



Wall 磚牆

健佳正在用大小相同的磚塊來疊起一面牆。這面牆由 n 列磚頭所組成，它們從左到右的編號為 0 至 $n - 1$ 。每列磚頭的高度可以不同的，而每列的高度就是該列磚塊的數量。

健佳用以下的方式來建造起這面牆。首先每列都是沒有磚塊的，然後，健佳通過 k 個階段的增加 (adding) 或 移除 (removing) 磚塊，而當這 k 個階段完成後，這面牆就砌好了。在每個階段中健佳都會知道一個磚塊的列範圍和一個高度值 h ，然後他就跟著以下的過程去做：

- 在 增加磚塊 (adding) 階段，健佳會在給出的列範圍中小於 h 的列增加磚塊，使它們的高度都恰好等於 h 個磚塊。而他不會去改變那些大於或等於 h 的列的磚塊數目。
- 在 移除磚塊 (removing) 階段，健佳會在給出的列範圍中大於 h 的列移除磚塊，使它們的高度都恰好等於 h 個磚塊。而他不會去改變那些小於或等於 h 的列的磚塊數目。

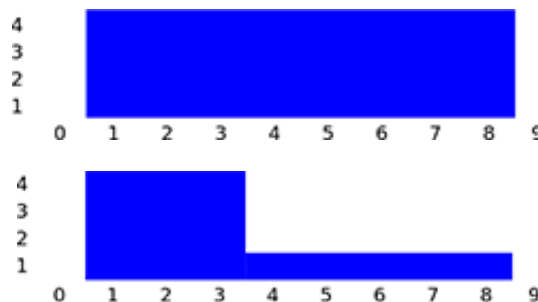
你的任務就是去計算出這面牆最後的形狀。

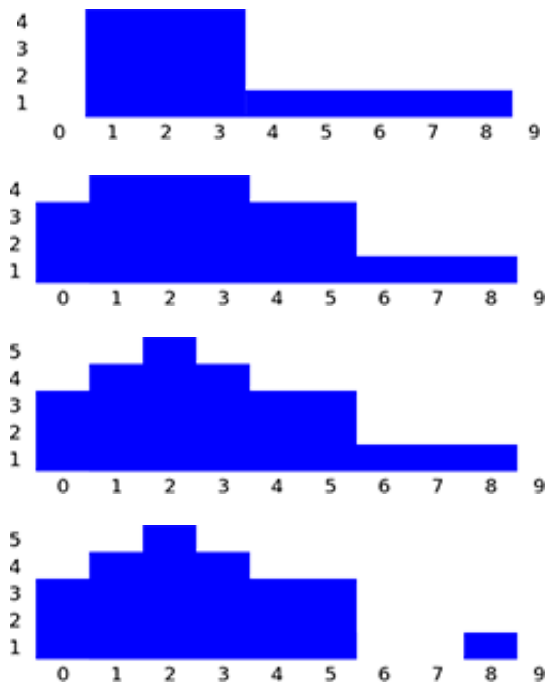
樣例

我們假設有 10 個磚塊列和 6 個建造牆的階段。全部的範圍如下表所示，在每個階段之後牆的圖像如下面的圖例中所示。

階段	階段類型	範圍	高度
0	增加add	列 1 至 8	4
1	移除remove	列 4 至 9	1
2	移除remove	列 3 至 6	5
3	增加add	列 0 至 5	3
4	增加add	列 2	5
5	移除remove	列 6 至 7	0

在開始時所有列都是空的，所以在階段 0 執行後列 1 至列 8 都會有 4 塊磚。列 0 和列 9 仍然是空的。在階段 1 裏，列 4 至列 8 將會被移除至 1 塊磚，而列 9 仍然是空的，列 0 至列 3 都不在這次的列範圍內，因此它們的高度是不變的。在階段 2 裏，因為列 3 至列 6 都沒有超過 5 塊磚，所以這階段不會改變這面牆的高度。在階段 3 進行後，列 0，列 4 和列 5 都有 3 塊磚。而在階段 4 進行後列 2 會有 5 塊磚。而階段 5 會移除列 6 和列 7 中的所有磚塊。





任務

給出這 k 個階段的描述後，請計算出當這 k 個建造階段完成之後每列的磚塊數目。你需要編寫函數 `buildWall`。

- `buildWall(n, k, op, left, right, height, finalHeight)`
 - n : 這面牆的列的數目。
 - k : 階段的數目。
 - op : 長度為 k 的陣列; $op[i]$ 是第 i 個階段的類型: 1 代表是增加階段(adding) 而 2 代表是移除階段(removing), 其中 $0 \leq i \leq k - 1$.
 - $left$ 和 $right$: 長度為 k 的陣列; 表示在第 i 個列範圍裏從列 $left[i]$ 開始到列 $right[i]$ 結束 (包含端點 $left[i]$ 和 $right[i]$), 其中 $0 \leq i \leq k - 1$, 而且 $left[i] \leq right[i]$ 。
 - $height$: 長度為 k 的陣列; $height[i]$ 表示在階段 i 的高度, 其中 $0 \leq i \leq k - 1$ 。
 - $finalHeight$: 長度為 n 的陣列; 你應該把第 i 列磚塊數目的最後計算結果存放到 $finalHeight[i]$ 中並且返回, 其中 $0 \leq i \leq n - 1$ 。

子任務

在所有子任務中，全部階段中的高度參數均為非負整數而且小於或等於 100,000。

子任務	分數	n	k	備註
1	8	$1 \leq n \leq 10,000$	$1 \leq k \leq 5,000$	沒有其他限制
2	24	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	全部增加階段均在全部移除階段之前
3	29	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	沒有其他限制
4	39	$1 \leq n \leq 2,000,000$	$1 \leq k \leq 500,000$	沒有其他限制

編程詳情

你必須提交唯一一個檔案，其名為 `wall.c`, `wall.cpp` 或 `wall.pas`。這檔案內需要實現如上面所述並包括下面的特徵的子程式。若你是編寫 C/C++ 程式的話，你亦需要加入標頭檔 `wall.h`。

C/C++ 程式

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

Pascal 程式

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

Sample grader 樣例評測器

樣例評測器需要讀入以下格式的資料：

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