Program 56.Check If All 1's Are at Least Length K Places Away
Given an binary array nums and an integer k, return true if all 1's are at least k places away from each other, otherwise return false.

Example 1:

Input: nums = [1,0,0,0,1,0,0,1], k = 2
Output: true
Explanation: Each of the 1s are at least 2 places away from each other.

Example 2:

Input: nums = [1,0,0,1,0,1], k = 2
Output: false
Explanation: The second 1 and third 1 are only one apart from each other.

Constraints:

- **1 <= nums.length <= 105**
- **0 <= k <= nums.length**
- **nums[i] is 0 or 1**

Program:

```
def kLengthApart(nums, k):
    prev_index = -1

    for i, num in enumerate(nums):
        if num == 1:
            if prev_index != -1 and i - prev_index - 1 < k:
                return False
            prev_index = i

    return True

# Example usage
nums1 = [1, 0, 0, 0, 1, 0, 0, 1]
k1 = 2
print(kLengthApart(nums1, k1))  # Output: True

nums2 = [1, 0, 0, 1, 0, 1]
k2 = 2
print(kLengthApart(nums2, k2))  # Output: False
```

Output:

```
"C:\Program Files\Python312\python.exe" "C:\Work Space\DAA COADS.PYTHON\program 57.py"
True
False

Process finished with exit code 0
```

Time complexity:
O(n)