



## Ломаная

Азербайджан известен своими коврами. Вы мастер рисунка по коврам, и вы хотите оформить новый ковёр с использованием **ломаной**. Ломаная — это последовательность из  $t$  отрезков на двумерной плоскости, которая задаётся последовательностью из  $t + 1$  вершин  $p_0, \dots, p_t$  следующим образом. Для каждого  $0 \leq j \leq t - 1$  между вершинами  $p_j$  и  $p_{j+1}$  проводится отрезок.

Приступая к рисунку, вы отметили  $n$  **точек** на плоскости. Точка  $i$  ( $1 \leq i \leq n$ ) имеет координаты  $(x[i], y[i])$ . **Не существует пары точек с совпадающими  $x$ -координатами или с совпадающими  $y$ -координатами.**

Теперь вы хотите найти последовательность вершин  $(sx[0], sy[0]), (sx[1], sy[1]), \dots, (sx[k], sy[k])$ , задающую ломаную со следующими свойствами:

- Ломаная начинается в  $(0, 0)$  (то есть,  $sx[0] = 0$  и  $sy[0] = 0$ ),
- Ломаная проходит через все точки (точки не обязаны совпадать с вершинами ломаной),
- Ломаная состоит только из горизонтальных и вертикальных отрезков (у любых двух последовательных вершин ломаной должны совпадать либо  $x$ -координаты, либо  $y$ -координаты).

Ломаная может самопересекаться и самонакладываться произвольным образом. Формально говоря, любая точка плоскости может содержаться в произвольном количестве отрезков ломаной.

Это задача с открытыми тестами и частичной системой оценивания каждого теста. Вам дано 10 входных файлов, описывающих положение точек. Для каждого входного файла вы должны отправить выходной файл, описывающий ломаную с требуемыми свойствами. Ваши баллы за каждый выходной файл зависят от **количества отрезков** в найденной ломаной.

В данной задаче не требуется отправлять какой-либо исходный код на проверку.

## Формат входных данных

Каждый входной файл содержит данные в следующем формате:

- строка 1:  $n$
- строка  $1 + i$  (для  $1 \leq i \leq n$ ):  $x[i] \ y[i]$

## Формат выходных данных

Каждый выходной файл должен содержать данные в следующем формате:

- строка 1:  $k$
- строка  $1 + j$  (для  $1 \leq j \leq k$ ):  $sx[j] \ sy[j]$

Обратите внимание, что вторая строка должна содержать  $sx[1]$  и  $sy[1]$  (то есть, **не требуется** выводить  $sx[0]$  и  $sy[0]$ ). Все числа  $sx[j]$  и  $sy[j]$  должны быть целыми.

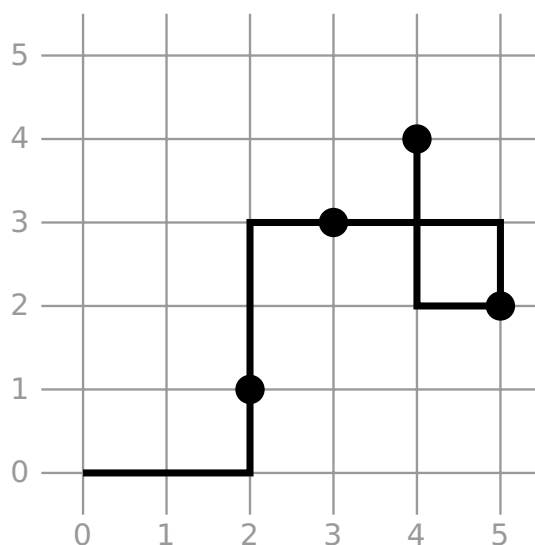
## Пример

Для примера входных данных:

```
4
2 1
3 3
4 4
5 2
```

возможен следующий вывод:

```
6
2 0
2 3
5 3
5 2
4 2
4 4
```



Обратите внимание, что данный пример не фигурирует среди тестов задачи.

## Ограничения

- $1 \leq n \leq 100\,000$
- $1 \leq x[i], y[i] \leq 10^9$
- Все  $x[i]$  и  $y[i]$  являются целыми числами.
- Ни у какой пары точек не совпадают  $x$ -координаты и  $y$ -координаты, то есть  $x[i_1] \neq x[i_2]$  и  $y[i_1] \neq y[i_2]$  для  $i_1 \neq i_2$ .
- $-2 \cdot 10^9 \leq sx[i], sy[i] \leq 2 \cdot 10^9$
- Размер каждого файла, отправляемого на проверку (либо выходного файла, либо zip-архива) не должен превосходить 15МВ.

## Система оценивания

За каждый тест вы можете получить до 10 баллов. Ваш вывод получит 0 баллов за тест, если он не задаёт ломаную с указанными свойствами. В противном случае ваши баллы будут определяться с использованием убывающей последовательности  $c_1, \dots, c_{10}$ , которая зависит от теста.

Предположим, что вы нашли ломаную, состоящую из  $k$  отрезков. В таком случае вы получите:

- $i$  баллов, если  $k = c_i$  (для  $1 \leq i \leq 10$ ),
- $i + \frac{c_i - k}{c_i - c_{i+1}}$  баллов, если  $c_{i+1} < k < c_i$  (для  $1 \leq i \leq 9$ ),
- 0 баллов, если  $k > c_1$ ,
- 10 баллов, если  $k < c_{10}$ .

Последовательности  $c_1, \dots, c_{10}$  для каждого теста представлены ниже.

Тест	01	02	03	04	05	06	07-10
$n$	20	600	5 000	50 000	72 018	91 891	100 000
$c_1$	50	1 200	10 000	100 000	144 036	183 782	200 000
$c_2$	45	937	7 607	75 336	108 430	138 292	150 475
$c_3$	40	674	5 213	50 671	72 824	92 801	100 949
$c_4$	37	651	5 125	50 359	72 446	92 371	100 500
$c_5$	35	640	5 081	50 203	72 257	92 156	100 275
$c_6$	33	628	5 037	50 047	72 067	91 941	100 050
$c_7$	28	616	5 020	50 025	72 044	91 918	100 027
$c_8$	26	610	5 012	50 014	72 033	91 906	100 015
$c_9$	25	607	5 008	50 009	72 027	91 900	100 009
$c_{10}$	23	603	5 003	50 003	72 021	91 894	100 003

## Визуализатор

Среди приложений к этой задаче имеется скрипт, позволяющий визуализировать входные и выходные файлы.

Чтобы визуализировать входной файл, используйте следующую команду:

```
python vis.py [input file]
```

Также вы можете визуализировать ваш ответ для некоторого теста, используя следующую команду. По техническим причинам визуализатор показывает только **первые 1000 отрезков** из выходного файла.

```
python vis.py [input file] --solution [output file]
```

Пример:

```
python vis.py examples/00.in --solution examples/00.out
```