

Traccia prova d'esame

Il postino del Villaggio Calbarico è un amante della Settimana Enigmatica, e ci chiede una mano per un gioco che non riesce a risolvere. Il gioco è costituito da n mappe. Ogni mappa ha un certo numero di castelli (che può differire da quello delle altre mappe) e tra due castelli può o non può esserci un tunnel sotterraneo che li collega; un tunnel serve ad andare da un castello all'altro (e viceversa). Inoltre, il gioco propone al lettore le seguenti informazioni, valide per ogni mappa:

- Il *fuoco* tra due castelli U e V è definito come il minimo numero di tunnel da attraversare per andare da U a V .
- L'*alto fuoco* è il valore di *fuoco* più grande.

L'obiettivo del gioco è controllare se tutte le n mappe hanno lo stesso valore di *alto fuoco*. Per aiutare il postino, bisogna scrivere un programma che, prese in input le n mappe, controlli tutte le n mappe abbiano lo stesso valore di *alto fuoco*: in caso positivo, il programma deve restituire questo valore; altrimenti, il programma restituisce **-1**.

Struttura dell'input

L'input consiste in una prima riga nel formato n dove n è il numero di mappe date in input. La riga successiva rappresenta l'inizio della prima mappa ed è nel formato $k\ q$ dove k è il numero di castelli e q il numero di tunnel per la prima mappa. I castelli sono identificati univocamente dai numeri $1..k$. Le successive q righe rappresentano i tunnel nel formato $\langle u = v \rangle$ dove u e v sono due castelli. Le successive $n-1$ mappe sono rappresentate nello stesso modo e seguono nell'input. Si può supporre che l'input sia sempre corretto e che ci siano almeno 2 mappe.

Struttura dell'output

L'output consiste nel valore di *alto fuoco* se quest'ultimo è uguale per tutte le mappe in input; altrimenti **-1**.

Esempi input – output

2 4 3 <1 = 2> <2 = 3> <3 = 4> 5 4 <2 = 1> <1 = 4> <5 = 1> <3 = 4>	3
3 2 1 <2 = 1> 3 2 <2 = 3> <1 = 2> 4 4 <1 = 2> <2 = 3> <1 = 3> <4 = 1>	-1

Regole e istruzioni

- Si può scegliere se usare C++ o Java; in entrambi i casi, si presuppone che lo studente sappia compilare il codice sorgente e avviare l'eseguibile ottenuto tramite terminale/prompt dei comandi.
- Si può assumere che l'input sia sempre corretto.
- Il vostro programma deve **leggere da stdin** e **scrivere su stdout**. La lettura da file o l'hardcoding di un input nel vostro programma **non rappresenta una soluzione corretta**.
- Per effettuare le (vostre) varie prove, potete creare dei file testuali contenenti input di prova e
 - o Scrivere o copiare riga per riga il vostro input al programma,
 - o Reindirizzare il contenuto del file di input al vostro programma (consigliato).
- Il vostro programma verrà valutato su vari input utilizzando il secondo metodo.

Come posso reindirizzare su stdin?

- Supponendo di utilizzare C++ e di essere su Linux/OS X

```
cat input.txt | ./programma
```

dove input.txt è un file testuale contenente un input e programma è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma devono essere nella stessa cartella)

- Supponendo di utilizzare C++ e di essere su Windows

```
type input.txt | programma.exe
```

dove input.txt è un file contenente un input e programma.exe è l'eseguibile ottenuto dalla compilazione del vostro codice sorgente (input.txt e programma.exe devono essere nella stessa cartella).