



## Szimurg

A legendás perzsa hős, Zal akkor veheti feleségül Rudabát, ha teljesít egy küldetést.

Perzsiában  $n$  város van (0-tól  $n - 1$ -ig sorszámozva), s közöttük  $m$  kétirányú út (0-tól  $m - 1$ -ig sorszámozva). Minden út két különböző várost köt össze és bármely két város között legfeljebb egy út van. Bármely városból bármely másik városba el lehet jutni utakon keresztül. Vannak királyi utak, de ezek titkosak. Zal feladata meghatározni ezeket.

Zal ismeri Perzsia városait és útjait, a királyi utak meghatározásához pedig segítséget kér Szimurgtól.

Szimurg egy aranyhalmazról meg tudja mondani, hogy abban hány királyi út van. Tudjuk, hogy a királyi utak halmaza is egy aranyhalmaz.

Utak egy halmaza aranyhalmaz, ha

- pontosan  $n - 1$  útból áll
- bármely városból bármely városba el lehet jutni a halmaz útjain keresztül.

Zal feltesz kérdéseket Szimurnak. Minden kérdésnél:

1. Választja utak egy aranyhalmazát.
2. Szimurg megmondja, hogy ebben hány királyi út van.

Írj programot, amely legfeljebb  $q$  kérdéssel meghatározza a királyi utak halmazát!

## Megvalósítás

Az alábbi függvényt kell megírnod:

```
int[] find_roads(int n, int[] u, int[] v)
```

- $n$ : a városok száma,
- $u$  és  $v$ :  $m$  elemű tömbök. Minden  $i$ -re  $u[i]$  és  $v[i]$  várost köti össze az  $i$ . út ( $0 \leq i \leq m - 1$ ).
- A függvény eredménye egy  $n - 1$  elemű tömb legyen, ami tartalmazza a királyi utak sorszámait, tetszőleges sorrendben.

A megoldásod legfeljebb  $q$ -szor hívhatja az alábbi függvényt:

```
int count_common_roads(int[] r)
```

- $r$ :  $n - 1$  elemű tömb, ami egy aranyhalmaz útjai sorszámát tartalmazza, tetszőleges sorrendben.
- A függvény megadja az  $r$  aranyhalmazban levő királyi utak számát.

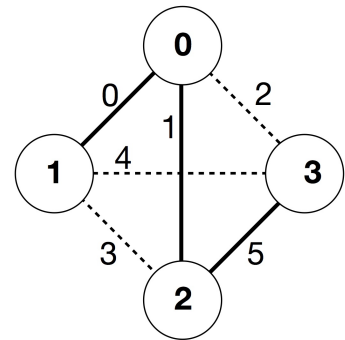
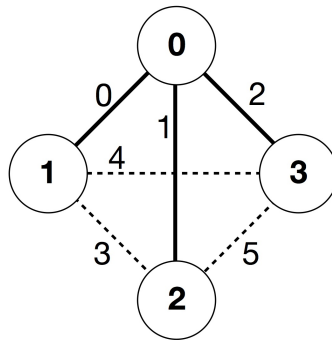
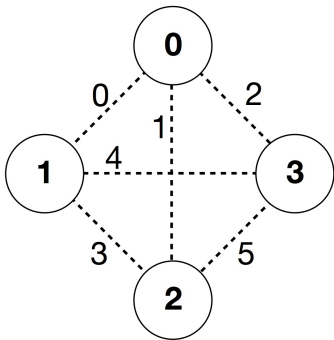
## Példa

```
find_roads(4, [0, 0, 0, 1, 1, 2], [1, 2, 3, 2, 3, 3])
```

`find_roads(...)`

`count_common_roads([0, 1, 2]) = 2`

`count_common_roads([5, 1, 0]) = 3`



Itt 4 város és 6 út van.  $(a, b)$  jelöli az  $a$  és  $b$  városokat összekötő utat. Az utakat 0-tól 5-ig sorszámozzuk, az alábbi sorrendben:  $(0, 1)$ ,  $(0, 2)$ ,  $(0, 3)$ ,  $(1, 2)$ ,  $(1, 3)$ , és  $(2, 3)$ . Minden aranyhalmaz  $n - 1 = 3$  utat tartalmaz.

Tegyük fel, hogy 0, 1, és 5, azaz a  $(0, 1)$ ,  $(0, 2)$ , és  $(2, 3)$  utak a királyi utak, ekkor a következő történik:

- `count_common_roads([0, 1, 2])` eredménye 2. A 0, 1, és 2 sorszámú utakkal, azaz a  $(0, 1)$ ,  $(0, 2)$  és  $(0, 3)$  úttal hívtuk meg. Közülük kettő királyi út.
- `count_common_roads([5, 1, 0])` eredménye 3. Tehát ez a halmaz a királyi utak halmaza.

A `find_roads` függvény eredménye az  $[5, 1, 0]$  elemeket tetszőleges sorrendben tartalmazó 3 elemű tömb legyen!

Az alábbi hívások hibásak:

- `count_common_roads([0, 1])`: az  $r$  elemszáma nem 3.
- `count_common_roads([0, 1, 3])`: itt  $r$  nem aranyhalmaz, mert nem lehet eljutni 0-ból 3-ba a halmaz  $(0, 1)$ ,  $(0, 2)$ ,  $(1, 2)$  útjain.

## Korlátok

- $2 \leq n \leq 500$
- $n - 1 \leq m \leq n(n - 1)/2$
- $0 \leq u[i], v[i] \leq n - 1$  ( $0 \leq i \leq m - 1$ )

- Minden  $i$ -re ( $0 \leq i \leq m - 1$ ),  $u[i] \neq v[i]$ .
- Bármely két város között legfeljebb egy út van.
- Bármely városból bármely másik városba el lehet jutni utakon keresztül.
- A királyi utak halmaza is egy aranyhalmaz.
- A `find_roads` a `count_common_roads` függvényt legfeljebb  $q$ -szor hívhatja. Az  $r$  minden hívásban aranyhalmaz kell legyen!

## Részfeladatok

1. (13 pont)  $n \leq 7$ ,  $q = 30\,000$
2. (17 pont)  $n \leq 50$ ,  $q = 30\,000$
3. (21 pont)  $n \leq 240$ ,  $q = 30\,000$
4. (19 pont)  $q = 12000$  és bármely két város között van út
5. (30 pont)  $q = 8000$

## Minta értékelő

A bemenete:

- Az 1. sor:  $n \ m$
- A  $2 + i$ . sorok ( $0 \leq i \leq m - 1$ ):  $u[i] \ v[i]$
- Az  $2 + m$ . sor:  $s[0] \ s[1] \ \dots \ s[n - 2]$

A királyi utak sorszámai:  $s[0], s[1], \dots, s[n - 2]$ .

A minta értékelő kimenete YES, ha a `find_roads` a `count_common_roads` függvényt legfeljebb 30 000-szer hívja és helyes királyi út halmazt ad meg, egyébként a kimenete NO.

A minta értékelő `count_common_roads` függvénye nem ellenőrzi, hogy helyes-e az  $r$  értéke.

Ha a benyújtott programodban `count_common_roads` függvény paraméterében  $r$  nem aranyhalmaz, akkor 'Wrong Answer' értékelést kapsz.

## Techikai megjegyzés

A `count_common_roads` függvény C++ és Pascal esetén változó paraméterű a hatékonyság miatt, de nem változtatja meg az  $r$  értékét.