Two pointers pattern-based problem that are mostly asked 10 interview question
give input and output of problem.

3/3

The two-pointer technique is a popular pattern used in algorithmic problems. It involves using two pointers to traverse data structures, often arrays or linked lists, to solve problems more efficiently. Here are 10 common two-pointer pattern-based problems along with their inputs and outputs:

1. **Two Sum II - Input Array Is Sorted**
   - **Problem:** Given a sorted array and a target sum, find the indices of the two numbers that add up to the target.
   - **Input:** `numbers = [2, 7, 11, 15]`, `target = 9`
   - **Output:** `[1, 2]` (1-based index of the numbers `2` and `7`)
2. **Container With Most Water**
   - **Problem:** Given an array representing the height of bars, find the maximum area of water that can be trapped between any two bars.
   - **Input:** `height = [1,8,6,2,5,4,8,3,7]`
   - **Output:** `49`
3. **3Sum**
   - **Problem:** Find all unique triplets in the array that add up to zero.
   - **Input:** `nums = [-1, 0, 1, 2, -1, -4]`
   - **Output:** `[[-1, -1, 2], [-1, 0, 1]]`
4. **Remove Duplicates from Sorted Array**
   - **Problem:** Remove duplicates from a sorted array and return the new length.
   - **Input:** `nums = [1, 1, 2]`
   - **Output:** `2` (The array after removing duplicates is `[1, 2]`)
5. **Valid Palindrome**
   - **Problem:** Check if a given string is a palindrome considering only alphanumeric characters and ignoring case.
   - **Input:** `s = "A man, a plan, a canal: Panama"`
   - **Output:** `true`
6. **Longest Substring Without Repeating Characters**
   - **Problem:** Given a string, find the length of the longest substring without repeating characters.
   - **Input:** `s = "abcabcbb"`
   - **Output:** `3` (The substring is `"abc"`)
7. **Find the Duplicate Number**
   - **Problem:** Given an array of `n+1` integers where each integer is between `1` and `n`, find the duplicate number.
   - **Input:** `nums = [1, 3, 4, 2, 2]`
   - **Output:** `2`
8. **Pair with Target Sum**
   - **Problem:** Given a sorted array and a target sum, find the indices of the two numbers that add up to the target.

Get citation

- **Input:** `arr = [1, 2, 3, 4, 6]`, `target = 6`
- **Output:** `[2, 4]` (The numbers are `2` and `4`)

9. **Squares of a Sorted Array**

   - **Problem:** Given a sorted array, return an array of the squares of each number sorted in non-decreasing order.
   - **Input:** `nums = [-4, -1, 0, 3, 10]`
   - **Output:** `[0, 1, 9, 16, 100]`

10. **Trapping Rain Water**

    - **Problem:** Given an array representing the height of bars in a histogram, calculate the total amount of water that can be trapped.
    - **Input:** `height = [0,1,0,2,1,0,1,3,2,1,2,1]`
    - **Output:** `6`

These problems can be solved efficiently using the two-pointer technique, which often reduces the time complexity compared to naive approaches.