

Gdje je Korijen?

Ovo je interaktivni zadatak

Dano je stablo s n čvorova. Stablo je graf u kojem postoji jednostavan put između svakog para čvorova. **Ovaj vam zadatak garantira i da postoji barem jedan čvor koji je direktno povezan s barem 3 druga čvora.** Jedan od čvorova je korijen i vaš je zadatak pronaći ga.

Dozvoljeno vam je postavljati pitanja slijedećeg oblika:

- Za dani set a_1, a_2, \dots, a_m čvorova, provjeri je li njihov najniži zajednički predak (lowest common ancestor) u setu.

Čvor v je zajednički predak od seta čvorova S ako putevi od svakaog čvora iz S do korijena prolaze kroz v . Najniži zajednički predak (LCA) od seta čvorova S je zajednički predak od S koji je najudaljeniji od korijena.

Interakcija

Prvo učitajte prirodan broj n ($4 \leq n \leq 500$) - broj čvorova.

Onda učitajte $n - 1$ linija. i -ta linija će sadržati dva broja a_i, b_i ($1 \leq a_i, b_i \leq n$), koja indiciraju da postoji edge između čvorova a_i, b_i u stablu.

Garantiramo vam da će tih $n - 1$ edgeva formirati stablo s barem jednim čvorom koji je direktno povezan s barem 3 druga čvora.

Da bi postavili upit, prvo ispišite "?", onda broj m , te m različitih brojeva a_1, a_2, \dots, a_m ($1 \leq m \leq n, 1 \leq a_i \leq n$, svi a_i su različiti) - čvorovi za koje želite provjeriti je li LCA među njima.

Kao odgovor, interaktor će ispisati "YES" ako je njihov LCA jedan od a_1, a_2, \dots, a_m , ili "NO" inače.

Možete postaviti najviše 1000 upita, ali ćete dobiti različit broj bodova ovisno o koliko upita iskoristite. Ispis odgovora se ne broji kao upit. Molimo vas da pogledate sekciju bodovanje za detalje.

Kada ste našli korijen, ispišite simbol "!" i onda broj v ($1 \leq v \leq n$) - indeks korijena. Tada završite program.

Nakon ispisa upita nemojte zaboraviti preći u novi red i flushati output. Za flushanje outputa koristite:

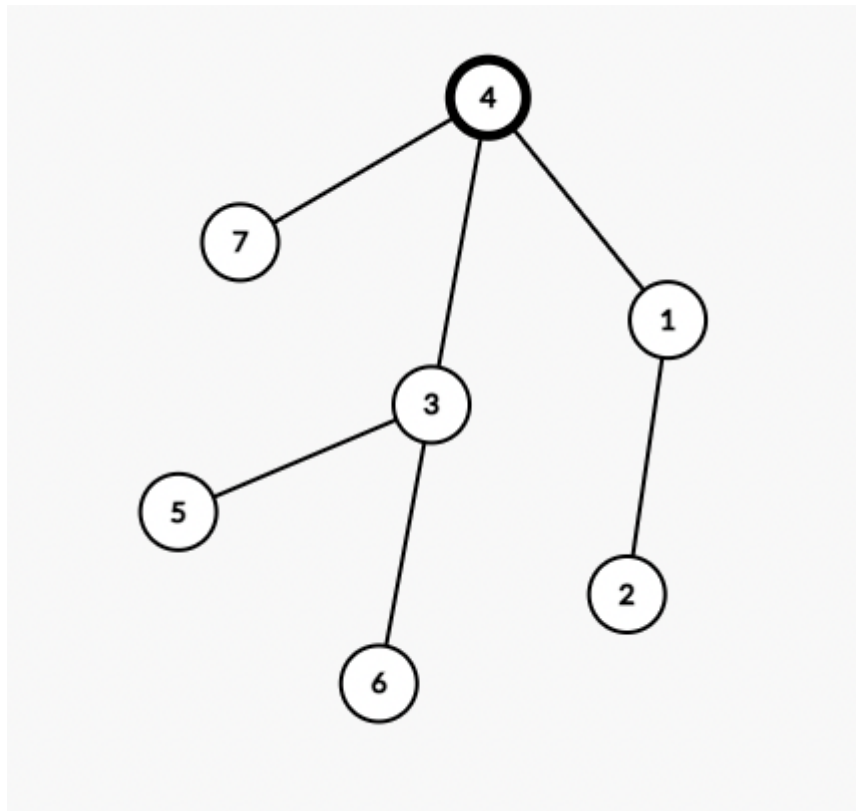
- `fflush(stdout)` ili `cout.flush()` u C++u;
- `stdout.flush()` u Pythonu;

Garantiramo vam da su za svaki test primjer stablo i njezin korijen fiksirani prije početka interakcije. Drugim riječima, **interaktor nije adaptivan**.

Primjer

```
Input:
7
4 1
1 2
4 3
3 5
3 6
4 7
Output:
? 2 5 6
Input:
NO
Output:
? 3 6 3 5
Input:
YES
Output:
? 2 1 7
Input:
NO
Output:
? 2 4 6
Input:
YES
Output:
! 4
```

Pojašnjenje



Skriveni korijen je čvor 4.

U prvom upitu, LCA čvorova 5 i 6 je čvor 3 koji nije među čvorovima 5 i 6 pa je odgovor "NO".

U drugom upitu, LCA čvorova 3, 5, i 6 je čvor 3 pa je odgovor "YES".

U trećem upitu, LCA čvorova 1 i 7 je čvor 4 pa je odgovor "NO".

U četvrtom upitu, LCA čvorova 4 i 6 je čvor 4 pa je odgovor "YES".

Nakon toga, možemo pogoditi da je korijen čvor 4 što je i točan odgovor.

Bodovanje

1. (7 bodova): $n \leq 9$
2. (10 bodova): $n \leq 30$
3. (do 83 boda): $n \leq 500$

U prvom i drugom podzadatku možete koristiti najviše 1000 upita.

U trećem podzadatku, neka je k najveći broj upita koji ste pitali u jednom primjeru u tom podzadatku. Ako je $k \leq 9$, dobit ćete 83 boda. Inače ćete dobiti $\lfloor \max(10, 83 \cdot (1 - \frac{\ln(k-6)}{7})) \rfloor$

bodova.

C++ kod koji računa broj bodova za treći podzadatak:

```
((k <= 9) ? 83: max(10, int(83 * (1 - log(k - 6.0) / 7))))
```