Problem 1708: Largest Subarray Length K

Problem Information

Difficulty: Easy
Acceptance Rate: 65.49%
Paid Only: Yes
Tags: Array, Greedy

Problem Description

An array `A` is larger than some array `B` if for the first index `i` where `A[i] != B[i]`, `A[i] > B[i]`.

For example, consider `0`-indexing:

* `[1,3,2,4] > [1,2,2,4]`, since at index `1`, `3 > 2`. * `[1,4,4,4] < [2,1,1,1]`, since at index `0`, `1 < 2`.

A subarray is a contiguous subsequence of the array.

Given an integer array `nums` of **distinct** integers, return the **largest** subarray of `nums` of length `k`.

Example 1:

Input: nums = [1,4,5,2,3], k = 3 **Output:** [5,2,3] **Explanation:** The subarrays of size 3 are: [1,4,5], [4,5,2], and [5,2,3]. Of these, [5,2,3] is the largest.

Example 2:

Input: nums = [1,4,5,2,3], k = 4 **Output:** [4,5,2,3] **Explanation:** The subarrays of size 4 are: [1,4,5,2], and [4,5,2,3]. Of these, [4,5,2,3] is the largest.

Example 3:

Input: nums = [1,4,5,2,3], k = 1 **Output:** [5]

Constraints:

* `1 <= k <= nums.length <= 105` * `1 <= nums[i] <= 109` * All the integers of `nums` are **unique**.

Follow up: What if the integers in `nums` are not distinct?

Code Snippets

C++:

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


}
};
```

Java:

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


}
}
```

Python3:

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