

Problem D. Stalin Sort

Time limit 1000 ms

Mem limit 262144 kB

Stalin Sort is a humorous sorting algorithm designed to eliminate elements which are out of place instead of bothering to sort them properly, lending itself to an $\mathcal{O}(n)$ time complexity.

It goes as follows: starting from the second element in the array, if it is strictly smaller than the previous element (ignoring those which have already been deleted), then delete it.

Continue iterating through the array until it is sorted in non-decreasing order. For example, the array $[1, 4, 2, 3, 6, 5, 5, 7, 7]$ becomes $[1, 4, 6, 7, 7]$ after a Stalin Sort.

We define an array as *vulnerable* if you can sort it in **non-increasing** order by repeatedly applying a Stalin Sort to **any of its subarrays**^{*}, as many times as is needed.

Given an array a of n integers, determine the minimum number of integers which must be removed from the array to make it *vulnerable*.

^{*}An array a is a subarray of an array b if a can be obtained from b by the deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

Each test consists of several test cases. The first line contains a single integer t ($1 \leq t \leq 500$) — the number of test cases. This is followed by descriptions of the test cases.

The first line of each test case contains a single integer n ($1 \leq n \leq 2000$) — the size of the array.

The second line of each test case contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$).

It is guaranteed that the sum of n over all test cases does not exceed 2000.

Output

For each test case, output a single integer — the minimum number of integers which must be removed from the array to make it *vulnerable*.

Examples

Input	Output
6	2
7	0
3 6 4 9 2 5 2	6
5	0
5 4 4 2 2	4
8	2
2 2 4 4 6 6 10 10	
1	
1000	
9	
6 8 9 10 12 9 7 5 4	
7	
3000000000 6000000000 4000000000 9000000000	
2000000000 4000000000 2000000000	

Note

In the first test case, the optimal answer is to remove the numbers 3 and 9. Then we are left with $a = [6, 4, 2, 5, 2]$. To show this array is vulnerable, we can first apply a Stalin Sort on the subarray $[4, 2, 5]$ to get $a = [6, 4, 5, 2]$ and then apply a Stalin Sort on the subarray $[6, 4, 5]$ to get $a = [6, 2]$, which is non-increasing.

In the second test case, the array is already non-increasing, so we don't have to remove any integers.