

## Recherche dans l'arbre (Tree Search)

On vous donne un arbre binaire enraciné (rooted binary tree) de  $N$  nœuds. Les nœuds sont numérotés de 1 à  $N$ , la racine (root) est le nœud numéro 1. Chacun des autres nœuds a un seul parent dans l'arbre. L'arbre est binaire, c'est-à-dire que chaque nœud peut être le parent d'au plus deux autres nœuds.

L'un des nœuds est spécial. Vous essayez de le deviner. Vous pouvez poser des questions du type : "Le nœud spécial est-il dans le sous-arbre du nœud  $x$  ?" Un nœud  $y$  est dans le sous-arbre de  $x$  si et seulement si le plus court chemin entre  $y$  et 1 passe par le nœud  $x$ . Observez que le nœud  $x$  est aussi dans son propre sous-arbre.

Vous êtes autorisé à poser cette question au plus 35 fois. Ensuite, vous devez donner votre réponse.

### Détails d'implémentation

Vous devez implémenter la procédure suivante :

```
int solve(int N, std::vector < int > p)
```

- $N$  : le nombre de nœuds
- $p$  contient exactement  $N - 1$  éléments qui décrivent l'arbre : le nœud  $p[i]$  (où  $1 \leq p[i] \leq i + 1$ ) est le parent du nœud  $i + 2$  pour chaque  $0 \leq i \leq N - 2$
- Aucun élément dans  $p$  n'apparaît plus de deux fois.
- Cette procédure doit renvoyer le numéro du nœud spécial.
- Cette procédure est appelée exactement une fois.

La procédure ci-dessus peut appeler la procédure suivante :

```
int ask(int x)
```

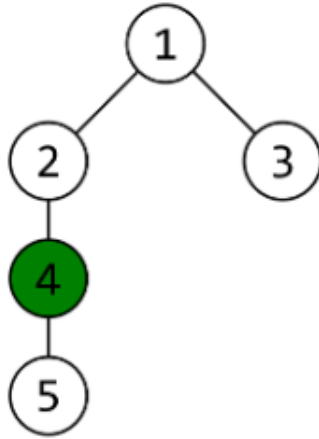
- $x$  : le numéro du nœud
- $1 \leq x \leq N$
- renvoie 1 si le nœud spécial est dans le sous-arbre de  $x$  et 0 sinon

### Exemple

Étudions l'appel suivant :

```
solve(5, [1, 1, 2, 4])
```

L'arbre contient les arêtes (1,2), (1,3), (2,4) et (4,5).



Votre programme a fait l'appel

```
ask(4)
```

qui a renvoyé 1. Après cela, votre programme a fait l'appel

```
ask(5)
```

qui a renvoyé 0.

Votre programme a conclu que le nœud 4 est spécial et a renvoyé 4.

## Contraintes

- $2 \leq N \leq 100\,000$

## Sous-tâches

1. (20 points)  $N \leq 35$
2. (30 points)  $p[i] = i + 1$  pour chaque  $0 \leq i \leq N - 2$
3. (15 points)  $p[i] = \lfloor i/2 \rfloor + 1$  pour chaque  $0 \leq i \leq N - 2$
4. (35 points) Aucune contrainte supplémentaire.

## Évaluateur d'exemples (Sample Grader)

L'évaluateur d'exemples lit l'entrée dans le format suivant :

- ligne 1 :  $N$
- ligne 2 :  $p[0], p[1], \dots, p[N - 2]$

L'évaluateur d'exemples affiche chaque question dans le format suivant :

- ligne 1 : ?  $x$

L'évaluateur d'exemples lit chaque réponse dans le format suivant :

- ligne 1 :  $y$

L'évaluateur d'exemples affiche la réponse dans le format suivant :

- ligne 1 : !  $x$