

Traccia prova d'esame

A causa di una grave siccità, nello stato del Burmini si sta cercando di ottimizzare la distribuzione dell'acqua. Si ha a disposizione una mappa in cui sono segnate le varie città, e per ogni coppia di città c'è una strada che le collega. In ogni città si può installare un serbatoio d'acqua, il quale ha la capacità di rifornire la città in cui è installato e le città collegate direttamente. Si vuole utilizzare un numero di serbatoi $\leq k$ da installare, in modo che tutte le città siano rifornite.

Restituire, se esiste, la disposizione dei serbatoi nelle città in modo da soddisfare i suddetti vincoli.

Struttura dell'input

L'input consiste in un certo numero di righe. La prima riga è nel formato $n \ k$ dove n è il numero di città presenti nello stato, mentre k è il numero massimo di serbatoi da utilizzare. Le successive n righe rappresentano le città, in particolare ogni riga contiene il nome di una città. Infine, ogni riga successiva è nel formato $a \ <-> \ b$ e indica che le città a e b sono collegate da una strada. Si può assumere che il nome di ciascuna città non contenga nessuno dei simboli $<->$ spazi o altri caratteri speciali.

Struttura dell'output

L'output della soluzione deve essere formato da due righe: la prima riga indica il numero minimo di serbatoi utilizzati, mentre la seconda riga è nel formato $c_1, \ c_2, \ \dots, \ c_k$ ed indica le città in cui sono stati installati i serbatoi. L'ordine in cui stampare le città segue l'ordine in cui si sono lette le stesse. Se non esiste alcuna disposizione dei serbatoi in modo da soddisfare i vincoli, stampare -1 .

Esempio input – output

```
4 3
Xinghua
Modiin
Apas
Ukiah
Xinghua <-> Modiin
Modiin <-> Apas
```

```
2
Modiin, Ukiah
```

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 contenete 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).