



Linea spezzata

L'Azerbaijan è un paese molto famoso per i suoi tappeti. Da esperto designer di tappeti, vuoi creare un nuovo design a **linea spezzata**. Una linea spezzata è una sequenza di t segmenti che congiungono $t + 1$ punti p_0, \dots, p_t in un piano bidimensionale come segue: per ogni $0 \leq j \leq t - 1$ esiste un segmento che connette i punti p_j e p_{j+1} .

Per la creazione del nuovo design hai già selezionato n punti in un piano bidimensionale, dove il punto i ($1 \leq i \leq n$) è alle coordinate $(x[i], y[i])$. **Non esistono due punti con la stessa coordinata x né con la stessa coordinata y.**

Il tuo compito è di trovare una linea spezzata, con estremi $(sx[0], sy[0])$, $(sx[1], sy[1])$, \dots , $(sx[k], sy[k])$, tale che:

- cominci nel punto $(0, 0)$ (ovvero, $sx[0] = 0$ e $sy[0] = 0$),
- contenga tutti i punti selezionati (anche non come estremi di uno dei suoi segmenti), e
- sia composta solamente di segmenti *orizzontali* o *verticali* (in altre parole, due estremi consecutivi della linea devono avere la stessa coordinata x o la stessa coordinata y).

La linea spezzata può auto-intersecarsi e sovrapporsi in qualsiasi modo: ogni punto del piano può appartenere ad un qualsiasi numero di segmenti della linea spezzata.

Questo task è output-only con punteggi parziali. In input avrai 10 file contenenti le coordinate dei punti. Per ogni file di input, dovrai sottoporre un file di output che descriva una linea spezzata con le proprietà richieste. Per ogni file di output che descrive una linea spezzata valida, il tuo punteggio dipenderà dal **numero di segmenti** della linea spezzata (vedi sezione Punteggio).

Non devi sottoporre nessun codice sorgente per questo task.

Formato di input

I file di input seguono il seguente formato:

- riga 1: n
- righe $1 + i$ (for $1 \leq i \leq n$): $x[i] \ y[i]$

Formato di output

I file di output devono seguire il seguente formato:

- riga 1: k
- righe $1 + j$ (for $1 \leq j \leq k$): $sx[j] \ sy[j]$

Nota che la seconda riga deve contenere $sx[1]$ e $sy[1]$: in altre parole, l'output **non deve** contenere $sx[0]$ e $sy[0]$. Ogni $sx[j]$ e $sy[j]$ deve essere un numero intero.

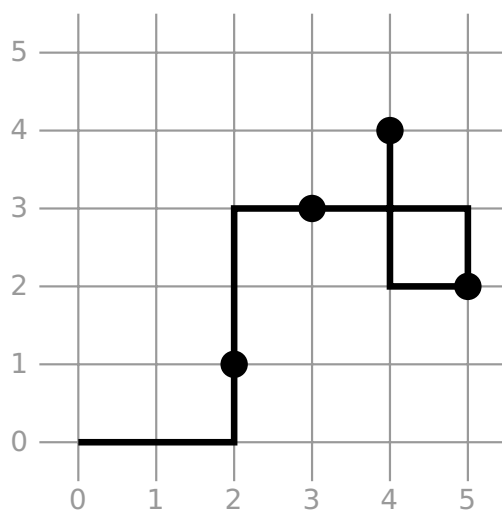
Esempio

Per l'input di esempio:

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

un possibile output è:

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



Nota che questo esempio non è tra gli input di questo task.

Assunzioni

- $1 \leq n \leq 100\,000$.
- $1 \leq x[i], y[i] \leq 10^9$.
- Tutti i valori $x[i]$ e $y[i]$ sono numeri interi.
- Non esistono due punti con la stessa coordinata x o la stessa coordinata y :
 $x[i_1] \neq x[i_2]$ e $y[i_1] \neq y[i_2]$ per $i_1 \neq i_2$.
- $-2 \cdot 10^9 \leq sx[j], sy[j] \leq 2 \cdot 10^9$.
- La dimensione del file sottoposto (che sia un output o un file compresso) non può superare i 15MB.

Punteggio

Per ogni test case, puoi ottenere fino a 10 punti. Se il tuo output non definisce una linea spezzata con le proprietà richieste otterrai 0 punti. Altrimenti, il punteggio sarà determinato usando una sequenza decrescente c_1, \dots, c_{10} che varia a seconda del test case.

Assumendo che la tua soluzione sia una linea spezzata valida di k segmenti, otterrai:

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

La sequenza c_1, \dots, c_{10} per ogni test case è riportata nella tabella sottostante.

Testcases	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

Visualizzatore

Tra gli allegati di ogni task, c'è uno script che permette di visualizzare i file di input e output. Per visualizzare un file di input, devi usare il seguente comando:

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

Puoi anche visualizzare la tua soluzione di un input usando il seguente comando. A causa di limiti tecnici, il visualizzatore mostra solamente **i primi 1000 segmenti** del file di output.

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

Esempio:

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