Problem 1852: Distinct Numbers in Each Subarray

Problem Information

Difficulty: Medium
Acceptance Rate: 77.16%
Paid Only: Yes
Tags: Array, Hash Table, Sliding Window

Problem Description

You are given an integer array `nums` of length `n` and an integer `k`. Your task is to find the number of **distinct** elements in **every** subarray of size `k` within `nums`.

Return an array `ans` such that `ans[i]` is the count of distinct elements in `nums[i..(i + k - 1)]` for each index `0 <= i < n - k`.

Example 1:

Input: nums = [1,2,3,2,2,1,3], k = 3 **Output:** [3,2,2,2,3] **Explanation:** The number of distinct elements in each subarray goes as follows: - nums[0..2] = [1,2,3] so ans[0] = 3 - nums[1..3] = [2,3,2] so ans[1] = 2 - nums[2..4] = [3,2,2] so ans[2] = 2 - nums[3..5] = [2,2,1] so ans[3] = 2 - nums[4..6] = [2,1,3] so ans[4] = 3

Example 2:

Input: nums = [1,1,1,1,2,3,4], k = 4 **Output:** [1,2,3,4] **Explanation:** The number of distinct elements in each subarray goes as follows: - nums[0..3] = [1,1,1,1] so ans[0] = 1 - nums[1..4] = [1,1,1,2] so ans[1] = 2 - nums[2..5] = [1,1,2,3] so ans[2] = 3 - nums[3..6] = [1,2,3,4] so ans[3] = 4

Constraints:

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

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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