Problem 1567: Maximum Length of Subarray With Positive Product

Problem Information

Difficulty: Medium
Acceptance Rate: 44.57%
Paid Only: No
Tags: Array, Dynamic Programming, Greedy

Problem Description

Given an array of integers `nums`, find the maximum length of a subarray where the product of all its elements is positive.

A subarray of an array is a consecutive sequence of zero or more values taken out of that array.

Return _the maximum length of a subarray with positive product_.

Example 1:

Input: nums = [1,-2,-3,4] **Output:** 4 **Explanation:** The array nums already has a positive product of 24.

Example 2:

Input: nums = [0,1,-2,-3,-4] **Output:** 3 **Explanation:** The longest subarray with positive product is [1,-2,-3] which has a product of 6. Notice that we cannot include 0 in the subarray since that'll make the product 0 which is not positive.

Example 3:

Input: nums = [-1,-2,-3,0,1] **Output:** 2 **Explanation:** The longest subarray with positive product is [-1,-2] or [-2,-3].

Constraints:

* `1 <= nums.length <= 105` * `-109 <= nums[i] <= 109`

Code Snippets

C++:

```
class Solution {
public:
int getMaxLen(vector<int>& nums) {

}
};
```

Java:

```
class Solution {
public int getMaxLen(int[] nums) {

}
}
```

Python3:

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