



## Wall

Marko zida zid slaganjem cigli iste veličine. Zid se sastoji od  $n$  stubova (kolona) cigli, koji su poređani u red i numerisani brojevima od 0 do  $n - 1$  sleva nadesno. Stubovi mogu biti različite visine. Visina stuba je broj cigli u njemu.

Marko zida zid na sledeći način. Na početku nema cigala ni u jednom stubu. Zatim, Marko izvodi  $k$  faza *dodavanja* ili *uklanjanja* cigli. Proces zidanja završava se izvođenjem svih  $k$  faza. U svakoj fazi Marku se zadaje interval uzastopnih stubova, kao i visina  $h$ , nakon čega on sprovodi sledeću proceduru:

- U fazi *dodavanja*, Marko, uz pomoć magičnog kišobrana, dodaje cigle na one stubove iz datog intervala koji imaju manje od  $h$  cigli, tako da nakon ove faze ti stubovi imaju tačno  $h$  cigli. On ne radi ništa sa stubovima koji imaju  $h$  ili više cigli.
- U fazi *uklanjanja*, Marko, uz pomoć magičnog kišobrana, uklanja cigle sa onih stubova iz datog intervala koji imaju više od  $h$  cigli, tako da nakon ove faze oni imaju tačno  $h$  cigli. On ne radi ništa sa stubovima koji imaju  $h$  ili manje cigli.

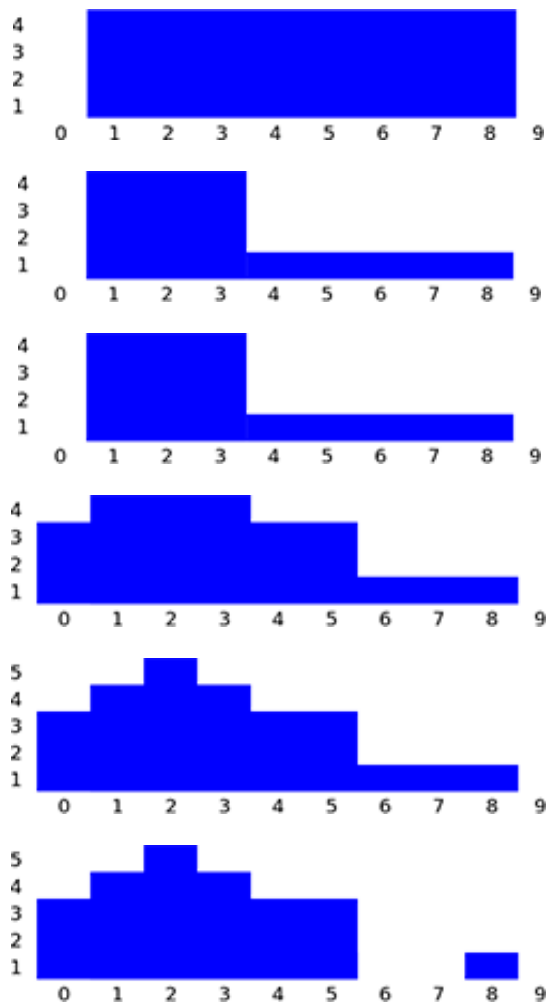
Vaš zadatak je da odredite konačan oblik zida.

## Primer

Pretpostavimo da postoji 10 stubova i 6 faza zidanja. Svi intervali u sledećoj tabeli uključuju i svoje krajeve. Ilustracije zida nakon svake faze date su niže.

faza	tip	opseg	visina
0	dodavanje	stubovi od 1 do 8	4
1	uklanjanje	stubovi od 4 do 9	1
2	uklanjanje	stubovi od 3 do 6	5
3	dodavanje	stubovi od 0 do 5	3
4	dodavanje	stub 2	5
5	uklanjanje	stubovi od 6 do 7	0

Kako su svi stubovi na početku prazni, nakon faze 0 svi stubovi od 1 do 8 imaće po 4 cigle. Stubovi 0 i 9 ostaju prazni. U fazi 1, cigle se uklanjaju sa stubova od 4 do 8 dok svaki od njih ne bude imao tačno 1 ciglu, a stub 9 ostaje prazan. Stubovi od 0 do 3, koji su van zadatog intervala, ostaju nepromenjeni. Faza 2 ne menja ništa jer stubovi od 3 do 6 nemaju više od 5 cigli. Nakon faze 3 broj cigli u stubovima 0, 4 i 5 povećava se na 3. Nakon faze 4 stub 2 ima 5 cigli. Faza 5 uklanja sve cigle iz stubova 6 i 7.



## Zadatak

Na osnovu datog opisa  $k$  faza, potrebno je izračunati broj cigli u svakom stubu na kraju, nakon završetka svih faza. Potrebno je implementirati funkciju `buildWall`.

- `buildWall(n, k, op, left, right, height, finalHeight)`
  - $n$ : broj stubova u zidu.
  - $k$ : broj faza.
  - `op`: niz dužine  $k$ ; `op[i]` je tip faze  $i$ : 1 za fazu dodavanja a 2 za fazu uklanjanja, za  $0 \leq i \leq k - 1$ .
  - `left` i `right`: nizovi dužine  $k$ ; interval stubova u fazi  $i$  počinje stubom `left[i]` i završava stubom `right[i]` (uključujući oba krajnja stuba `left[i]` i `right[i]`), za  $0 \leq i \leq k - 1$ . Uvek važi `left[i] ≤ right[i]`.
  - `height`: niz dužine  $k$ ; `height[i]` je parametar visine faze  $i$ , za  $0 \leq i \leq k - 1$ .
  - `finalHeight`: niz dužine  $n$ ; u ovom nizu vraćate vaš rezultat tako što ćete konačan broj cigli u stubu  $i$  upisati u `finalHeight[i]`, za svako  $0 \leq i \leq n - 1$ .

## Podzadaci

Za sve podzadatke parametar visine u svim fazama je nenegativan ceo broj manji ili jednak **100,000**.

podzadatak	poeni	$n$	$k$	napomena
1	8	$1 \leq n \leq 10,000$	$1 \leq k \leq 5,000$	nema dodatnih ograničenja
2	24	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	sve faze dodavanja su pre svih faza uklanjanja
3	29	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	nema dodatnih ograničenja
4	39	$1 \leq n \leq 2,000,000$	$1 \leq k \leq 500,000$	nema dodatnih ograničenja

## Detalji implementacije

Morate poslati tačno jedan fajl sa nazivom `wall.c`, `wall.cpp` ili `wall.pas`. Taj fajl mora implementirati gore opisane funkcije/procedure koristeći sledeće potpise. Takođe morate include-ovati header fajl `wall.h` za C/C++ programe.

### C/C++ programi

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

### Pascal programi

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

### Opis grejdera

Grejder koji je dostupan na vašem računaru čita ulaz u sledećem formatu:

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