

Тренировки по алгоритмам 5.0 от Яндекса — Занятие 3 (Множества и словари)

20 мар 2024, 23:12:19
старт: 15 мар 2024, 20:30:00
финиш: 27 мар 2024, 18:00:00
до финиша: 6д. 18ч.
начало: 15 мар 2024, 20:30:00
конец: 27 мар 2024, 18:00:00
длительность: 11д. 21ч.

G. Построить квадрат

Ограничение времени	2 секунды
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Задано множество, состоящее из N различных точек на плоскости. Координаты всех точек — целые числа. Определите, какое минимальное количество точек нужно добавить во множество, чтобы нашлось четыре точки, лежащие в вершинах квадрата.

Формат ввода

В первой строке вводится число N ($1 \leq N \leq 2000$) — количество точек.
В следующих N строках вводится по два числа x_i, y_i ($-10^8 \leq x_i, y_i \leq 10^8$) — координаты точек.

Формат вывода

В первой строке выведите число K — минимальное количество точек, которые нужно добавить во множество.
В следующих K строках выведите координаты добавленных точек x_i, y_i через пробел. Координаты должны быть целыми и не превышать 10^9 по модулю.
Если решений несколько — выведите любое из них.

Пример 1

Ввод	Вывод
2	2
0 1	0 0
1 0	1 1

Пример 2

Ввод	Вывод
3	1
0 2	0 0
2 0	
2 2	

Пример 3

Ввод

Вывод

4

-1 1

1 1

-1 -1

1 -1

0

Язык Python 3.9 (PyPy 7.3.11)

Набрать здесь Отправить файл

```
1 dots_set = set()
2
3 with open('input.txt', 'r') as file:
4     k = int(file.readline())
5     for i in range(k):
6         x, y = map(int, file.readline().split())
7         dots_set.add((x, y))
8
9 dots = list(dots_set)
10 length = len(dots)
11 answer = []
12
13 if length == 0:
14     answer.append((0, 0))
15     answer.append((1, 0))
16     answer.append((0, 1))
17     answer.append((1, 1))
18 elif length == 1:
19     answer.append((dots[0][0]+1, dots[0][1]))
20     answer.append((dots[0][0], dots[0][1]+1))
21     answer.append((dots[0][0]+1, dots[0][1]+1))
22 elif length == 2:
23     a = dots[0]
24     b = dots[1]
25
26     if (a[0] < b[0] and a[1] == b[1]) or (a[0] < b[0] and a[1] > b[1]) or (a[0] > b[0] and a[1] < b[1]) or (a[0] == b[0] and a[1]
27         abx = abs(b[0] - a[0])
28         aby = abs(b[1] - a[1])
29         cx = a[0] + aby
30         cy = a[1] + abx
31         c = (cx, cy)
32         dx = b[0] + aby
33         dy = b[1] + abx
34         d = (dx, dy)
35         answer = [c, d]
36     else:
37         abx = abs(a[0] - b[0])
38
```

Отправить

Предыдущая

Следующая