



Siena

Jian-Jia būvē sienu, mūrējot vienāda izmēra ķieģeļus vienu pie otra. Šī siena sastāv no n ķieģeļu kolonnām, kur kolonnas virzienā no kreisās uz labo pusi sanumurētas ar numuriem no 0 to $n - 1$. Katrā no kolonnām sienas augstums var būt atšķirīgs. Kolonas augstums ir vienāds ar tajā esošo ķieģeļu skaitu.

Jian-Jia būvē sienu šādā veidā. Sākumā nevienā kolonnā neviena ķieģeļa nav. Tad Jian-Jia izpilda k *pievienošanas* vai *noņemšanas* darbības. Sienas būvēšanas process beidzas, kad visas k darbības ir izpildītas. Katram darbības veidam Jian-Jia-am tiek norādīts secīgu kolonnu intervāls, augstums h , un viņš izpilda sekojošo:

- *pievienošanas* darbībai Jian-Jia pievieno ķieģeļus norādītā intervāla kolonnām, kurās ir mazāk par h ķieģeļiem, tā, lai pēc darbības tajās būtu tieši h ķieģeļi. Viņš nedara neko tām norādītā intervāla kolonnām, kurās ir h vai vairāk ķieģeļu.
- *noņemšanas* darbībai Jian-Jia noņem ķieģeļus no norādītā intervāla kolonnām, kurās ir vairāk par h ķieģeļiem, tā, lai tajās paliktu tieši h ķieģeļi. Viņš nedara neko tām norādītā intervāla kolonnām, kurās ir h vai mazāk ķieģeļu.

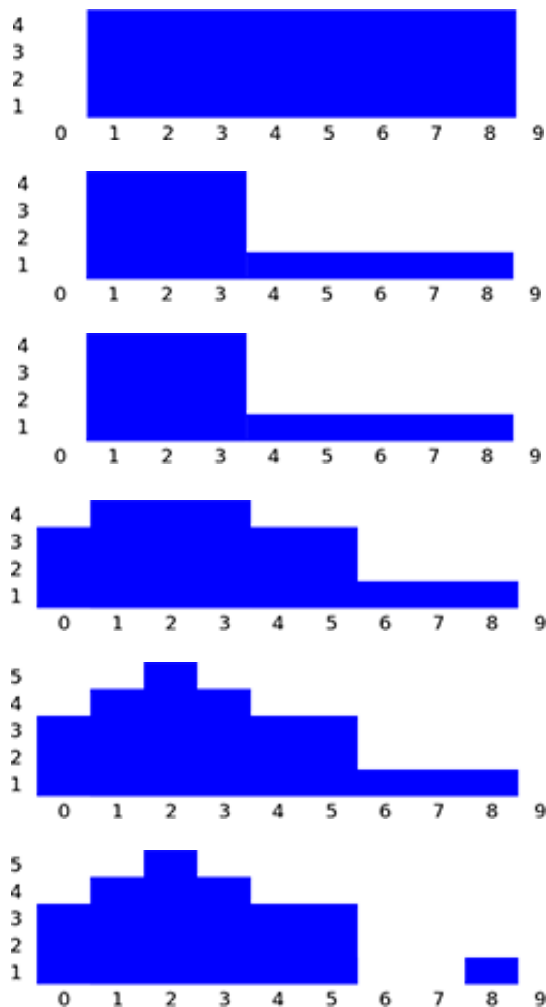
Jūsu uzdevums ir noskaidrot sienas formu, kad sienas būvēšana beigusies.

Piemērs

Pieņemsim, ka sienai ir 10 kolonnas un 6 būvēšanas darbības. Visi norādītie intervāli iekļauj galapunktus. Sienas forma pēc katras darbības beigām ir norādīta zemāk.

darbība	tips	intervāls	augstums
0	pievieno	kolonnas 1 līdz 8	4
1	noņem	kolonnas 4 līdz 9	1
2	noņem	kolonnas 3 līdz 6	5
3	pievieno	kolonnas 0 līdz 5	3
4	pievieno	kolonnu 2	5
5	noņem	kolonnas 6 līdz 7	0

Tā kā visas kolonnas sākumā ir tukšas, tad pēc nultās darbības katrā no pirmās līdz astotajai kolonnai būs 4 ķieģeļi. Nultā un devītā kolonna joprojām ir tukša. Pirmajā darbībā no ceturtās līdz astotajai kolonnai ķieģeļi tiks noņemti līdz katrā no tām būs atlicis tikai viens ķieģelis un devītā kolonna paliek tukša. Kolonnas no nultās līdz trešajai nav ieļautas intervālā, tāpēc tās netiek mainītas. Otrā darbība neizraisa nekādas izmaiņas, jo no trešās līdz sestajai kolonnai nevienā nav vairāk par 5 ķieģeļiem. Pēc trešās darbības ķieģeļu skaits nultajā, ceturtajā un piektajā kolonnā tiek palielināts līdz 3. Pēc ceturtās darbības otrajā kolonnā ir pieci ķieģeļi. Piektajā darbībā visi ķieģeļi tiek izņemti no sestās un septītās kolonnas.



Uzdevums

Dotam k darbību aprakstam, lūdzu noskaidrojiet cik ķieģeļu atradīsies katrā no kolonnām pēc tam kad visas darbības būs izpildītas. Jums jārealizē funkcija `buildWall`.

- `buildWall(n, k, op, left, right, height, finalHeight)`
 - n : sienas kolonnu skaits.
 - k : darbību skaits.
 - `op`: masīvs garumā k ; `op[i]` ir i -tās darbības veids : 1 norāda, ka tā ir pievienošanas darbība, bet 2, ka noņemšanas darbība, for $0 \leq i \leq k - 1$.
 - `left` un `right`: masīvi garumā k ; kolonnu itervāls i -tajai darbībai sākas ar kolonnu `left[i]` un beidzas ar `right[i]` kolonnu (ieskaitot abus gala punktus `left[i]` un `right[i]`), for $0 \leq i \leq k - 1$. Vienmēr būs spēkā `left[i] \leq right[i]`.
 - `height`: masīvs garumā k ; `height[i]` ir i -tās darbības augstuma parametrs h , for $0 \leq i \leq k - 1$.
 - `finalHeight`: masīvs garumā n ; Jūsu iegūto i -tajā kolonnā esošo ķieģeļu skaits jāieraksta masīva elementā `finalHeight[i]`, for $0 \leq i \leq n - 1$.

Apakšuzdevumi

Visiem apakšuzdevumiem augstuma parametrs visām darbībām ir nenegatīvs vesels skaitlis mazāks vai vienāds ar **100,000**.

apakšuzdevums	punkti	n	k	piezīmes
1	8	$1 \leq n \leq 10,000$	$1 \leq k \leq 5,000$	nav papildus ierobežojumu
2	24	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	visas pievienošanas darbības ir pirms visām noņemšanas darbībām
3	29	$1 \leq n \leq 100,000$	$1 \leq k \leq 500,000$	nav papildus ierobežojumu
4	39	$1 \leq n \leq 2,000,000$	$1 \leq k \leq 500,000$	nav papildus ierobežojumu

Realizācijas detaļas

Jums jāiesūta tieši viens fails ar nosaukumu `wall.c`, `wall.cpp` vai `wall.pas`. Šim failam jārealizē apakšprogramma, kas aprakstīta iepriekš ar norādīto signatūru. C/C++ programmas gadījumā Jums programmas tekstā jāiekļauj `wall.h`.

C/C++ programma

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

Pascal programma

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

Paraugtestētājs

Paraugtestētājs lasa ievadu šādā formātā:

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