



Ligne brisée (Broken Line)

L'Azerbaïdjan est connu pour ses tapis. Étant maître tapissier, vous souhaitez concevoir un nouveau style de tapis en dessinant une **ligne brisée**. Une ligne brisée est une séquence de t segments dans le plan, définie par une séquence de $t + 1$ points p_0, \dots, p_t . Pour chaque $0 \leq j \leq t - 1$, un segment connecte les points p_j et p_{j+1} .

Afin de concevoir ce nouveau style, vous avez déjà placé n **marques** dans un plan en deux dimensions. Les coordonnées de la marque i ($1 \leq i \leq n$) sont $(x[i], y[i])$. **Deux marques ne peuvent avoir ni la même coordonnée x, ni la même coordonnée y.**

Vous souhaitez trouver une séquence de points $(sx[0], sy[0]), (sx[1], sy[1]), \dots, (sx[k], sy[k])$, définissant une ligne brisée qui

- commence en position $(0, 0)$ (c'est à dire, $sx[0] = 0$ et $sy[0] = 0$),
- contient toutes les marques (pas forcément au niveau des extrémités de segments), et
- n'est constitué que de segments verticaux ou de segments horizontaux (deux points consécutifs ont une coordonnée x ou y en commun).

La ligne brisée peut s'intersecter ou se superposer à elle même de n'importe quelle manière. Formellement, chaque point du plan peut appartenir à un nombre quelconque de segments de la ligne brisée.

Ce sujet est un problème de type "output-only" avec un score partiel. On vous fournit 10 fichiers tests spécifiant les positions des marques. Pour chaque fichier d'entrée, vous devez soumettre un fichier de sortie qui décrit une ligne brisée satisfaisant les conditions énoncées. Pour chaque fichier de sortie qui décrit une ligne brisée valide, votre score dépend du **nombre de segments** de la ligne brisée (voir la section Score ci-dessous).

Vous ne devez pas soumettre de code source pour ce problème.

Format d'entrée

Chaque fichier d'entrée a le format suivant :

- ligne 1 : n
- ligne $1 + i$ (pour $1 \leq i \leq n$) : $x[i] \ y[i]$

Format de sortie

Chaque fichier de sortie doit avoir le format suivant :

- ligne 1 : k
- ligne $1 + j$ (pour $1 \leq j \leq k$) : $sx[j] \ sy[j]$

Notez que la deuxième ligne de l'entrée doit contenir $sx[1]$ et $sy[1]$ (i.e., la sortie **ne doit pas** contenir $sx[0]$ et $sy[0]$). Les valeurs $sx[j]$ et $sy[j]$ doivent être des entiers.

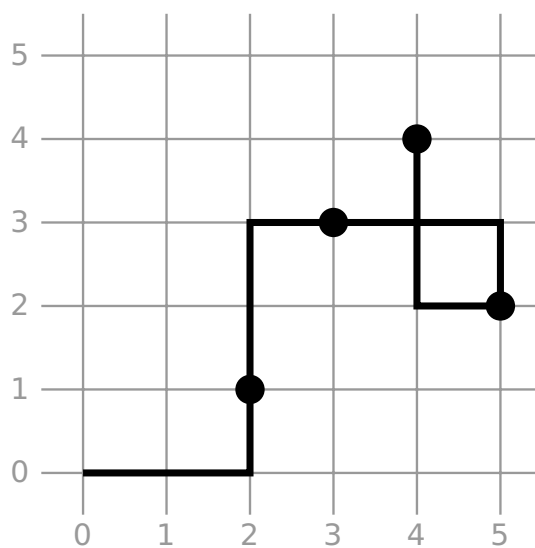
Exemple

Pour le fichier d'entrée :

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

la sortie suivante est une sortie valide possible :

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



Notez que cet exemple ne fait pas partie des fichiers tests du problème.

Contraintes

- $1 \leq n \leq 100\,000$
- $1 \leq x[i], y[i] \leq 10^9$
- Toutes les valeurs $x[i]$ et $y[i]$ sont des entiers.
- Deux points ne peuvent avoir ni la même coordonnée x , ni la même coordonnée y , i.e. $x[i_1] \neq x[i_2]$ **et** $y[i_1] \neq y[i_2]$ pour $i_1 \neq i_2$.
- $-2 \cdot 10^9 \leq sx[j], sy[j] \leq 2 \cdot 10^9$
- Il est impossible de soumettre un fichier (soit un fichier de sortie, soit une archive zip) dont la taille dépasse 15Mo.

Score

Pour chaque fichier test, vous pouvez obtenir jusqu'à 10 points. Un fichier de sortie obtiendra 0 point s'il ne décrit pas une ligne brisée qui satisfait les conditions énoncées. Sinon, le score sera déterminé à partir d'une séquence décroissante c_1, \dots, c_{10} , qui dépend du fichier test.

Si on considère que votre solution est une ligne brisée valide constituée de k segments, votre score sera de

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

La séquence c_1, \dots, c_{10} de chaque fichier test est donnée dans le tableau ci-dessous.

Fichier test	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

Visualisation

Parmi les fichiers joints à ce sujet se trouve un script qui vous permet de visualiser les fichiers d'entrée et de sortie.

Pour visualiser un fichier d'entrée, utilisez la commande suivante :

```
python vis.py [fichier d'entrée]
```

Vous pouvez également visualiser une solution pour un fichier test, en utilisant la commande suivante. À cause de limitations techniques, le script ne trace que **les 1000 premiers segments** du fichier de sortie.

```
python vis.py [fichier d'entrée] --solution [fichier de sortie]
```

Exemple :

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