

Prehistoria

Limit pamięci: 64 MB

Na dziewiczych, zamieszkałych przez dinozaury, mamuty, wiewiórki biegające za żółędziem i inne dziwne stwory terenach rozciąga się kraina zwana przez jej pierwotnych mieszkańców Bajtolandią. Na owym obszarze żyje n plemion. Między niektórymi z nich są dwukierunkowe drogi, których jest w sumie m . Wodzowie postanowili połączyć ich plemiona w sojusze, aby łatwiej było im przetrwać. Zasada łączenia jest prosta: plemię a zawiąże sojusz z plemieniem b jeśli:

- istnieje droga łącząca plemię a i plemię b ,

lub

- istnieje takie plemię c , że zarówno plemię a i plemię b mają z nim sojusz.

Proces zawierania sojuszy trwa tak długo, aż nie da się zawrzeć żadnego innego.

Ludzie zamieszkujący te tereny są prymitywni, aczkolwiek bardzo inteligentni, więc zgadali się i założyli Koło Informatyczne Bajtolandii (KIB). Zasada przyjęcia do KIBu jest prosta. Obecni członkowie zadają kandydatowi q pytań o treści "Czy plemię p jest w sojuszu z plemieniem k ". Jaskiniowcy mają już swoje komputery stworzone z kamieni oraz śladowych ilości drewna, lecz mimo to odpowiadanie na owe pytania sprawia im trudność. Napisz odpowiedni program i pomóż im dostać się do KIBu.

Zadanie

Napisz program, który:

- wczyta ze standardowego wejścia ilość plemion, dróg, opisy tych dróg, ilość pytań KIBu, oraz ich opisy.
- wyliczy odpowiedzi na zapytania,
- wypisze wynik na standardowe wyjście.

Wejście

W pierwszym wierszu standardowego wejścia zapisane są dwie liczby całkowite n i m ($1 \leq n, m \leq 1000000$) oznaczające odpowiednio ilość plemion i dróg w Bajtolandii. W następnych m wierszach znajdują się opisy tych dróg, składające się z dwóch liczb całkowitych a i b ($1 \leq a, b \leq n$, $a \neq b$) oznaczające, że plemię a i plemię b łączy droga. W następnym wierszu znajduje się jedna liczba całkowita q ($1 \leq q \leq 1000000$), a w następnych q wierszach po dwie liczby całkowite p i k ($1 \leq p, k \leq n$, $p \neq k$), oznaczające pytanie KIBu. W testach wartych około 40% punktów zachodzi warunek ($1 \leq n, m, q \leq 1000$).

Wyjście

Twój program powinien wypisać na wyjście q wierszy, w i -tym z nich powinno znajdować się pojedyncze słowo TAK, jeśli odpowiedź na i -te pytanie KIBu jest twierdząca, lub NIE w przeciwnym wypadku.

Przykład

Wejście	Wyjście
8 6	TAK
1 2	NIE
5 7	TAK
3 2	NIE
5 8	
1 3	
2 4	
4	
1 4	
6 7	
7 5	
3 8	

Ilustracja przykładu:

