Problem 643: Maximum Average Subarray I

Problem Information

Difficulty: Easy
Acceptance Rate: 46.36%
Paid Only: No
Tags: Array, Sliding Window

Problem Description

You are given an integer array `nums` consisting of `n` elements, and an integer `k`.

Find a contiguous subarray whose **length is equal to** `k` that has the maximum average value and return _this value_. Any answer with a calculation error less than `10-5` will be accepted.

Example 1:

Input: nums = [1,12,-5,-6,50,3], k = 4 **Output:** 12.75000 **Explanation:** Maximum average is (12 - 5 - 6 + 50) / 4 = 51 / 4 = 12.75

Example 2:

Input: nums = [5], k = 1 **Output:** 5.00000

Constraints:

* `n == nums.length` * `1 <= k <= n <= 105` * `-104 <= nums[i] <= 104`

Code Snippets

C++:

```
class Solution {
public:
```

```
double findMaxAverage(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public double findMaxAverage(int[] nums, int k) {

}
}
```

Python3:

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