Problem 1438: Longest Continuous Subarray With Absolute Diff Less Than or Equal to Limit

Problem Information

Difficulty: Medium
Acceptance Rate: 57.07%
Paid Only: No
Tags: Array, Queue, Sliding Window, Heap (Priority Queue), Ordered Set, Monotonic Queue

Problem Description

Given an array of integers `nums` and an integer `limit`, return the size of the longest **non-empty** subarray such that the absolute difference between any two elements of this subarray is less than or equal to `limit` _._

Example 1:

Input: nums = [8,2,4,7], limit = 4 **Output:** 2 **Explanation:** All subarrays are: [8] with maximum absolute diff |8-8| = 0 <= 4. [8,2] with maximum absolute diff |8-2| = 6 > 4. [8,2,4] with maximum absolute diff |8-2| = 6 > 4. [8,2,4,7] with maximum absolute diff |8-2| = 6 > 4. [2] with maximum absolute diff |2-2| = 0 <= 4. [2,4] with maximum absolute diff |2-4| = 2 <= 4. [2,4,7] with maximum absolute diff |2-7| = 5 > 4. [4] with maximum absolute diff |4-4| = 0 <= 4. [4,7] with maximum absolute diff |4-7| = 3 <= 4. [7] with maximum absolute diff |7-7| = 0 <= 4. Therefore, the size of the longest subarray is 2.

Example 2:

Input: nums = [10,1,2,4,7,2], limit = 5 **Output:** 4 **Explanation:** The subarray [2,4,7,2] is the longest since the maximum absolute diff is |2-7| = 5 <= 5.

Example 3:

Input: nums = [4,2,2,2,4,4,2,2], limit = 0 **Output:** 3

Constraints:

* `1 <= nums.length <= 105` * `1 <= nums[i] <= 109` * `0 <= limit <= 109`

Code Snippets

C++:

```
class Solution {
public:
int longestSubarray(vector<int>& nums, int limit) {

}
};
```

Java:

```
class Solution {
public int longestSubarray(int[] nums, int limit) {

}
}
```

Python3:

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