Problem 3638: Maximum Balanced Shipments

Problem Information

Difficulty: Medium
Acceptance Rate: 59.56%
Paid Only: No
Tags: Array, Dynamic Programming, Stack, Greedy, Monotonic Stack

Problem Description

You are given an integer array `weight` of length `n`, representing the weights of `n` parcels arranged in a straight line. A **shipment** is defined as a contiguous subarray of parcels. A shipment is considered **balanced** if the weight of the **last parcel** is **strictly less** than the **maximum weight** among all parcels in that shipment.

Select a set of **non-overlapping** , contiguous, balanced shipments such that **each parcel appears in at most one shipment** (parcels may remain unshipped).

Return the **maximum possible number** of balanced shipments that can be formed.

Example 1:

Input: weight = [2,5,1,4,3]

Output: 2

Explanation:

We can form the maximum of two balanced shipments as follows:

* Shipment 1: `[2, 5, 1]` * Maximum parcel weight = 5 * Last parcel weight = 1, which is strictly less than 5. Thus, it's balanced. * Shipment 2: `[4, 3]` * Maximum parcel weight = 4 * Last parcel weight = 3, which is strictly less than 4. Thus, it's balanced.

It is impossible to partition the parcels to achieve more than two balanced shipments, so the answer is 2.

Example 2:

Input: weight = [4,4]

Output: 0

Explanation:

No balanced shipment can be formed in this case:

* A shipment `[4, 4]` has maximum weight 4 and the last parcel's weight is also 4, which is not strictly less. Thus, it's not balanced. * Single-parcel shipments `[4]` have the last parcel weight equal to the maximum parcel weight, thus not balanced.

As there is no way to form even one balanced shipment, the answer is 0.

Constraints:

* `2 <= n <= 105` * `1 <= weight[i] <= 109`

Code Snippets

C++:

```
class Solution {
public:
int maxBalancedShipments(vector<int>& weight) {

}
};
```

Java:

```
class Solution {
public int maxBalancedShipments(int[] weight) {

}
}
```

Python3:

```
class Solution:
def maxBalancedShipments(self, weight: List[int]) -> int:
```