

## Αναζήτηση Δέντρου

Σας δίνεται ένα ριζωμένο δυαδικό δέντρο (rooted binary tree) που αποτελείται από  $N$  κορυφές. Οι κορυφές αριθμούνται από 1 έως  $N$ , η ρίζα είναι η κορυφή 1. Κάθε μία από τις υπόλοιπες κορυφές έχει έναν μόνο γονέα στο δέντρο. Το δέντρο είναι δυαδικό, δηλαδή κάθε κορυφή μπορεί να είναι γονέας το πολύ δύο άλλων κορυφών.

Μία από τις κορυφές είναι ιδιαίτερη. Προσπαθείτε να την μαντέψετε. Μπορείτε να κάνετε τις ερωτήσεις του ακόλουθου είδους: "Περιλαμβάνεται η ειδική κορυφή στο υποδέντρο της κορυφής  $x$ "; Ένας κόμβος  $y$  βρίσκεται στο υποδέντρο της κορυφής  $x$  αν και μόνο αν το συντομότερο μονοπάτι μεταξύ  $y$  και 1 περνάει από την κορυφή  $x$ . Σημειώστε ότι η κορυφή  $x$  βρίσκεται επίσης στο δικό της υποδέντρο.

Επιτρέπεται να κάνετε αυτήν την ερώτηση το πολύ 35 φορές. Μετά από αυτό θα πρέπει να αναφέρετε την μαντεψιά σας.

## Λεπτομέρειες υλοποίησης

Θα πρέπει να υλοποιήσετε την ακόλουθη διαδικασία:

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

- $N$ : ο αριθμός των κορυφών
- $p$ : περιέχει ακριβώς  $N - 1$  στοιχεία που περιγράφουν το δέντρο: η κορυφή  $p[i]$  (όπου  $1 \leq p[i] \leq i + 1$ ) είναι ο γονέας της κορυφής  $i + 2$  για κάθε  $0 \leq i \leq N - 2$
- Κανένα στοιχείο στο  $p$  δεν εμφανίζεται περισσότερο από δύο φορές
- Αυτή η διαδικασία θα πρέπει να επιστρέψει τον αριθμό της ειδικής κορυφής
- Αυτή η διαδικασία καλείται ακριβώς μία φορά

Η παραπάνω διαδικασία μπορεί να κάνει κλήσεις στην ακόλουθη διαδικασία:

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

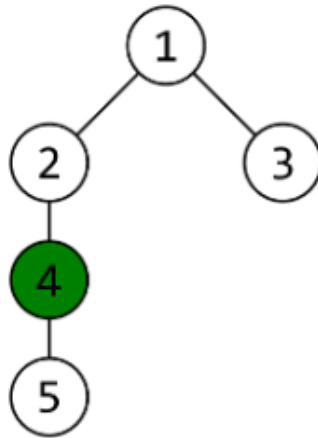
- $x$ : ο αριθμός της κορυφής
- $1 \leq x \leq N$
- επιστρέφει 1 εάν η ειδική κορυφή βρίσκεται στο υποδέντρο του  $x$  και 0 διαφορετικά.

## Παράδειγμα

Στην ακόλουθη κλήση:

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

Το δέντρο αποτελείται από τις κορυφές (1,2), (1,3), (2,4) και (4,5).



Το πρόγραμμά σας έκανε μια κλήση

```
ask(4)
```

η οποία επέστρεψε 1. Στη συνέχεια το πρόγραμμά σας έκανε μια κλήση

```
ask(5)
```

η οποία επέστρεψε 0.

Το πρόγραμμά σας κατέληξε στο συμπέρασμα ότι η κορυφή 4 είναι ιδιαίτερη και επέστρεψε 4.

## Περιορισμοί

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

## Subtasks

1. (20 πόντοι)  $N \leq 35$
2. (30 πόντοι)  $p[i] = i + 1$  για κάθε  $0 \leq i \leq N - 2$
3. (15 πόντοι)  $p[i] = \lfloor i/2 \rfloor + 1$  για κάθε  $0 \leq i \leq N - 2$
4. (35 πόντοι) Κανένας περιορισμός.

## Sample Grader

Ο Sample Grader διαβάζει την είσοδο με την ακόλουθη μορφή:

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

Ο Sample Grader εξάγει κάθε ερώτηση με την ακόλουθη μορφή:

- γραμμή 1:  $? x$

Ο Sample Grader διαβάζει κάθε απάντηση με την ακόλουθη μορφή:

- γραμμή 1:  $y$

Ο Sample Grader καταλήγει στο συμπέρασμα με την ακόλουθη μορφή:

- line 1:  $! x$