Meetings

Imamo ukupno N planina poredanih u jednom horizontalnom redu i numerisanih od 0 do $N-1$ sa lijeva na desno. Visina planine i je H_i ($0 \leq i \leq N-1$). Tačno jedna osoba živi na vrhu svake planine.

Treba da organizujete tačno Q sastanaka numerisanih od 0 do $Q-1$. Na sastanku j ($0 \leq j \leq Q-1$) moraju učestvovati sve osobe koje žive na planinama od L_j do R_j, uključivo ($0 \leq L_j \leq R_j \leq N-1$). Za ovaj sastanak i morate izabrati planinu x gdje će se sastanak održati ($L_j \leq x \leq R_j$). Ovaj sastanak uključuje troškove koji se računaju na sljedeći način:

- Trošak cijelog sastanka je zbir troškova za svakog od učesnika. Troškovi se računaju u odnosu na izbor planine za ovaj sastanak.
- Trošak jednog učesnika sa svake od planina y ($L_j \leq y \leq R_j$) je maksimalna visina planina koje se nalaze između planina x i y, uključivo.
- U specijalnom slučaju, trošak učesnika sa planine x je H_x, tj. visina planine x.

Za svaki sastanak želite naći najmanji mogući trošak njegovog održavanja.

Primjetimo da se svi učesnici vraćaju na svoje planine nakon svakog sastanka tako da na troškove nekog sastanka ne utiču prethodni sastanci.

Detalji implementacije

Vi treba da implementirate sljedeću funkciju:

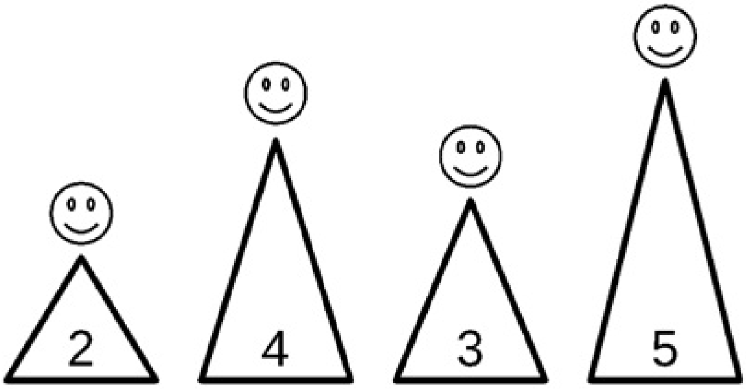
```
int64[] minimum_costs(int[] H, int[] L, int[] R)
```

- `H`: jedan niz dužine N koji predstavlja visine planina.
- `L` i `R`: nizovi dužine Q koji predstavljaju interval niza učesnika u sastanku.
- Ova funkcija treba da vrati jedan niz C dužine Q. Vrijednost C_j ($0 \leq j \leq Q-1$) mora biti jednaka minimalnom mogućem trošku održavanja sastanka j.
- Primjetimo da su vrijednosti N i Q dužine nizova i da se mogu dobiti kako je to opisano u napomeni o implementaciji.

Primjer

Neka su $N = 4$, $H = [2, 4, 3, 5]$, $Q = 2$, $L = [0, 1]$, i $R = [2, 3]$.

Grader poziva `minimum_costs([2, 4, 3, 5], [0, 1], [2, 3])`.



Za sastanak $j = 0$ imamo da je $L_j = 0$ i $R_j = 2$, tako da će ovom sastanku prisustovati osobe koje žive na planinama 0, 1, i 2. Ako bi se planina 0 izabrala kao mjesto sastanka onda bi se trošak tog sastanka računao na sljedeći način:

- Trošak učesnika sa planine 0 je $\max\{H_0\} = 2$.
- Trošak učesnika sa planine 1 je $\max\{H_0, H_1\} = 4$.
- Trošak učesnika sa planine 2 je $\max\{H_0, H_1, H_2\} = 4$.
- Prema tome, trošak cijelog sastanka 0 je $2 + 4 + 4 = 10$.

Nije moguće održati sastanak 0 sa manjim troškom tako da je minimalan mogući trošak sastanka 0 jednak 10.

Za sastanak $j = 1$ imamo da je $L_j = 1$ i $R_j = 3$ tako da će ovom sastanku prisustovati osobe koje žive na planinama 1, 2, i 3. Ukoliko bi planina 2 bila izabrana kao mjesto sastanka onda bi se trošak sastanka 1 računao na sljedeći način:

- Trošak učesnika sa planine 1 je $\max\{H_1, H_2\} = 4$.
- Trošak učesnika sa planine 2 je $\max\{H_2\} = 3$.
- Trošak učesnika sa planine 3 je $\max\{H_2, H_3\} = 5$.
- Prema tome, trošak cijelog sastanka 1 je $4 + 3 + 5 = 12$.

Nije moguće održati sastanak 1 sa manjim troškom tako da je minimalan mogući trošak sastanka 1 jednak 12.

U zip-datoteci u prilogu ovom zadatku, datoteke sample-01-in.txt i sample-01-out.txt odgovaraju ovom primjeru. U zip-datoteci postoji i nekoliko drugih primjera sa datim parovima ulaznih i izlaznih datoteka.

Ograničenja

- $1 \leq N \leq 750\,000$
- $1 \leq Q \leq 750\,000$
- $1 \leq H_i \leq 1\,000\,000\,000$ $(0 \leq i \leq N - 1)$

- $0 \leq L_j \leq R_j \leq N-1$ ($0 \leq j \leq Q-1$)
- $(L_j, R_j) \neq (L_k, R_k)$ ($0 \leq j < k \leq Q-1$)

Podzadaci

1. (4 boda) $N \leq 3\,000$, $Q \leq 10$
2. (15 bodova) $N \leq 5\,000$, $Q \leq 5\,000$
3. (17 bodova) $N \leq 100\,000$, $Q \leq 100\,000$, $H_i \leq 2$ ($0 \leq i \leq N-1$)
4. (24 boda) $N \leq 100\,000$, $Q \leq 100\,000$, $H_i \leq 20$ ($0 \leq i \leq N-1$)
5. (40 bodova) Nema dodatnih ograničenja

Testni grader

Testni grader čita ulaz u sljedećem formatu:

- linija 1: $N\ Q$
- linija 2: $H_0\ H_1 \cdots H_{N-1}$
- linije $3+j$ ($0 \leq j \leq Q-1$): $L_j\ R_j$

Testni grader štampa vraćenu vrijednost funkcije `minimum_costs` u sljedećem formatu:

- linije $1+j$ ($0 \leq j \leq Q-1$): C_j