Problem 1546: Maximum Number of Non-Overlapping Subarrays With Sum Equals Target

Problem Information

Difficulty: Medium
Acceptance Rate: 48.55%
Paid Only: No
Tags: Array, Hash Table, Greedy, Prefix Sum

Problem Description

Given an array `nums` and an integer `target`, return _the maximum number of**non-empty** **non-overlapping** subarrays such that the sum of values in each subarray is equal to_ `target`.

Example 1:

Input: nums = [1,1,1,1,1], target = 2 **Output:** 2 **Explanation:** There are 2 non-overlapping subarrays [**1,1** ,1,**1,1**] with sum equals to target(2).

Example 2:

Input: nums = [-1,3,5,1,4,2,-9], target = 6 **Output:** 2 **Explanation:** There are 3 subarrays with sum equal to 6. ([5,1], [4,2], [3,5,1,4,2,-9]) but only the first 2 are non-overlapping.

Constraints:

* `1 <= nums.length <= 105` * `-104 <= nums[i] <= 104` * `0 <= target <= 106`

Code Snippets

C++:

```
class Solution {
public:
int maxNonOverlapping(vector<int>& nums, int target) {

}
};
```

Java:

```
class Solution {
public int maxNonOverlapping(int[] nums, int target) {

}
}
```

Python3:

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