

## Мозаїка

Сальма планує розфарбувати глиняну мозаїку на стіні. Мозаїка являє собою  $N \times N$  сітку, що складається з  $N^2$  спочатку нерозфарбованих  $1 \times 1$  квадратних плиток. Рядки мозаїки пронумеровані від 0 до  $N - 1$  зверху вниз, і стовпці пронумеровані від 0 до  $N - 1$  зліва направо. Плитка в рядку  $i$  і стовпці  $j$  ( $0 \leq i < N$ ,  $0 \leq j < N$ ) позначається  $(i, j)$ . Кожна плитка також повинна бути зафарбована білим (позначається 0) або чорним (позначається 1) кольором.

Щоб розфарбувати мозаїку, Сальма спочатку вибирає два масиви  $X$  і  $Y$  довжини  $N$ , кожне з яких складається зі значень 0 і 1 таких, що  $X[0] = Y[0]$ . Вона розфарбовує плитки самого верхнього рядка (рядок 0) відповідно до масиву  $X$  так, що колір плитки  $(0, j)$  дорівнює  $X[j]$  ( $0 \leq j < N$ ). Вона також розфарбовує плитки самого лівого стовпця (стовпець 0) відповідно до масиву  $Y$  так, що колір плитки  $(i, 0)$  дорівнює  $Y[i]$  ( $0 \leq i < N$ ).

Потім вона повторює наступні кроки, доки всі плитки не зафарбуються:

- Вона знаходить будь-яку *незафарбовану* плитку  $(i, j)$  таку, що її верхній сусід (плитка  $(i - 1, j)$ ) і лівий сусід (плитка  $(i, j - 1)$ ) обидва *вже пофарбовані*.
- Потім вона зафарбовує плитку  $(i, j)$  в чорний колір, якщо обидва ці сусіди білі; інакше вона зафарбовує плитку  $(i, j)$  в білий колір.

Можна показати, що кінцеві кольори плитки не залежать від порядку, у якому Сальма їх розфарбовує.

Ясмін дуже цікавиться кольорами плиток у мозаїці. Вона ставить Сальмі  $Q$  запитів, пронумерованих від 0 до  $Q - 1$ . У запиті  $k$  ( $0 \leq k < Q$ ), Ясмін визначає підпрямокутник мозаїки за допомогою:

- Самого верхнього рядка  $T[k]$  і самого нижнього рядка  $B[k]$  ( $0 \leq T[k] \leq B[k] < N$ ),
- Самого лівого стовпця  $L[k]$  і самого правого стовпця  $R[k]$  ( $0 \leq L[k] \leq R[k] < N$ ).

Відповіддю на запит є кількість чорних плиток у цьому підпрямокутнику. Зокрема, Сальма повинна знайти, скільки плиток  $(i, j)$  існує таких, що  $T[k] \leq i \leq B[k]$ ,  $L[k] \leq j \leq R[k]$ , а колір плитки  $(i, j)$  — чорний.

Напишіть програму, яка відповідає на запити Ясмін.

## Деталі реалізації

Ви повинні реалізувати наступну функцію.

```
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)
```

- $X, Y$ : масиви довжиною  $N$ , що описують кольори плиток у верхньому рядку та лівому стовпці відповідно.
- $T, B, L, R$ : масиви довжини  $Q$ , що описують запити, які ставить Ясмін.
- Функція має повернути масив  $C$  довжини  $Q$ , такий, що  $C[k]$  дає відповідь на запит  $k$  ( $0 \leq k < Q$ ).
- Ця функція викликається рівно один раз для кожного тесту.

## Обмеження

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 200\,000$
- $X[i] \in \{0, 1\}$  і  $Y[i] \in \{0, 1\}$  для кожного  $i$  такого, що  $0 \leq i < N$
- $X[0] = Y[0]$
- $0 \leq T[k] \leq B[k] < N$  і  $0 \leq L[k] \leq R[k] < N$  для кожного  $k$  такого, що  $0 \leq k < Q$

## Підзадачі

Підзадачі	Балів	Додаткові обмеження
1	5	$N \leq 2; Q \leq 10$
2	7	$N \leq 200; Q \leq 200$
3	7	$T[k] = B[k] = 0$ (для кожного $k$ такого, що $0 \leq k < Q$ )
4	10	$N \leq 5000$
5	8	$X[i] = Y[i] = 0$ (для кожного $i$ такого, що $0 \leq i < N$ )
6	22	$T[k] = B[k]$ і $L[k] = R[k]$ (для кожного $k$ такого, що $0 \leq k < Q$ )
7	19	$T[k] = B[k]$ (для кожного $k$ такого, що $0 \leq k < Q$ )
8	22	Без додаткових обмежень

## Приклад

Розглянемо наступний виклик.

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

Цей приклад показано на малюнках нижче. Ліворуч показано кольори плиток мозаїки. Середній і правий малюнки показують підпрямокутники, які Ясмін запитала у першому та другому запиті відповідно.

	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

Відповіді на запит (тобто кількість одиниць у заштрихованих прямокутниках) - 7 і 3 відповідно. Отже, функція має повернути [7,3]

## Приклад градера

Формат вхідних даних:

```
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]
```

Формат вихідних даних:

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

Тут  $S$  — це довжина масиву  $C$ , яку повертає `mosaic`.