



## Wall

Джан строи стена от еднакви тухли, като ги реди една над друга в  $n$  колони, номерирани отляво надясно с числата от 0 до  $n - 1$ . Височина на всяка колона е броят на тухлите в нея и различните колони могат да имат различни височини.

Джан строи стената в  $k$  фази, всяка от които е или *добавяне*, или *премахване*. Строенето завършва, когато всички предвидени  $k$  фази се изпълнят. Отначало всички колони са празни. Всяка фаза е с параметър  $h$ , който е възможна височина на колона и протича както следва:

- Ако фазата е добавяне, във всички колони с височина по-малка от  $h$  се добавят тухли, за да станат височините на тези колони точно  $h$ . Колоните с височини по-големи или равни на  $h$  не се променят.
- Ако фазата е премахване, от всички колони с височина по-голяма от  $h$  се махат тухли, за да станат височините на тези колони точно  $h$ . Колоните с височини по-малки или равни на  $h$  не се променят.

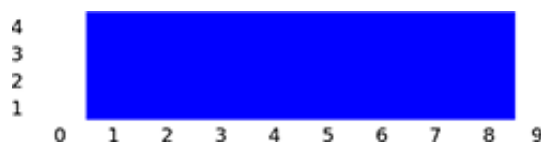
Задачата ви е да намерите окончателната височина на стената.

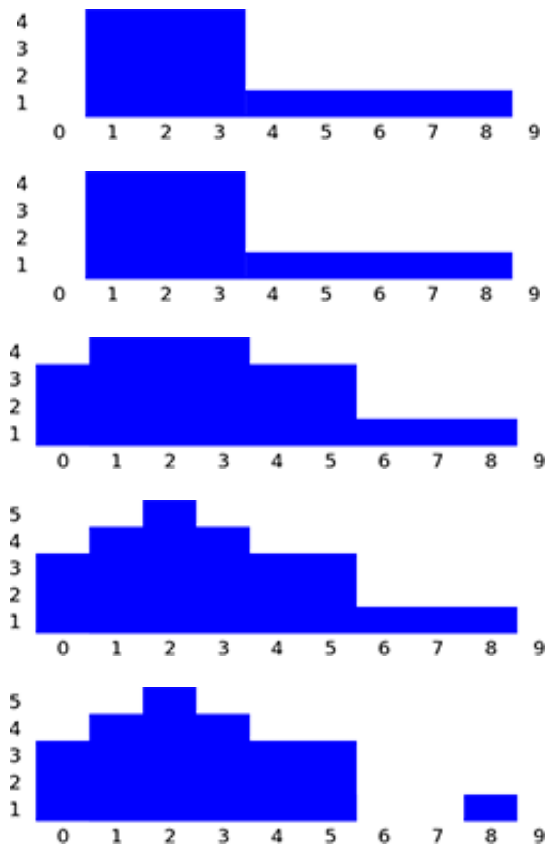
## Пример

Нека имаме стена с 10 колони и 6 фази на построяване на стената както следва:

фаза	тип	интервал	височина
0	добави	колони от 1 до 8	4
1	премахни	колони от 4 до 9	1
2	премахни	колони от 3 до 6	5
3	добави	колони от 0 до 5	3
4	добави	колона 2	5
5	премахни	колони от 6 to 7	0

Тъй като отначало всички колони са празни, след фаза 0 колоните с номера от 1 до 8 са с височина 4, а колоните 0 и 9 са празни. Във фаза 1 се махат тухли от колоните с номера от 4 до 8 така, че във всяка от тях да остане само 1 тухла, а колони 0 и 9 остават празни. Във фаза 2 нищо не се променя, защото никоя от колоните с номера от 3 до 6 няма повече от 5 тухли. След фаза 3 броят на тухлите в колоните 0, 4 и 5 нараства до 3. След фаза 4 броят на тухлите в колона 2 става 5, а след фаза 5 в колоните 6 и 7 не остава нито една тухла (виж Фигурата).





## Задача

По зададени описания на  $k$ -те фази, намерете броя на тухлите във всяка от колоните след края на последната фаза. Напишете функция:

- `buildWall(n, k, op, left, right, height, finalHeight)`
  - $n$ : брой на колоните в стената.
  - $k$ : брой на фазите.
  - `op`: масив с  $k$  елемента; `op[i]` е типът на фазата с номер  $i$ : 1, за фаза от тип добавяне или 2, за фаза от тип премахване,  $0 \leq i \leq k - 1$ .
  - `left` и `right`: масиви с  $k$  елемента; интервалът от колони които ще бъдат прегледани във фазата с номер  $i$  започва от колона `left[i]` и завършва в колона `right[i]` (включително `left[i]` и `right[i]`),  $0 \leq i \leq k - 1$ . Гарантирано е, че `left[i] ≤ right[i]`.
  - `height`: масив с  $k$  елемента; `height[i]` е параметърът за височина на колоните във фазата с номер  $i$ ,  $0 \leq i \leq k - 1$ .
  - `finalHeight`: масив с  $n$  елемента; в него трябва да бъдат поставени резултатите от работата на вашата функция, т.е. в `finalHeight[i]` трябва да е броят на тухлите в колоната с номер  $i$ ,  $0 \leq i \leq n - 1$ .

## Подзадачи

За всички подзадачи параметърът за височина във всички фази е неотрицателно цяло, по-малко от или равно на **100000**.

подзадача	точки	$n$	$k$	Допълнително условие
1	8	$1 \leq n \leq 10000$	$1 \leq k \leq 5000$	няма допълнително условие
2	24	$1 \leq n \leq 100000$	$1 \leq k \leq 500000$	Всички фази на добавяне са преди всички фази на изтриване.
3	29	$1 \leq n \leq 100000$	$1 \leq k \leq 500000$	няма допълнително условие
4	39	$1 \leq n \leq 2000000$	$1 \leq k \leq 500000$	няма допълнително условие

## Детайли на имплементацията на C/C++

Изпратете само един файл с име `wall.c` или `wall.cpp`. Той трябва да съдържа имплементация на функцията `buildWall` такава, каквато е описана по-горе, със следната спецификация:

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

като не забравите да включите в него файла `wall.h`.

### Примерен грейдър

Примерният грейдър чете вход във формат:

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