UNIVERSITÀ DELLA CALABRIA

## Traccia prova d'esame

I ladri della gilda Ravenholdt sono stati assoldati dall'Alto Re di Ironforge per delle operazioni di spionaggio a discapito dell'Alto Concilio di Dalaran. Durante un lungo periodo di tempo, i ladri hanno rubato delle conversazioni tra individui, che adesso devono essere analizzate per carpire possibili segreti. Per ogni conversazione, sappiamo solo che è avvenuta tra due individui $X$ e $Y$. Dati in input una lista di conversazioni $C$, dove ogni conversazione è una coppia $(X,Y)$, un intero positivo $q$ e $t$ insiemi disgiunti $P_1, \ldots, P_t$ (dove ogni elemento di $P_i$ è una conversazione appartenente a $C$, con $1 \leq i \leq t$, e l'unione di tutti gli insiemi è uguale a $C$), bisogna restituire in output **SI** se esiste un sottoinsieme $F$ di $C$, che contenga almeno $q$ conversazioni e che soddisfi entrambe le seguenti condizioni:

1. non possono esserci in $F$ due conversazioni che hanno un individuo in comune, e
2. $F$ deve contenere al massimo una sola conversazione da ogni insieme $P_i$ ($1 \leq i \leq t$).

Se non esiste tale sottoinsieme $F$ così formato, si deve restituire **NO**.

### Struttura dell'input

La prima riga contiene i valori $n$ $q$ $t$, dove $n$ indica il numero di conversazioni contenute nell'insieme $C$, $q$ è un intero positivo e $t$ è il numero dei sottoinsiemi disgiunti. Le seguenti $n$ righe (dove la prima riga ha indice $1$ e l'ultima $n$) sono nel formato $X$ @ $Y$, ognuna rappresentante una conversazione tra gli individui $X$ e $Y$. Le successive $t$ righe rappresentano gli insiemi $P_1, \ldots, P_t$ (dove la prima riga ha indice $1$ e l'ultima $t$). Ogni insieme è nel formato $c_1$ $c_2$ … $c_k$, indicante che le conversazioni $c_1, c_2, \ldots, c_k$ appartengono all'insieme; ogni $c_j$ è un numero rappresentante la $j$-esima conversazione in $C$. Nota bene: gli insiemi possono essere di grandezza differente. L'output è da considerarsi sempre corretto.

### Struttura dell'output

L'output consiste nella stringa **SI** se esiste un sottoinsieme $F$ di $C$ che contenga almeno $q$ conversazioni e che soddisfi entrambe le condizioni precedenti, **NO** altrimenti.

### Esempi input – output

| | |
|---|---|
| 8 3 4<br>a @ b<br>b @ d<br>a @ d<br>d @ c<br>b @ c<br>e @ f<br>e @ d<br>f @ d<br>2 3 4<br>1 7<br>5<br>6 8 | SI |
| 4 4 2<br>a @ b<br>b @ c<br>b @ d<br>b @ e<br>1 3<br>4 2 | NO |

UNIVERSITÀ DELLA CALABRIA

## 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
  - Scrivere o copiare riga per riga il vostro input al programma,
  - 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).