# Átülés

Egy tanteremben N diák számára van szék, mindenkihez hozzárendelve a sajátja (az X sorszámú diák széke az X sorszámú szék). A diákok ezt sajnos nem tudták és leültek valahova. Behozunk egy további széket (ami üres) és a következő szabályt hozzuk: azon diákok közül, akik nem a saját székükön ülnek, egy feláll és átül az egyetlen üres székre.

Készíts programot, amely megadja, hogy minimum hány lépés kell, hogy mindenki a saját helyére kerüljön!

## Bemenet

A *standard bemenet* első sorában a diákok száma van (1≤N≤100000). A következő sorban az egyes székeken ülő diákok sorszámai találhatók (1≤Di≤N).

## Kimenet

A *standard kimenet* első sorába a minimális lépésszámot kell írni, amivel mindenki a helyére kerülhet!

## Példa

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 5 2 4 5 1 3 | 7 |
| Egy lehetséges megoldás: az első diák átül 4-ről 6-ra. A negyedik diák átül 2-ről 4-re. A második diák átül 1-ről 2-re. Az első diák átül 6-ról 1-re. Ezután a harmadik diák átül 5-ől 6-ra, az ötödik átül 3-ról 5-re, majd a harmadik átül 6-ról 3-ra. | |

## Korlátok

Időlimit: 0.3 mp.

Memórialimit: 32 MB

# Gabonaraktár

Egy út mentén N helyen termelnek gabonát. Ismerjük a termelők helyét és a termelt mennyiségeket. A gabona szállítási költsége a megtett út hossza szorozva a szállított mennyiséggel. Egy raktárt építhetünk valamely termelő helyre.

Készíts programot, amely megadja, hogy hol legyen a raktár, hogy a szállítás összköltsége a lehető legkisebb legyen!

## Bemenet

A *standard bemenet* első sorában a termelő helyek száma van (1≤N≤10000). A következő N sorban egy-egy termelő hely távolsága az előzőtől (1≤Ai≤100), az első mindig 0, valamint az ott termelt gabona mennyisége (1≤Bi≤1000) van.

## Kimenet

A *standard kimenet* első sorába a minimális költséget kell írni, amivel a gabona a raktárba összegyűjthető! A második sorba egy ilyen raktár sorszámát kell írni!

## Példa

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 5 0 1 3 1 10 10 5 10 1 10 | 93 4 |

## Korlátok

Időlimit: 0.1 mp.

Memórialimit: 32 MB

# Síugró verseny

*Hó Elemér* síugró versenyen vesz részt, ahol az N versenyző rajtszám szerinti sorrendben indul. Minden ugrás után megnézhetjük, hogy *Hó Elemér* hányadik helyen áll.

Készíts programot, amely megadja, hogy hányszor volt Hó Elemér az első, a második, ..., az N. helyen a verseny során!

## Bemenet

A *standard bemenet* első sorában a síugrók száma (1≤N≤100000) és *Hó Elemér* rajtszáma (1≤M≤N) van. A következő sorban az egyes versenyzők ugrására kapott pontszám található (1≤Pi≤1000), rajtszám szerinti sorrendben.

## Kimenet

A *standard kimenet* első sorába azt kell írni, hogy hányszor volt *Hó Elemér* az első, a második, ..., az N. helyen a verseny során!

## Példa

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 10 3 5 8 6 3 9 2 6 6 1 8 | 0 2 5 1 0 0 0 0 0 0  Magyarázat: A harmadik és a negyedik ugrás után a második helyen állt, az 5..9. ugrás között harmadik volt (azaz 5 alkalommal), az utolsó ugrás után pedig negyedik helyen állt. |

## Korlátok

Időlimit: 0.3 mp.

Memórialimit: 32 MB

# Zsetonok[[1]](#footnote-1)

Egy kétszemélyes játékban különböző színű zsetonok vannak egy kupacba lerakva (a színeket 1 és 100 közöti egész számokkal adjuk meg). A játékosok felváltva lépnek úgy, hogy a kupac tetejéről elvesznek tetszőleges számú (de legalább 1) azonos színű zsetont. Az nyer, aki az utolsó zsetont elveszi.

Készíts programot, amely M játék esetén megadja, hogy melyikben nyerhet az első játékos!

## Bemenet

A *standard bemenet* első sorában a zsetonok száma (1≤N≤100000) és a játékok száma(1≤M≤10) van. A következő M sorban az egyes játékokbeli zsetonsorrend található (1≤Zsi≤100), a legfelsőtől kezdődően.

## Kimenet

A *standard kimenet* M sorába az IGEN szót kell írni, ha az adott játékban a kezdő játékos nyerhet, egyébként a NEM szót! A kiírt szót mindkét esetben kövesse a két játékos által optimális stratégia esetén elvett zsetonok számai, az elvétel sorrendjében!

## Példa

|  |  |
| --- | --- |
| Bemenet | Kimenet |
| 10 3 1 1 1 1 1 1 1 1 1 2 3 1 1 1 1 1 1 1 1 2 1 2 3 4 5 6 7 8 9 9 | IGEN 8 1 1 NEM 1 7 1 1 IGEN 1 1 1 1 1 1 1 1 2  Magyarázat: piros színűek az első játékos által elvett zsetonok számai. |
| Az első játékban a kezdő játékos elvesz 8 darab 1-es zsetont, a másik ezután csak a 9. helyen levő 1-est tudja elvenni, így a kezdőé lesz az utolsó zseton.  A második játékban a kezdő csak az első helyen levő 3-ast tudja elvenni, a másik a 8 darab 1-esből elvesz 7-et, mert utána a kezdő csak az utolsó 1-est veheti el, így a legalul levő 2-es a másik játékosé lesz. | |

## Korlátok

Időlimit: 0.8 mp.

Memórialimit: 32 MB

1. Kátai Zoltán feladata. [↑](#footnote-ref-1)