

Problem 2832: Maximal Range That Each Element Is Maximum in It

Problem Information

Difficulty: Medium

Acceptance Rate: 75.36%

Paid Only: Yes

Tags: Array, Stack, Monotonic Stack

Problem Description

You are given a **0-indexed** array `nums` of **distinct** integers.

Let us define a **0-indexed** array `ans` of the same length as `nums` in the following way:

* `ans[i]` is the **maximum** length of a subarray `nums[l..r]`, such that the maximum element in that subarray is equal to `nums[i]`.

Return _the array_ `ans`.

Note that a **subarray** is a contiguous part of the array.

Example 1:

Input: nums = [1,5,4,3,6] **Output:** [1,4,2,1,5] **Explanation:** For nums[0] the longest subarray in which 1 is the maximum is nums[0..0] so ans[0] = 1. For nums[1] the longest subarray in which 5 is the maximum is nums[0..3] so ans[1] = 4. For nums[2] the longest subarray in which 4 is the maximum is nums[2..3] so ans[2] = 2. For nums[3] the longest subarray in which 3 is the maximum is nums[3..3] so ans[3] = 1. For nums[4] the longest subarray in which 6 is the maximum is nums[0..4] so ans[4] = 5.

Example 2:

Input: nums = [1,2,3,4,5] **Output:** [1,2,3,4,5] **Explanation:** For nums[i] the longest subarray in which it's the maximum is nums[0..i] so ans[i] = i + 1.

****Constraints:****

* `1 <= nums.length <= 105` * `1 <= nums[i] <= 105` * All elements in `nums` are distinct.

Code Snippets

C++:

```
class Solution {  
public:  
    vector<int> maximumLengthOfRanges(vector<int>& nums) {  
  
    }  
};
```

Java:

```
class Solution {  
public int[] maximumLengthOfRanges(int[] nums) {  
  
}  
}
```

Python3:

```
class Solution:  
    def maximumLengthOfRanges(self, nums: List[int]) -> List[int]:
```