

HHPC1 P2 - Uniformity Challenge

Time limit: 1.0s **Memory limit:** 256M

In the corporate realm of Raymond's company, employees occasionally slack during work hours. Raymond's vigilant boss conducts surprise checks to ensure that his employees' tasks are on track. Raymond is tasked with ensuring all numbers in an array a of length N are the same.

Each surprise inspection by his boss focuses on a specific subarray of a . Given the sudden nature of these checks, Raymond must act swiftly to showcase the selected section of his work is uniform and consistent. In the i -th inspection, Raymond's boss focuses on the subarray $a[l_i, \dots, r_i]$.

Raymond has two quick-fix strategies to make the elements in the selected subarray $[l_i, r_i]$ appear the same:

1. Choose an index x ($l_i \leq x \leq r_i$). For every index j ($l_i \leq j \leq r_i$) in the array a , increase a_j by a_x .
2. Choose an index x ($l_i \leq x \leq r_i$). For every index j ($l_i \leq j \leq r_i$) in the array a , multiply a_j by a_x .

Help Raymond determine, for each of these sudden inspections, whether it is possible to make all numbers in the subarray $a[l_i, \dots, r_i]$ the same using the strategies above **at most twice**.

Any changes Raymond makes will revert at the end of each inspection, so the array a will return to its original state before any subsequent inspection.

Constraints

For all subtasks:

$$1 \leq N \leq 5 \times 10^5$$

$$1 \leq Q \leq 5 \times 10^5$$

$$-10^6 \leq a_i \leq 10^6$$

$$1 \leq l_i \leq r_i \leq N$$

Subtask 1 [30%]

$$1 \leq N \leq 5 \times 10^3$$

$$1 \leq Q \leq 5 \times 10^3$$

Subtask 2 [70%]

No additional constraints.

Input Specification

The first line contains two integers N and Q , representing the size of the array a and the number of sudden inspections.

The second line contains N space-separated integers, a_1, a_2, \dots, a_N .

The next Q lines each contain two integers, l_i and r_i , describing the i -th sudden inspection.

Output Specification

For each sudden inspection, print YES if it is possible for all elements in the subarray to become the same after performing the strategies at most twice, and NO otherwise.

Sample Input

```
6 5
-2 3 3 3 2 0
1 6
2 3
3 5
1 5
1 4
```

Sample Output

```
YES
YES
NO
YES
NO
```