

Mozaïek

Salma is van plan om een kleimozaïek op een muur te kleuren. Het mozaïek is een $N \times N$ grid, gemaakt van N^2 van tevoren ongekleurde 1×1 vierkante tegels. De rijen van het mozaïek zijn van boven naar beneden genummerd van 0 tot en met $N - 1$, en de kolommen zijn van links naar rechts genummerd van 0 tot en met $N - 1$. De tegel in rij i en kolom j ($0 \leq i < N$, $0 \leq j < N$) wordt aangeduid met (i, j) . Elke tegel moet gekleurd zijn: wit (aangeduid met 0) of zwart (aangeduid met 1).

Om het mozaïek te kleuren, kiest Salma eerst twee arrays X en Y van lengte N , elk bestaande uit de waarden 0 en 1, zodanig dat $X[0] = Y[0]$. Ze kleurt de tegels van de bovenste rij (rij 0) volgens array X , zodat de kleur van de tegel $(0, j)$ $X[j]$ is ($0 \leq j < N$). Ze kleurt ook de tegels van de meest linker kolom (kolom 0) volgens array Y , zodat de kleur van de tegel $(i, 0)$ $Y[i]$ is ($0 \leq i < N$).

Vervolgens herhaalt ze de volgende stappen totdat alle tegels gekleurd zijn:

- Ze vindt een *ongekleurde* tegel (i, j) zodanig dat de bovenbuur (tegel $(i - 1, j)$) en linkerbuur (tegel $(i, j - 1)$) beide *al gekleurd* zijn.
- Vervolgens kleurt ze tegel (i, j) zwart als beide burens wit zijn; anders kleurt ze tegel (i, j) wit.

Er kan worden aangetoond dat de uiteindelijke kleuren van de tegels niet afhankelijk zijn van de volgorde waarin Salma ze kleurt.

Yasmin is erg nieuwsgierig naar de kleuren van de tegels in het mozaïek. Ze stelt Salma Q vragen, genummerd van 0 tot en met $Q - 1$. In vraag k ($0 \leq k < Q$), specificeert Yasmin een subrechthoek van het mozaïek door zijn:

- Bovenste rij $T[k]$ en onderste rij $B[k]$ ($0 \leq T[k] \leq B[k] < N$),
- Meest linker kolom $L[k]$ en meest rechter kolom $R[k]$ ($0 \leq L[k] \leq R[k] < N$).

Het antwoord op de vraag is het aantal zwarte tegels in deze subrechthoek. Om precies te zijn moet Salma uitzoeken hoeveel tegels (i, j) er bestaan, zodanig dat $T[k] \leq i \leq B[k]$, $L[k] \leq j \leq R[k]$, en de kleur van tegel (i, j) zwart is.

Schrijf een programma dat de vragen van Yasmin beantwoordt.

Implementatiedetails

Je moet de volgende procedure implementeren.

```
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 : arrays van lengte N die de kleuren van de tegels beschrijven respectievelijk in de bovenste rij en de meest linker kolom.
- T, B, L, R : arrays met lengte Q die de door Yasmin gestelde vragen beschrijven.
- De procedure moet een array C van lengte Q returnen, zodat $C[k]$ het antwoord geeft op vraag k ($0 \leq k < Q$).
- Deze procedure wordt voor elke testcase precies één keer aangeroepen.

Beperkingen

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 200\,000$
- $X[i] \in \{0, 1\}$ en $Y[i] \in \{0, 1\}$ voor elke i zodat $0 \leq i < N$
- $X[0] = Y[0]$
- $0 \leq T[k] \leq B[k] < N$ en $0 \leq L[k] \leq R[k] < N$ voor elke k zodat $0 \leq k < Q$

Subtaken

Subtaak	Score	Extra beperkingen
1	5	$N \leq 2; Q \leq 10$
2	7	$N \leq 200; Q \leq 200$
3	7	$T[k] = B[k] = 0$ (voor elke k zodat $0 \leq k < Q$)
4	10	$N \leq 5000$
5	8	$X[i] = Y[i] = 0$ (voor elke i zodat $0 \leq i < N$)
6	22	$T[k] = B[k]$ en $L[k] = R[k]$ (voor elke k zodat $0 \leq k < Q$)
7	19	$T[k] = B[k]$ (voor elke k zodat $0 \leq k < Q$)
8	22	Geen extra beperkingen.

Voorbeeld

Bekijk de volgende call.

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

Dit voorbeeld wordt geïllustreerd in de onderstaande afbeeldingen. De linker afbeelding toont de kleuren van de tegels in het mozaïek. De middelste en rechter afbeeldingen tonen de subrechthoeken waar Yasmin naar vroeg in respectievelijk de eerste en tweede vraag.

	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

De antwoorden op de vragen (dat wil zeggen, het aantal enen in de getinte rechthoeken) zijn respectievelijk 7 en 3. Daarom moet de procedure `[7, 3]` returnen.

Voorbeeldgrader

Invoerformaat:

```
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]
```

Uitvoerformaat:

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

Hierbij is S de lengte van de array C die wordt gereturned door `mosaic`.