

Tree Search

Dano vam je rootano binarno stablo s N čvorova. Čvorovi su numerirani od 1 do N , root je čvor s brojem 1. Svaki drugi čvor ima jednog roditelja u stablu. Stablo je binarno, pa svaki čvor može imati najviše dvoje djece.

Jedan je čvor u stablu poseban. Vi ga želite pogoditi. Možete pitati pitanja sljedećeg oblika: "Je li poseban čvor u podstablu čvora x ?" Podstablo čvora x je stablo koje se sastoji od čvora x i svih njegovih potomaka.

Dopušteno vam je pitati najviše 35 pitanja. Nakon toga trebate vratiti odgovor.

Implementacijski detalji

Trebate implementirati sljedeću funkciju:

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

- N : broj čvorova
- vektor p sadrži točno $N - 1$ elemenata koja opisuju stablo: čvor $p[i]$ (gdje je $1 \leq p[i] \leq i + 1$) je roditelj čvora $i + 2$ za svaki $0 \leq i \leq N - 2$
- Niti jedan element iz p se ne pojavljuje više od dvaput
- Ova funkcija treba vratiti indeks specijalnog čvora
- Funkcija se poziva samo jedanput po primjeru

Gornja funkcija može pozivati sljedeću funkciju:

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

- x : indeks čvora
- $1 \leq x \leq N$
- Vraća 1 ako je specijalni čvor u podstablu od x i 0 inače

Ograničenja

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

Bodovanje

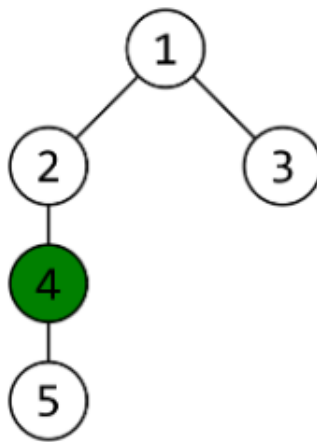
1. (20 bodova) $N \leq 35$
2. (30 bodova) $p[i] = i + 1$ za svaki $0 \leq i \leq N - 2$
3. (15 bodova) $p[i] = \lfloor i/2 \rfloor + 1$ za svaki $0 \leq i \leq N - 2$
4. (35 bodova) Bez dodatnih ograničenja.

Probni primjeri

Pogledajmo sljedeći primjer poziva funkcije:

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

Stablo sadrži edgeve: (1,2), (1,3), (2,4) i (4,5).



Vaš program je pozvao funkciju ask:

```
ask(4)
```

Koja je vratila 1. Nakon toga vaš je program još jednom pozvao funkciju ask:

```
ask(5)
```

Taj poziv je vratio 0.

Vaš je program zaključio da je čvor s indeksom 4 specijalan i vratio 4.

Sample Grader

Sample grader čita podatke o stablu u sljedećem formatu:

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

Sample grader ispisuje svako pitanje u sljedećem formatu:

- linija 1: ? x

Nakon svakog pitanja potrebno je upisati odgovor na to pitanje. Sample grader čita svaki odgovor u sljedećem formatu:

- linija 1: y

Sample grader ispisuje vaš odgovor u sljedećem formatu:

- linija 1: ! x