



Seats

Organizați un concurs internațional de programare într-un hol dreptunghiular cu HW locuri aranjate în H rânduri și W coloane. Rândurile sunt numerotate de la 0 la $H - 1$ și coloanele de la 0 la $W - 1$. Locul din rândul r și coloana c este notat (r, c) . Ați invitat HW concurenți, numerotați de la 0 la $HW - 1$. Ați elaborat de asemenea o schemă de amplasare care atribuie concurentul cu numărul i ($0 \leq i \leq HW - 1$) locului (R_i, C_i) . Schema atribuie exact câte un concurent fiecărui loc.

O mulțime de locuri din holul S este **dreptunghiulară** dacă există numerele întregi r_1 , r_2 , c_1 , și c_2 care satisfac următoarele condiții:

- $0 \leq r_1 \leq r_2 \leq H - 1$.
- $0 \leq c_1 \leq c_2 \leq W - 1$.
- S este exact mulțimea tuturor locurilor (r, c) astfel încât $r_1 \leq r \leq r_2$ și $c_1 \leq c \leq c_2$.

O mulțime dreptunghiulară formată din k ($1 \leq k \leq HW$) locuri se consideră **frumoasă** dacă concurenții, cărora le sunt atribuite locurile din mulțime, sunt numerotați de la 0 la $k - 1$. **Gradul de frumusețe** al schemei de amplasare este numărul mulțimilor dreptunghiulare frumoase din schemă.

După elaborarea schemei de amplasare ați primit câteva cereri de interschimbare a locurilor unor perechi de concurenți. Mai exact, sunt Q astfel de cereri, numerotate de la 0 la $Q - 1$ în ordine cronologică. Cererea j ($0 \leq j \leq Q - 1$) este de a interschimba locurile asociate concurenților A_j și B_j . Fiecare cerere se prelucrează imediat și se actualizează schema de amplasare. După fiecare actualizare scopul este să calculați gradul de frumusețe al schemei de amplasare curente.

Detalii de implementare

Trebuie să implementați următoarea procedură și funcție:

```
give_initial_chart(int H, int W, int[] R, int[] C)
```

- H, W : numărul de rânduri și numărul de coloane.
- R, C : tablouri unidimensionale cu HW elemente reprezentând schema de amplasare inițială.
- Aastă procedură este apelată o singură dată, înainte de orice apel al funcției `swap_seats`.

```
int swap_seats(int a, int b)
```

- Această funcție descrie o cerere de interschimbare a două locuri.
- a, b : concurenții a căror locuri urmează a fi interschimbate.
- Această funcție este apelată de Q ori.
- Această funcție va întoarce gradul de frumusețe al schemei de amplasare după interschimbare.

Exemplu

Fie $H = 2$, $W = 3$, $R = [0, 1, 1, 0, 0, 1]$, $C = [0, 0, 1, 1, 2, 2]$, și $Q = 2$.

Grader-ul va apela mai întâi `give_initial_chart(2, 3, [0, 1, 1, 0, 0, 1], [0, 0, 1, 1, 2, 2])`.

Inițial, schema de amplasare este următoarea:

0	3	4
1	2	5

Să presupunem că grader-ul apelează `swap_seats(0, 5)`. După cererea 0, schema de amplasare este următoarea:

5	3	4
1	2	0

Mulțimile de locuri care corespund concurenților $\{0\}$, $\{0, 1, 2\}$, și $\{0, 1, 2, 3, 4, 5\}$ sunt dreptunghiulare și frumoase. Astfel, gradul de frumusețe al schemei de amplasare este 3, și `swap_seats` va returna 3.

Să zicem că grader-ul apelează `swap_seats(0, 5)` din nou. După cererea 1, schema de amplasare revine la starea inițială. Mulțimile de locuri care corespund concurenților $\{0\}$, $\{0, 1\}$, $\{0, 1, 2, 3\}$, și $\{0, 1, 2, 3, 4, 5\}$ sunt dreptunghiulare și frumoase. Prin urmare, gradul de frumusețe al schemei de amplasare este 4, și `swap_seats` va întoarce 4.

Fișierele `sample-01-in.txt` și `sample-01-out.txt` în pachetul arhivat anexat corespund acestui exemplu. Pachetul conține și alte exemple de intrări/ieșiri.

Restricții

- $1 \leq H$
- $1 \leq W$
- $HW \leq 1\,000\,000$
- $0 \leq R_i \leq H - 1$ ($0 \leq i \leq HW - 1$)
- $0 \leq C_i \leq W - 1$ ($0 \leq i \leq HW - 1$)
- $(R_i, C_i) \neq (R_j, C_j)$ ($0 \leq i < j \leq HW - 1$)
- $1 \leq Q \leq 50\,000$
- $0 \leq a \leq HW - 1$ pentru orice apel `swap_seats`
- $0 \leq b \leq HW - 1$ pentru orice apel `swap_seats`
- $a \neq b$ pentru orice apel `swap_seats`

Subtask-uri

1. (5 puncte) $HW \leq 100$, $Q \leq 5\,000$
2. (6 puncte) $HW \leq 10\,000$, $Q \leq 5\,000$
3. (20 puncte) $H \leq 1\,000$, $W \leq 1\,000$, $Q \leq 5\,000$
4. (6 puncte) $Q \leq 5\,000$, $|a - b| \leq 10\,000$ pentru orice apel `swap_seats`
5. (33 puncte) $H = 1$
6. (30 puncte) Fără constrângeri adiționale.

Grader local

Grader-ul local citește datele de intrare în următoarea formă:

- linia 1: $H W Q$
- linia $2 + i$ ($0 \leq i \leq HW - 1$): $R_i C_i$
- linia $2 + HW + j$ ($0 \leq j \leq Q - 1$): $A_j B_j$

Aici, A_j și B_j sunt parametrii pentru apelul `swap_seats` în cererea j .

Grader-ul local afișează răspunsurile în următoarea formă:

- linia $1 + j$ ($0 \leq j \leq Q - 1$): valoarea întoarsă de `swap_seats` pentru cererea j