

# Zadanie: BIE

## Bieg na orientację



Eliminacje do IOI, dzień trzeci. Plik źródłowy `bie.*` Dostępna pamięć: 512 MB.

14.08.2020

Bajtazar organizuje Bajtocki Bieg na Orientację. Bieg będzie odbywał się ulicami Bajtogradu. Sieć drogowa tego miasta składa się z  $n$  skrzyżowań połączonych  $n - 1$  dwukierunkowymi ulicami (z każdego skrzyżowania da się przejechać ulicami do każdego innego).

Bieg będzie zaczynał się na skrzyżowaniu początkowym  $v_1$ , po czym zawodnicy będą musieli dobiec do skrzyżowania  $v_2$ , przy którym stoi pierwszy obiekt opisany w zadaniu orientacyjnym, następnie muszą dobiec do skrzyżowania  $v_3$ , przy którym stoi drugi obiekt, i w końcu wrócić do skrzyżowania początkowego.

Bajtazar zastanawia się właśnie, jak wybrać skrzyżowania  $v_1$ ,  $v_2$  i  $v_3$ , by bieg był jak najciekawszy. Jednym z kryteriów jest odległość pomiędzy tymi skrzyżowaniami – nie może być ani za mała (by zawodnikom nie było zbyt łatwo odnaleźć obiekt), ani za duża (by za bardzo nie znużyli się biegiem). Przygotował sobie kilka wariantów możliwych odległości i dla każdego z wariantów zastanawia się, czy istnieje taki wybór skrzyżowań, który realizuje te odległości. Napisz program, który mu w tym pomoże.

## Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby całkowite  $n$  i  $q$  ( $3 \leq n \leq 300\,000$ ,  $1 \leq q \leq 200\,000$ ) oznaczające liczbę skrzyżowań w Bajtogradzie i liczbę wariantów. Skrzyżowania numerujemy dla uproszczenia liczbami od 1 do  $n$ .

W kolejnych  $n - 1$  wierszach znajduje się opis ulic: w każdym z nich znajdują się dwie liczby całkowite  $a$  i  $b$  ( $1 \leq a \neq b \leq n$ ) oznaczające dwukierunkową ulicę pomiędzy skrzyżowaniami o numerach  $a$  oraz  $b$ .

W kolejnych  $q$  wierszach znajdują się opisy wariantów rozważanych przez Bajtazara; w każdym z nich znajdują się trzy liczby całkowite  $d_{12}$ ,  $d_{23}$  i  $d_{31}$  z przedziału  $[0, n - 1]$ , które oznaczają odległości kolejno pomiędzy skrzyżowaniami  $v_1$  i  $v_2$ , pomiędzy skrzyżowaniami  $v_2$  i  $v_3$  oraz pomiędzy skrzyżowaniami  $v_3$  i  $v_1$ . Czasami Bajtazar rozważa sytuację, w której odległość jest równa zero (więc niektóre skrzyżowania mogą się wtedy powtarzać).

## Wyjście

Na wyjście należy wypisać dokładnie  $q$  wierszy zawierające odpowiedzi do kolejnych zapytań z wejścia. Odpowiedzią jest albo jedno słowo NIE, jeśli poszukiwana trójka skrzyżowań nie istnieje, albo jedno słowo TAK, po którym następują trzy liczby całkowite  $v_1$ ,  $v_2$  i  $v_3$ .

Jeśli istnieje więcej niż jedno rozwiązanie, Twój program może wypisać dowolne z nich.

## Przykład

Dla danych wejściowych:

```
6 4
1 2
2 3
2 5
2 4
5 6
3 2 3
1 2 3
1 1 1
3 0 3
```

jednym z poprawnych wyników jest:

```
TAK 6 4 3
TAK 4 2 6
NIE
TAK 4 6 6
```

### Testy „ocen”:

**1ocen:**  $n = 10$ ,  $q = 10$ ;

**2ocen:**  $n = 7$ ,  $q = 343$ ; ścieżka, wszystkie możliwe zapytania;

**3ocen:**  $n = 1000$ ,  $q = 1000$ ; gąsienica długości 500, wierzchołki  $1, 3, \dots, 499$  mają doczepione po dwa wierzchołki;

**4ocen:**  $n = 300\,000$ ,  $q = 200\,000$ ;  $d_{12} = 0$  we wszystkich wariantach.

## Ocenianie

Jeżeli Twój program wypisze niepoprawne numery skrzyżowań dla odpowiedzi TAK lub nie wypisze ich wcale, wciąż otrzyma on 50% punktów za daną grupę testów.

Zestaw testów dzieli się na następujące podzadania. Testy do każdego podzadania składają się z jednej lub większej liczby osobnych grup testów.

Podzadanie	Warunki	Liczba punktów
1	$n \leq 100$	8
2	$n, q \leq 1000$	12
3	$q \leq 100$	14
4	$d_{12} = 0$ we wszystkich rozważanych wariantach	12
5	sieć drogowa jest ścieżką	8
6	sieć drogowa składa się ze ścieżki oraz wierzchołków w odległości 1 od niej („gąsienica”)	10
7	brak dodatkowych warunków	36