56. Longest Continuous Subarray With Absolute Diff Less Than or Equal to Limit

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

AIM: To find the Longest Continuous Subarray With Absolute Diff Less Than or Equal to Limit

PROGRAM:

```
from collections import deque


def longest_subarray(nums, limit):
    min_queue = deque()
    max_queue = deque()
    left = 0
    max_length = 0
    for right, num in enumerate(nums):
        while min_queue and num < nums[min_queue[-1]]:
            min_queue.pop()
        while max_queue and num > nums[max_queue[-1]]:
            max_queue.pop()
        min_queue.append(right)
        max_queue.append(right)
        while nums[max_queue[0]] - nums[min_queue[0]] > limit:
            if max_queue[0] == left:
                max_queue.popleft()
            if min_queue[0] == left:
                min_queue.popleft()
            left += 1
        max_length = max(max_length, right - left + 1)
    return max_length
```

```
print(longest_subarray([8,2,4,7], 4))
print(longest_subarray([10,1,2,4,7,2], 5))
```

OUTPUT:

```
2
4
```

TIME COMPLEXITY: O(n)