



Διακεκομμένη γραμμή

Το Αζερμπαϊτζάν είναι διάσημο για τα χαλιά του. Είστε ένας διάσημος σχεδιαστής χαλιών και θέλετε να σχεδιάσετε μια νέα δημιουργία χρησιμοποιώντας μια **διακεκομμένη γραμμή**. Μια διακεκομμένη γραμμή είναι μια ακολουθία από t ευθύγραμμα τμήματα στο διδιάστατο επίπεδο, η οποία ορίζεται από μια ακολουθία $t + 1$ σημείων p_0, \dots, p_t , ως εξής. Για κάθε $0 \leq j \leq t - 1$ υπάρχει ένα ευθύγραμμο τμήμα που ενώνει τα σημεία p_j και p_{j+1} .

Για να μπορέσετε να σχεδιάσετε τη νέα δημιουργία, έχετε ήδη επιλέξει n **σημεία** στο διδιάστατο επίπεδο. Οι συντεταγμένες του σημείου i ($1 \leq i \leq n$) είναι $(x[i], y[i])$. **Δύο σημεία δεν μπορούν να έχουν την ίδια συντεταγμένη x ούτε την ίδια συντεταγμένη y .**

Θέλετε να βρείτε μια ακολουθία σημείων $(sx[0], sy[0]), (sx[1], sy[1]), \dots, (sx[k], sy[k])$ που να ορίζει μια διακεκομμένη γραμμή η οποία να:

- ξεκινά από το $(0, 0)$, δηλαδή $sx[0] = 0$ και $sy[0] = 0$,
- περιλαμβάνει όλα τα σημεία (όχι απαραίτητως ως άκρα των ευθυγράμμων τμημάτων της), και
- αποτελείται αποκλειστικά από οριζόντια και κατακόρυφα τμήματα (δύο διαδοχικά σημεία που ορίζουν τη διακεκομμένη γραμμή πρέπει να έχουν ίδια τη συντεταγμένη x ή την y).

Η διακεκομμένη γραμμή επιτρέπεται να τέμνει ή να επικαλύπτει τον εαυτό της με οποιονδήποτε τρόπο. Πιο τυπικά, κάθε σημείο του επιπέδου μπορεί να ανήκει σε οσαδήποτε τμήματα της διακεκομμένης γραμμής.

Αυτό το πρόβλημα είναι output-only με μερική βαθμολογία (partial scoring). Θα σας δοθούν 10 αρχεία εισόδου (input files) που θα καθορίζουν τις θέσεις των σημείων. Για κάθε αρχείο εισόδου πρέπει να υποβάλετε ένα αρχείο εξόδου (output file), το οποίο να περιγράφει τη διακεκομμένη γραμμή με τις απαιτούμενες ιδιότητες. Για κάθε αρχείο εξόδου που περιγράφει μια έγκυρη διακεκομμένη γραμμή, η βαθμολογία σας θα εξαρτάται από το **πλήθος των τμημάτων** της διακεκομμένης γραμμής (δείτε τη Βαθμολογία παρακάτω).

Δεν πρέπει να υποβάλετε κώδικα για αυτό το πρόβλημα.

Δεδομένα εισόδου

Κάθε αρχείο εισόδου θα έχει την εξής μορφή:

- γραμμή 1: n
- γραμμή $1 + i$ (για $1 \leq i \leq n$): $x[i] \ y[i]$

Δεδομένα εξόδου

Κάθε αρχείο εξόδου πρέπει να έχει την εξής μορφή:

- γραμμή 1: k
- γραμμή $1 + j$ (για $1 \leq j \leq k$): $sx[j] \ sy[j]$

Να σημειωθεί ότι η δεύτερη γραμμή πρέπει να περιέχει τα $sx[1]$ και $sy[1]$ (δηλαδή, το αρχείο **δεν πρέπει** να περιέχει τα $sx[0]$ και $sy[0]$). Οι συντεταγμένες $sx[j]$ και $sy[j]$ πρέπει να είναι ακέραιοι αριθμοί.

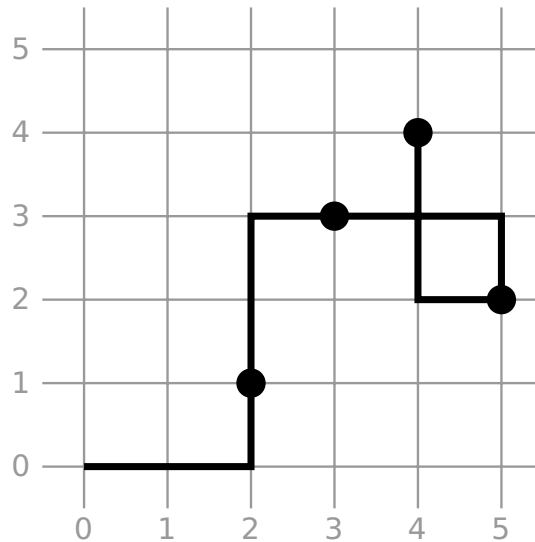
Παράδειγμα

Για το αρχείο εισόδου:

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

ένα πιθανό έγκυρο αρχείο εξόδου είναι:

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



Να σημειωθεί ότι αυτό το παράδειγμα δεν συμπεριλαμβάνεται ανάμεσα στα πραγματικά αρχεία εισόδου του συγκεκριμένου προβλήματος.

Περιορισμοί

- $1 \leq n \leq 100\,000$
- $1 \leq x[i], y[i] \leq 10^9$
- Όλες οι τιμές των $x[i]$ και $y[i]$ είναι ακέραιες.
- Δύο σημεία δεν μπορούν να έχουν την ίδια συντεταγμένη x ή την ίδια συντεταγμένη y , δηλαδή, $x[i_1] \neq x[i_2]$ και $y[i_1] \neq y[i_2]$ για $i_1 \neq i_2$.
- $-2 \cdot 10^9 \leq sx[j], sy[j] \leq 2 \cdot 10^9$
- Το μέγεθος του υποβληθέντος αρχείου (είτε κειμένου είτε συμπιεσμένου) δεν μπορεί να υπερβαίνει τα 15MB.

Βαθμολογία

Κάθε test case βαθμολογείται το πολύ με 10 βαθμούς. Το αρχείο εξόδου για κάποιο test case θα πάρει 0 βαθμούς αν δεν ορίζει μια διακεκομμένη γραμμή με τις απαιτούμενες ιδιότητες. Σε διαφορετική περίπτωση, η βαθμολογία θα καθορίζεται χρησιμοποιώντας μια μειούμενη ακολουθία c_1, \dots, c_{10} , η οποία διαφέρει για κάθε test case.

Έστω ότι η λύση σας περιλαμβάνει μια έγκυρη διακεκομμένη γραμμή, η οποία αποτελείται από k τμήματα. Τότε θα βαθμολογηθείτε με:

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

Η ακολουθία c_1, \dots, c_{10} για κάθε test case δίνεται παρακάτω.

Test case	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

Πρόγραμμα απεικόνισης (visualizer)

Στο συμπεσμένο πακέτο αυτού του προβλήματος υπάρχει ένα πρόγραμμα (script) που σας επιτρέπει να απεικονίσετε τα αρχεία εισόδου και εξόδου.

Για να απεικονίσετε ένα αρχείο εισόδου, χρησιμοποιήστε την ακόλουθη εντολή:

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

Μπορείτε επίσης να απεικονίσετε τη λύση σας για κάποιο αρχείο εισόδου, χρησιμοποιώντας την ακόλουθη εντολή. Λόγω τεχνικών περιορισμών, το πρόγραμμα απεικόνισης θα εμφανίζει μόνο **τα πρώτα 1000 τμήματα** του αρχείου εξόδου.

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

Παράδειγμα:

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