



Radiopyloner

I Yogya finns N radiopyloner. Dessa är belägna längs en rak linje och är numrerade mellan 0 och $N - 1$ i ordning. För varje i där $0 \leq i \leq N - 1$, är pylon nummer i $H[i]$ famnar hög. Dessa höjder är alla **olika**.

För en viss problemfrekvens δ , så kan 2 pyloner i och j (där $0 \leq i < j \leq N - 1$) kommunicera med varandra om och bara om en annan pylon k finns, där

- pylon i är ej höger om pylon k medan pylon j är höger om pylon k , med andra ord, $i < k < j$, och
- höjderna av pylon i och pylon j är båda ej mer än $H[k] - \delta$ famnar.

Paul Degkvarn vill hyra några radiopyloner för sin nya radioanordning. Du ska svara på Q frågor från Paul Degkvarn som är på följande form: Given värdena L, R och D ($0 \leq L \leq R \leq N - 1$ och $D > 0$), vad är maximala mängden pyloner som Paul Degkvarn kan hyra, under följande premisser:

- Paul Degkvarn kan bara hyra pyloner vars index är mellan L och R (inklusive), och
- problemfrekvensen δ är D , och
- varje par av pyloner som Paul Degkvarn hyr skall kunna kommunicera med varandra.

Obs: 2 hyrda pyloner kan kommunicera genom en mellanpylon k , även om k ej är hyrd.

Implementation Details

You should implement the following procedures:

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

- N : the number of radio towers.
- H : an array of length N describing the tower heights.
- This procedure is called exactly once, before any calls to `max_towers`.

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

- L, R : the boundaries of a range of towers.
- D : the value of δ .

- This procedure should return the maximum number of radio towers Pak Dengklek can lease for his new radio network if he is only allowed to lease towers between tower L and tower R (inclusive) and the value of δ is D .
- This procedure is called exactly Q times.

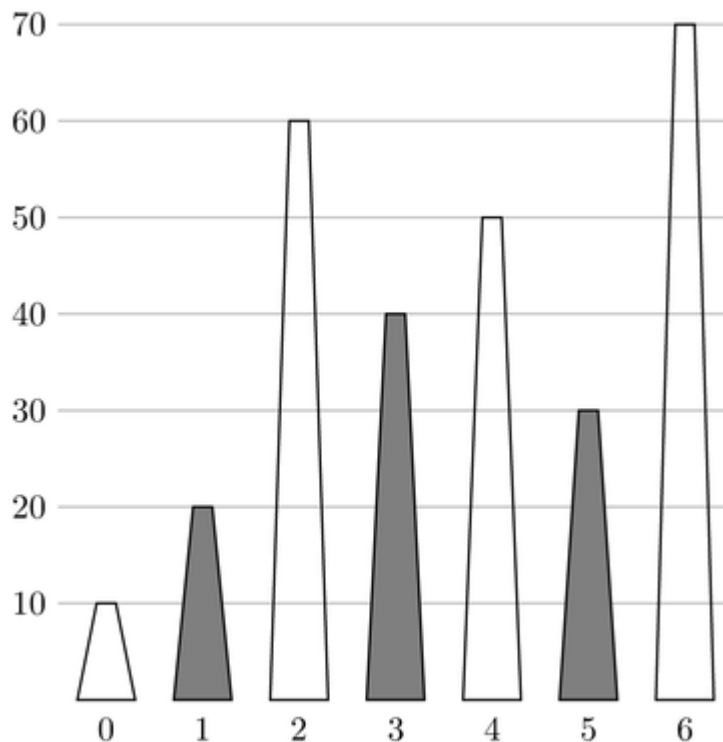
Example

Consider the following sequence of calls:

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

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

Pak Dengklek can lease towers 1, 3, and 5. The example is illustrated in the following picture, where shaded trapezoids represent leased towers.



Towers 3 and 5 can communicate using tower 4 as an intermediary, since $40 \leq 50 - 10$ and $30 \leq 50 - 10$. Towers 1 and 3 can communicate using tower 2 as an intermediary. Towers 1 and 5 can communicate using tower 3 as an intermediary. There is no way to lease more than 3 towers, therefore the procedure should return 3.

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

There is only 1 tower in the range, thus Pak Dengklek can only lease 1 tower. Therefore the procedure should return 1.

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

Pak Dengklek can lease towers 1 and 3. Towers 1 and 3 can communicate using tower 2 as an intermediary, since $20 \leq 60 - 17$ and $40 \leq 60 - 17$. There is no way to lease more than 2 towers, therefore the procedure should return 2.

Constraints

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq H[i] \leq 10^9$ (for each i such that $0 \leq i \leq N - 1$)
- $H[i] \neq H[j]$ (for each i and j such that $0 \leq i < j \leq N - 1$)
- $0 \leq L \leq R \leq N - 1$
- $1 \leq D \leq 10^9$

Subtasks

1. (4 points) There exists a tower k ($0 \leq k \leq N - 1$) such that
 - for each i such that $0 \leq i \leq k - 1$: $H[i] < H[i + 1]$, and
 - for each i such that $k \leq i \leq N - 2$: $H[i] > H[i + 1]$.
2. (11 points) $Q = 1$, $N \leq 2000$
3. (12 points) $Q = 1$
4. (14 points) $D = 1$
5. (17 points) $L = 0$, $R = N - 1$
6. (19 points) The value of D is the same across all `max_towers` calls.
7. (23 points) No additional constraints.

Sample Grader

The sample grader reads the input in the following format:

- line 1: $N Q$
- line 2: $H[0] H[1] \dots H[N - 1]$
- line $3 + j$ ($0 \leq j \leq Q - 1$): $L R D$ for question j

The sample grader prints your answers in the following format:

- line $1 + j$ ($0 \leq j \leq Q - 1$): the return value of `max_towers` for question j