

Pugilato

Durante una lezione di Algoritmi e Strutture Dati, gli studenti di una certa Università hanno deciso di organizzare un torneo di pugilato (possono iscriversi sia membri del personale, che professori e gli stessi studenti). Per stabilire quali aspiranti pugili sono rivali tra loro, gli studenti hanno scritto un programma in C decisamente disonesto. In particolare, nella prima fase di qualificazione, un pugile può avere un numero arbitrario di rivali: qualcuno deve partecipare a numerosi scontri, mentre altri potrebbero anche avere zero rivali!

Ci sono N pugili, ognuno descritto da un intero $0 \leq i < N$. Le qualificazioni sono descritte da una lista di M coppie (i, j) con $i \neq j$, che stabiliscono la rivalità tra i pugili i e j .

Per favorire la raccolta delle scommesse, gli studenti vogliono partizionare i pugili in due gruppi in modo che, presa una coppia qualsiasi di pugili appartenenti ad un gruppo, non vi sia rivalità tra di essi. Data la descrizione di un torneo, stampa TRUE se il partizionamento è possibile, FALSE altrimenti.

Note

La descrizione del problema è azzeccata come espediente didattico, ma non vuole favorire il gioco d'azzardo.

Input

L'input deve essere letto da un file "input.txt".

La prima riga dell'input consiste di due numeri interi separati da uno spazio, N (il numero totale di pugili, numerati da 0 a $N - 1$) e M (il numero di rivalità tra pugili).

Seguono M righe, ognuna delle quali contiene due numeri interi i e j separati da uno spazio. Queste coppie stabiliscono tra quali coppie di pugili sussiste rivalità.

Output

L'output deve essere scritto su un file "output.txt".

L'output consiste nella stringa TRUE se è possibile suddividere i pugili in due gruppi, tali che non vi siano pugili rivali tra loro nello stesso gruppo. Altrimenti, FALSE.

Vincoli

- $1 \leq N, M \leq 10^6$;
- $0 \leq i, j < N$
- **tempo limite:** 1 s;
- **memoria limite:** 256 MiB.

Punteggio

Risolvere questo problema garantisce un punto all'esame di algoritmi.

Input (input.txt)	Output (output.txt)
6 5 0 5 1 5 3 5 4 5 2 1	TRUE
5 3 0 1 1 2 2 0	FALSE

Esempi

Nel primo caso d'esempio, i due gruppi sono costituiti dai pugili $\{0, 1, 3, 4\}$ e $\{2, 5\}$.

Nel secondo caso d'esempio, i pugili 3 e 4, non avendo nessun rivale, vincono a tavolino. Non è comunque possibile suddividere i pugili come richiesto. Entrambi i pugili 1 e 2 devono scontrarsi con 0, ma sono anche rivali tra loro.