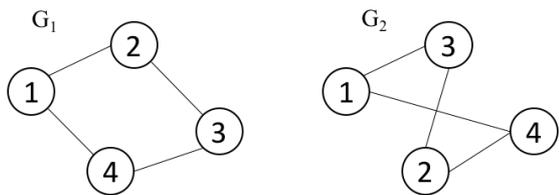
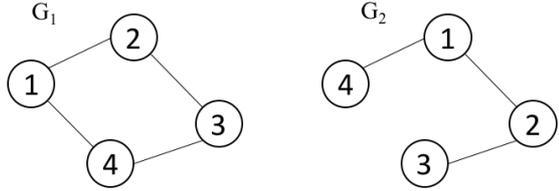


Traccia prova d'esame

Nella sessione d'esami di settembre, nel Corso di Laurea in Informatica alle Università Riunite di Quel'Thalas, per l'esame di Algoritmi e Strutture Dati è stato proposto il seguente problema: presi due grafi G_1, G_2 in input, si deve verificare se i due grafi sono isomorfi. Due grafi sono isomorfi se procedendo ad una ri-numerazione dei suoi nodi il secondo grafo diventa identico al primo. Complice l'interessante problema, si è scelto di provare a risolverlo. Si crei quindi un programma che presi in input due grafi G_1 e G_2 , restituisca SI se i grafi sono isomorfi, altrimenti NO. Entrambi i grafi in input sono non orientati e senza alcun peso tra gli archi.

Due esempi sono i seguenti:

<p>G1 e G2 sono isomorfi</p>	
<p>G1 e G2 NON sono isomorfi</p>	

Struttura dell'input

La struttura dell'input è come segue. Una prima riga contiene i valori $N_1 N_2 E_1 E_2$, dove N_1 indica il numero di nodi del primo grafo e N_2 il numero di nodi del secondo grafo (i nodi sono etichettati con i numeri interi da 1 a N), E_1 indica il numero di archi del grafo G_1 ed E_2 indica il numero di archi del grafo G_2 . Le successive E_1 righe sono nel formato $X <> Y$, indicando la connessione tra il nodo X e Y nel grafo G_1 . Le successive E_2 righe seguono lo stesso formato, indicando la connessione tra due nodi nel grafo G_2 . L'input si può considerare sempre corretto.

Struttura dell'output

L'output consisterà nella stringa SI se i due grafi in input sono isomorfi, altrimenti nella stringa NO.

Esempio input – output

<pre>4 4 4 4 1 <> 2 2 <> 3 3 <> 4 4 <> 1 1 <> 4 1 <> 3 3 <> 2 2 <> 4</pre>	<p>SI</p>
--	-----------

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).