

## Túra

Magyarország folyóiról feljegyeztük, hogy milyen másik folyóba folynak bele. Minden folyó legfeljebb egy másikba folyhat bele, de lehet, hogy egybe sem (mert nem folyóba folyik bele). Csónak-túrákat szeretnénk szervezni, de a könnyebbség kedvéért csak úgy, hogy minden folyón a folyás irányában haladjunk.

Készíts programot, amely megadja, hogy

- 1.) két különböző folyón indult túra hol találkozhat;
- 2.) az első túrát bevárhatja-e egy második úgy, hogy nem indul el addig, amíg az első oda nem ér, és ha igen, akkor az elsőnek hány folyón kell addig haladnia (ha ugyanazon a folyón indulnak, akkor 1, ha az egyik folyó éppen befolyik a másikba, akkor 2, és így tovább).

## Bemenet

A *standard bemenet* első sorában a folyók száma ( $1 \leq N \leq 10\,000$ ), a másik folyóba torkoló folyók száma ( $0 \leq M < N$ ) és a két megvizsgálandó folyó sorszáma ( $1 \leq E \neq F \leq N$ ) van. A következő  $M$  sor két egész számot ( $1 \leq A \neq B \leq N$ ) tartalmaz egy szóközzel elválasztva, ennek jelentése: az  $A$  folyóba befolyik a  $B$  folyó.

## Kimenet

A *standard kimenet* első sorába annak a folyónak a sorszámát kell írni, ahol a két túra először találkozhat; a sor legyen üres, ha a két túra Magyarországon nem találkozhat! A második sorba azt az egész számot kell írni, ahány folyón az első túrának át kell haladnia, hogy a második túra kezdetéhez érjen! Ha a második nem tudja bevárni az első, akkor ez a szám 0 legyen!

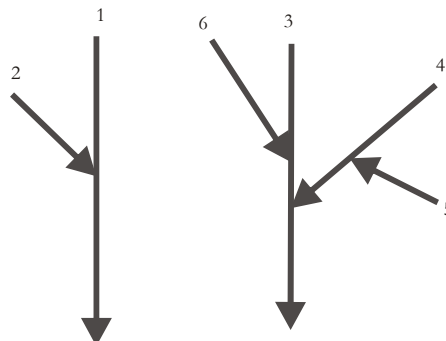
## Példa

Bemenet

```
6 4 6 5
1 2
3 4
4 5
3 6
```

Kimenet

```
3
0
```



## Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MiB

Pontozás: A tesztek 40%-ában a folyók száma  $N \leq 500$