



Kırık Çizgi

Azerbaycan halıları ile ünlüdür. Usta bir halı tasarımcısı olarak, bir **kırık çizgi** çizerek yeni bir tasarım yapmak istiyorsunuz. Bir kırık çizgi, iki boyutlu bir düzlemde t adet doğru parçası dizisidir. t adet doğru parçası dizisi de, p_0, \dots, p_t şeklinde $t + 1$ tane nokta ile şu şekilde tanımlanabilir. Her $0 \leq j \leq t - 1$ için, p_j and p_{j+1} noktalarını bağlayan bir doğru parçası vardır.

Yeni bir tasarım yapmak için, iki boyutlu bir düzlemde n adet **nokta**yı işaretlediniz. Nokta i 'nin ($1 \leq i \leq n$) koordinatları, $(x[i], y[i])$ 'dir. **Herhangi iki nokta aynı x veya y koordinatına sahip değildir.**

Şimdi, aşağıdaki koşulları sağlayan, $(sx[0], sy[0]), (sx[1], sy[1]), \dots, (sx[k], sy[k])$ şeklinde gösterilen ve bir kırık çizgi tanımlayan bir nokta dizisi bulmak istiyorsunuz:

- $(0, 0)$ 'dan başlar (yani, $sx[0] = 0$ ve $sy[0] = 0$),
- bütün noktaları içerir (doğru parçalarının başlangıç veya bitiş noktaları olmak zorunda değildir) ve
- sadece yatay veya dikey parçalardan oluşabilir (kırık çizgiyi tanımlayan iki ardışık nokta aynı x veya y koordinatına sahiptir).

Kırık çizgi kendisiyle herhangi bir şekilde çakışabilir veya örtüşebilir. Formal olarak, düzlemdeki her nokta, kırık çizgideki herhangi sayıda doğru parçasına ait olabilir.

Bu, kısmi puanlamalı bir sadece-çıkı problemi. Size noktaların yerlerini gösteren 10 adet girdi dosyası verilecektir. Her girdi dosyası için, istenen özellikleri sağlayan kırık çizgileri ifade eden bir çıktı dosyası göndermelisiniz. Geçerli bir kırık çizgi içeren her çıktı dosyası için puanınız kırık çizgideki **doğru parçası sayısına** göre hesaplanacaktır. (Aşağıdaki puanlamaya bakınız)

Bu problem için bir kod dosyası göndermenize gerek yoktur.

Girdi formatı

Her girdi dosyası aşağıdaki formatta olacaktır:

- satır 1: n
- satır $1 + i$ ($1 \leq i \leq n$ için): $x[i] \ y[i]$

Çıktı formatı

Her çıktı dosyası aşağıdaki formata sahip olmalıdır:

- satır 1: k
- satır $1 + j$ ($1 \leq j \leq k$ için): $sx[j] \ sy[j]$

İkinci satırın $sx[1]$ ve $sy[1]$ 'i içermesi gerektiğine dikkat ediniz (yani, çıktı $sx[0]$ ve $sy[0]$ **içermemelidir.**). Her $sx[j]$ ve $sy[j]$ birer tamsayı olmalıdır.

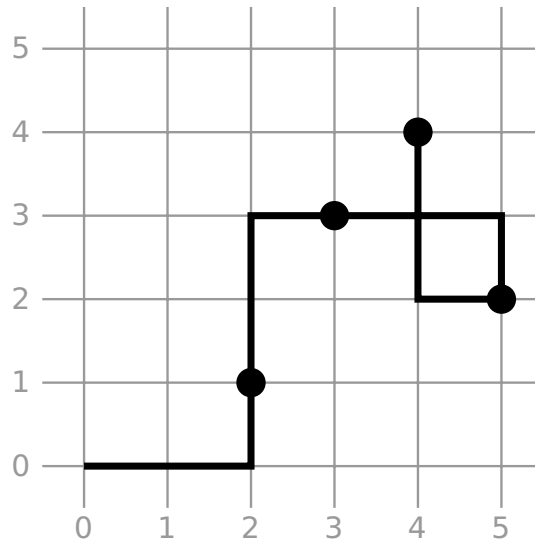
Örnek

Aşağıdaki girdi için:

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

olası geçerli bir çıktı aşağıdaki gibidir:

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



Bu örneğin, bu problemin test girdilerinden biri olmadığına dikkat ediniz.

Kısıtlar

- $1 \leq n \leq 100\,000$
- $1 \leq x[i], y[i] \leq 10^9$
- $x[i]$ ve $y[i]$ 'nin bütün deęerleri tamsayıdır.
- Herhangi iki nokta aynı x veya y koordinatına sahip deęildir, yani her $i_1 \neq i_2$ için, $x[i_1] \neq x[i_2]$ ve $y[i_1] \neq y[i_2]$ 'dir.
- $-2 \cdot 10^9 \leq sx[j], sy[j] \leq 2 \cdot 10^9$
- Gönderilen dosyaların boyutu (çıkıtı veya zip dosyası olarak) 15MB'yi geçemez.

Puanlama

Her test için, en fazla 10 puan alabilirsiniz. Bir test için, çıktınız istenen özelliklerde bir kırık çizgi içermiyorsa 0 puan alacaktır. Aksi halde, puan, teste göre deęişen c_1, \dots, c_{10} azalan dizisi kullanılarak hesaplanacaktır.

Çözümünüzün, k doğru parçası içeren bir geçerli kırık çizgi olduğunu varsayalım. Buna göre alacağınız puanlar aşağıdaki gibi olacaktır.

- i puan, eęer $k = c_i$ (for $1 \leq i \leq 10$) ise,
- $i + \frac{c_i - k}{c_i - c_{i+1}}$ puan, eęer $c_{i+1} < k < c_i$ (for $1 \leq i \leq 9$) ise,
- 0 puan, eęer $k > c_1$ ise,
- 10 puan, eęer $k < c_{10}$ ise.

Her test için c_1, \dots, c_{10} dizisi aşağıda verilmiştir.

| 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 |

Görselleştirici

Bu problemin eklentilerinde, girdi ve çıktı dosyalarını görselleştirmenizi sağlayacak bir script bulacaksınız.

Bir girdi dosyasını görselleştirmek için aşağıdaki komutu kullanınız:

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

Bir girdi için çözümünüzü de aşağıdaki komutu kullanarak görselleştirebilirsiniz. Teknik sınırlamalar yüzünden, sağlanan görselleştirici çıktının sadece ilk 1000 doğru parçasını göstermektedir.

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

Örnek:

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