

# Пребарување на дрво

Дадено е кореново бинарно дрво (бинарно дрво со корен, англ. rooted binary tree), кое се состои од  $N$  темиња. Темињата се нумерирани со целите броеви од 1 до  $N$ , каде коренот е темето со број 1. Секое друго теме во дрвото (освен коренот) има точно еден родител. Дрвото е бинарно, т.е. секое теме може да биде родител на најмногу две други темиња.

Едно од темињата е специјално теме. Вие се обидуваат да го погодите. Може да поставувате прашања од следниот вид: "Дали специјалното теме се наоѓа во поддрвото на темето  $x$ "? Темето  $y$  се наоѓа во поддрвото на темето  $x$  ако и само ако најкраткиот пат помеѓу  $y$  и 1 минува низ темето  $x$ . Да забележиме дека темето  $x$  е исто така и во своето поддрво.

Дозволено е ова прашање да го поставите најмногу 35 пати. После ова, треба да го дадете вашиот обид за точен одговор (англ. guess).

## Имплементациски детали

Треба да ја имплементирате следната процедура:

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

- $N$ : бројот на темиња
- $p$  содржи точно  $N - 1$  елементи што го опишуваат дрвото: темето  $p[i]$  (каде  $1 \leq p[i] \leq i + 1$ ) е родителот на темето  $i + 2$ , за секое  $0 \leq i \leq N - 2$
- Ниту еден елемент во  $p$  не се појавува повеќе од два пати
- Оваа процедура треба да го врати бројот на специјалното теме
- Оваа процедура се повикува точно еднаш

Горната процедура може да прави повици до следната процедура:

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

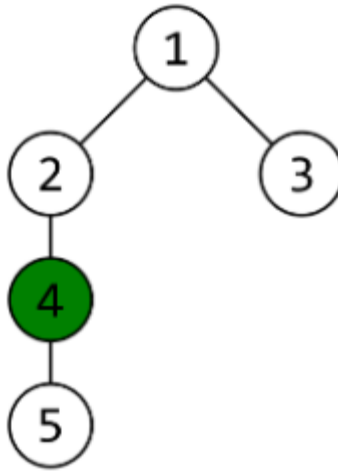
- $x$ : бројот на темето
- $1 \leq x \leq N$
- враќа 1 ако специјалното теме се наоѓа во поддрвото на темето  $x$ , а во спротивно враќа 0.

## Пример

Да го разгледаме следниот повик:

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

Дрвото се состои од ребрата (1, 2), (1, 3), (2, 4) и (4, 5).



Вашата програма го направила повикот

```
ask(4)
```

кој вратил вредност 1. После ова вашата програма го направила повикот

```
ask(5)
```

кој вратил вредност 0.

Вашата програма заклучила дека темето 4 е специјалното теме, па вратила вредност 4.

## Ограничувања

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

## Подзадачи

1. (20 поени)  $N \leq 35$
2. (30 поени)  $p[i] = i + 1$ , за секое  $0 \leq i \leq N - 2$
3. (15 поени)  $p[i] = \lfloor i/2 \rfloor + 1$ , за секое  $0 \leq i \leq N - 2$
4. (35 поени) Без дополнителни ограничувања.

## Пример-оценувач

Пример-оценувачот го чита влезот во следниот формат:

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

Пример-оценувачот го печати секое прашање во следниот формат:

- линија 1:  $? x$

Пример-оценувачот го чита секој одговор во следниот формат:

- линија 1:  $y$

Пример-оценувачот го печати обидот за точен одговор (анг. guess) во следниот формат:

- линија 1:  $! x$