

## Problem M. M

**Time limit** 1000 ms

**Mem limit** 262144 kB

Given an array  $a = [a_1, a_2, \dots, a_n]$  of  $n$  positive integers, you can do operations of two types on it:

1. Add 1 to **every** element with an **odd** index. In other words change the array as follows:  
 $a_1 := a_1 + 1, a_3 := a_3 + 1, a_5 := a_5 + 1, \dots$
2. Add 1 to **every** element with an **even** index. In other words change the array as follows:  
 $a_2 := a_2 + 1, a_4 := a_4 + 1, a_6 := a_6 + 1, \dots$

Determine if after any number of operations it is possible to make the final array contain only even numbers or only odd numbers. In other words, determine if you can make all elements of the array have the same parity after any number of operations.

Note that you can do operations of both types any number of times (even none). Operations of different types can be performed a different number of times.

### Input

The first line contains an integer  $t$  ( $1 \leq t \leq 100$ ) — the number of test cases.

The first line of each test case contains an integer  $n$  ( $2 \leq n \leq 50$ ) — the length of the array.

The second line of each test case contains  $n$  integers  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^3$ ) — the elements of the array.

Note that after the performed operations the elements in the array can become greater than  $10^3$ .

### Output

Output  $t$  lines, each of which contains the answer to the corresponding test case. As an answer, output "YES" if after any number of operations it is possible to make the final array contain only even numbers or only odd numbers, and "NO" otherwise.

You can output the answer in any case (for example, the strings "yEs", "yes", "Yes" and "YES" will be recognized as a positive answer).

Examples

Input	Output
4	YES
3	NO
1 2 1	YES
4	YES
2 2 2 3	
4	
2 2 2 2	
5	
1000 1 1000 1 1000	

Note

For the first test case, we can increment the elements with an even index, obtaining the array [1, 3, 1], which contains only odd numbers, so the answer is "YES".

For the second test case, we can show that after performing any number of operations we won't be able to make all elements have the same parity, so the answer is "NO".

For the third test case, all elements already have the same parity so the answer is "YES".

For the fourth test case, we can perform one operation and increase all elements at odd positions by 1, thus obtaining the array [1001, 1, 1001, 1, 1001], and all elements become odd so the answer is "YES".