



## Radio Towers

Sunt  $N$  turnuri radio în Jakarta. Turnurile sunt amplasate în linie dreaptă și numerotate de la 0 la  $N - 1$  de la stânga la dreapta. Pentru fiecare  $i$  astfel încât  $0 \leq i \leq N - 1$ , înălțimea turnului  $i$  este de  $H[i]$  metri. Înălțimile turnurilor sunt **distincte**.

Pentru o oarecare valoare de interferență pozitivă  $\delta$ , turnurile din perechea  $i$  și  $j$  (unde  $0 \leq i < j \leq N - 1$ ) pot comunica între ele dacă și numai dacă există un turn intermediar  $k$ , astfel încât

- turnul  $i$  este amplasat în stânga turnului  $k$  și turnul  $j$  în dreapta turnului  $k$ , adică,  $i < k < j$ , și
- înălțimile turnurilor  $i$  și  $j$  sunt ambele cel mult  $H[k] - \delta$  metri.

Pak Dengklek dorește să închirieze unele turnuri pentru rețeaua sa radio. Sarcina voastră este să răspundeți la  $Q$  întrebări ale lui Pak Dengklek, care au următoarea formă: fiind cunoscuți parametrii  $L, R$  și  $D$  ( $0 \leq L \leq R \leq N - 1$  și  $D > 0$ ), care este numărul maximal de turnuri pe care Pak Dengklek le poate închiria, presupunând că:

- Pak Dengklek poate închiria doar turnuri ai căror indici sunt între  $L$  și  $R$  (inclusiv), și
- valoarea de interferență  $\delta$  este  $D$ , și
- pentru orice pereche de turnuri radio dintre cele închiriate de Pak Dengklek, turnurile din acea pereche să poată comunica între ele.

De notat că două turnuri închiriate pot comunica folosind un turn intermediar  $k$  indiferent dacă turnul  $k$  este închiriat sau nu.

## Detalii de Implementare

Veți implementa următoarele proceduri:

```
void init(int N, int[] H)
```

- $N$ : numărul de turnuri radio.
- $H$ : un tablou unidimensional de lungime  $N$ , elementele căruia descriu înălțimile turnurilor.
- Această procedură este apelată exact o dată, înainte de orice apel a `max_towers`.

```
int max_towers(int L, int R, int D)
```

- $L, R$ : capetele unui interval de turnuri.
- $D$ : valoarea  $\delta$ .
- Această procedură trebuie să returneze numărul maxim de turnuri radio, pe care Pak Dengklek le poate închiria pentru rețeaua sa radio, dacă are permisiunea de a închiria doar turnuri dintre turnul  $L$  și turnul  $R$  (inclusiv) cu valoarea  $\delta$  egală cu  $D$ .
- Această procedură este apelată de exact  $Q$  ori.

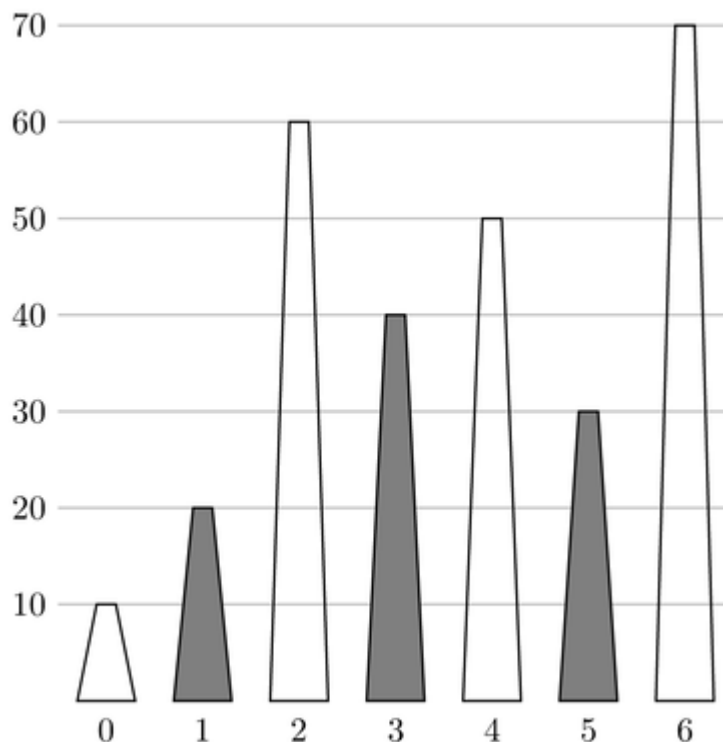
## Exemplu

Considerăm următoarea secvență de apeluri:

```
init(7, [10, 20, 60, 40, 50, 30, 70])
```

```
max_towers(1, 5, 10)
```

Pak Dengklek poate închiria turnurile 1, 3, și 5. Acest exemplu este ilustrat în următoarea imagine, unde trapezele evidențiate reprezintă turnurile închiriate.



Turnurile 3 și 5 pot comunica folosind în calitate de intermediar turnul 4, deoarece  $40 \leq 50 - 10$  și  $30 \leq 50 - 10$ . Turnurile 1 și 3 pot comunica folosind în calitate de intermediar turnul 2. Turnurile 1 și 5 pot comunica folosind în calitate de intermediar turnul 3. Nu există o cale de a închiria mai mult decât 3 turnuri, astfel procedura va returna 3.

```
max_towers(2, 2, 100)
```

Există doar un turn în interval, prin urmare Pak Dengklek poate închiria doar un turn. Astfel procedura va returna 1.

```
max_towers(0, 6, 17)
```

Pak Dengklek poate închiria turnurile 1 și 3. Turnurile 1 și 3 pot comunica între ele folosind în calitate de intermediar turnul 2, deoarece  $20 \leq 60 - 17$  și  $40 \leq 60 - 17$ . Nu există o cale de a închiria mai mult decât 2 turnuri, astfel procedura va returna 2.

## Restricții

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq H[i] \leq 10^9$  (pentru fiecare  $i$  astfel încât  $0 \leq i \leq N - 1$ )
- $H[i] \neq H[j]$  (pentru fiecare  $i$  și  $j$  astfel încât  $0 \leq i < j \leq N - 1$ )
- $0 \leq L \leq R \leq N - 1$
- $1 \leq D \leq 10^9$

## Subtask-uri

1. (4 puncte) Există un turn  $k$  ( $0 \leq k \leq N - 1$ ) astfel încât
  - pentru fiecare  $i$  astfel încât  $0 \leq i \leq k - 1$ :  $H[i] < H[i + 1]$  și
  - pentru fiecare  $i$  astfel încât  $k \leq i \leq N - 2$ :  $H[i] > H[i + 1]$ .
2. (11 puncte)  $Q = 1$ ,  $N \leq 2000$
3. (12 puncte)  $Q = 1$
4. (14 puncte)  $D = 1$
5. (17 puncte)  $L = 0$ ,  $R = N - 1$
6. (19 puncte) Valoarea  $D$  este aceeași pentru toate apelurile procedurii `max_towers`.
7. (23 puncte) Fără alte restricții.

## Grader-ul Local

Grader-ul local citește datele de intrare în următorul format:

- linia 1:  $N$   $Q$
- linia 2:  $H[0]$   $H[1]$  ...  $H[N - 1]$
- linia  $3 + j$  ( $0 \leq j \leq Q - 1$ ):  $L$   $R$   $D$  pentru întrebarea  $j$

Grader-ul local afișează răspunsurile voastre în următorul format:

- linia  $1 + j$  ( $0 \leq j \leq Q - 1$ ): valoarea returnată de `max_towers` pentru întrebarea  $j$

