



Prisoner Challenge

Într-o închisoare sunt 500 de prizonieri. Într-o zi, directorul închisorii le oferă prizonierilor o șansa de a se elibera. Acesta pune doi saci cu bani într-o cameră: sacul A și, respectiv, sacul B. Fiecare sac conține între 1 și N monede, inclusiv. Numărul de monede din sacul A este **diferit** de numărul de monede din sacul B. Directorul le prezintă prizonierilor o provocare. Scopul prizonierilor este să identifice sacul cu numărul mai mic de monede.

Camera, pe lângă sacii cu bani, conține, de asemenea, o tablă. La fiecare moment de timp pe tablă va fi scris un singur număr. Inițial, numărul de pe tablă este 0.

Apoi, directorul invită prizonierii în cameră, unul câte unul. Prizonierul intrat în cameră nu cunoaște care și nici câți prizonieri au mai intrat în cameră înaintea lui. De fiecare dată când un prizonier intră în camera, acesta citește numărul scris pe tablă. După ce citește numărul, acesta alege între sacii A și B și **inspectează** sacul ales, aflând astfel numărul de monede din interiorul acestuia. Apoi, prizonierul trebuie să aleagă una dintre următoarele **acțiuni** posibile:

- Suprascrie numărul scris pe tablă cu un întreg nenegativ ales de el și iese apoi din cameră. De notat că prizonierul poate fie să schimbe, fie să lase neschimbat numărul de pe tablă. Provocarea apoi continuă (excepție cazul în care toți cei 500 de prizonieri au intrat deja în cameră).
- Alege unul dintre cei doi saci și declară că acesta are mai puține monede. Această acțiune încheie imediat provocarea.

Directorul nu va invita niciodată un prizonier care a ieșit deja din cameră înapoi în cameră pe parcursul provocării.

Se consideră că prizonierii câștigă provocarea, dacă unul dintre ei identifică corect sacul cu numărul mai mic de monede. Pe de altă parte, prizonierii pierd dacă oricare dintre ei identifică greșit sacul cu mai puține monede, sau dacă toți 500 intră în cameră, fără a încerca să identifice sacul cu mai puține monede.

Înainte ca provocarea să înceapă, prizonierii se adună în holul închisorii și decid o **strategie** comună în trei pași pentru provocarea directorului.

- Aceștia aleg un întreg nenegativ x , reprezentând cel mai mare număr pe care au dreptul să îl scrie vreodată pe tablă.
- Decid, pentru fiecare număr i posibil scris pe tablă ($0 \leq i \leq x$), ce sac va fi inspectat de către un prizonier care citește numărul i de pe tablă când intra în cameră.

- Decid ce acțiune trebuie să efectueze un prizonier intrat în cameră după ce află numărul de monede din sacul ales. Mai exact, pentru fiecare număr i posibil scris pe tablă ($0 \leq i \leq x$) și fiecare număr j posibil văzut în sacul inspectat ($1 \leq j \leq N$), aceștia decid fie
 - ce număr cuprins între 0 și x (inclusiv) trebuie scris pe tablă înainte de a ieși din cameră, fie
 - ce sac trebuie identificat ca fiind acela cu mai puține monede.

După ce câștigă provocarea, directorul îi va elibera pe prizonieri după ce își mai ispașesc încă x zile din pedeapsă.

Sarcina voastră este să alcătuiți o strategie pentru prizonieri, care să garanteze faptul că aceștia câștigă provocarea (indiferent de numărul de monede din sacii A și B). Punctajul acordat soluției voastre va depinde de valoarea lui x (mai multe detalii în secțiunea Subtask-uri).

Detalii de Implementare

Veti implementa urmatoarea procedură.

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

- N : numărul maxim posibil de monede prezente în fiecare sac.
- Procedura va returna un tablou unidimensional s cu elemente tablouri unidimensionale formate din câte $N + 1$ numere intregi, reprezentând strategia alcătuită. Valoarea lui x va fi lungimea lui din s , minus 1. Pentru fiecare număr i astfel încât $0 \leq i \leq x$, tabloul $s[i]$ reprezintă ce trebuie un prizonier să facă, dacă vede scris numărul i pe tablă când intra în cameră:
 1. Valoarea $s[i][0]$ este 0 dacă prizonierul trebuie să inspecteze sacul A, și 1 dacă prizonierul trebuie să inspecteze sacul B.
 2. Fie j numărul de monede văzute în sacul ales. Prizonierul va efectua apoi următoarea acțiune:
 - Dacă valoarea $s[i][j]$ este -1 , prizonierul va identifica sacul A ca având mai puține monede.
 - Dacă valoarea $s[i][j]$ este -2 , prizonierul va identifica sacul B ca având mai puține monede.
 - Dacă valoarea $s[i][j]$ este un intreg nenegativ, prizonierul va scrie acest număr pe tablă. De notat că $s[i][j]$ trebuie să fie cel mult x .
- Procedura este apelată exact o singură dată.

Exemplu

Se consideră următorul apel.

```
devise_strategy(3)
```

Fie v numărul citit de către prizonier la intrarea în cameră. Atunci, o strategie corectă posibilă ar fi următoarea:

- Dacă $v = 0$ (inclusiv numărul inițial), inspectează sacul A.
 - Dacă conține o singură monedă, identifică sacul A ca având mai puține monede.
 - Dacă conține 3 monede, identifică sacul B ca având mai puține monede.
 - Dacă conține 2 monede, scrie numărul 1 pe tablă (suprascriind 0).
- Dacă $v = 1$, inspectează sacul B.
 - Dacă conține o singură monedă, identifică sacul B ca având mai puține monede.
 - Dacă conține 3 monede, identifică sacul A ca având mai puține monede.
 - Dacă conține 2 monede, scrie numărul 0 pe tablă (suprascriind 1). De notat că acest caz nu se poate, de fapt, întâmpla, deoarece s-ar deduce că ambii saci conțin câte 2 monede, acest lucru nefiind permis.

Pentru a raporta aceasta strategie, procedura returnează $[[0, -1, 1, -2], [1, -2, 0, -1]]$. Lungimea tabloului returnat este 2, deci valoarea lui x în acest caz este $2 - 1 = 1$.

Restricții

- $2 \leq N \leq 5000$

Subtask-uri

1. (5 puncte) $N \leq 500$, valoarea x nu trebuie să depășească 500.
2. (5 puncte) $N \leq 500$, valoarea x nu trebuie să depășească 70.
3. (90 puncte) Valoarea x nu trebuie să depășească 60.

Dacă în oricare dintre testele asociate unui subtask tabloul returnat de procedura `devise_strategy` nu reprezintă o strategie corectă, punctajul acordat soluției pentru subtask-ul respectiv va fi 0.

Pentru subtask-ul 3 se poate obține punctaj parțial. Fie m valoarea maximă a lui x pentru tablourile returnate pentru toate testele din subtask. Scorul acordat pentru subtask se calculează conform următorului tabel:

Condiție	Puncte
$40 \leq m \leq 60$	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

Grader-ul Local

Grader-ul local citește intrarea în următorul format:

- linia 1: N
- linia $2 + k$ ($0 \leq k$): $A[k] B[k]$
- ultima linie: -1

Fiecare linie, excepție prima și ultima, reprezintă un scenariu. Ne vom referi la scenariul descris pe linia $2 + k$ ca scenariul k . În scenariul k sacul A conține $A[k]$ monede și sacul B conține $B[k]$ monede.

Grader-ul local apelează prima dată `devise_strategy(N)`. Valoarea lui x este lungimea tabloului returnat de apel, minus 1. Apoi, dacă grader-ul local depistează că tabloul returnat de `devise_strategy` nu este conform cu constrângerile descrise în Detalii de Implementare, acesta afișează unul dintre următoarele mesaje de eroare și apoi termină execuția:

- `s is an empty array`: tabloul s este gol (și astfel nu reprezintă o strategie validă).
- `s[i] contains incorrect length`: Există un indice i ($0 \leq i \leq x$) astfel încât lungimea tabloului $s[i]$ nu este $N + 1$.
- `First element of s[i] is non-binary`: Există un indice i ($0 \leq i \leq x$) astfel încât $s[i][0]$ nu este nici 0, nici 1.
- `s[i][j] contains incorrect value`: Există doi indici i, j ($0 \leq i \leq x, 1 \leq j \leq N$) astfel încât $s[i][j]$ nu este cuprins între -2 și x .

Altfel, grader-ul local produce doua ieșiri.

Mai întâi, grader-ul local afișează rezultatul aplicării strategiei produse de programul vostru, în următorul format:

- linia $1 + k$ ($0 \leq k$): rezultatul aplicării strategiei în scenariul k . Dacă aplicarea strategiei duce ca un prizonier să identifice sacul A ca având mai puține monede, atunci rezultatul este caracterul A. Dacă aplicarea strategiei duce ca un prizonier să identifice sacul A ca având mai puține monede, atunci rezultatul este caracterul B. Dacă aplicarea strategiei face ca niciun prizonier să nu identifice sacul cu mai puține monede, atunci rezultatul este caracterul X.

Apoi, grader-ul local crează un fișier `log.txt` în directorul curent, al cărui conținut are următorul format:

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

Secvența de pe linia $1 + k$ corespunde scenariului k și descrie numerele scrise pe tablă. Mai exact, $w[k][l]$ este numărul scris de către al $(l + 1)$ -lea prizonier intrat în cameră.