



Postazioni

Devi organizzare una competizione internazionale di programmazione in una sala rettangolare con HW postazioni disposte su H righe e W colonne (numerata rispettivamente da 0 a $H - 1$ e da 0 a $W - 1$). La postazione in riga r e colonna c è indicata come (r, c) . Hai invitato HW atleti, numerati da 0 a $HW - 1$, e hai preparato una mappa delle postazioni che assegna l'atleta i ($0 \leq i \leq HW - 1$) alla postazione (R_i, C_i) , in modo che ad ogni posto sia assegnato esattamente un atleta.

Un insieme di postazioni S viene detto **rettangolare** se esistono interi r_1, r_2, c_1, c_2 tali per cui:

- $0 \leq r_1 \leq r_2 \leq H - 1$,
- $0 \leq c_1 \leq c_2 \leq W - 1$, ed
- S coincide con l'insieme delle postazioni (r, c) tali che $r_1 \leq r \leq r_2$ e $c_1 \leq c \leq c_2$.

Inoltre, un insieme rettangolare di k posti ($1 \leq k \leq HW$) si dice **bello** quando gli atleti che contiene sono esattamente quelli numerati da 0 a $k - 1$. La **bellezza** di una mappa di postazioni è quindi definita come il numero di insiemi rettangolari e belli di postazioni nella mappa.

Dopo aver preparato la mappa, hai ricevuto Q diverse richieste di scambio tra i posti assegnati a due atleti diversi. Le richieste sono numerate da 0 a $Q - 1$ in ordine cronologico, e la j -esima richiesta ($0 \leq j \leq Q - 1$) corrisponde allo scambio di posto tra gli atleti A_j e B_j . Ogni richiesta viene immediatamente accettata e la mappa aggiornata di conseguenza. Calcola, di volta in volta, la bellezza della nuova mappa che si ottiene dopo ciascuno di questi aggiornamenti.

Dettagli implementativi

Devi implementare la seguente funzione:

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

- H, W : il numero di righe e colonne.
- R, C : array di lunghezza HW che rappresentano le coordinate dei posti inizialmente assegnati a ciascuno studente.
- Questa funzione è chiamata esattamente una volta, prima di tutte le chiamate a `swap_seats`.

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

- Questa funzione rappresenta una richiesta di scambio tra due postazioni.
- a, b: gli atleti che si vogliono scambiare.
- Questa funzione viene chiamata Q volte, e deve restituire la bellezza della mappa dopo lo scambio.

Esempio

Siano $H = 2$, $W = 3$, $R = [0, 1, 1, 0, 0, 1]$, $C = [0, 0, 1, 1, 2, 2]$, e $Q = 2$. Il grader inizialmente chiama `give_initial_chart(2, 3, [0, 1, 1, 0, 0, 1], [0, 0, 1, 1, 2, 2])`. All'inizio, la mappa delle postazioni è la seguente.

0	3	4
1	2	5

Poniamo che il grader chiami `swap_seats(0, 5)`. Dopo questa richiesta (numero 0), la mappa diventa:

5	3	4
1	2	0

Gli insiemi di posti corrispondenti agli atleti $\{0\}$, $\{0, 1, 2\}$, e $\{0, 1, 2, 3, 4, 5\}$ sono rettangolari e belli. Quindi, la bellezza di questa mappa è 3 e `swap_seats` dovrebbe restituire 3.

Poniamo ora che il grader chiami `swap_seats(0, 5)` nuovamente. Dopo questa richiesta (numero 1), la mappa ritorna allo stato iniziale. Gli insiemi di posti corrispondenti agli atleti $\{0\}$, $\{0, 1\}$, $\{0, 1, 2, 3\}$, e $\{0, 1, 2, 3, 4, 5\}$ sono rettangolari e belli. Quindi, la bellezza di questa mappa è 4 e `swap_seats` dovrebbe restituire 4.

I file `sample-01-in.txt` e `sample-01-out.txt` nell'archivio compresso in allegato corrispondono a questo esempio. Altri input/output di esempio sono inoltre disponibili in questo archivio.

Assunzioni

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

Subtask

1. (5 punti) $HW \leq 100$, $Q \leq 5\,000$
2. (6 punti) $HW \leq 10\,000$, $Q \leq 5\,000$
3. (20 punti) $H \leq 1\,000$, $W \leq 1\,000$, $Q \leq 5\,000$
4. (6 punti) $Q \leq 5\,000$, $|a - b| \leq 10\,000$ in ogni chiamata a `swap_seats`
5. (33 punti) $H = 1$
6. (30 punti) Nessuna limitazione aggiuntiva

Grader di esempio

Il grader di esempio legge l'input nel seguente formato:

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

dove A_j e B_j sono i parametri della chiamata j -esima a `swap_seats`.

Il grader di esempio stampa l'output nel seguente formato:

- righe $1 + j$ ($0 \leq j \leq Q - 1$) : il valore che viene restituito da `swap_seats` per la richiesta j -esima