

# Teste di serie (**serie**)

## Difficoltà $D = 2$

### Descrizione del problema

Un torneo è composto da  $K$  gironi, con  $N$  squadre partecipanti in ciascun girone (per un totale di  $K \times N$  squadre nel torneo). Dopo le eliminatorie, passa soltanto la prima classificata di ogni girone.

A ogni squadra è associato un "coefficiente di bravura", ovvero un intero positivo che è tanto maggiore quanto più la squadra è forte. Per rendere più vivace il torneo, gli organizzatori vogliono far gareggiare le squadre più forti tra loro soltanto dopo le eliminatorie: in altre parole, le  $K$  squadre con i coefficienti di bravura più alti devono giocare in gironi distinti.

Aiutate gli organizzatori a verificare che la composizione del torneo rispetti il loro volere: prese le  $K$  squadre con il più alto coefficiente di bravura, ciascun girone deve contenere esattamente una di esse (da notare che due o più squadre possono avere lo stesso coefficiente).

### Dati di input

Il file `input.txt` è composto da  $K+1$  righe.

La prima riga contiene due interi positivi separati da uno spazio: il numero  $K$  di gironi e il numero  $N$  di squadre per girone.

Le successive  $K$  righe contengono i coefficienti di bravura delle squadre: la  $j$ -esima di tale righe contiene  $N$  interi positivi separati da uno spazio che sono i coefficienti di bravura delle  $N$  squadre nel  $j$ -esimo girone, per  $1 \leq j \leq K$ .

### Dati di output

Il file `output.txt` è composto di una riga contenente un solo intero: 1 se il torneo rispetta i vincoli imposti dagli organizzatori, 0 altrimenti.

### Assunzioni

- $1 < N \leq 100$ .
- $1 < K \leq 100$ .

### Esempi di input/output

File <code>input.txt</code>	File <code>output.txt</code>
3 4 3 5 7 9 3 6 78 90 43 78 71 32	0

File input.txt	File output.txt
3 4 2 2 2 1 2 1 3 1 2 4 2 1	1

## Note

- Un programma che restituisce sempre lo stesso valore, indipendentemente dai dati in `input.txt`, non totalizza alcun punteggio rilevante.
- Attenzione se usate la piattaforma di sviluppo software basata sul compilatore Turbo Pascal e sul sistema operativo Windows: visto che i vostri programmi potrebbero essere valutati in una piattaforma diversa dalla vostra, avete garanzia di uniformità di comportamenti soltanto se utilizzate *sempre* il tipo `LongInt` al posto del tipo `Integer` (quest'ultimo permette di rappresentare gli interi nell'intervallo [-32768...32767] mentre `LongInt` ne permette la rappresentazione in [-2147483648...2147483647]).