



Gebrochene Linie

Azerbaïjan ist berühmt für seine Teppiche. Als Meister der Teppichkunst, möchtest du ein neues Muster entwerfen, indem du eine **gebrochene Linie** zeichnest. Eine gebrochene Linie ist eine Folge von t Segmenten auf einer zweidimensionalen Ebene, welche durch eine Folge von $t+1$ Punkten p_0, \dots, p_t gegeben ist. Für jedes j ($0 \leq j \leq t-1$) gibt es ein Segment, welches die Punkte p_j und p_{j+1} verbindet.

Zu diesem Zweck hast du bereits n **Punkte** auf der Ebene eingezeichnet. Die Koordinaten des Punktes i ($1 \leq i \leq n$) sind durch das Paar $(x[i], y[i])$ gegeben. **Keine zwei Punkte haben die gleiche x - oder die gleiche y -Koordinate.**

Dein Ziel ist jetzt, eine Folge von Punkten $(sx[0], sy[0]), (sx[1], sy[1]), \dots, (sx[k], sy[k])$ zu finden, welche eine gebrochene Linie definiert, sodass Folgendes gilt.

- Sie beginnt bei $(0, 0)$ (das heißt $sx[0] = 0$ and $sy[0] = 0$).
- Sie enthält alle Punkte (nicht unbedingt als Endpunkte der Segmente).
- Sie besteht ausschließlich aus horizontalen oder vertikalen Segmenten (das heißt, die beiden Punkte, die ein Segment definieren, haben gleiche x - oder gleiche y -Koordinate). Die gebrochene Linie darf sich in jeglicher Art und Weise selbst schneiden oder überlappen. Mit anderen Worten, jeder Punkt der Ebene darf zu beliebig vielen Segmenten gehören.

Diese Aufgabe ist eine Output-Only Aufgabe mit partieller Bewertung. Du erhältst 10 Eingabedateien, welche die Orte der Punkte angeben. Zu jeder Eingabedatei sollst du eine Ausgabedatei einsenden. Diese Datei soll eine gebrochene Linie mit den angegebenen Eigenschaften beschreiben. Falls dies erfüllt ist, erhältst du Punkte, entsprechend der **Anzahl benutzter Segmente** (siehe Abschnitt *Bewertung*).

Du brauchst keinen Quellcode einzureichen.

Format der Eingabe

Die Eingabe hat folgende Form:

- Zeile 1: n
- Zeile $1 + i$ (für $1 \leq i \leq n$): $x[i] \ y[i]$

Format der Ausgabe

Die Ausgabe hat folgende Form:

- Zeile 1: k
- Zeile $1 + j$ (für $1 \leq j \leq k$): $sx[j] \ sy[j]$

Beachte, dass die zweite Zeile der Ausgabe mit $sx[1]$ and $sy[1]$ beginnt (das heißt, die Ausgabedatei soll $sx[0]$ und $sy[0]$ **nicht** enthalten). Jedes $sx[j]$ und jedes $sy[j]$ muss eine ganze Zahl sein.

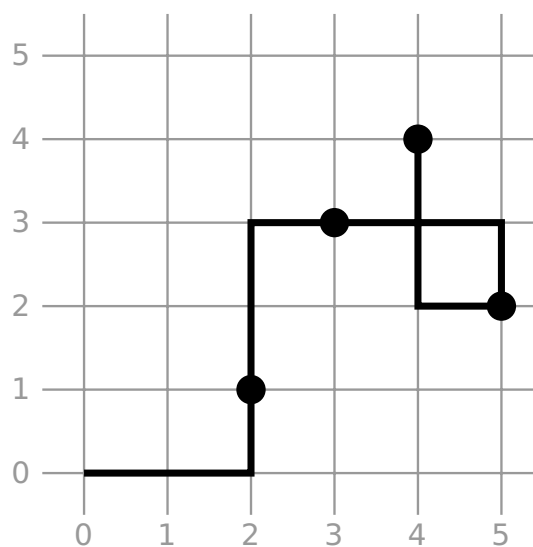
Beispiel

Eingabe:

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

Eine korrekte Ausgabe wäre:

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



Beachte, dass dieses Beispiel nicht unter den tatsächlichen Eingaben dieser Aufgabe ist.

Beschränkungen

- $1 \leq n \leq 100\,000$
- $1 \leq x[i], y[i] \leq 10^9$
- $x[i]$ und $y[i]$ sind ganze Zahlen.
- Keine zwei Punkte haben die gleiche x - oder die gleiche y -Koordinate, das heißt, $x[i_1] \neq x[i_2]$ **und** $y[i_1] \neq y[i_2]$ für $i_1 \neq i_2$.
- $-2 \cdot 10^9 \leq sx[j], sy[j] \leq 2 \cdot 10^9$
- Jede eingesendete Datei (sei es eine Ausgabedatei oder ein ZIP-Archiv) darf nicht größer als 15 MB sein.

Bewertung

Du kannst bis zu 10 Punkte für jede Eingabe erhalten.

Du erhältst *keine* Punkte, falls deine Ausgabe keine gebrochene Linie mit den angegebenen Eigenschaften beschreibt. Ansonsten wird deine Punktezahl anhand einer absteigenden Folge c_1, \dots, c_{10} ermittelt. Diese Folge ist für jede Eingabe unterschiedlich.

Falls deine Lösung eine gültige, aus k Segmenten bestehende, gebrochene Linie ist, dann erhältst du:

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

Die Folge c_1, \dots, c_{10} der einzelnen Eingaben ist gegeben durch:

| Eingabe | 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 |

Visualizer

Du findest im Anhang dieser Aufgabe ein Skript zur Visualisierung der Eingabe- und Ausgabedateien.

Benutze folgenden Befehl, um eine Eingabedatei zu visualisieren:

```
python vis.py [Eingabedatei]
```

Wenn du deine Lösung einer Eingabe visualisieren möchtest, kannst du folgenden Befehl aufrufen:

```
python vis.py [Eingabedatei] --solution [Ausgabedatei]
```

Aus technischen Gründen zeigt dir der Visualizer nur die **ersten** 1000 **Segmente** der Ausgabe an.

Beispiel:

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