



Стена

Джан-Джи строит стену из кирпичей одинакового размера. Стена состоит из n столбцов кирпичей, пронумерованных слева направо от 0 до $(n - 1)$. Высотой столбца называется количество кирпичей в нем. У столбцов могут быть разные высоты.

Джан-Джи строит стену так. Сначала ни в одном из столбцов нет кирпичей. Далее Джан-Джи выполняет k действий, каждое из которых может быть действием *добавления* или *удаления* кирпичей. Строительство считается законченным, когда выполнены все k действий. Перед каждым действием Джан-Джи выбирает интервал из последовательно стоящих столбцов и высоту h . После этого он выполняет одно из следующих действий:

- действие *добавления*: Джан-Джи добавляет кирпичи в столбцы из выбранного интервала, высота которых меньше чем h , так, чтобы она стала равной h . Со столбцами, высота которых не меньше, чем h , он ничего не делает;
- действие *удаления*: Джан-Джи убирает кирпичи из столбцов из выбранного интервала, высота которых больше чем h , так, чтобы она стала равной h . Со столбцами, высота которых не больше, чем h , он ничего не делает.

Требуется определить конечную форму стены.

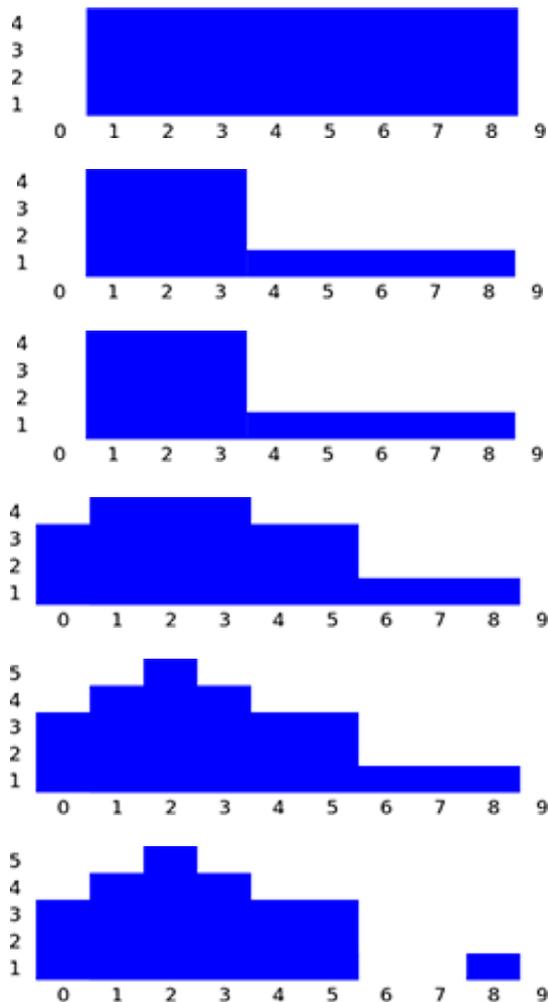
Пример

Будем считать, что стена состоит из 10 столбцов и строится за шесть действий. Все приведенные интервалы включают границы. Изображения стены после каждого действия представлены на рисунке ниже.

действие	вид	интервал	высота
0	добавление	столбцы с 1 до 8	4
1	удаление	столбцы с 4 до 9	1
2	удаление	столбцы с 3 до 6	5
3	добавление	столбцы с 0 до 5	3
4	добавление	столбец 2	5
5	удаление	столбцы с 6 до 7	0

Так как изначально все столбцы не содержат кирпичей, то после действия 0 каждый из столбцов с номерами от 1 до 8 будет содержать по 4 кирпича, столбцы с номерами 0 и 9 будут пустыми. После действия 1 в столбцах с номерами с 4 по 8 остается один кирпич, 9-ый столбец остается пустым. Столбцы с номерами с 0 по 3 лежат вне интервала и не меняются. Действие 2 не меняет столбцы с номерами с 3 по 6, так как в них и так меньше пяти кирпичей. После действия 3 количество кирпичей в столбцах с номерами 0, 4 и 5 увеличивается до 3. После действия 4 во втором столбце становится пять кирпичей. За 5-ое действие из столбцов

с номерами 6 и 7 убираются все кирпичи.



Постановка задачи

По заданному описанию k действий определите количество кирпичей в каждом столбце после того, как все действия будут выполнены. Вы должны реализовать функцию `buildWall`.

- `buildWall(n, k, op, left, right, height, finalHeight)`
 - n : количество столбцов;
 - k : количество действий;
 - `op`: массив длины k ; `op[i]` тип i -го действия: 1, если это действие добавления кирпичей, и 2, если это действие удаления кирпичей;
 - `left` и `right`: массивы длины k ; интервал для i -го действия начинается со столбца `left[i]` и заканчивается столбцом `right[i]` (включая оба конца `left[i]` и `right[i]`); `left[i]` всегда не превосходит `right[i]`;
 - `height`: массив длины k ; `height[i]` высота для i -го действия;
 - `finalHeight`: массив длины n ; вы должны записать конечное количество кирпичей в столбце i в `finalHeight[i]`.

Подзадачи

Во всех подзадачах все высоты не превосходят 100 000.

Подзадача	Баллы	n	k	Комментарий
1	8	$1 \leq n \leq 10\,000$	$1 \leq k \leq 5\,000$	нет дополнительных ограничений
2	24	$1 \leq n \leq 100\,000$	$1 \leq k \leq 500\,000$	все действия добавления будут до действий удаления
3	29	$1 \leq n \leq 100\,000$	$1 \leq k \leq 500\,000$	нет дополнительных ограничений
4	39	$1 \leq n \leq 2\,000\,000$	$1 \leq k \leq 500\,000$	нет дополнительных ограничений

Детали реализации

Вы должны послать ровно один файл, названный `wall.c`, `wall.cpp` или `wall.pas`. В этом файле должна быть реализована функция, описанная выше, с указанным ниже прототипом. На языках C/C++ вы должны подключить заголовочный файл `wall.h`.

Язык C/C++

```
void buildWall(int n, int k, int op[], int left[], int right[],
int height[], int finalHeight[]);
```

Язык Pascal

```
procedure buildWall(n, k : longint; op, left, right, height :
array of longint; var finalHeight : array of longint);
```

Пример проверяющего модуля

Предоставленный пример проверяющего модуля имеет следующий формат ввода данных:

- строка 1: n, k .
- строки $2 + i$ ($0 \leq i \leq k - 1$): $op[i], left[i], right[i], height[i]$.