



Kihívás a szabadulásért

Egy börtönben 500 rab van. Egy nap a börtönőr ajánlott tett a kiszabadulásukra. Két pénzeszsákot helyezett el egy szobában, amiket A-val, illetve B-vel jelölünk. Mindkét zsák legalább 1 és legfeljebb N darab pénzérmét tartalmaz. Az A zsákban lévő érmék száma **különbözik** a B zsákban lévő érmék számától. A börtönőr ajánlatot tett a raboknak, hogy ha felismerik azt a zsákot, amelyik kevesebb pénzérmét tartalmaz, akkor kiszabadulnak.

A szoba a két zsákon kívül tartalmaz egy táblát is. Ezen bármely pillanatban egyetlen szám állhat. Kezdetben a táblán a 0 szám áll.

Ezután a börtönőr arra kéri a rabokat, hogy egyesével menjenek be a szobába. A szobába lépő rab nem tudja, hogy előtte hány társa és kik voltak már a szobában. Minden alkalommal, amikor egy rab a szobába lép, elolvassa a táblán álló számot, majd vagy az A, vagy a B zsákot választja. **Megállapítja**, hogy a választott zsákban hány pénzérmé van. Ezután a következő két **művelet** valamelyikét kell végrehajtania:

- Átírja a táblán lévő számot egy nemnegatív egész számra és elhagyja a szobát. Megjegyezzük, hogy ugyanazt a számot is felírhatja, mint ami a táblán volt. A kihívás folytatódik (hacsak nem mind az 500 rab sorra került már).
- Rámutat az egyik zsákra, amely szerinte a kevesebb érmét tartalmazza, és ezzel befejeződik a kihívás.

A börtönőr ugyanazt a rabot nem engedi be még egyszer a szobába.

A rabok nyerik a kihívást (és kiszabadulnak) ha egyikük a második művelet végrehajtásával a kevesebb érmét tartalmazó zsákot választja. Veszítenek viszont, ha bármelyikük rosszul választ zsákot, vagy mind az 500 rab volt már a szobában és egyikük sem próbálta megmondani, melyik zsákban van kevesebb.

A kihívás előtt a rabok összegyűlnek és megbeszélnek egy három részből álló közös **stratégiát**.

- Választanak egy nemnegatív x egész számot, és vállalják, hogy nem írnak ennél nagyobb számot a táblára.
- Eldöntik, hogy ha a táblán az i szám áll ($0 \leq i \leq x$), akkor a szobába belépő rab melyik zsákot vizsgálja meg.
- Eldöntik, melyik műveletet hajtja végre a rab, miután megállapította, hány érme van a választott zsákban. Pontosabban, bármely i szám áll a táblán ($0 \leq i \leq x$) és bármely j számú érme van a választott zsákban ($1 \leq j \leq N$), meghatározzák, hogy a szobában lévő rab

- melyik 0 és x közötti számot írja fel a táblára, vagy
- melyik zsákra mutasson rá.

Ha megnyerik a kihívást, akkor x nap után mindegyikük kiszabadul.

Az a feladatod, hogy kidolgozz egy nyerő stratégiát, amellyel megnyerhetik a kihívást (függetlenül attól, hogy hány érmét tartalmaznak a zsákok). A pontszámod az x értékétől függ (lásd a Részfeladatok fejezetet a részletekért).

Megvalósítás

A következő függvényt kell megvalósítanod:

```
int[][] devise_strategy(int N)
```

- N : a zsákokban lévő érmék maximális száma.
- A függvénynek egy olyan s tömböt kell visszaadni, amelynek minden eleme egy $N + 1$ elemű, egész számokat tartalmazó tömb. Az x értéke az s tömb hossza mínusz 1. Minden i -re $0 \leq i \leq x$ között, $s[i]$ tartalmazza, hogy a rab mit csináljon, ha a szobába lépésekor a táblán az i szám áll:
 1. $s[i][0]$ értéke 0 legyen, ha az A zsák tartalmát, és 1, ha a B zsák tartalmát kell megállapítania.
 2. Legyen j a vizsgált zsákban lévő érmék száma. Ekkor a rab az alábbiakat teszi:
 - Ha $s[i][j]$ értéke -1 , akkor rámutat az A zsákra, hogy szerinte az tartalmaz kevesebbet.
 - Ha $s[i][j]$ értéke -2 , akkor rámutat a B zsákra, hogy szerinte az tartalmaz kevesebbet.
 - Ha $s[i][j]$ értéke egy nemnegatív egész szám, akkor a rab ezt a számot írja a táblára. Megjegyezzük, hogy ennek értéke legfeljebb x lehet.
- A függvényt legfeljebb egyszer hívják meg.

Példa

Tekintsük a következő függvényhívást:

```
devise_strategy(3)
```

Jelölje v a rab szobába lépésekor a táblán álló számot. Az egyik helyes stratégia a következő:

- Ha $v = 0$ (ami a kezdeti érték), az A zsákot vizsgálja meg.
 - Ha ez 1 érmét tartalmaz, mutasson rá az A zsákra mint a kevesebb érmét tartalmazóra.
 - Ha ez 3 érmét tartalmaz, mutasson rá a B zsákra mint a kevesebb érmét tartalmazóra.
 - Ha ez 2 érmét tartalmaz, írjon 1-est a táblára (felülírva a 0-t).
- Ha $v = 1$, a B zsákot vizsgálja meg.

- Ha ez 1 érmét tartalmaz, mutasson rá a B zsákra mint a kevesebb érmét tartalmazóra.
- Ha ez 3 érmét tartalmaz, mutasson rá az A zsákra mint a kevesebb érmét tartalmazóra.
- Ha ez 2 érmét tartalmaz, írjon 0-t a táblára (felülírva az 1-et). Megjegyezzük, hogy ez az eset nem fordulhat elő, mert azt jelentené, hogy mindkét zsákban 2 érme van, ami nem megengedett.

Tehát a stratégiát leíró tömbként a következőt kell a függvénynek visszaadni: $[[0, -1, 1, -2], [1, -2, 0, -1]]$. A visszaadott tömb hossza 2, így az x értéke $2 - 1 = 1$.

Korlátok

- $2 \leq N \leq 5000$

Részfeladatok

1. (5 pont) $N \leq 500$, az x értéke nem lehet 500-nál nagyobb.
2. (5 pont) $N \leq 500$, az x értéke nem lehet 70-nél nagyobb.
3. (90 pont) Az x értéke nem lehet 60-nál nagyobb.

Ha bármelyik tesztetben a `devise_strategy` által visszaadott tömb nem helyes stratégiát tartalmaz, akkor a részfeladatra 0 pontot kapsz.

A 3. részfeladatban részpontszámot kaphatsz. Legyen m a részfeladat teszteteiben kapott x értékek maximuma. A részfeladatra a következő táblázat szerinti pontszámot kapod:

Feltétel	Pontszám
$40 \leq m \leq 60$	20
$26 \leq m \leq 39$	$25 + 1.5 \times (40 - m)$
$m = 25$	50
$m = 24$	55
$m = 23$	62
$m = 22$	70
$m = 21$	80
$m \leq 20$	90

Mintaértékelő

A mintaértékelő a standard bemenetről a következő formában olvas be:

- 1. sor: N
- $2 + k$. sor ($0 \leq k$): $A[k] \ B[k]$

- utolsó sor: -1

Az első és az utolsó sor kivételével mindegyik sor egy esetet tartalmaz. A $2 + k$. sorban lévő esetre k . esetként hivatkozunk. A k . esetben az A zsák $A[k]$, a B zsák pedig $B[k]$ értémet tartalmaz.

A mintaértékelő először a `devise_strategy(N)` függvényt hívja meg. Az x értéke a visszaadott tömb mérete mínusz 1. Ekkor ha a mintaértékelő úgy érzékeli, hogy a `devise_strategy` függvény által visszaadott tömb nem a Megvalósítás fejezetben lévő feltételeknek megfelelő, az alábbi hibaüzenetek egyikét írja ki és kilép:

- `s is an empty array`: s üres tömb (ami nem képvisel érvényes stratégiát).
- `s[i] contains incorrect length`: Van olyan i index ($0 \leq i \leq x$), hogy $s[i]$ hossza nem $N + 1$.
- `First element of s[i] is non-binary`: Van olyan i index ($0 \leq i \leq x$), hogy $s[i][0]$ se nem 0, se nem 1.
- `s[i][j] contains incorrect value`: Vannak olyan i, j indexek ($0 \leq i \leq x, 1 \leq j \leq N$), hogy $s[i][j]$ nem -2 és x közötti érték.

Egyébként a mintaértékelő két kimenetet készít.

Az elsőben a stratégiád alkalmazásának eredményét írja ki az alábbi formában:

- $1 + k$. sor ($0 \leq k$): a k . eset eredménye. Ha a stratégiát alkalmazva egy rab az A zsákot jelöli meg mint a kevesebb értémet tartalmazót, akkor a sorban az A karakter van. Ha a stratégiát alkalmazva egy rab a B zsákot jelöli meg mint a kevesebb értémet tartalmazót, akkor a sorban a B karakter van. Ha a stratégiát alkalmazva egyik rab sem jelöl meg zsákot, akkor a sorban az X karakter van.

Másik kimenetként a mintaértékelő az aktuális könyvtárban a `log.txt` fájlt írja a következő formában:

- $1 + k$. sor ($0 \leq k$): $w[k][0] w[k][1] \dots$

Az $1 + k$. sorban lévő sorozat a k . esetnek felel meg és azt tartalmazza, hogy milyen számokat írtak a táblára. Pontosabban, az $l + 1$. rab szobába lépésekor a táblán a $w[k][l]$ szám áll.