

# Forbici e Scotch (scissors)

Giorno	2
Lingua	Italiano
Limite di tempo	1 secondo
Limite di memoria	1024 megabyte

Ti viene dato un pezzo di carta della forma di un singolo poligono  $S$ . Il tuo compito è di convertirlo in un poligono semplice  $T$  che ha la stessa area di  $S$ .

Puoi utilizzare due strumenti: forbici e scotch. Le forbici possono essere utilizzate per tagliare qualsiasi poligono in poligoni più piccoli. Lo scotch serve a unire più poligoni in poligoni più grandi. Puoi utilizzare ogni strumento molteplici volte (2000 al massimo), in ogni ordine.

I poligoni dati in input hanno coordinate intere, ma è consentito a produrre forme con **coordinate non intere** come output.

Segue una descrizione formale del problema.

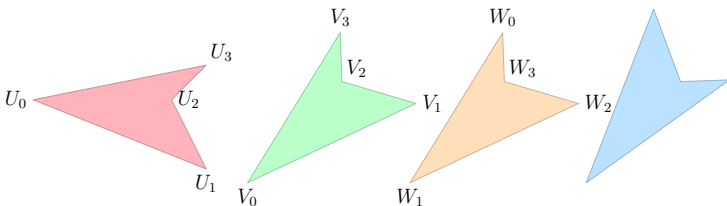
Una **forma**  $Q = (Q_0, \dots, Q_{n-1})$  è una sequenza di tre o più punti nel piano tali che:

- La polilinea chiusa  $Q_0Q_1Q_2 \dots Q_{n-1}Q_0$  non tocca o interseca se stessa, e quindi forma il contorno di un poligono semplice.
- La polilinea gira attorno al contorno del poligono in senso antiorario.

Il poligono il cui contorno è la forma  $Q$  sarà identificato con  $P(Q)$ .

Due forme sono considerate **equivalenti** se una può essere traslata e/o ruotata per diventare identica all'altra.

Nota che specchiare la forma non è consentito. Inoltre nota che l'ordine dei punti è importante: la forma  $(Q_1, \dots, Q_{n-1}, Q_0)$  non è equivalente alla forma  $(Q_0, \dots, Q_{n-1})$ .



Nella figura a sinistra: Le forme  $U$  e  $V$  sono equivalenti. La forma  $W$  non è equivalente a loro perché i punti di  $W$  sono dati in un ordine differente. Anche ignorando l'ordine dei punti, la quarta forma non è equivalente con le precedenti visto che specchiare una forma non è consentito.

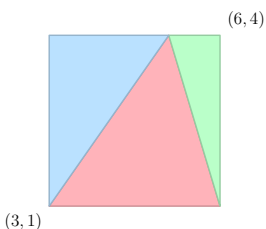
Sia in input che in output, una forma con  $n$  punti è rappresentata come una singola linea che contiene  $2n + 1$  interi separati da spazi. Il primo di questi numeri è  $n$ . I rimanenti sono le coordinate dei punti:  $Q_{0,x}, Q_{0,y}, Q_{1,x}, \dots$

Le forme hanno **numeri identificativi** (IDs). La forma data in input  $S$  ha ID 0, le forme che produrrà nella tua soluzione avranno indici crescenti 1, 2, 3, ..., nell'ordine in cui sono prodotti.

Le forme  $B_1, \dots, B_k$  formano una **suddivisione** della forma  $A$  se:

- The union of all  $P(B_i)$  is exactly  $P(A)$ .
- L'unione di tutte le  $P(B_i)$  è esattamente  $P(A)$ .
- Per ogni  $i \neq j$ , l'area di intersezione tra  $P(B_i)$  e  $P(B_j)$  è zero.

L'operazione **forbici** distrugge una forma esistente  $A$  e produce una o più forme  $B_1, \dots, B_k$  che siano una suddivisione di  $A$ .



Nella figura a sinistra: La forma  $A$  (quadrato) è suddivisa nelle forme  $B_1, B_2, B_3$  (tre triangoli). Uno dei modi validi per descrivere una forma  $B_i$  è "3 3 1 6 1 5.1 4".



L'operazione **scotch** distrugge una o più forme esistenti  $A_1, \dots, A_k$  e produce una nuova forma  $B$ . Per eseguire questa operazione, devi prima specificare le forme  $C_1, \dots, C_k$  e solo successivamente la forma finale  $B$ . Queste forme devono soddisfare le seguenti condizioni:

- Per ogni  $i$ , la forma  $C_i$  è equivalente alla forma  $A_i$ .
- Le forme  $C_1, \dots, C_k$  formano una suddivisione della forma  $B$ .

In modo informale, puoi scegliere una forma  $B$  e mostrare come muovere ogniuna delle forme esistenti  $A_i$  nella loro corretta posizione  $C_i$  contenuta in  $B$ . Nota che solo la forma  $B$  ottiene un nuovo ID, le forme  $C_i$  non lo ottengono.

## Dati di input

La prima linea contiene la forma di partenza  $S$ .

La seconda linea contiene la forma finale  $T$ .

Ogni forma ha tra 3 e 10 punti, estremi inclusi. Entrambe le forme sono date nel formato specificato sopra.

Tutte le coordinate in input sono interi compresi tra  $-10^6$  e  $10^6$ , inclusi.

In ogni forma, non esistono tre punti qualsiasi che formano un angolo inferiore di 3 gradi. (Questo quindi include punti non consecutivi, ed implica che non esistono tre punti collineari).

I poligoni  $P(S)$  e  $P(T)$  hanno la stessa area.

## Dati di output

Ogni qualvolta usi l'operazione **forbici** scrivi in output un blocco di righe nella seguente forma:

```
scissors
id(A) k
B_1
B_2
...
B_k
```

dove  $id(A)$  è l'ID della forma che vuoi distruggere,  $k$  è il numero di nuove forme da produrre, e  $B_1, \dots, B_k$  sono queste forme.

Ogni qualvolta usi l'operazione **scotch** scrivi in output un blocco di righe nella seguente forma:

```
tape
k id(A_1) ... id(A_k)
C_1
C_2
...
C_k
B
```

dove  $k$  è il numero di forme che vuoi attaccare insieme,  $id(A_1), \dots, id(A_k)$  sono i loro ID,  $C_1, \dots, C_k$  sono forme equivalenti che mostrano la loro posizione in  $B$ , e  $B$  è la forma finale ottenuta attaccandole assieme.

È consigliabile scrivere in output le coordinate dei punti con almeno 10 cifre decimali.

L'output deve soddisfare i seguenti requisiti:

- Tutte le coordinate di punti nell'output devono essere comprese fra  $-10^7$  e  $10^7$ , inclusi.
- Ogni forma dell'output deve avere al più 100 punti.
- In ogni operazione il numero  $k$  di forme deve essere compreso fra 1 e 100, inclusi.
- Il numero totale di operazioni non può eccedere 2000.
- Il numero totale di punti in tutte le forme nell'output non deve eccedere 20000.
- Alla fine deve esserci esattamente una forma (che non è stata distrutta), e tale forma deve essere equivalente a  $T$ .



- Tutte le operazioni devono risultare valide al correttore. Soluzioni con piccoli errori di arrotondamento saranno accettate (internamente tutti i confronti controllano per un errore assoluto o relativo fino a  $10^{-3}$  quando viene verificata ogni condizione).

## Handouts

- Istruzioni su come stampare i numeri a virgola mobile sono disponibili negli appunti del tuo linguaggio di programmazione.
- Puoi scaricare il binario `scissors-checker`, renderlo eseguibile (`chmod a+x scissors-checker`) e usarlo in locale per controllare la correttezza del tuo output (`./scissors-checker input your_output`).

## Assegnazione del punteggio

Una forma è chiamata un **bel rettangolo** se ha la forma  $((0,0), (x,0), (x,y), (0,y))$  con  $x$  e  $y$  interi positivi.

Una forma è chiamata un **bel quadrato** se aggiuntivamente  $x = y$ .

Una forma  $A$  è chiamata **strettamente convessa** se tutti gli angoli interni del poligono  $P(A)$  sono meno ampi di 180 gradi.

Subtask 1 (5 punti):  $S$  e  $T$  sono bei rettangoli. Tutte le coordinate di tutti i punti sono comprese fra 0 e 10, inclusi.

Subtask 2 (13 punti):  $S$  è un bel rettangolo con  $x > y$ , e  $T$  è un bel quadrato.

Subtask 3 (12 punti):  $S$  e  $T$  sono bei rettangoli.

Subtask 4 (14 punti):  $S$  è un triangolo e  $T$  è un bel rettangolo.

Subtask 5 (10 punti):  $S$  e  $T$  sono triangoli.

Subtask 6 (16 punti):  $S$  è un poligono strettamente convesso e  $T$  è un bel rettangolo.

Subtask 7 (11 punti):  $T$  è un bel rettangolo.

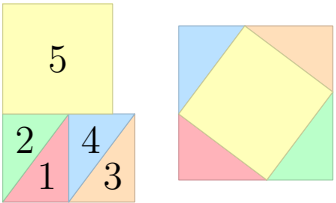
Subtask 8 (19 punti): nessuna ulteriore restrizione.

# Esempi di input ed output

standard input	standard output
6 0 0 6 0 6 4 5 4 5 9 0 9 4 0 0 7 0 7 7 0 7	scissors 0 5 3 0 0 3 0 3 4 3 3 4 0 4 0 0 3 3 0 6 0 6 4 3 6 4 3 4 3 0 4 0 4 5 4 5 9 0 9 tape 5 1 2 5 3 4 3 0 3 0 0 4 0 3 4 0 7 0 7 4 4 0 3 4 0 7 4 3 7 3 7 4 7 7 3 7 3 3 7 0 7 0 3 4 0 0 7 0 7 7 0 7
4 0 0 3 0 3 3 0 3 4 7 -1 10 -1 11 2 8 2	scissors 0 2 3 0 0 1 3 0 3 4 1 3 0 0 3 0 3 3 tape 2 1 2 3 110 -1 111 2 110 2 4 108 2 107 -1 110 -1 110 2 4 107 -1 110 -1 111 2 108 2

standard input	standard output
4 0 0 9 0 9 1 0 1 4 0 0 3 0 3 3 0 3	scissors 0 2 4 1.4700000000 0 9 0 9 1 1.470000000 1 4 0 0 1.470000000 0 1.470000000 1 0 1 scissors 1 2 4 1.470000000 0 6 0 6 1 1.470000000 1 4 9 0 9 1 6 1 6 0 tape 2 4 3 4 3 2 3 1 6 1 6 2 4 6 1 1.470000000 1 1.470000000 0 6 0 6 1.470000000 0 6 0 6 2 3 2 3 1 1.47 scissors 5 4 4 1.470000000 0 3 0 3 1 1.470000000 1 4 3 0 4 0 4 2 3 2 4 4 2 4 0 5 0 5 2 4 5 0 6 0 6 2 5 2 tape 5 2 6 7 8 9 4 0 0 1.470000000 0 1.470000000 1 0 1 4 1.470000000 0 3 0 3 1 1.470000000 1 4 0 2 0 1 2 1 2 2 4 0 2 2 2 2 3 0 3 4 3 3 2 3 2 1 3 1 4 0 0 3 0 3 3 0 3

# Note



La figura sulla sinistra mostra l'output del primo esempio. Sulla sinistra c'è la figura originale alterata usando le forbici, sulla destra ci sono i corrispondenti  $C_i$  quando attacchiamo questi pezzi insieme con lo scotch.

Nel secondo output di esempio, nota che è sufficiente che la figura finale sia equivalente a quella voluta, non è necessario che siano uguali.

La figura sotto mostra tre stadi del terzo output di esempio. Prima tagliamo il rettangolo di input in due rettangoli più piccoli, dopo di che tagliamo i più grandi di questi due rettangoli in ulteriori due rettangoli. Lo stato dopo aver effettuato questi tagli è mostrato in alto a sinistra nella figura. Continuando attacchiamo i due nuovi rettangoli assieme con il nastro adesivo per formare un poligono di sei lati, e quindi tagliamo quel poligono in tre rettangoli 2 per 1 ed in un rettangolo più piccolo. Questo è mostrato in basso a sinistra nella figura.

Infine prendiamo il rettangolo che ancora abbiamo dal primo passaggio ed i quattro nuovi rettangoli, e li assembliamo insieme nel quadrato 3 per 3 desiderato.

