



Встречи

N гор расположены в ряд и пронумерованы целыми числами от 0 до $N - 1$ слева направо. Высота горы с номером i равна H_i ($0 \leq i \leq N - 1$). На вершине каждой горы живёт ровно один человек.

Вы собираетесь организовать Q встреч, пронумерованных целыми числами от 0 до $Q - 1$. Во встрече с номером j ($0 \leq j \leq Q - 1$) будут участвовать все люди, живущие на горах с номерами от L_j до R_j включительно ($0 \leq L_j \leq R_j \leq N - 1$). В качестве места встречи вам необходимо выбрать номер x ($L_j \leq x \leq R_j$) горы, на которой произойдёт встреча. Стоимость такой встречи, в зависимости от выбора горы, вычисляется следующим образом:

- Стоимость встречи для участника, который живет на горе с номером y ($L_j \leq y \leq R_j$), равна максимальной высоте среди гор с номерами между x и y включительно. В частности, стоимость встречи для участника с горы x равна H_x — высоте горы x .
- Стоимость встречи равна сумме стоимостей встречи для каждого из её участников.

Для каждой встречи вам необходимо найти минимальную возможную стоимость её проведения.

Обратите внимание, что после встречи все участники возвращаются обратно на вершины гор, где они живут. Таким образом, стоимость встречи не зависит от предыдущих встреч.

Детали реализации

Вам следует реализовать одну функцию :

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

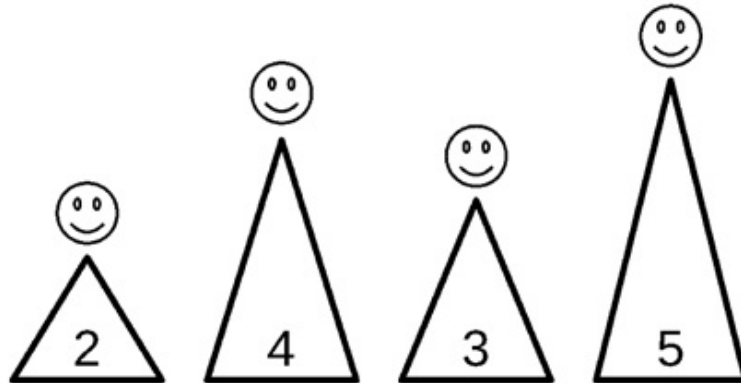
- H : массив длины N , описывающий высоты гор.
- L и R : массивы длины Q , описывающие диапазоны участников встреч.
- Эта функция должна возвращать массив C длины Q . Значение C_j ($0 \leq j \leq Q - 1$) должно равняться минимальной возможной стоимости проведения встречи с номером j .
- Обратите внимание, что значения N и Q являются длинами массивов и могут

быть получены, как указано в памятке о деталях реализации.

Пример

Пусть $N = 4$, $H = [2, 4, 3, 5]$, $Q = 2$, $L = [0, 1]$ и $R = [2, 3]$.

Проверяющий модуль (grader) делает вызов функции `minimum_costs([2, 4, 3, 5], [0, 1], [2, 3])`.



Для встречи с номером $j = 0$ имеем $L_j = 0$ и $R_j = 2$, таким образом, в ней примут участие жители гор с номерами 0, 1 и 2. Если гора с номером 0 выбрана в качестве места встречи, то стоимость встречи с номером 0 вычисляется следующим образом:

- Стоимость для участника, живущего на горе с номером 0, равна $\max\{H_0\} = 2$.
- Стоимость для участника, живущего на горе с номером 1, равна $\max\{H_0, H_1\} = 4$.
- Стоимость для участника, живущего на горе с номером 2, равна $\max\{H_0, H_1, H_2\} = 4$.
- Следовательно, стоимость встречи с номером 0 равна $2 + 4 + 4 = 10$.

Невозможно организовать встречу с номером 0 с меньшей стоимостью, поэтому минимальная стоимость этой встречи равна 10.

Для встречи с номером $j = 1$ имеем $L_j = 1$ и $R_j = 3$, таким образом, в ней примут участие жители гор с номерами 1, 2 и 3. Если гора с номером 2 выбрана в качестве места встречи, то стоимость встречи с номером 1 вычисляется следующим образом:

- Стоимость для участника, живущего на горе с номером 1, равна $\max\{H_1, H_2\} = 4$.
- Стоимость для участника, живущего на горе с номером 2, равна $\max\{H_2\} = 3$.
- Стоимость для участника, живущего на горе с номером 3, равна $\max\{H_2, H_3\} = 5$.
- Следовательно, стоимость встречи с номером 1 равна $4 + 3 + 5 = 12$.

Невозможно организовать встречу с номером 1 с меньшей стоимостью, поэтому минимальная стоимость этой встречи равна 12.

Файлы `sample-01-in.txt` и `sample-01-out.txt` в приложенном zip-архиве соответствуют этому примеру. В архиве есть также другие примеры ввода и вывода.

Ограничения

- $1 \leq N \leq 750\,000$
- $1 \leq Q \leq 750\,000$
- $1 \leq H_i \leq 1\,000\,000\,000$ ($0 \leq i \leq N - 1$)
- $0 \leq L_j \leq R_j \leq N - 1$ ($0 \leq j \leq Q - 1$)
- $(L_j, R_j) \neq (L_k, R_k)$ ($0 \leq j < k \leq Q - 1$)

Подзадачи

1. (4 балла) $N \leq 3\,000$, $Q \leq 10$
2. (15 баллов) $N \leq 5\,000$, $Q \leq 5\,000$
3. (17 баллов) $N \leq 100\,000$, $Q \leq 100\,000$, $H_i \leq 2$ ($0 \leq i \leq N - 1$)
4. (24 балла) $N \leq 100\,000$, $Q \leq 100\,000$, $H_i \leq 20$ ($0 \leq i \leq N - 1$)
5. (40 баллов) Нет дополнительных ограничений

Пример проверяющего модуля

Пример проверяющего модуля считывает входные данные в следующем формате:

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

Пример проверяющего модуля выводит ваши ответы в следующем формате:

- строка $1 + j$ ($0 \leq j \leq Q - 1$): возвращаемое значение `minimum_costs` для запроса j