



## Hálózati vibe (vibingnetwork)

Van egy belsős poén a rendszergazdák csatornáján: „Ha az eszközök legalább 67%-a elérhető, a rendszer vibe-ol.”

IoT eszközöket figyeltek meg  $T$  percen keresztül. Minden percben látjátok, hogy az  $N$  eszközből mennyi elérhető. Szeretnétek megtalálni az **első pillanatot**, amikor a hálózat vibe-olni kezd, és  $K$  percen keresztül folyamatosan így marad.

Az  $i$ . perc akkor **megfelelő**, ha az eszközök legalább 67%-a elérhető, azaz

$$a_i \cdot 100 \geq 67 \cdot N$$

teljesül, ahol  $a_i$  az  $i$ . percben elérhető eszközök száma.

**Feladat.**  $N$ ,  $T$ ,  $K$ , és az  $a_1, a_2, \dots, a_T$  értékek ismeretében keressd meg a **legkorábbi** (1-alapú)  $i$  indexet, amire teljesül, hogy az  $i, i+1, \dots, i+K-1$  percek mind megfelelőek. Ha nem létezik ilyen  $i$ , akkor a válasz  $-1$ .

Az értékelő rendszerből letölthető csatolmányok közt találhatsz **vibingnetwork.\*** nevű fájlokat, melyek a bemeneti adatok beolvasását valósítják meg az egyes programnyelveken. A megoldásodat ezekből a hiányos minta implementációkból kiindulva is elkészítheted.

## Bemenet

Az első sor az  $N$ ,  $T$  és  $K$  egész számokat tartalmazza: az eszközök számát, a megfigyelt percek számát, és a minimális sorozat hosszát.

A második sor  $T$  számot tartalmaz, a működő eszközök  $a_1, a_2, \dots, a_T$  számát az egyes percekben.

## Kimenet

A kimenet egyetlen egész szám: a legkorábbi  $i$  index úgy, hogy  $i, i+1 \dots i+K-1$  percekben a hálózat vibe-ol, vagy  $-1$ , ha nem létezik ilyen index.

## Korlátok

- $1 \leq N \leq 1\,000\,000$ .
- $1 \leq T \leq 200\,000$ .
- $1 \leq K \leq T$ .
- $0 \leq a_i \leq N$  minden  $i = 1 \dots T$ -re.

**Megjegyzés.** Lebegőpontos műveletek használata helyett egész számokkal vizsgáld a 67%-os határt. Az  $a_i \cdot 100 \geq 67 \cdot N$  egyenlőtlenség ekvivalens az  $a_i \geq \left\lceil N \cdot \frac{67}{100} \right\rceil$ -al.

## Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely

az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

- **0. Részfeladat** (0 pont)      Példák.  

- **1. Részfeladat** (15 pont)       $T \leq 2000$ .  

- **2. Részfeladat** (20 pont)       $K = 1$ .  

- **3. Részfeladat** (65 pont)      Nincsenek további megkötések.  


## Példák

| input                 | output |
|-----------------------|--------|
| 10 6 3<br>6 7 8 7 7 5 | 2      |
| 4 5 2<br>3 2 3 2 3    | -1     |

## Magyarázat

Az **első példában**  $N = 10$ , aminek a 67%-a 6.7, így egy perc akkor, és csak akkor megfelelő, ha legalább 7 eszköz elérhető.

Az elérhető eszközök száma percenként:  $[6, 7, 8, 7, 7, 5]$ .

A percek „megfelelősége”:  $[rossz, jó, jó, jó, jó, rossz]$ .

$K = 3$  egymás utáni megfelelő percre van szükségünk. A legkorábbi ilyen sorozat a második percben kezdődik (hiszen a 2., 3., 4. percek mind megfelelnek).

A **második példában**  $N = 4$ . Ahhoz, hogy egy perc megfeleljen,  $a_i \cdot 100 \geq 67 \cdot 4 = 268$  kell, amit csak 3, vagy annál nagyobb  $a_i$ -k elégítenek ki.

Az elérhető eszközök száma percenként:  $[3, 2, 3, 2, 3]$ .

Csak az 1., 3. és 5. percek megfelelőek, így nem fordul elő  $K = 2$  egymást követő megfelelő perc. Így a válasz -1.