

Padavine

Ljeto je počelo, ali kiša ne odustaje. Svako malo u Sarajevu se formira neki oblak iznad određenog naselja, te ravnomjerno po naselju rasporedi određenu količinu kiše.

Naime, Sarajevo je relativno izdužen grad i u ovom zadatku ćemo ga predstavljati u jednoj dimenziji, kao liniju sa **N** polja. Tačnije sastoji se od polja s indeksima 0, 1, 2, ..., N-1. Naselje ćemo predstaviti uzastopnim nizom polja unutar grada, tako npr. jedno može biti 4, 5, 6, dok drugo može biti 1, 2, 3, 4, 5 ..., N-1, tj. čitav grad. Pojam naselja i zapravo svaku kombinaciju brojeva (**A, B**) ($0 \leq A \leq B \leq N - 1$) možemo smatrati naseljem koje počinje u **A** i završava u **B** inkluzivno.

Sada kada znamo situaciju s vremenom i opis grada, studenti Elektrotehničkog fakulteta Sarajevo (ETF) su odlučili da se poigraju s vremenom, tačnije da simuliraju padavine. Oni su shvatili da se u gradu dešavaju samo dvije stvari s padavinama, a to su:

- Oblak se formira nad nekim naseljem (A, B), te svako polje u tom naselju dodatno dopuni kišom za tačno **S** kapljica kiše,
- Vjetar puhne u nekom naselju (A, B), te se kiša u tom naselju raspodijeli tako da svako polje naselja na kraju ima istu količinu kapljica kiše. Kako su kapljice kiše samo cijeli brojevi, vjetar će skroz odnijeti višak kiše ako drugačije ne bude moguće da imaju sva polja isto kiše. Naravno vjetar će odbaciti minimalno moguće kiše da se može gornji uslov ispuniti (ispod na primjeru objašnjeno).

Grad na početku uvijek ima **0** jedinica kiše na svakom svom polju.

Npr. recimo da je grad sačinjen od $N = 6$ polja. Neka se formirao oblak nad naseljem (1, 3) i dopunio njegova polja sa $S = 6$ jedinica kiše. Nakon toga je došao drugi oblak nad (2, 5) i imao $S = 3$. Sada grad izgleda ovako:

Polje	0	1	2	3	4	5
Jedinice kiše	0	6	9	9	3	3

Sada recimo da puhne vjetar nad naseljem (2, 4). U tom naselju imamo ukupno $9+9+3$ kapljice kiše, a 3 polja, te će vjetar puhnuti tako da se po dvije kapljice iz polja 2 i 3 premjeste na polje 4, i onda će sva polja imati po tačno 7 kapljica kiše. Kako sva polja imaju isto kapljica s istom količinom kao i prije vjetra, nema potrebe da vjetar nosi ikakve kapljice iz naselja. Stanje grada će sada biti:

Polje	0	1	2	3	4	5
Jedinice kiše	0	6	7	7	7	3

Neka sada vjetar puhne u naselju (0, 2). Tu imamo ukupno 13 kapljica kiše, ali 3 polja. Polje 0 je prazno, pa će s polja 1 dvije kapljice da odlete na polje 0 i još dvije s polja 2 na polje 0 (jedna od mogućnosti). Nakon ovoga imali bi nepravilno stanje grada:

Polje	0	1	2	3	4	5
Jedinice kiše	4	4	5	7	7	3

Naime na polju 2 je ostalo viška kapljica, te će isti vjetar to da odnese kako bi sva polja u naselju imala isto kapljica, i finalno stanje nakon ovog vjetra će biti:

Polje	0	1	2	3	4	5
Jedinice kiše	4	4	4	7	7	3

Vjetar nikad neće nositi iz naselja više kapljica nego što je potrebno, tj. količina kapljica u naselju u kojem je puhao vjetar će uvijek biti maksimalna moguća tako da svako polje ima istu količinu na sebi.

Studenti su sada uzeli da mjere padavine, te mogu se pitati koja je ukupna količina kapljica u nekom naselju (A, B). Iz gornjeg primjera, ako je grad u stanju kao na zadnjoj tabeli, onda su odgovori na neka pitanja:

- u naselju (0, 3) ima tačno 19 kapljica vode,
- u naselju (2, 2) ima tačno 4 kapljice vode,
- u naselju (0, 5) ima tačno 29 kapljica vode.

Vaš zadatak je sada da pomognete studentima ETF-a (jer imaju ionako previše obaveza) tako što im isprogramirate ovu simulaciju.

Detalji implementacije

Vaš zadatak je da implementirate četiri funkcije i to prva je:

```
void init(int N, int T);
```

Ona će se po pokretanju programa tačno jednom pozvati i to prije svih drugih funkcija koje vi implementirate.

N je veličina grada iz teksta zadatka,

T je broj podzadatka i može biti broj iz skupa {1, 2, 3, 4, 5, 6}, ovaj parametar vam samo treba služiti kao pomoć pri izradi zadatka, te nije nužno ga uzimati u obzir.

Druga funkcija je:

```
void oblak(int A, int B, int S);
```

Ovu funkciju ćemo vam pozvati kad god se formira oblak nad naseljem.

A i **B** su granice naselja (A, B) nad kojim se formirao oblak,

S je broj kapljica koje će pasti na svako od polja u naselju zbog ovog oblaka.

Treća funkcija je:

```
void vjetar(int A, int B);
```

Ovu funkciju ćemo vam pozvati kad god puhne vjetar nad nekim naseljem.

A i **B** su granice naselja (A, B) u kojem puše vjetar.

Četvrta, koja ima i povratnu vrijednost je:

```
long long sum(int A, int B);
```

Ovu funkciju pozivamo kad god studenti hoće da mjere količinu kapljica u nekom naselju

A i **B** su granice naselja (A, B) u kojem mjerimo ukupnu količinu kapljica.

Povratna vrijednost ove funkcije treba da odgovara tačnom broju kapljica u naselju (A, B) u trenutku kada je mjerenje izvršeno.

Primjer

<i>Pozivi funkcija</i>	<i>Povrat funkcije</i>
<code>init(6, 1);</code>	-
<code>oblak(1, 3, 6);</code>	-
<code>oblak(2, 5, 3);</code>	-
<code>vjetar(2, 4);</code>	-
<code>vjetar(0, 2);</code>	-
<code>sum(0, 3);</code>	19
<code>sum(2, 2)</code>	4
<code>sum(0, 5);</code>	29
<code>init(5, 1);</code>	-
<code>oblak(2, 4, 5);</code>	-
<code>oblak(1, 3, 7);</code>	-
<code>sum(1, 3);</code>	31
<code>sum(3, 4);</code>	17
<code>vjetar(3, 4);</code>	-
<code>sum(3, 4);</code>	16
<code>vjetar(0, 2);</code>	-
<code>sum(0, 2);</code>	18
<code>sum(0, 0);</code>	6
<code>sum(1, 1);</code>	6
<code>oblak(0, 4, 7);</code>	-
<code>vjetar(4, 4);</code>	-

Ograničenja na resurse i opis subtaskova

Generalno ograničenje je A i B u svim funkcijama uvijek pripadaju gradu, te je:

$$0 \leq A \leq B \leq N - 1$$

Također je svaka količina padavine iz oblaka S :

$$S \leq 1000$$

Za potrebe ograničenja označit ćemo sa M broj akcija koje će se desiti. Akcija je poziv svake od vaših funkcija, ne uključujući funkciju `ini`. T iz `init` funkcije će uvijek odgovarati broju podzadatka iz opisa ispod.

Zadatak se sastoji od 6 podzadatka sa sljedećim dodatnim ograničenjima i bodovanjem:

1. Podzadatak 1 (5 bodova)

$$10 \leq N \leq 5000$$
$$10 \leq M \leq 1000$$

2. Podzadatak 2 (13 bodova)

$$10 \leq N \leq 500000$$
$$10 \leq M \leq 100000$$

Dodatno vrijedi da nikada neće puhati vjetar, te da će oblaci biti samo nad naseljima veličine 1, tj. $A = B$ za svaki poziv funkcije `oblak`.

3. Podzadatak 3 (15 bodova)

$$10 \leq N \leq 500000$$
$$10 \leq M \leq 10000$$

Dodatno vrijedi da će prvo se formirati svi oblaci, tj. prvo će se izvršiti svi pozivi funkcije `oblak`, nakon čega idu samo pozivi funkcije `vjetar`, te tek na kraju studenti mjere sa pozivima funkcije `sum`. Pored toga vjetar će biti jako rijedak i očekujte da će manje od 11 svih M akcija biti vjetar.

4. Podzadatak 4 (23 bodova)

$$10 \leq N \leq 10000000$$
$$10 \leq M \leq 1000$$

5. Podzadatak 4 (23 bodova)

$$10 \leq N \leq 500000$$
$$10 \leq M \leq 100000$$

6. Podzadatak 4 (21 bodova)

$$10 \leq N \leq 10000000$$
$$10 \leq M \leq 100000$$

Vremenska i memorijska ograničenja su dostupna na sistemu za ocjenjivanje.