

# Mosaic

Салма планирует нарисовать глиняную мозаику на стене. Мозаика это таблица  $N \times N$ , состоящая из  $N^2$  изначально бесцветных квадратных клеток  $1 \times 1$ . Строки мозаики пронумерованы сверху вниз от 0 до  $N - 1$ , столбцы пронумерованы слева направо от 0 до  $N - 1$ . Клетка в строке  $i$  и столбце  $j$  ( $0 \leq i < N$ ,  $0 \leq j < N$ ) обозначается как  $(i, j)$ . Каждая клетка должна быть покрашена либо в белый (обозначим за 0) либо в черный (обозначим за 1) цвет.

Для того чтобы раскрасить мозаику, Салма сначала выбирает два массива  $X$  и  $Y$  длины  $N$ , каждый состоящий из 0 и 1, такие что  $X[0] = Y[0]$ . Она красит клетки самой верхней строки (строки 0) в соответствии с массивом  $X$ , так что цвет клетки  $(0, j)$  это  $X[j]$  ( $0 \leq j < N$ ). Она также красит клетки самого левого столбца (столбец 0) в соответствии с массивом  $Y$ , так что цвет клетки  $(i, 0)$  это  $Y[i]$  ( $0 \leq i < N$ ).

Она повторяет следующие шаги, до тех пор пока все клетки не будут покрашены:

- Она находит любую *непокрашенную* клетку  $(i, j)$ , такую что ее верхний сосед (клетка  $(i - 1, j)$ ) и левый сосед (клетка  $(i, j - 1)$ ) оба уже покрашены
- Затем она красит клетку  $(i, j)$  в черный, если оба соседа белые; иначе она красит клетку  $(i, j)$  в белый.

Можно показать, что итоговая раскраска клеток не зависит от порядка, в котором Салма будет их красить.

Ясмин очень интересуется цветами клеток в мозаике. Он задал Салме  $Q$  запросов, пронумерованных от 0 до  $Q - 1$ . В запросе  $k$  ( $0 \leq k < Q$ ) Ясмин задает подпрямоугольник мозаики с помощью:

- его самой верхней строки  $T[k]$  и его самой нижней строки  $B[k]$  ( $0 \leq T[k] \leq B[k] < N$ ),
- его самого левого столбца  $L[k]$  и его самого правого столбца  $R[k]$  ( $0 \leq L[k] \leq R[k] < N$ ).

Ответом на запрос является количество черных клеток в подпрямоугольнике. Формально Салма должна найти сколько клеток  $(i, j)$  существует, таких что  $T[k] \leq i \leq B[k]$ ,  $L[k] \leq j \leq R[k]$  и клетка  $(i, j)$  раскрашена в черный.

Напишите программу, которая ответит на запросы Ясмينا.

## Implementation Details

Вы должны реализовать следующую функцию:

```
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)
```

- $X, Y$ : массивы длины  $N$ , описывающие цвета клеток в самой верхней строке и в самом левом столбце, соответственно.
- $T, B, L, R$ : массивы длины  $Q$ , описывающие запросы, заданные Ясмином.
- Функция должна вернуть массив  $C$  длины  $Q$ , такой что  $C[k]$  равно ответу на запрос  $k$  ( $0 \leq k < Q$ ).
- Эта функция будет вызвана ровно один раз для каждого теста.

## Constraints

- $1 \leq N \leq 200\,000$
- $1 \leq Q \leq 200\,000$
- $X[i] \in \{0, 1\}$  и  $Y[i] \in \{0, 1\}$  для всех  $i$  таких что  $0 \leq i < N$
- $X[0] = Y[0]$
- $0 \leq T[k] \leq B[k] < N$  и  $0 \leq L[k] \leq R[k] < N$  для всех  $k$  таких что  $0 \leq k < Q$

## Subtasks

Подзадача	Балл	Дополнительные ограничения
1	5	$N \leq 2; Q \leq 10$
2	7	$N \leq 200; Q \leq 200$
3	7	$T[k] = B[k] = 0$ (для всех $k$ таких что $0 \leq k < Q$ )
4	10	$N \leq 5000$
5	8	$X[i] = Y[i] = 0$ (для всех $i$ таких что $0 \leq i < N$ )
6	22	$T[k] = B[k]$ и $L[k] = R[k]$ (для всех $k$ таких что $0 \leq k < Q$ )
7	19	$T[k] = B[k]$ (для всех $k$ таких что $0 \leq k < Q$ )
8	22	Нет дополнительных ограничений.

## Example

Рассмотрим следующий вызов.

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

Этот пример нарисован снизу. Слева нарисованы цвета клеток в мозаике. На центральной и правой картинках выделены подпрямоугольники, про которые Ясмин спросил в первом и втором запросах, соответственно.

	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

Ответы на запросы (то есть, количества единиц в выделенных подпрямоугольниках) это 7 и 3, соответственно. Таким образом, функция должна вернуть [7, 3].

## Sample Grader

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]
```

Здесь  $S$  это длина массива  $C$ , который вернула функция `mosaic`.