

Régen, amikor a számítógép-hálózatok többnyire még koaxiális kábelekből és BNC csatlakozókból álló láncok voltak, a Pokoli Rendszergazda egyik kedvenc hobbjá volt a hálózati topológia menet közbeni átalakítása, és az ezt követő felhasználói reakciók monitorozása.

Sajnos az újabb és újabb hálózatépítési módszerek elterjedésével a fenti szórakozás sokat veszített értékéből, így a Pokoli Rendszergazda elhatározta, hogy a régi szép idők emlékére hobbját virtuális környezetben újraéleszti. Minket kért fel annak a programkomponensnek a kifejlesztésére, amely a hálózat állapotát, illetve a felhasználói visszajelzéseket szimulálja.

A szimulátor egy  $N$  darab összeköttetés nélküli számítógépből álló „üres” hálózatból indul ki, majd összesen  $M$  darab hálózatépítési és diagnosztikai műveletet vár. Feladata, hogy az egyes műveletekről rendre eldöntse, hogy azok megvalósíthatóak-e, és a megvalósítható hálózatépítési műveleteket sorra végrehajtsa.

A hálózatépítési műveletek részletesen a következők:

- **add** <A> <B>

ahol  $1 \leq A \neq B \leq N$  a két számítógép sorszáma, melyek között új (kétirányú) összeköttetést kell kiépíteni. A művelet megvalósítható, ha jelenleg mindkét számítógéphez legfeljebb 1 kábel csatlakozik, és az új összekötés nem eredményez kört.

- **remove** <A> <B>

ahol  $1 \leq A \neq B \leq N$  a két számítógép sorszáma, melyek közötti közvetlen összeköttetést meg kell szüntetni. A művelet megvalósítható, ha  $A$  és  $B$  között vezet kábel.

Az alábbi diagnosztikai művelet pedig annak vizsgálatára szolgál, hogy az adott felhasználóktól várható-e visszajelzés a közeljövőben:

- **ping** <A> <B>

ahol  $1 \leq A \neq B \leq N$  a két számítógép sorszáma, melyek közötti elérhetőséget vizsgáljuk. A művelet megvalósítható, ha  $A$  és  $B$  között vezet út a hálózatban.

Mindhárom műveletre a megvalósíthatóságtól függően „yes”, illetve „no” választ kell adni.

A program  $2 \leq N \leq 1\,000\,000$  és  $0 \leq M \leq 1\,000\,000$  értékét a standard bemenet első sorában kapja, egyetlen szóközzel elválasztva. Az ezt követő  $M$  darab sor pedig egy-egy műveletet ír le a fenti formátumban.

A standard kimenetre szintén pontosan  $M$  darab sor kerüljön, minden sorba rendre a megfelelő művelet eredménye. A maximális pontszám eléréséhez a programnak legfeljebb néhány másodpercen belül le kell futnia a legnagyobb bemenetekre is.

| Példa bemenet | Példa kimenet |
|---------------|---------------|
| 3 8           | no            |
| ping 1 2      | yes           |
| add 1 2       | yes           |
| add 2 3       | yes           |
| ping 1 3      | no            |
| add 3 1       | no            |
| remove 1 3    | yes           |
| remove 1 2    | no            |
| ping 1 3      |               |

Beküldendő egy tömörített `s67.zip` állományban a program forráskódja (`s67.pas`, `s67.cpp`, ...) az `.exe` és más, fordító által generált állományok nélkül, valamint a program rövid dokumentációja (`s67.txt`, `s67.pdf`, ...), amely tartalmazza a megoldás rövid leírását, és megadja, hogy a forrás melyik fejlesztő környezetben fordítható.