

## Traccia prova d'esame

Nelle fredde terre del Northrend, un gruppo di sciamani ha ripreso in mano le redini di una cittadina perduta, con l'intenzione di ricostruirla. La cittadina è formata da  $P$  piazze; alcune di queste piazze sono collegate da delle strade, e ogni strada si può percorrere in entrambi i versi. Gli sciamani desiderano trovare un sottoinsieme di piazze  $H$ , e ad ogni piazza in questo sottoinsieme assegnare uno di due totem magici ( $T_1$  e  $T_2$ ), in modo tale che due piazze collegate non abbiano lo stesso totem. Dati in input  $P$ , l'insieme  $S$  contenente le strade e un intero positivo  $z$ , si deve dire se esiste un sottoinsieme  $H$  di  $P$  tale che:

- $H$  contenga almeno  $z$  piazze,
- Assegnando o il totem  $T_1$  o il totem  $T_2$  ad ogni piazza in  $H$ , non esistano due piazze collegate fra loro (da una strada in  $S$ ) aventi lo stesso totem.

Si deve restituire **SI** in output se esiste un sottoinsieme  $H$  di  $P$  fatto in questo modo; **NO** altrimenti.

## Struttura dell'input

L'input consiste in una prima riga nel formato  $P$   $z$   $S$  dove  $P$  è il numero di piazze (le piazze sono indicate con i numeri da 1 a  $P$ ),  $z$  è il numero minimo di piazze che il sottoinsieme  $H$  deve contenere ed  $S$  è il numero di strade. Le successive  $S$  righe sono nel formato  $\{x\ y\}$ , dove  $x$  e  $y$  sono delle piazze, indicante che esiste una strada tra la piazza  $x$  e la  $y$  (e viceversa).

## Struttura dell'output

L'output consiste nella stringa **SI** se è possibile formare un sottoinsieme di piazze  $H$  che rispetti i vincoli sopra elencati, altrimenti **NO**.

## Esempi input – output

6 4 8 {1 3} {1 2} {3 2} {5 1} {4 2} {4 5} {4 6} {6 5}	SI
---	----

L'esempio precedente ha come soluzione (una delle possibili) il sottoinsieme formato dalle piazze 1, 2, 4, 5.

7 3 8 {6 5} {3 4} {4 7} {7 3} {7 6} {5 7} {1 7} {7 2}	SI
---	----

L'esempio precedente ha come soluzione (una delle possibili) il sottoinsieme formato dalle piazze 1, 2, 7.

*(segue un ulteriore esempio nel retro del foglio)*

```
4 3 6
{2 1}
{3 2}
{1 4}
{4 3}
{2 4}
{3 1}
```

NO

### 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).