

## Benzinkút üzemeltetés

Egy autópálya mentén  $N$  pihenő helyet képeztek ki. Egyes pihenő helyekre benzinkutat építhetünk, de két benzinkút távolsága környezetvédelmi okok miatt legalább  $K$  kilométer kell legyen. Tudjuk, hogy melyik pihenőnél mekkora haszonnal működhet benzinkút.

Készíts programot, amely kiszámolja a maximális elérhető hasznot és meghatározza, hogy ehhez hova kell benzinkutakat építeni!

### Bemenet

A *standard bemenet* első sorában a pihenő helyek száma ( $1 \leq N \leq 1000$ ) és a benzinkutak minimális távolsága ( $1 \leq K \leq 1000$ ) van. A következő  $N$  sorban az egyes pihenőhelyek távolsága az autópálya elejétől ( $1 \leq T_i \leq 1\ 000\ 000$ ) és a várható haszon értéke ( $1 \leq H_i \leq 10\ 000$ ) szerepel, távolság szerint növekvő sorrendben.

### Kimenet

A *standard kimenet* első sorába a maximálisan elérhető hasznot kell írni! A második sorba az ehhez megépítendő benzinkutak  $B$  száma kerüljön, amit a  $B$  pihenőhely sorszáma kövessen, ahova benzinkutat építhetünk, növekvő sorrendben, a sorszámozást 1-től kezdve! Több megoldás esetén bármelyik megadható.

### Példa

Bemenet

5 20  
10 10  
20 40  
30 10  
40 20  
50 30

Kimenet

70  
2 2 5

Megjegyzés: a távolságok miatt 1,3,5 is lehetne, de az csak 50 értékű.

### Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MB

## Bimm, bamm, bumm

Matekszakkörre várva a tanulók egy egyszerű játékkal ütik el az időt. Körbe állnak, majd 1-től kezdve számolnak. Ha a szám osztható 3-mal, akkor a szám helyett azt kell mondani, hogy BIMM, ha a szám osztható 5-tel, akkor helyette a BAMM szót kell mondani, ha pedig 3-mal és 5-tel is osztható, akkor BUMM-ot.

Készíts programot, amely a bemenetben megadott játéokra megadja, hogy melyik volt az első helytelen bemondás!

### Bemenet

A *standard bemenet* első sorában a játék hossza van ( $1 \leq N \leq 1\,000$ ). A következő  $N$  sorban egy-egy bemondás található, abban a sorrendben, ahogy a játékban elhangzott. Egy bemondás négyféle értéket vehet fel: BIMM, BAMM, BUMM vagy SZAM. Ha a SZAM szó szerepel, akkor az adott játékos a körének megfelelő számot mondta ki.

### Kimenet

A *standard kimenet* első sorába az első helytelen bemondásnak a sorszámát kell írni! Ha senki sem hibázott, akkor 0-t kell kiírni!

### Példák

Bemenet

4  
SZAM  
SZAM  
BIMM  
BIMM

Kimenet

4  
(A 4. játékosnak 4-et kellett volna mondania, mert az sem 3-mal, sem 5-tel nem osztható, azaz SZAM szónak kellene szerepelnie a bemenetben.)

Bemenet

5  
SZAM  
SZAM  
BAMM  
SZAM  
SZAM

Kimenet

3  
(A 3. játékosnak BIMM-et kellett volna mondania, mert a száma osztható 3-mal.)

### Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MB

## Fasor

A Százholdas Pagonyban van egy  $N$  fából álló fasor, a szomszédos fák távolsága 1 pagométer. Bagoly akkor boldog, ha olyan fa tetején ül, ahonnan nem lát magasabb fát. Mivel Bagoly öregszik, ezért csak a legfeljebb  $K$  pagométer távolságra lévő fákat látja. Egy sajátjánál magasabb fát tehát pontosan akkor lát, ha a fasorban a sorszámuk különbsége nem nagyobb, mint  $K$ .

Készíts programot, amely megadja a sorban az első olyan fát, amelynek tetején Bagoly boldogan ücsöröghet!

### Bemenet

A *standard bemenet* első sorában a fák száma ( $1 \leq N \leq 200\,000$ ) és Bagoly látótávolsága ( $1 \leq K \leq 200\,000$ ) található. A második sorban az egyes fák magassága van sorrendben, amelyek pozitív egész számok ( $1 \leq H_i \leq 1\,000\,000$ ).

### Kimenet

A *standard kimenet* első sorába a legelső Bagoly számára megfelelő fa sorszámát kell írni. Ha nincs ilyen, akkor -1-et.

### Példa

Bemenet	Kimenet
10 3	4
6 2 1 8 4 8 7 12 9 3	

### Korlátok

Időlimit: 0.2 mp.

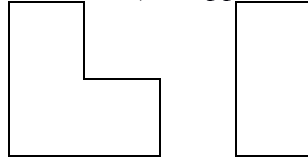
Memórialimit: 32 MB

### Pontozás

A pontok 50%-a szerezhető olyan tesztekre, ahol  $N \leq 10\,000$  és  $K \leq 100$ .

## Járda I és L alakú járólapokkal

Egy  $N \times 2$ -es hosszúságú járdát az alábbi kétféle járólappal kell lefedni:



Készíts programot, amely megadja, hogy a járdát hányféleképpen fedhetjük le velük!

### Bemenet

A *standard bemenet* első sorában a járda hossza van ( $1 \leq N \leq 28$ ).

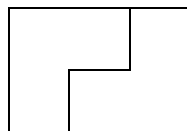
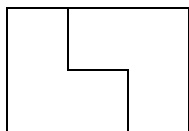
### Kimenet

A *standard kimenet* első sorába a lehetséges járdalefedések számát kell írni!

### Példa

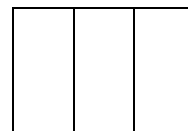
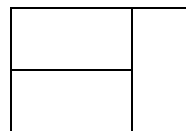
Bemenet

3



Kimenet

5



### Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MB

## Síkság

Egy egyenes mentén ismerjük pontok tengerszint feletti magasságát. Síkságnak nevezzük azokat a szakaszokat, amelyekben belül bármely két pont magassága legfeljebb 1-gyel tér el egymástól.

Készíts programot, amely megadja a leghosszabb síkság hosszát és helyét!

### Bemenet

A *standard bemenet* első sorában a mérések száma van ( $1 \leq N \leq 100\,000$ ). A következő sorban az  $N$  mérés értéke szerepel ( $1 \leq M_i \leq 100\,000$ ).

### Kimenet

A *standard kimenet* egyetlen sorába a leghosszabb síkság hosszát, valamint a síkság kezdetéhez tartozó mérés sorszámát kell írni! Több lehetséges megoldás esetén a legkisebb sorszámút kell megadni (a méréseket 1-től sorszámozzuk)!

### Példa

Bemenet	Kimenet
16	6 9
2 3 3 4 3 5 5 4 3 3 2 3 2 2 1 3	

### Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MB

### Pontozás

A pontok 20%-a szerezhető olyan tesztekre, ahol  $N \leq 100$ .

A pontok további 25%-a szerezhető olyan tesztekre, ahol  $N \leq 10\,000$ .

## Tom és Jerry 1

Tom a macska, és Jerry az egér egy labirintusban él, amely csomópontjait különböző szélességű (1 vagy 2) járatok kötik össze. Jerry bármely szélességű járatban tud menni, Tom azonban csak a 2 szélességűekben. Azonos sebességgel haladnak. Ismerjük, hogy kezdetben hol van Tom és hol van Jerry, valamint azt is, hogy hol található az egérluk, amiben Jerry el tud bújni Tom elől. Jerry-nek ide kell eljutnia. Jerry több helyről is próbálkozik. Tom mindig ugyanarról a helyről indul, és ezt Jerry is tudja. Tom látja Jerry-t, de Jerry nem látja, hogy Tom merre mozog, tehát olyan útvonalon kell haladnia, amelynek csomópontjaiba mindenképpen Tom előtt tud odaérni.

Készíts programot, amely az egyes kezdőhelyekre megadja, hogy Jerry el tud-e jutni az egérlukhoz, úgy, hogy Tom biztosan nem kapja el!

### Bemenet

A *standard bemenet* első sorában a csomópontok száma ( $1 \leq N \leq 100\,000$ ), a járatok száma ( $1 \leq M \leq 100\,000$ ), Tom pozíciója ( $1 \leq T \leq N$ ), Jerry próbálkozásainak száma ( $1 \leq P \leq 10\,000$ ), valamint az egérluk pozíciója ( $1 \leq E \leq N$ ) van. A következő  $M$  sorban az egyes járatok két végpontjának sorszáma ( $1 \leq A_i \neq B_i \leq N$ ) és a járat szélessége ( $1 \leq S_i \leq 2$ ) szerepel. A következő  $P$  sorban Jerry egy-egy kezdőpozíciója szerepel ( $1 \leq K_i \neq T \leq N$ ).

### Kimenet

A *standard kimenet*  $P$  sorába az IGEN vagy a NEM szót kell írni! Az  $i$ . sorban IGEN szerepeljen, ha az  $i$ . kezdőpozícióból Jerry el tud jutni az egérlukig úgy, hogy közben biztosan nem találkozik Tom-mal, egyébként pedig a NEM szó.

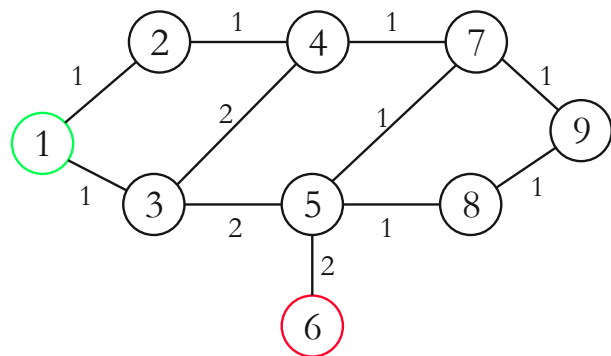
### Példa

Bemenet

```
9 11 6 3 1
1 2 1
1 3 1
2 4 1
3 4 2
3 5 2
4 7 1
3 5 2
5 6 2
6 8 1
7 9 1
8 9 1
7
8
9
```

Kimenet

```
IGEN
NEM
IGEN
```



### Korlátok

Időlimit: 0.2 mp.

Memórialimit: 32 MB

### Pontozás

A pontok 20%-a szerezhető olyan tesztekre, ahol  $N \leq 100$  és  $P \leq 10$ .

A pontok további 40%-a szerezhető olyan tesztekre, ahol  $P \leq 10$ .