



Seats

Međunarodno takmičenje iz programiranja održava se u sali u kojoj su sjedišta poređana u pravougaonik sa H redova i W kolona. Redovi su označeni brojevima od 0 do $H - 1$, a kolone su uoznačene od 0 do W . Sjedište u redu r i koloni c označavamo sa (r, c) . Na takmičenju učestvuje HW takmičara koji su numerisani brojevima od 0 do $HW - 1$. Napravljen je plan sjedenja pa je takmičaru i ($0 \leq i \leq HW - 1$) dodijeljeno sjedište (R_i, C_i) tako da svaki takmičar zauzima po jedno sjedište.

Skup sjedišta S nazivamo **pravougaonim** ako postoje cijeli brojevi r_1, r_2, c_1 i c_2 takvi da važi:

- $0 \leq r_1 \leq r_2 \leq H - 1$.
- $0 \leq c_1 \leq c_2 \leq W - 1$.
- Skupu S pripadaju sva sjedišta (r, c) takva da je $r_1 \leq r \leq r_2$ i $c_1 \leq c \leq c_2$.

Za pravougaoni skup od k ($1 \leq k \leq HW$) sjedišta kažemo da je **lijep** ako su na tih k sjedišta smješteni takmičari hronološki od 0 do $k - 1$. **Ljepota** plana sjedenja se definiše kao broj lijepih pravougaonih skupova sjedišta.

Nakon objavljivanja plana sjedenja dobili ste više zahtjeva za međusobnu zamjenu mjesta dva takmičara. Preciznije, postoji Q zahtjeva koji su hronološki numerisani od 0 do $Q - 1$. U zahtjevu j ($0 \leq j \leq Q - 1$) takmičari A_j i B_j međusobno mijenjaju sjedišta. Svaki zahtjev se odmah izvršava i mijenja se plan sjedenja. Vaš zadatak je da izračunate ljepotu plana sjedenja poslije svake izmjene.

Detalji implementacije

Potrebno je implementirati sljedeće funkcije:

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

- H, W : broj redova i broj kolona.
- R, C : nizovi dužine HW koji predstavljaju početni raspored sjedenja.
- Ova funkcija se poziva tačno jednom, i to prije poziva funkcije `swap_seats`.

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

- Ova funkcija opisuje zahtjev kojim dva takmičara međusobno zamjenjuju sjedišta

- a, b: takmičari koji međusobno mijenjaju sjedišta
- Ova funkcija se poziva Q puta
- Funkcija vraća ljepotu plana sjedenja poslije ispunjavanja zahtjeva za zamjenu sjedišta dva takmičara.

Primjer

Neka je $H = 2$, $W = 3$, $R = [0, 1, 1, 0, 0, 1]$, $C = [0, 0, 1, 1, 2, 2]$, i $Q = 2$.

Program za ocjenjivanje (grader) prvo poziva `give_initial_chart(2, 3, [0, 1, 1, 0, 0, 1], [0, 0, 1, 1, 2, 2])`.

Na početku, plan sjedenja je sljedeći:

0	3	4
1	2	5

Pretpostavimo da program za ocjenjivanje poziva `swap_seats(0, 5)`. Ovo je zahtjev 0, pa poslije ispunjavanja ovog zahtjeva, plan sjedenja je sljedeći:

5	3	4
1	2	0

Sljedeći skupovi sjedišta su pravougaoni i lijepi: $\{0\}$, $\{0, 1, 2\}$ i $\{0, 1, 2, 3, 4, 5\}$. Ljepota ovog plana sjedenja je 3, pa funkcija `swap_seats` treba da vrati 3.

Pretpostavimo da sada program za ocjenjivanje (grader) još jednom poziva `swap_seats(0, 5)`. Ovo je zahtjev 1, pa poslije ispunavanja ovog zahtjeva, plan sjedenja je isti kao na početku. Sada postoje 4 pravougaona i lijepa skupa sjedišta: $\{0\}$, $\{0, 1\}$, $\{0, 1, 2, 3\}$, and $\{0, 1, 2, 3, 4, 5\}$. Dakle, ljepota ovog plana sjedenja je 4, pa funkcija `swap_seats` treba da vrati 4.

U zip-fajlu uz ovaj zadatak, fajlovi `sample-01-in.txt` i `sample-01-out.txt` odgovaraju ovom primjeru. U zip-fajlu postoje i drugi primjeri.

Ograničenja

- $1 \leq H$
- $1 \leq W$
- $HW \leq 1\,000\,000$
- $0 \leq R_i \leq H - 1$ ($0 \leq i \leq HW - 1$)
- $0 \leq C_i \leq W - 1$ ($0 \leq i \leq HW - 1$)
- $(R_i, C_i) \neq (R_j, C_j)$ ($0 \leq i < j \leq HW - 1$)
- $1 \leq Q \leq 50\,000$
- $0 \leq a \leq HW - 1$ za bilo koji poziv `swap_seats`
- $0 \leq b \leq HW - 1$ za bilo koji poziv `swap_seats`
- $a \neq b$ za bilo koji poziv `swap_seats`

Podzadaci

1. (5 bodova) $HW \leq 100$, $Q \leq 5\,000$
2. (6 bodova) $HW \leq 10\,000$, $Q \leq 5\,000$
3. (20 bodova) $H \leq 1\,000$, $W \leq 1\,000$, $Q \leq 5\,000$
4. (6 bodova) $Q \leq 5\,000$, $|a - b| \leq 10\,000$ za bilo koji poziv `swap_seats`
5. (33 boda) $H = 1$
6. (30 bodova) Nema dodatnih ograničenja

Primjer programa za ocjenjivanje (sample grader)

Program za ocjenjivanje (grader) učitava podatke u sljedećem formatu:

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

A_j i B_j su parametri funkcije `swap_seats` za zahtjev j .

Program za ocjenjivanje štampa rezultate u sljedećem formatu:

- red $1 + j$ ($0 \leq j \leq Q - 1$) : vrijednost koju vraća `swap_seats` za zahtjev j