



Mosaik

Salma planerar att färglägga en lermosaik på en vägg. Mosaiken består av ett $N \times N$ rutnät, gjord av N^2 stycken 1×1 fyrkantiga brickor, som alla är ofärgade i början. Mosaikens rader är numrerade från 0 till $N - 1$ uppifrån och ned, och kolumnerna är numrerade från 0 till $N - 1$ från vänster till höger. Brickan i rad i och kolumn j ($0 \leq i < N$, $0 \leq j < N$) betecknas med (i, j) . Varje bricka måste vara färgad antingen vitt (betecknad med 0) eller svart (betecknad med 1).

För att färglägga mosaiken väljer Salma först två arrayer X och Y med längd N . Var och en av dessa arrayer består av värdena 0 och 1, så att $X[0] = Y[0]$. Hon färgar brickorna på den översta raden (rad 0) enligt array X , så att färgen på brickan $(0, j)$ är $X[j]$ ($0 \leq j < N$). Hon färgar också brickorna i kolumnen längst till vänster (kolumn 0) enligt array Y , så att färgen på brickan $(i, 0)$ är $Y[i]$ ($0 \leq i < N$).

Sedan upprepar hon följande steg tills alla brickor är färgade:

- Hon väljer en *ofärgad* bricka (i, j) så att dess övre granne (bricka $(i - 1, j)$) och vänster granne (bricka $(i, j - 1)$) båda är *färgade*.
- Sedan färgar hon brickan (i, j) svart om båda dessa grannar är vita; annars färgar hon brickan (i, j) vit.

Det kan bevisas att de slutliga färgerna på brickorna inte beror på ordningen som Salma färgar dem.

Yasmin är väldigt nyfiken på färgerna på plattorna i mosaiken. Hon ställer Q frågor till Salma, numrerade från 0 till $Q - 1$. I fråga k ($0 \leq k < Q$) specificerar Yasmin en delrektangel av mosaiken genom att nämna dess:

- Översta rad $T[k]$ och nedersta rad $B[k]$ ($0 \leq T[k] \leq B[k] < N$),
- Kolumnen längst till vänster $L[k]$ och kolumnen längst till höger $R[k]$ ($0 \leq L[k] \leq R[k] < N$).

Svaret på en fråga är antalet svarta brickor i denna delrektangel. Specifikt bör Salma hitta hur många brickor (i, j) det finns, så att $T[k] \leq i \leq B[k]$, $L[k] \leq j \leq R[k]$, och färgen på brickan (i, j) är svart.

Skriv ett program som svarar på Yasmin's frågor.

Implementationsdetaljer

Joshua och Harry är nere på alla 8 för att be er att implementera följande funktion:

```
std::vector<long long> mosaic(
    std::vector<int> X, std::vector<int> Y,
    std::vector<int> T, std::vector<int> B,
    std::vector<int> L, std::vector<int> R)
```

- X , Y : arrayer med längd N som beskriver färgerna på brickorna i den översta raden respektive den vänstra kolumnen.
- T , B , L , R : arrayer med längd Q som beskriver frågorna från Yasmin.
- Funktionen skall returnera en array C med längden Q , så att $C[k]$ ger svaret på frågan k ($0 \leq k < Q$).
- Denna funktion anropas exakt en gång för varje testfall.

Gränser

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 200\,000$
- $X[i] \in \{0, 1\}$ och $Y[i] \in \{0, 1\}$ för alla i sådan att $0 \leq i < N$
- $X[0] = Y[0]$
- $0 \leq T[k] \leq B[k] < N$ and $0 \leq L[k] \leq R[k] < N$ för alla k sådan att $0 \leq k < Q$

Delpoäng

Grupp	Poäng	Ytterligare begränsningar
1	5	$N \leq 2; Q \leq 10$
2	7	$N \leq 200; Q \leq 200$
3	7	$T[k] = B[k] = 0$ (för alla k sådan att $0 \leq k < Q$)
4	10	$N \leq 5000$
5	8	$X[i] = Y[i] = 0$ (för alla i sådan att $0 \leq i < N$)
6	22	$T[k] = B[k]$ och $L[k] = R[k]$ (för alla k sådan att $0 \leq k < Q$)
7	19	$T[k] = B[k]$ (för alla k sådan att $0 \leq k < Q$)
8	22	Inga ytterligare begränsningar.

Exempel

Vi skall nu noggrant examinera följande exemplar 🤖👉:

```
mosaic([1, 0, 1, 0], [1, 1, 0, 1], [0, 2], [3, 3], [0, 0], [3, 2])
```

Detta exempel illustreras i bilderna nedan. Den vänstra bilden visar färgerna på brickorna i mosaiken. De mittersta och högra bilderna visar delrektanglarna Yasmin frågade om i den första respektive den andra frågan.

	0	1	2	3
0	1	0	1	0
1	1	0	0	1
2	0	1	0	0
3	1	0	1	0

	0	1	2	3
0	1	0	1	0
1	1	0	0	1
2	0	1	0	0
3	1	0	1	0

	0	1	2	3
0	1	0	1	0
1	1	0	0	1
2	0	1	0	0
3	1	0	1	0

Svaren på frågorna (det vill säga antalet ettor i de skuggade rektanglarna) är 7 respektive 3. Därför bör funktionen returnera $[7, 3]$.

Exempelgraderare

Input-format:

```
N
X[0]  X[1]  ...  X[N-1]
Y[0]  Y[1]  ...  Y[N-1]
Q
T[0]  B[0]  L[0]  R[0]
T[1]  B[1]  L[1]  R[1]
...
T[Q-1] B[Q-1] L[Q-1] R[Q-1]
```

Output-format:

```
C[0]
C[1]
...
C[S-1]
```

Här är S längden på arrayen C som returneras av `mosaic`.