Problem 548: Split Array with Equal Sum

Problem Information

Difficulty: Hard
Acceptance Rate: 50.09%
Paid Only: Yes
Tags: Array, Hash Table, Prefix Sum

Problem Description

Given an integer array `nums` of length `n`, return `true` if there is a triplet `(i, j, k)` which satisfies the following conditions:

* `0 < i, i + 1 < j, j + 1 < k < n - 1` * The sum of subarrays `(0, i - 1)`, `(i + 1, j - 1)`, `(j + 1, k - 1)` and `(k + 1, n - 1)` is equal.

A subarray `(l, r)` represents a slice of the original array starting from the element indexed `l` to the element indexed `r`.

Example 1:

Input: nums = [1,2,1,2,1,2,1] **Output:** true **Explanation:** i = 1, j = 3, k = 5. sum(0, i - 1) = sum(0, 0) = 1 sum(i + 1, j - 1) = sum(2, 2) = 1 sum(j + 1, k - 1) = sum(4, 4) = 1 sum(k + 1, n - 1) = sum(6, 6) = 1

Example 2:

Input: nums = [1,2,1,2,1,2,1,2] **Output:** false

Constraints:

* `n == nums.length` * `1 <= n <= 2000` * `-106 <= nums[i] <= 106`

Code Snippets

C++:

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

}
};
```

Java:

```
class Solution {
public boolean splitArray(int[] nums) {

}
}
```

Python3:

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