



Asientos

Una competencia internacional de programación será llevada a cabo en un salón rectangular, que tiene HW asientos distribuidos en H filas y W columnas. Las filas se numeran desde 0 hasta $H - 1$ y las columnas se numeran desde 0 hasta $W - 1$. Al asiento de la fila r y de la columna c se le denota (r, c) . Se han invitado HW participantes, numerados desde 0 hasta $HW - 1$. También se ha creado una tabla de asientos, que le asigna al participante i ($0 \leq i \leq HW - 1$) el asiento (R_i, C_i) . La tabla ubica exactamente un participante en cada asiento.

Un conjunto de asientos S se dice **rectangular** si hay enteros $r_1, r_2, c_1, y c_2$ que satisfacen las siguientes condiciones:

- $0 \leq r_1 \leq r_2 \leq H - 1$.
- $0 \leq c_1 \leq c_2 \leq W - 1$.
- S es exactamente el conjunto de todos los asientos (r, c) tales que $r_1 \leq r \leq r_2$ y $c_1 \leq c \leq c_2$.

Un conjunto rectangular de k ($1 \leq k \leq HW$) asientos es **hermoso** si los participantes cuyos asientos asignados están en el conjunto tienen los números desde 0 hasta $k - 1$. La **belleza** de una tabla de asientos es la cantidad de conjuntos rectangulares hermosos de asientos en la tabla.

Luego de confeccionar la tabla de asientos, se reciben varios pedidos de intercambiar dos asientos asignados a dos participantes. Más precisamente, hay Q pedidos numerados desde 0 hasta $Q - 1$ en orden cronológico. El pedido j ($0 \leq j \leq Q - 1$) consiste en intercambiar los asientos asignados a los participantes A_j y B_j . Cada pedido es aceptado inmediatamente, y la tabla es actualizada. Luego de cada actualización, el objetivo es calcular la belleza de la tabla de asientos actual.

Detalles de implementación

Debes implementar las siguientes funciones:

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

- H, W : la cantidad de filas y la cantidad de columnas.
- R, C : arreglos de longitud HW que representan la tabla de asientos inicial.
- Esta función se llama una única vez, antes de cualquier llamada a `swap_seats`.

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

- Esta función describe un pedido de intercambiar dos asientos.
- a, b : los participantes cuyos asientos deben ser intercambiados.
- Esta función se llamará Q veces.
- Esta función debe retornar la belleza de la tabla de asientos luego del intercambio.

Ejemplo

Sea $H = 2, W = 3, R = [0, 1, 1, 0, 0, 1], C = [0, 0, 1, 1, 2, 2]$, y $Q = 2$.

El evaluador primero realiza la llamada `give_initial_chart(2, 3, [0, 1, 1, 0, 0, 1], [0, 0, 1, 1, 2, 2])`.

Al comenzar, la tabla de asientos es la siguiente:

0	3	4
1	2	5

Digamos que el evaluador realiza la llamada `swap_seats(0, 5)`. Luego del pedido 0, la tabla de asientos es la siguiente:

5	3	4
1	2	0

Los conjuntos de asientos correspondientes a los participantes $\{0\}$, $\{0, 1, 2\}$, y $\{0, 1, 2, 3, 4, 5\}$ son rectangulares y hermosos. Por lo tanto, la belleza de esta tabla de asientos es 3, y `swap_seats` debería retornar 3.

Digamos que el evaluador vuelve a realizar la llamada `swap_seats(0, 5)`. Luego del pedido 1, la tabla de asientos vuelve al estado original. Los conjuntos de asientos correspondientes a los participantes $\{0\}$, $\{0, 1\}$, $\{0, 1, 2, 3\}$, y $\{0, 1, 2, 3, 4, 5\}$ son rectangulares y hermosos. Por lo tanto, la belleza de esta tabla de asientos es 4, y `swap_seats` debería retornar 4.

Los archivos `sample-01-in.txt` y `sample-01-out.txt` en el archivo `.zip` adjunto corresponden a este ejemplo. Hay otros ejemplos de entrada/salida disponibles en el `.zip`.

Restricciones

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

Subtareas

1. (5 puntos) $HW \leq 100$, $Q \leq 5\,000$
2. (6 puntos) $HW \leq 10\,000$, $Q \leq 5\,000$
3. (20 puntos) $H \leq 1\,000$, $W \leq 1\,000$, $Q \leq 5\,000$
4. (6 puntos) $Q \leq 5\,000$, $|a - b| \leq 10\,000$ para cualquier llamada a `swap_seats`
5. (33 puntos) $H = 1$
6. (30 puntos) Sin restricciones adicionales

Evaluador de ejemplo

El evaluador de ejemplo lee la entrada con el siguiente formato:

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

Aquí, A_j y B_j son los parámetros para la llamada a `swap_seats` para el pedido j .

El evaluador de ejemplo escribe sus respuestas con el siguiente formato:

- línea $1 + j$ ($0 \leq j \leq Q - 1$): el valor retornado por `swap_seats` para el pedido j