

Kiegyensúlyozott keresőfa

Rendezett adatsorozatban egy adatelemet bináris kereséssel is megkereshetünk. Ha gyakran kell ilyen keresést végezni, akkor elhelyezhetjük az adatokat bináris fa adatszerkezetben. Egy bináris fát keresőfának nevezünk, ha minden p pontjára teljesül, hogy ha a q pont p bal-részfájában van, akkor q -ban kisebb adat van, mint p -ben, és ha az r pont p jobb-részfájában van, akkor r -ben nagyobb adat van, mint p -ben. Bináris keresőfa esetén egy adat megkereséséhez legrosszabb esetben annyi összehasonlítást kell tenni, ahány pontja van a fának a leghosszabb, gyökértől levél-ig tartó úton. Ezért érdemes kiegyensúlyozottá tenni a fát. Azt mondjuk, hogy egy bináris fa kiegyensúlyozott, ha bármely p pontjára teljesül, hogy a bal-részfájában és a jobb-részfájában levő pontok száma legfeljebb eggyel tér el.

Ha az $a_1 < a_2 < \dots < a_N$ rendezett sorozat elemeit kell elhelyezni egy bináris keresőfában, akkor elegendő megadni azt, hogy a fa minden pontjában melyik indexű elem van.

Készíts programot, amely elkészít egy N -pontú kiegyensúlyozott bináris keresőfát!

Bemenet

A *standard bemenet* egyetlen sora egy egész számot tartalmaz, a bináris keresőfa pontjainak számát ($1 \leq N \leq 20\,000$). A keresőfa pontjait az $1, \dots, N$ számokkal azonosítjuk.

Kimenet

A *standard kimenet* pontosan N sort tartalmazzon, soronként három egész számot, szóközzel elválasztva! Az i -edik sorban a bináris keresőfa i . azonosítójú pontjának adatai legyenek! Az első szám a pontba helyezendő adatelem sorszáma legyen! A második szám a bal gyerek sorszáma, a harmadik pedig a jobb gyerek sorszáma! Ha a pontnak valamelyik gyereke hiányzik, akkor a 0 számot kell megadni! Több megoldás esetén bármelyik megadható.

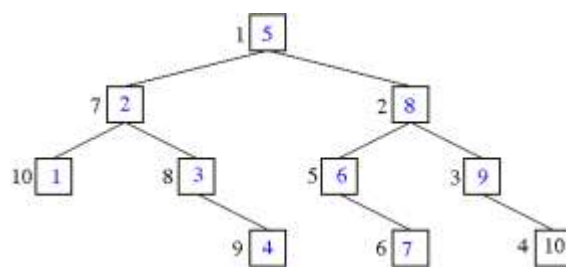
Példa

Bemenet

10

Kimenet

```
5 7 2
8 5 3
9 0 4
10 0 0
6 0 6
7 0 0
2 10 8
3 0 9
4 0 0
1 0 0
```



Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 40%-ában a pontok száma $N \leq 1000$