

Distributing Candies

Fifi priprema n kutija bombona koje će podijeliti ostatku cvjetnog društva. Kutije će označiti cijelim brojevima između 0 i $n - 1$. Sve su kutije na početku prazne, a i -ta ($0 \leq i \leq n - 1$) kutija ima kapacitet $c[i]$ (u nju stane najviše $c[i]$ bombona).

Fifi će točno q dana pripremati kutije. Preciznije, j -toga ($0 \leq j \leq q - 1$) će dana odabrati brojeve $l[j]$, $r[j]$ i $v[j]$ ($0 \leq l[j] \leq r[j] \leq n - 1$ i $v[j] \neq 0$) temeljem kojih će promijeniti bombončano stanje u kutijama, i to tako da će za svaku kutiju k za koju vrijedi $l[j] \leq k \leq r[j]$ napraviti sljedeće:

- Ako je $v[j] > 0$, Fifi će dodavati k -toj kutiji bombone (jedan po jedan), sve dok nije dodao točno $v[j]$ bombona ili je kutija dosegla svoj kapacitet. Drugim riječima, ako se prije ovoga dana u kutiji nalazilo p bombona, na kraju dana će kutija sadržavati $\min(c[k], p + v[j])$ bombona.
- Ako je $v[j] < 0$, Fifi će micati bombone iz k -te kutije (jedan po jedan), sve dok nije maknuo točno $-v[j]$ bombona iz kutije ili je kutija postala prazna. Drugim riječima, ako se prije ovoga dana u kutiji nalazilo p bombona, a kraju dana će kutija sadržavati $\max(0, p + v[j])$ bombona.

Vaš je zadatak odrediti broj bombona u svakoj kutiji nakon q dana.

Implementacijski detalji

Potrebno je implementirati sljedeću proceduru:

```
int[] distribute_candies(int[] c, int[] l, int[] r, int[] v)
```

- c : polje duljine n . Za svaki $0 \leq i \leq n - 1$, $c[i]$ označava kapacitet kutije i .
- l , r i v : tri polja duljine q . Na dan j , za svaki $0 \leq j \leq q - 1$, Fifi će odraditi operaciju objašnjenu u tekstu zadatka temeljem parametara $l[j]$, $r[j]$ i $v[j]$.
- Procedura treba vratiti polje duljine n . Označimo to polje sa s . Za svaki $0 \leq i \leq n - 1$, $s[i]$ treba predstavljati broj bombona u kutiji i nakon q dana.

Primjeri

Primjer 1

Promotrimo sljedeći poziv procedure:

```
distribute_candies([10, 15, 13], [0, 0], [2, 1], [20, -11])
```

Vidimo da kutija 0 ima kapacitet od 10 bombona, kutija 1 ima kapacitet od 15 bombona i kutija 2 ima kapacitet od 13 bombona.

Na kraju dana 0, kutija 0 sadrži $\min(c[0], 0 + v[0]) = 10$ bombona, kutija 1 sadrži $\min(c[1], 0 + v[0]) = 15$ bombona, a kutija 2 sadrži $\min(c[2], 0 + v[0]) = 13$ bombona.

Na kraju dana 1, kutija 0 sadrži $\max(0, 10 + v[1]) = 0$ bombona, kutija 1 sadrži $\max(0, 15 + v[1]) = 4$ bombona. Budući da je $2 > r[1]$, nema promjene bombončanog stanja u kutiji 2. Broj bombona na kraju svakog dana sažet je u sljedećoj tablici:

| Dan | Kutija 0 | Kutija 1 | Kutija 2 |
|-----|----------|----------|----------|
| 0 | 10 | 15 | 13 |
| 1 | 0 | 4 | 13 |

Shodno tome, procedura treba vratiti $[0, 4, 13]$.

Ograničenja

- $1 \leq n \leq 200\,000$
- $1 \leq q \leq 200\,000$
- $1 \leq c[i] \leq 10^9$ (za svaki $0 \leq i \leq n - 1$)
- $0 \leq l[j] \leq r[j] \leq n - 1$ (za svaki $0 \leq j \leq q - 1$)
- $-10^9 \leq v[j] \leq 10^9, v[j] \neq 0$ (za svaki $0 \leq j \leq q - 1$)

Podzadaci

1. (3 boda) $n, q \leq 2000$
2. (8 bodova) $v[j] > 0$ (za svaki $0 \leq j \leq q - 1$)
3. (27 bodova) $c[0] = c[1] = \dots = c[n - 1]$
4. (29 bodova) $l[j] = 0$ i $r[j] = n - 1$ (za svaki $0 \leq j \leq q - 1$)
5. (33 boda) Nema dodatnih ograničenja.

Ogledni ocjenjivač

Ogledni ocjenjivač čita ulaz u sljedećem obliku:

- redak 1: n
- redak 2: $c[0] \ c[1] \ \dots \ c[n - 1]$
- redak 3: q
- redak $4 + j$ ($0 \leq j \leq q - 1$): $l[j] \ r[j] \ v[j]$

Ogledni ocjenjivač ispisuje vaše odgovore u sljedećem obliku:

- redak 1: $s[0] \ s[1] \ \dots \ s[n-1]$