Problem 915: Partition Array into Disjoint Intervals

Problem Information

Difficulty: Medium
Acceptance Rate: 49.27%
Paid Only: No
Tags: Array

Problem Description

Given an integer array `nums`, partition it into two (contiguous) subarrays `left` and `right` so that:

* Every element in `left` is less than or equal to every element in `right`. * `left` and `right` are non-empty. * `left` has the smallest possible size.

Return _the length of_`left` _after such a partitioning_.

Test cases are generated such that partitioning exists.

Example 1:

Input: nums = [5,0,3,8,6] **Output:** 3 **Explanation:** left = [5,0,3], right = [8,6]

Example 2:

Input: nums = [1,1,1,0,6,12] **Output:** 4 **Explanation:** left = [1,1,1,0], right = [6,12]

Constraints:

* `2 <= nums.length <= 105` * `0 <= nums[i] <= 106` * There is at least one valid answer for the given input.

Code Snippets

C++:

```
class Solution {
public:
int partitionDisjoint(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public int partitionDisjoint(int[] nums) {

}
}
```

Python3:

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