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.

Program:

```
def longest_subarray(nums, limit):
    from collections import deque

    max_queue = deque()
    min_queue = deque()
    left = 0
    result = 0

    for right, num in enumerate(nums):
        while max_queue and num > max_queue[-1]:
            max_queue.pop()
        while min_queue and num < min_queue[-1]:
            min_queue.pop()

        max_queue.append(num)
        min_queue.append(num)

        if max_queue[0] - min_queue[0] > limit:
            if max_queue[0] == nums[left]:
                max_queue.popleft()
            if min_queue[0] == nums[left]:
                min_queue.popleft()
            left += 1

        result = max(result, right - left + 1)

    return result
```

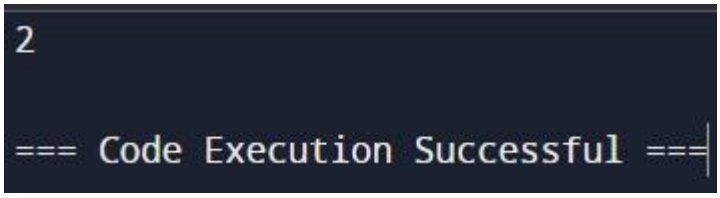
Example usage:

nums = [8, 2, 4, 7]

limit = 4

print(longest_subarray(nums, limit)) # Output: 2

output:



Time complexity:O(nlogn)