Problem Statement **Suggest Edit**

You are given an array 'ARR' of 'N' positive integers. You need to find the length of the longest switching contiguous subarray.

An array is called Switching if all the elements at even indices are equal and all the elements at odd indices are also equal.

For Example :

```
If the given 'ARR' is [1, 4, 1, 4, 3, 2, 3, 0]. Then {1, 4, 1, 4}, {3, 2, 3}, {3, 0}, {0} are some of the switching subarrays. But {1, 4, 3}, {1, 4, 1, 4, 3, 2, 3} are not.
```

Input Format :

```
The first line of input contains a single integer T, representing the number of test cases or queries to be run.
Then the T test cases follow:

The first line of each test case contains a positive integer 'N', where 'N' is the size of the given array.

The next line contains 'N' single space-separated positive integers representing the elements of the array.
```

Output Format :

```
For each test case, print an integer denoting the length of the longest switching subarray of the given array in a single line.

Output for each test case will be printed in a separate line.
```

Note:

```
You do not need to print anything. It has already been taken care of. Just implement the given function.
```

Constraint :

```
1 <= T <= 10
1 <= N <= 10^5
1 <= ARR[i] <= 10^8

Time Limit: 1 sec
```

Sample Input 1 :

```
1
6
5 2 3 5 2 5
```

Sample Output 1:

```
3
```

Explanation For Sample Output 1:

```
The longest switching subarray is {5, 2, 5} having length 3.
```

Sample Input 2 :

```
1
8
1 5 6 0 1 0 1 3
```

Sample Output 2 :

```
4
```