

Titkos társaság 4

Egy titkos társaság hierarchikusan épül fel, minden tagja csak a felettesét és a hozzá közvetlenül beosztott legfeljebb két tagot ismeri. A társaságnak pontosan egy olyan tagja van, akinek nincs főnöke. Bármelyik tag küldhet levelet bármelyik tagnak. Minden levél csak úgy juthat el a feladótól a címzetthez, hogy egy lépésben vagy a közvetlen főnökhöz, vagy közvetlen beosztotthoz továbbítódik.

Készíts programot, amely adott két, X és Y tagra kiszámítja, hogy

- 1.) hány beosztottja – nem csak közvetlen – van az X és az Y tagnak;
- 2.) hány lépéssel továbbítódik egy levél, ha X küld levelet Y-nak;
- 3.) mennyi a legkevesebb lépésszám, ami alatt biztosan odaér egy levél, bárki legyen is a feladó, illetve a címzett?

Bemenet

A *standard bemenet* első sorában a társaság tagjainak száma ($1 \leq N \leq 10\,000$), valamint a két tag sorszáma ($1 \leq X, Y \leq N$) van. A társaság tagjai olyan sorszámot kaptak 1 és N között, hogy mindenkinek nagyobb a sorszáma, mint a közvetlen főnökéé. A második sor pontosan N db 0 és N közötti egész számot tartalmaz egy-egy szóközzel elválasztva. Az i -edik szám a társaság i sorszámú tagjának közvetlen főnökét adja. A sorban az első szám 0, mivel pontosan egy tagnak, az 1 sorszámúnak nincs főnöke. Minden tag legfeljebb két másik tagnak lehet a közvetlen főnöke.

Kimenet

A *standard kimenet* első sorába az X és az Y tag beosztottjainak a számát kell írni! A második sorba azt a lépésszámot kell írni, amely alatt egy levél eljut az X tagtól a Y taghoz! A harmadik sorba a legkevesebb lépésszámot kell írni, ami alatt biztosan odaér egy levél, bárki legyen is a feladó, illetve a címzett!

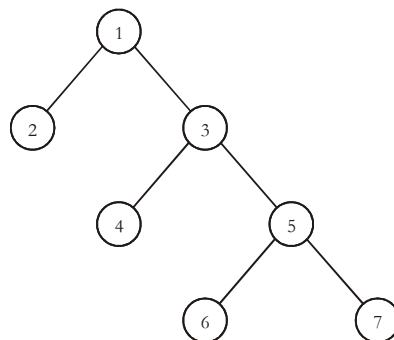
Példa

Bemenet

```
7 2 5
0 1 1 3 3 5 5
```

Kimenet

```
0 2
3
4
```



Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 40%-ában a tagok száma $N \leq 100$