Problem 2420: Find All Good Indices

Problem Information

Difficulty: Medium
Acceptance Rate: 40.42%
Paid Only: No
Tags: Array, Dynamic Programming, Prefix Sum

Problem Description

You are given a **0-indexed** integer array `nums` of size `n` and a positive integer `k`.

We call an index `i` in the range `k <= i < n - k` **good** if the following conditions are satisfied:

* The `k` elements that are just **before** the index `i` are in **non-increasing** order. * The `k` elements that are just **after** the index `i` are in **non-decreasing** order.

Return _an array of all good indices sorted in**increasing** order_.

Example 1:

Input: nums = [2,1,1,1,3,4,1], k = 2 **Output:** [2,3] **Explanation:** There are two good indices in the array: - Index 2. The subarray [2,1] is in non-increasing order, and the subarray [1,3] is in non-decreasing order. - Index 3. The subarray [1,1] is in non-increasing order, and the subarray [3,4] is in non-decreasing order. Note that the index 4 is not good because [4,1] is not non-decreasing.

Example 2:

Input: nums = [2,1,1,2], k = 2 **Output:** [] **Explanation:** There are no good indices in this array.

Constraints:

* `n == nums.length` * `3 <= n <= 105` * `1 <= nums[i] <= 106` * `1 <= k <= n / 2`

Code Snippets

C++:

```
class Solution {
public:
vector<int> goodIndices(vector<int>& nums, int k) {

}
};
```

Java:

```
class Solution {
public List<Integer> goodIndices(int[] nums, int k) {

}
}
```

Python3:

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