Container With Most Water

Solving the classic "Container With Most Water" problem – best approached with the two-pointer technique.

Key Idea: To maximise the area between two vertical lines:

- Area = min (height [i], height[j]) * (j - i)
- Start with two pointers:
  - left = 0, right = n-1
- Move the pointer pointing to the shorter height inward;
  - Moving the taller one can't give a better result (width shrinks but height doesn't improve)

Algorithm Steps:

1. Initialize: left = 0, right = n-1
   maxArea = 0
2. While left < right:
   - Calculate area: area = min (height [left], height[right]) * (right - left)
   - Update maxArea
   - Move the shorter line inward:
     - If height [left] < height [right]:
       left++
     - Else: right--
3. Return maxArea

Time Complexity:

- O(n) - single pass from both ends.
- O(1) space

Example: For [1, 8, 6, 2, 5, 4, 8, 3, 7]:

- Max area is between indices 1 and 8:

→ min (8, 7) * (8 - 1) = 7 * 7 = 49