



Miejsca

Masz przeprowadzić międzynarodowe zawody w prostokątnej hali mającej HW miejsc ułożonych w planszę o H wierszach i W kolumnach. Wiersze mają numery od 0 do $H - 1$, a kolumny od 0 do $W - 1$. Miejsce w wierszu r i kolumnie c oznaczamy przez (r, c) . Zaprosiłeś HW zawodników o numerach od 0 do $HW - 1$. Przygotowałeś też schemat, który usadza zawodnika o numerze i ($0 \leq i \leq HW - 1$) na miejscu (R_i, C_i) . Schemat każdemu miejscu przyporządkowuje dokładnie jednego zawodnika.

Podzbiór miejsc S nazwiemy **prostokątnym**, jeśli istnieją takie liczby całkowite r_1, r_2, c_1 oraz c_2 , że

- $0 \leq r_1 \leq r_2 \leq H - 1$.
- $0 \leq c_1 \leq c_2 \leq W - 1$.
- S jest zbiorem wszystkich miejsc (r, c) takich, że $r_1 \leq r \leq r_2$ oraz $c_1 \leq c \leq c_2$.

Prostokątny zbiór k miejsc ($1 \leq k \leq HW$) jest **śliczny**, jeśli zawodnicy mu przypisani mają numery od 0 do $k - 1$. **Śliczność** schematu, to liczba prostokątnych ślicznych obszarów znajdujących się w nim.

Już po sporządzeniu schematu dostajesz kilka żądań zamiany miejsc pomiędzy parami zawodników. Dokładniej: mamy Q takich żądań numerowanych chronologicznie od 0 do $Q - 1$. Żądanie j ($0 \leq j \leq Q - 1$) dotyczy zawodników A_j oraz B_j . Realizujesz żądania natychmiastowo i modyfikujesz zgodnie z nimi schemat zamieniając ich miejscami. Po każdym takim żądaniu powinieneś określić śliczność aktualnie obowiązującego schematu.

Szczegóły implementacyjne

Powinieneś zaimplementować następującą procedurę oraz funkcję

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

- H, W : liczba wierszy i liczba kolumn.
- R, C : tablice długości HW zawierające początkowy schemat.
- Ta procedura wywoływana jest raz, zanim nastąpi jakiegokolwiek wywołanie funkcji `swap_seats`.

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

- Funkcja ta opisuje żądanie zamiany dwóch zawodników
- a, b: zawodnicy, których miejsca mają być zamienione
- Funkcja ta jest wywoływana Q razy
- Wynikiem działania funkcji powinna być śliczność schematu po opisanej zamianie.

Przykład

Niech $H = 2$, $W = 3$, $R = [0, 1, 1, 0, 0, 1]$, $C = [0, 0, 1, 1, 2, 2]$, oraz $Q = 2$.

Sprawdzaczka wywołuje najpierw `give_initial_chart(2, 3, [0, 1, 1, 0, 0, 1], [0, 0, 1, 1, 2, 2])`.

Początkowy schemat jest więc taki:

0	3	4
1	2	5

Założmy, że sprawdzaczka wywołuje `swap_seats(0, 5)`. Po żądaniu nr 0 usadzenie zawodników jest więc następujące:

5	3	4
1	2	0

Zbiory miejsc przypisane zbiorom zawodników $\{0\}$, $\{0, 1, 2\}$ i $\{0, 1, 2, 3, 4, 5\}$ są prostokątne i śliczne. Zatem śliczność tego schematu, to 3 i funkcja `swap_seats` powinna zwrócić 3.

Wyobraźmy sobie, że wywołujemy ponownie `swap_seats(0, 5)`. Zatem po żądaniu nr 1 schemat wraca do początkowego ustawienia. Zbiory miejsc odpowiadające zbiorom zawodników $\{0\}$, $\{0, 1\}$, $\{0, 1, 2, 3\}$, oraz $\{0, 1, 2, 3, 4, 5\}$ są prostokątne i śliczne. Zatem śliczność tego schematu to 4 i funkcja `swap_seats` powinna zwrócić 4.

Pliki `sample-01-in.txt` i `sample-01-out.txt` w załączonym zzipowanym pakiecie odpowiadają temu przykładowi. W pakiecie tym znajdziesz też inne przykładowe wejścia i wyjścia.

Ograniczenia

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

Podzadania

1. (5 punktów) $HW \leq 100$, $Q \leq 5\,000$
2. (6 punktów) $HW \leq 10\,000$, $Q \leq 5\,000$
3. (20 punktów) $H \leq 1\,000$, $W \leq 1\,000$, $Q \leq 5\,000$
4. (6 punktów) $Q \leq 5\,000$, $|a - b| \leq 10\,000$ dla każdego wywołania `swap_seats`
5. (33 punkty) $H = 1$
6. (30 punktów) brak ograniczeń.

Przykładowa sprawdzaczka

Przykładowa sprawdzaczka czyta dane wejściowe w następującym formacie:

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

Tutaj A_j oraz B_j są parametrami wywołania `swap_seats` dla żądania o numerze j .

Sprawdzaczka wypisuje Twoje odpowiedzi w następującym formacie:

- wiersz $1 + j$ ($0 \leq j \leq Q - 1$) : wartość zwrócona przez funkcję `swap_seats` dla żądania o numerze j