

Gdje je korijen?

Ovo je interaktivni problem.

Dato vam je drvo od n čvorova. Drvo je graf takav da postoji samo jedan jednostavan put između svakog para čvorova. **Također je garantovano da je najmanje jedan čvor spojen sa najmanje 3 druga čvora.** Jedan od čvorova je korijen i tvoj zadatak je da ga pronađeš. Da bi to postigao dozvoljeni su ti upiti u sljedećoj formi:

- Za dati skup a_1, a_2, \dots, a_m čvorova, provjeri da li je njihov najniži zajednički predak (lowest common ancestor) u ovom skupu.

Čvor v je zajednički predak skupa S čvorova ako putevi od svih tačaka u S prolaze kroz v . Najniži zajednički predak skupa S je onaj koji je najdalji od korijena drveta.

Interakcija

Počni interakciju čitajući jedan cijeli broj n ($4 \leq n \leq 500$) - broj čvorova.

Zatim pročitaj sljedećih $n - 1$ linija. i -ta linija će sadržavati dva cijela broja a_i, b_i ($1 \leq a_i, b_i \leq n$), ukazujući na to da postoji veza između čvorova a_i, b_i u drvetu.

Garantovano je da ovih $n - 1$ veza formiraju drvo i da postoji najmanje jedan čvor koji je spojen sa najmanje 3 druga čvora.

Da postavite upit, prvo ispišite "?", zatim cijeli broj m , a potom m različitih cijelih 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 da li je njihov NZP među njima.

Kao odgovor, interaktor će ispisati "YES" ako je njihov NZP jedan od a_1, a_2, \dots, a_m , a "NO" ako nije.

Možete postaviti najviše 1000 upita, ali ćete dobiti različit broj bodova u zavisnosti od toga koliko upita postavite. Ispisivanje odgovora se ne računa kao upit. Molimo pogledajte dio *bodovanje* za više informacija.

Nakon što pronađete korijen ispišite simbol "!" i zatim jedan cijeli broj v ($1 \leq v \leq n$) - korijen. Zatim završite vaš program.

Nakon ispisivanja upita ne zaboravite ispisati kraj linije (end of line) i *flush*-ati izlaz. Da biste to uradili, koristite:

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

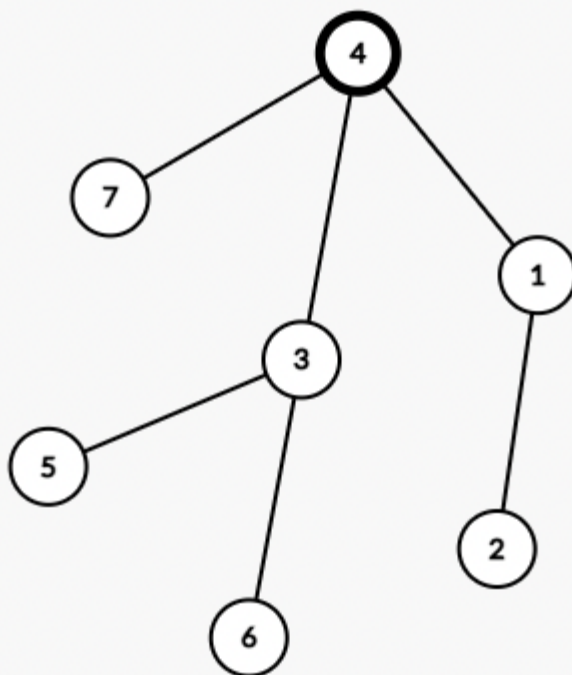
Zagarantovano je da za svaki testni primjer, drvo i njegov korijen je fiksiran prije početka interakcije. Drugim riječima, **interaktor neće mijenjati iskaz**.

Ne zaboravite *flush*-ati izlaz prije svakog upita.

Primjer

```
Ulaz:
7
4 1
1 2
4 3
3 5
3 6
4 7
Izlaz:
? 2 5 6
Ulaz:
NO
Izlaz:
? 3 6 3 5
Ulaz:
YES
Izlaz:
? 2 1 7
Ulaz:
NO
Izlaz:
? 2 4 6
Ulaz:
YES
Izlaz:
! 4
```

Objašnjenje



Skriveni korijen je čvor 4.

U prvom pitanju, NZP vrhova 5 i 6 je čvor 3 koje nije među vrhovima 5 i 6 pa je odgovor "NO".

U drugom pitanju, NZP vrhova 3, 5 i 6 je čvor 3, pa je odgovor "YES".

U trećem pitanju, NZP vrhova 1 i 7 je čvor 4, stoga je odgovor "NO".

U četvrtom pitanju, NZP vrhova 4 i 6 je čvor 4, pa je odgovor "YES".

Nakon ovoga, možemo pretpostaviti da je korijen tjeme 4, što i jeste tačan odgovor.

Bodovanje

Uzmimo za primjer da je k maksimaln broj upita koje si postaivo za jedan podzadatak. Možeš postaviti najviše 1000 upita, znači $k \leq 1000$.

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

U trećem podzadatku, ako je $k \leq 9$, dobit ćete 83 boda. U suprotnom, dobit ćete $\lfloor \max(10, 83 \cdot (1 - \frac{\ln(k-6)}{7})) \rfloor$ bodova.

Prema tome, za sve bodove, tvoje rješenje treba da napravi najviše 9 upita na svakom testu podzatka.

C++ kod koji računa broj bodova:

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