Problem Link:

<https://leetcode.com/problems/number-of-ways-to-split-array/submissions/1496384084/?envType=daily-question&envId=2025-01-03>

Solution:

class Solution {

public:

int waysToSplitArray(vector<int>& nums) {

int n = nums.size();

vector<long long> psum(n, 0), ssum(n, 0);

psum[0] = nums[0];

for(int i = 1; i < n; ++i)

{

psum[i] = psum[i - 1] + nums[i];

}

ssum[n - 1] = nums[n - 1];

for(int i = n - 2; i >= 0; --i)

{

ssum[i] = ssum[i + 1] + nums[i];

}

int count = 0;

for(int i = 0; i < n - 1; ++i)

{

if (psum[i] >= ssum[i + 1])

{

count++;

}

}

return count;

}

};