

# Another Palindromes Problem

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            **3 seconds**  
Memory limit:         **1024 megabytes**

Given a sequence  $A$  of length  $n$ , denote the  $i$  th element in the sequence  $A$  as  $a_i$ . Next there are  $q$  operations, which are categorized as modification and query operations. Each modification operation is given an interval  $[l, r]$  and a positive integer  $v$ , which is denoted as add  $v$  to  $a_l, a_{l+1}, \dots, a_r$ . Each query operation, given an interval  $[l, r]$ , asks whether the sequence  $a_l, a_{l+1}, \dots, a_r$  can be formed into a sequence of palindromes by a number of “swaps”, **which are guaranteed to be of even length**. For each “swap”, a subscript  $i$  can be chosen that satisfies  $l \leq i \leq r - 2$ , and  $a_i$  and  $a_{i+2}$  are swapped. Note that for the query operation, we are only determining whether the sequences can be “swapped” to form a palindrome sequence, and not changing the  $A$  sequence.

## Input

The first line contains two integers  $n, q$  ( $1 \leq n, q \leq 2 \times 10^5$ ) which represent the length of the sequence  $A$  and the number of operations.

The second line contains  $n$  integers, the  $i$  th number representing the  $i$ th element  $a_i$  ( $1 \leq a_i \leq 10^9$ ) in the sequence  $A$ .

The following  $q$  lines, each represents an operation, if it is a modification operation, it contains 4 integers  $op, l, r, v$ , where  $op = 0$  and the meaning of  $l, r, v$  is the same as the description of the statement; otherwise, it contains 3 integers  $op, l, r$ , where  $op = 1$ , which represents the query operation, and the meaning of  $l, r$  is the same as that of the description of the statement ( $op \in \{0, 1\}$ ,  $1 \leq l \leq r \leq n$ ,  $1 \leq v \leq 10^9$ , and  $r - l + 1$  is guaranteed to be even when  $op = 1$ ).

## Output

For each query, output “YES” if the sequence can be “swapped” to form a palindrome, otherwise output “NO”. You can output the answer in any form (upper or lower case), e.g. “yEs”, “yes”, “Yes”, and “YES” are all considered positive answers.

## Examples

standard input	standard output
5 5	YES
2 1 2 1 2	NO
0 2 4 1	NO
1 1 2	YES
1 2 3	
1 3 4	
1 4 5	
9 9	YES
1 2 3 3 2 1 1 2 3	YES
1 1 6	YES
1 4 9	NO
1 2 5	NO
0 1 5 12	YES
1 1 6	YES
1 4 9	
0 8 9 12	
1 4 9	
1 3 8	