии от двух вершин, выведите ту вершину, номер которой меньше.

Пример 1

Ввод <input type="text"/>	Вывод <input type="text"/>
5	0
1 1	

Пример 2

Ввод <input type="text"/>	Вывод <input type="text"/>
3	1
-1 -1	

Пример 3

Ввод <input type="text"/>	Вывод <input type="text"/>
4	2
4 4	

Пример 4

Ввод

Вывод

4
2 2

0

Примечания

Комментарии к примерам тестов

1. Точка лежит внутри треугольника.
2. Точка лежит вне треугольника и ближе всего к ней вершина А
3. Точка лежит на равном расстоянии от вершин В и С, в этом случае нужно вывести ту вершину, у которой номер меньше, т.е. выведено должно быть число 2
4. Точка лежит на стороне треугольника.

Язык Python 3.12.1

Набрать здесь

Отправить файл

```
1 # считываем данные
2 d = int(input().strip()) # катет треугольника
3 x = list(map(int, input().split()))
4
5 if 0 <= x[0] <= d and 0 <= x[1] <= d and x[0] + x[1] <= d:
6     print(0)
7 else:
8     coords = ((1, 0, 0), (2, d, 0), (3, 0, d))
9     dists = []
10    for coord in coords:
11        dists.append(((x[0]-coord[1])**2 + (x[1]-coord[2])**2, coord[0]))
12    dists.sort()
13    print(dists[0][1])
```

Отправить

Предыдущая