Problem 795: Number of Subarrays with Bounded Maximum

Problem Information

Difficulty: Medium
Acceptance Rate: 54.45%
Paid Only: No
Tags: Array, Two Pointers

Problem Description

Given an integer array `nums` and two integers `left` and `right`, return _the number of contiguous non-empty**subarrays** such that the value of the maximum array element in that subarray is in the range _`[left, right]`.

The test cases are generated so that the answer will fit in a **32-bit** integer.

Example 1:

Input: nums = [2,1,4,3], left = 2, right = 3 **Output:** 3 **Explanation:** There are three subarrays that meet the requirements: [2], [2, 1], [3].

Example 2:

Input: nums = [2,9,2,5,6], left = 2, right = 8 **Output:** 7

Constraints:

* `1 <= nums.length <= 105` * `0 <= nums[i] <= 109` * `0 <= left <= right <= 109`

Code Snippets

C++:

```
class Solution {
public:
int numSubarrayBoundedMax(vector<int>& nums, int left, int right) {

}
};
```

Java:

```
class Solution {
public int numSubarrayBoundedMax(int[] nums, int left, int right) {

}
}
```

Python3:

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