

Où est la racine ?

Ceci est un sujet interactif

On vous donne un arbre constitué de n noeuds. L'arbre est un graphe tel qu'il y a exactement un chemin simple entre chaque paire de noeuds. **On vous garantit aussi qu'au moins un noeud est directement relié par une arête à au moins 3 noeuds.** L'un des noeuds est la racine, et votre objectif est de trouver lequel. Pour cela, vous pouvez poser des questions de la forme suivante :

- Pour un ensemble de noeuds donné a_1, a_2, \dots, a_m , déterminer si leur plus petit ancêtre commun est dans cet ensemble.

Un noeud v est un ancêtre commun d'un ensemble de noeuds S , si les chemins de tous les noeuds de S à la racine passent par v . Le plus petit ancêtre commun (LCA) d'un ensemble S de noeuds est l'ancêtre commun de S qui est le plus éloigné de la racine.

Interaction

Commencez l'interaction en lisant un entier n ($4 \leq n \leq 500$) seul sur la première ligne - le nombre de noeuds.

Lisez ensuite les $n - 1$ lignes suivantes. La i -ème de ces lignes contiendra deux entiers a_i, b_i ($1 \leq a_i, b_i \leq n$), indiquant que l'arbre contient une arête entre les noeuds a_i et b_i .

On vous garantit que ces $n - 1$ arêtes forment un arbre et qu'il y a au moins un noeud qui est relié directement par une arête à au moins trois noeuds.

Pour poser une question, commencez par afficher "?", puis l'entier m , puis m entiers distincts a_1, a_2, \dots, a_m ($1 \leq m \leq n, 1 \leq a_i \leq n$) - les noeuds de l'ensemble dont vous voulez déterminer si leur LCA est parmi eux.

Comme réponse, l'évaluateur affichera "YES" si leur LCA est l'un des noeuds parmi a_1, a_2, \dots, a_m , et "NO" sinon.

Vous pouvez poser au maximum 1000 questions, mais vous obtiendrez un nombre de points différent selon le nombre de questions que vous posez. Afficher la réponse ne compte pas comme une question. Lisez la section score pour les détails.

Lorsque vous avez identifié la racine, affichez le symbole "!" puis un entier v ($1 \leq v \leq n$) - la racine. Terminez ensuite votre programme.

Après avoir affiché une question, n'oubliez pas d'afficher une fin de ligne et de flusher la sortie. Pour cela, utilisez :

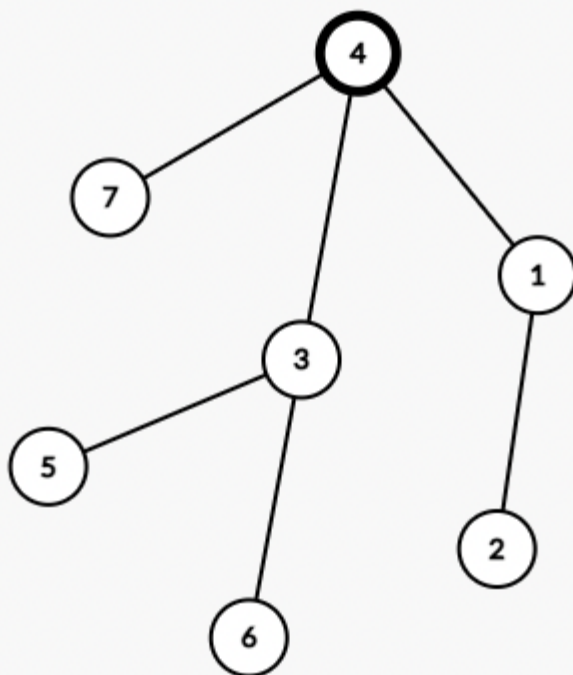
- `fflush(stdout)` ou `cout.flush()` in C++;
- `stdout.flush()` en Python;

On vous garantit que pour chaque test, l'arbre et sa racine sont choisis avant le début de l'interaction. Autrement dit, **l'interaction n'est pas adaptative**.

Exemple

```
Entrée :  
7  
4 1  
1 2  
4 3  
3 5  
3 6  
4 7  
Sortie :  
? 2 5 6  
Entrée :  
NO  
Sortie :  
? 3 6 3 5  
Entrée :  
YES  
Sortie :  
? 2 1 7  
Entrée :  
NO  
Sortie :  
? 2 4 6  
Entrée :  
YES  
Sortie :  
! 4
```

Commentaires



La racine cachée est le noeud 4.

Dans la première question, le LCA des noeuds 5 et 6 est le noeud 3, qui n'est pas parmi les noeuds 5 et 6, donc la réponse est "NO".

Dans la deuxième question, le LCA des noeuds 3, 5 et 6 est le noeud 3, donc la réponse est "YES".

Dans la troisième question, le LCA des noeuds 1 et 7 est le noeud 4, donc la réponse est "NO".

Dans la quatrième question, le LCA des noeuds 4 et 6 est le noeud 4, donc la réponse est "YES".

Après cela, on peut deviner que la racine est le noeud 4, ce qui est la bonne réponse.

Score

1. (7 points) : $n \leq 9$
2. (10 points) : $n \leq 30$
3. (jusqu'à 83 points) : $n \leq 500$

Dans les première et deuxième sous-tâches, vous pouvez poser au maximum 1000 questions.

Dans la troisième sous-tâche, soit k le nombre maximum de questions que vous posez dans un même test. Si $k \leq 9$, vous obtiendrez 83 points. Sinon, vous obtiendrez $\lfloor \max(10, 83 \cdot (1 - \frac{\ln(k-6)}{7})) \rfloor$ points.

Voici le code C++ qui calcule le nombre de points pour la troisième sous-tâche :

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