

Učiteľská hra

Na vychýrenom, bližšie nešpecifikovanom, bratislavskom gymnáziu sa jeden bližšie nešpecifikovaný (ale inak veľmi pohľadný) učiteľ rozhodol zorganizovať medzi učiteľmi a žiakmi hru. Na začiatku sa do učebne posadilo N žiakov, ktorí boli očíslovaní od 0 po $N - 1$. Každý žiak mal v ruke papier, na ktorom bolo napísané pole celých čísel. Na začiatku boli všetky papiere prázdne, obsahovali teda prázdne pole.

Do hry sa zapojilo aj M učiteľov, ktorí sú očíslovaní od 0 po $M - 1$, a navyše aj riaditeľka. Hra má M kôl, ktoré sú tiež očíslované od 0 po $M - 1$. V kole j vstúpi do učebne učiteľ j a postupne sa stane nasledovné:

- Niekoľko (potenciálne nula) žiakov zdvihne ruku. Je zaručené, že v každom kole bude aspoň jeden žiak, ktorý ruku **nezdvihne**. Tiež je zaručené, že každý žiak zdvihne ruku **v nanajvýš jednom kole**.
- Učiteľ sa môže pozrieť na všetky papiere, ktoré majú žiaci v rukách. Následne môže **zmeniť** obsah papierov tým žiakom, ktorí sa **neprihlásili**. Každému žiakovi môže učiteľ vygumovať obsah jeho papiera a nahradiť ho novým (potenciálne prázdny) poľom čísel. Čísla v tomto poli musia byť celé čísla od 0 po 63 vrátane a pole na papieri môže mať nanajvýš 63 prvkov.
- Keď učiteľ j spraví všetky modifikácie, ktoré chcel, odíde z učebne. Následne si žiaci povymieňajú svoje papiere na základe **tajnej** permutácie P . Teda, žiak i (pre $0 \leq i < N$) podá svoj papier žiakovi $P[i]$. Platí, že $P[0], P[1], \dots, P[N - 1]$ je permutácia čísel 0 až $N - 1$. Táto permutácia je fixná a počas hry sa nemení. Učitelia ani riaditeľka túto permutáciu nepoznajú.

Po úplnom odohraní všetkých M kôl (vrátane posunu papierov po poslednom z nich) vjde do učebne riaditeľka. Na základe polí, ktoré majú žiaci napísané na svojich papieroch, musí pre každého žiaka určiť, v ktorom kole zdvihol ruku, prípadne, že daný žiak ruku počas celej hry nezdvihol.

Učitelia a riaditeľka si pred hrou môžu dohodnúť spoločnú stratégiu. Následne však hrajú nezávisle od seba. Po začatí hry nemôžu učitelia komunikovať medzi sebou ani s riaditeľkou. Ich jediný spôsob komunikácie je teda prostredníctvom čísel, ktoré napíšu na žiacke papiere. Každý učiteľ vie, v ktorom kole vstupuje do učebne.

Vašou úlohou je navrhnúť a implementovať stratégiu pre učiteľov a riaditeľku, pomocou ktorej dokáže riaditeľka správne určiť, kedy zdvihli ktorí žiaci ruky. Výsledné skóre za túto úlohu bude závisieť od dĺžky najdlhšieho poľa napísaného na papieri kedykoľvek počas hry – čím bude kratšie, tým viac bodov môžete získať. (Pre detaily pozrite časť Podúlohy a bodovanie.)

Implementačné detaily

Implementujte dve funkcie: jednu simulujúcu hru učiteľov, druhú simulujúcu riaditeľku.

Hru **učiteľa** realizuje funkcia:

```
std::vector<std::vector<int>> process_step(  
    int N, int M, int R,  
    std::vector<int> T,  
    std::vector<std::vector<int>> A)
```

- N : počet žiakov.
- M : počet učiteľov a tiež počet kôl hry.
- R : aktuálne kolo hry (medzi 0 až $M - 1$).
- T : pole (potenciálne prázdne) obsahujúce indexy žiakov, ktorí v tomto kole zdvihli ruku. Toto pole je zoradené v rastúcom poradí.
- A : pole dĺžky N popisujúce žiacke papiere. Prvok $A[i]$ ($0 \leq i < N$) je pole čísel na papieri žiaka i na začiatku aktuálneho kola.
- Funkcia je pre jednu hru zavolaná presne M -krát, postupne s hodnotami $R = 0, 1, \dots, M - 1$ v tomto poradí.

Táto funkcia má na výstup vrátiť pole B dĺžky N , ktoré reprezentuje žiacke papiere po modifikácii daným učiteľom. Formát poľa B má byť rovnaký ako formát A . Navyše musí platiť:

- Pre každého žiaka i , ktorý v danom kole zdvihol ruku (teda i je prvkom poľa T) je papier nezmenený, teda $B[i] = A[i]$.
- Pre každého žiaka i ($0 \leq i < N$) dĺžka poľa $B[i]$ nepresahuje 63 prvkov a hodnoty v poli $B[i]$ sú medzi 0 a 63 (vrátane).

Pre **riaditeľku** implementujte nasledujúcu funkciu:

```
std::vector<int> determine_steps(  
    int N, int M, std::vector<std::vector<int>> A)
```

- N, M : počet žiakov a počet učiteľov.
- A : pole dĺžky N , kde $A[i]$ reprezentuje papier žiaka i po úplnom odohratí všetkých M kôl.
- Táto funkcia je pre jednu hru zavolaná raz, a to až po poslednom volaní funkcie `process_step`.

Táto funkcia má vrátiť pole D dĺžky N . Pre každé i ($0 \leq i < N$) musí platiť:

- $D[i] = j$ ak žiak i zdvihol ruku počas kola j ;
- $D[i] = -1$ ak žiak i nezdvihol ruku počas celej hry.

Váš program si nesmie ukladať alebo odovzdávať informácie medzi jednotlivými volaniami funkcie `process_step` inak ako cez výstupnú hodnotu tejto funkcie. V prípade, že váš program bude komunikovať nedovoleným spôsobom, body v CMS môžu byť zobrazované nesprávne a po konci súťaže môžu byť znížené.

Testovač môže spustiť viacero inštancií vášho programu naraz. To znamená, že volania funkcie `process_step` a `determine_steps` jednej hry sa môžu spustiť v rôznych inštanciách a naopak, v rámci jednej inštancie môžu byť spustené funkcie `process_step` a `determine_steps` z rôznych hier.

Každý testovací vstup obsahuje nanajvýš 5 hier.

Obmedzenia

- $2 \leq N \leq 63$
- $1 \leq M \leq 63$
- Každý žiak môže zdvihnúť ruku **najviac raz** počas celej hry. To znamená, že pre každého žiaka i existuje nanajvýš jedno kolo j , v ktorom zdvihol ruku.
- V každom kole aspoň jeden žiak ruku **nezdvihne**.

Podúlohy a bodovanie

Podúloha	Body	Dodatočné obmedzenia
1	4	$M = 1$
2	6	$N = 2$
3	9	$P[i] = i$ pre všetky $0 \leq i \leq N - 1$.
4	25	V každom kole zdvihne ruku nanajvýš jeden žiak.
5	56	Bez dodatočných obmedzení.

Za vstup dostanete 0 bodov a verdikt `Output isn't correct` v CMS, ak ľubovoľné volanie funkcie `process_step` vráti pole v nesprávnom formáte alebo ak je výsledok volania funkcie `determine_steps` nesprávny.

Inak, nech C je maximálna dĺžka poľa na žiackom papieri vráteného v B v **ľubovoľnom** volaní funkcie `process_step`. Nech S je maximálny počet bodov udeľovaný za danú podúlohu. Výsledné skóre je potom vypočítané ako $S \cdot X$, kde X je hodnota určená hodnotou C na základe nasledujúcej tabuľky:

C	X
$C \leq 2$	1.00
$C = 3$	0.75
$C = 4$	0.55
$5 \leq C \leq 13$	$0.50 - 0.03 \cdot (C - 5)$
$14 \leq C \leq 63$	$0.19 \cdot \frac{64-C}{64-14} + 0.04$

Ukážkový príklad

Predstavme si situáciu, kde máme $N = 4$ žiakov, $M = 2$ učiteľov a tajná permutácia je $P = [0, 3, 1, 2]$. Na začiatku sú všetky papiere prázdne.

Testovač spraví volanie:

```
process_step(4, 2, 0, [1], [[], [], [], []])
```

Žiak 1 zdvihol ruku, jeho papier sa teda meniť nebude. Učiteľ 0 sa rozhodol napísať pole $[10]$ na papier žiaka 0, pole $[1, 63, 4]$ na papier žiaka 3 a nechať papier žiaka 2 prázdny. Výstupom tejto funkcie bude preto $B = [[10], [], [], [1, 63, 4]]$.

Keď učiteľ 0 odíde z učebne, žiaci si vymenia svoje papiere na základe permutácie P . Po výmene sú ich papiere popísané poľom $A = [[10], [], [1, 63, 4], []]$.

Testovač ďalej zavolá:

```
process_step(4, 2, 1, [0, 3], [[10], [], [1, 63, 4], []])
```

Žiaci 0 a 3 zdvihli ruky, ich papiere sa teda meniť nebudú. Učiteľ 1 sa rozhodne napísať pole $[0, 40]$ na papier žiaka 1 a napísať $[1, 50]$ na papier žiaka 2. Výstupom tejto funkcie bude preto $B = [[10], [0, 40], [1, 50], []]$.

Keď učiteľ 1 odíde, žiaci si znova vymenia svoje papiere na základe permutácie P . Po tejto výmene sú ich papiere popísané poľom $A = [[10], [1, 50], [], [0, 40]]$.

Nakoniec testovač zavolá:

```
determine_steps(4, 2, [[10], [1, 50], [], [0, 40]])
```

Výstupom tejto funkcie musí byť pole $D = [1, 0, -1, 1]$, keďže žiaci 0 a 3 zdvihli ruku v kole 1, žiak 1 zdvihol ruku v kole 0 a žiak 2 počas hry ruku nezdvihol. V tomto prípade je $C = 3$, keďže počas kola

0 učiteľ zapísal na jeden z papierov trojprvkové pole.

Ukážkový testovač

Formát vstupu:

```
N M
Q[0] Q[1] ... Q[N-1]
P[0] P[1] ... P[N-1]
```

Pole Q má dĺžku N a popisuje, kedy žiaci zdvihnú ruky. Konkrétne $Q[i] = j$ ak žiak i zdvihne ruku v kole j a $Q[i] = -1$ ak žiak ruku počas celej hry nezdvihne.

Formát výstupu:

Po každom zavolaní funkcie `process_step` vypíše ukážkový testovač obsah žiackych papierov. K označuje dĺžku poľa B a $L[i]$ označuje dĺžku poľa $B[i]$ (pre $0 \leq i < K$). Ukážkový testovač vypíše B v nasledovnom formáte, za ním nasleduje **prázdny riadok**.

```
K
L[0] B[0][0] B[0][1] ... B[0][L[0]-1]
L[1] B[1][0] B[1][1] ... B[1][L[1]-1]
:
L[K-1] B[K-1][0] B[K-1][1] ... B[K-1][L[K-1]-1]
```

Ukážkový testovač potom povymieňa papiere podľa určenej permutácie P . Ak by sa $K \neq N$, ukážkový testovač vypíše chybu a skončí.

Po zavolaní `determine_steps`, ukážkový testovač vypíše:

```
C H
D[0] D[1] ... D[H-1]
```

H je dĺžka poľa D , ktoré vrátila funkcia `determine_steps`.