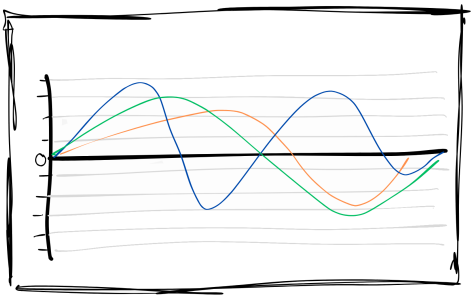
Problem H
History in Numbers



Algorithms and data structures find their place in various arts, crafts, and sciences. In this problem, we are discussing one potential application in history.

Historians are studying the urban development of a city over a period of n years. The urban development is quantified using an aggregate measure they call an urban development index (UDI), which could take integer values, including negative. There is an initial estimation e_i of this index for each of the years. However, during the research process, due to the new documents and evidence discovered, these estimates are revised and updated. More formally, throughout the historians' work process, they could want to update the UDI estimates for all years between s_j and f_j (inclusive) by adding d_j to each of them.

The main question historians are interested in is if the UDI has been increasing over a certain period of time. However, due to the noisy nature of estimates, we will not be using the standard definition. Instead, the following procedure is used:

- we collapse all consequent equal numbers into one. For instance, `1 1 2 2 2 3 3 3` becomes `1 2 3`;
- the UDI is considered *increasing* from year i to year j if the sequence of local minima in the UDI sequence after the above transformation is strictly increasing. An element p_i is considered to be a local minimum if it is less than both of its neighbors (one neighbor for the first or last element).

You are to implement the system able to store the UDI estimates and process the requests to update them and to check if the UDI has been increasing over a certain period.

Input

- One line containining an integer n ($1 \le n \le 3 \cdot 10^5$) denoting the length of the time period being considered.
- One line containing n space separated integers e_i ($-10^8 \le e_i \le 10^8$).
- One line containing an integer number m ($1 \le m \le 3 \cdot 10^5$).
- m further lines each describing one request in one of the following two formats:
  - `update` followed by three integers s_j, f_j, and d_j ($1 \le s_j \le f_j \le n$, $|d_j| \le 10^8$).
  - `check` followed by two integers s and f ($1 \le s \le f \le n$).

Output

For each `check` request output `YES` if the UDI has been increasing during the corresponding period and `NO` otherwise.

Sample Input 1

```
5
10 4 10 6 10
5
check 1 5
update 2 3 1
check 1 5
update 2 3 1
check 1 5
```

Sample Output 1

```
YES
YES
NO
```

Sample Input 2

```
8
10 -5 -5 -5 11 6 6 12
1
check 1 8
```

Sample Output 2

```
YES
```