

# Trapping Rain Water

This problem is solvable in  $O(n)$  time and  $O(1)$  space using the two-pointers technique.

## [Key Idea:]

The water trapped at position  $i$  is determined by:

$$\text{water}[i] = \min(\text{max\_left}[i], \text{max\_right}[i]) - \text{height}[i]$$

But instead of computing two arrays, we can do this in one pass using two pointers.

## [Steps:]

1. Initialize two pointers:
  - $\text{left} = 0, \text{right} = n-1$
2. Maintain two variables:
  - $\text{left\_max} = 0, \text{right\_max} = 0$
3. While  $\text{left} < \text{right}$ :
  - If  $\text{height}[\text{left}] < \text{height}[\text{right}]$ :
    - If  $\text{height}[\text{left}] \geq \text{left\_max}$ , update  $\text{left\_max}$
    - Else add  $\text{left\_max} - \text{height}[\text{left}]$  to result
    - Move  $\text{left}++$
  - Else:
    - If  $\text{height}[\text{right}] \geq \text{right\_max}$ , update  $\text{right\_max}$
    - Else add  $\text{right\_max} - \text{height}[\text{right}]$  to result
    - Move  $\text{right}--$

## [Complexity:]

- Time:  $O(n)$  — single pass
- Space:  $O(1)$  — constant extra space