

Specchio (specchio)

Descrizione del problema

Un *albero* è formato da una *radice* e da un insieme finito e ordinato (eventualmente vuoto) di *figli*, che sono anch'essi alberi. Ad esempio, la Figura 1 rappresenta un albero: per convenzione, la radice viene disegnata in cima (cioè al contrario rispetto agli alberi veri). Come ogni albero, lo potete descrivere dicendo quanti figli ha la radice (in questo caso, quattro) e poi descrivendo uno a uno, nell'ordine da sinistra a destra, i quattro sottoalberi. In questo modo, ad esempio, l'albero in Figura 1 sarebbe identificato dalla seguente sequenza:

4 2 0 3 0 0 1 0 0 0 0

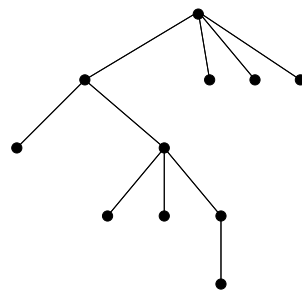


Figura 1: Un albero.

Infatti, l'albero ha quattro figli sotto la radice, per cui il primo numero è 4. A questo 4 segue poi subito la sequenza 2 0 3 0 0 1 0, che è la descrizione del primo figlio, mentre gli ultimi tre figli sono ciascuno descritti da 0 (dato che non hanno figli).

Supponete ora di guardare l'albero riflesso in uno specchio. L'ordine dei figli di ciascun nodo risulta invertito e l'albero di Figura 1 appare come in Figura 2.

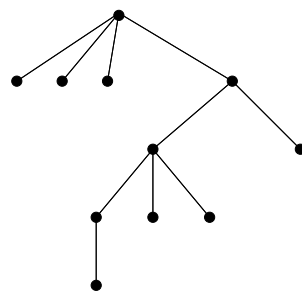


Figura 2: L'albero di Figura 1 allo specchio.

Questo nuovo albero è descritto dalla sequenza

4 0 0 0 2 3 1 0 0 0 0

Dati di input

Il vostro programma legge da `stdin` una sola riga contenente una sequenza di interi non negativi, separati da uno spazio. La sequenza descrive correttamente un albero: quindi il primo numero è il numero di figli della radice, ed è seguito dalle descrizioni dei figli, uno dopo l'altro, da sinistra a destra.

Dati di output

Il vostro programma deve restituire su `stdout` una sola riga, costituita da una sequenza di interi non negativi separati da uno o più spazi. Questa sequenza è la descrizione dell'albero dato in input, visto allo specchio.

Esempio di input/output

input (da <code>stdin</code>)	output (su <code>stdout</code>)
4 2 0 3 0 0 1 0 0 0 0	4 0 0 0 2 3 1 0 0 0 0

Assunzioni

- dove N è il numero di nodi dell'albero, vale sempre che $1 \leq N \leq 1\,000\,000$;
- tempo limite: un secondo.

Subtask

- **Subtask 1 [0 punti]:** l'esempio del testo.
- **Subtask 2 [20 punti]:** ogni nodo ha massimo due figli.
- **Subtask 3 [20 punti]:** $N \leq 100$.
- **Subtask 4 [20 punti]:** $N \leq 10\,000$.
- **Subtask 5 [20 punti]:** $N \leq 100\,000$.
- **Subtask 6 [20 punti]:** nessuna limitazione specifica.