



Torres de Rádio

Há N torres de rádio em Jakarta. As torres estão localizadas ao longo de uma linha reta e numeradas de 0 a $N - 1$ da esquerda para a direita. Para cada i tal que $0 \leq i \leq N - 1$, a altura da torre i é $H[i]$ metros. As alturas das torres são **distintas**.

Para algum valor positivo de interferência δ , um par de torres i e j (onde $0 \leq i < j \leq N - 1$) podem se comunicar uma com a outra se e somente se houver uma torre intermediária k tal que

- a torre i está à esquerda da torre k e a torre j está à direita da torre k , isto é, $i < k < j$ e
- as alturas da torre i e da torre j são ambas no máximo $H[k] - \delta$ metros.

Pak Dengklek quer alugar algumas torres de rádio para sua nova rede de rádio. Sua tarefa é responder Q perguntas de Pak Dengklek que são da seguinte forma: dados os parâmetros L , R e D ($0 \leq L \leq R \leq N - 1$ e $D > 0$), qual é o número máximo de torres que Pak Dengklek pode alugar, assumindo que

- Pak Dengklek só pode alugar torres com índices entre L e R (inclusive) e
- o valor de interferência δ é D e
- qualquer par de torres de rádio que Pak Dengklek aluga deve ser capaz de se comunicar uma com a outra.

Note que duas torres alugadas podem se comunicar usando uma torre intermediária k , independentemente de a torre k ser alugada ou não.

Detalhes de Implementação

Você deve implementar os seguintes procedimentos:

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

- N : o número de torres de rádio.
- H : um vetor de tamanho N descrevendo as alturas das torres.
- Este procedimento é chamado exatamente uma vez, antes de qualquer chamada para `max_towers`.

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

- L , R : os limites de um intervalo de torres.

- D : o valor de δ .
- Este procedimento deve retornar o número máximo de torres de rádio que Pak Dengklek pode alugar para sua nova rede de rádio se ele só puder alugar torres entre a torre L e a torre R (inclusive) e o valor de δ for D .
- Este procedimento é chamado exatamente Q vezes.

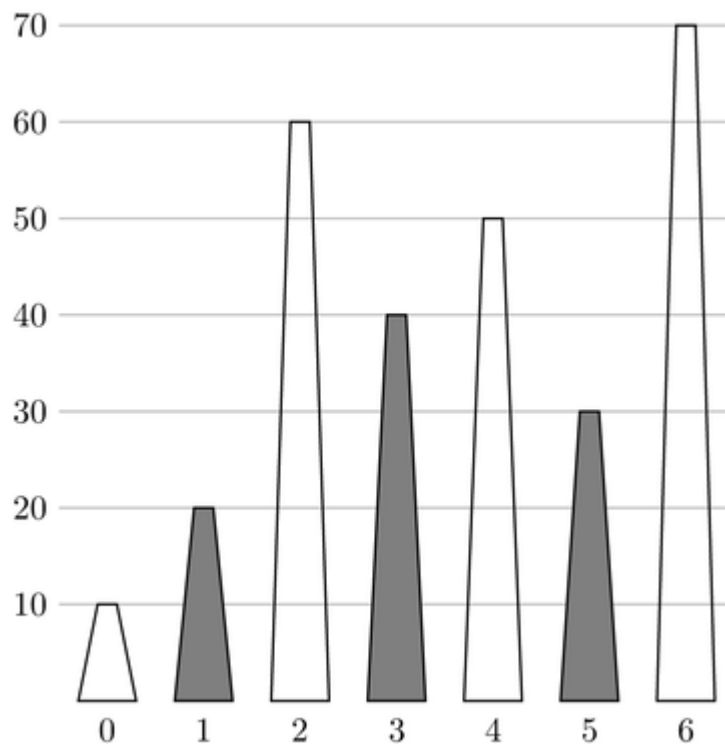
Exemplo

Considere a seguinte sequência de chamadas:

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

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

Pak Dengklek pode alugar as torres 1, 3 e 5. O exemplo está ilustrado na figura a seguir, onde trapézios sombreados representam torres alugadas.



As torres 3 e 5 podem se comunicar usando a torre 4 como uma intermediária, pois $40 \leq 50 - 10$ e $30 \leq 50 - 10$. As torres 1 e 3 podem se comunicar usando a torre 2 como uma intermediária. As torres 1 e 5 podem se comunicar usando a torre 3 como uma intermediária. Não há maneira de alugar mais do que 3 torres, portanto, o procedimento deve retornar 3.

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

Há apenas 1 torre no intervalo, então Pak Dengklek só consegue alugar 1 torre. Portanto, o procedimento deve retornar 1.

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

Pak Dengklek consegue alugar as torres 1 e 3. As torres 1 e 3 podem se comunicar usando a torre 2 como uma intermediária, pois $20 \leq 60 - 17$ e $40 \leq 60 - 17$. Não há maneira de alugar mais do que 2 torres, portanto, o procedimento deve retornar 2.

Restrições

- $1 \leq N \leq 100\,000$
- $1 \leq Q \leq 100\,000$
- $1 \leq H[i] \leq 10^9$ (para cada i tal que $0 \leq i \leq N - 1$)
- $H[i] \neq H[j]$ (para cada i e j tais que $0 \leq i < j \leq N - 1$)
- $0 \leq L \leq R \leq N - 1$
- $1 \leq D \leq 10^9$

Subtarefas

1. (4 pontos) Existe uma torre k ($0 \leq k \leq N - 1$) tal que
 - para cada i tal que $0 \leq i \leq k - 1$: $H[i] < H[i + 1]$ e
 - para cada i tal que $k \leq i \leq N - 2$: $H[i] > H[i + 1]$.
2. (11 pontos) $Q = 1$, $N \leq 2000$
3. (12 pontos) $Q = 1$
4. (14 pontos) $D = 1$
5. (17 pontos) $L = 0$, $R = N - 1$
6. (19 pontos) O valor de D é o mesmo em todas as chamadas a `max_towers`.
7. (23 pontos) Nenhuma restrição adicional.

Corretor Exemplo

O corretor exemplo lê a entrada no seguinte formato:

- linha 1: N Q
- linha 2: $H[0]$ $H[1]$ \dots $H[N - 1]$
- linha $3 + j$ ($0 \leq j \leq Q - 1$): L R D para a pergunta j

O corretor exemplo escreve suas respostas no seguinte formato:

- linha $1 + j$ ($0 \leq j \leq Q - 1$): o valor de retorno de `max_towers` para a pergunta j