Trapping Rain Water - Code Solutions

```
// Approach 1: Precompute leftMax and rightMax
var trap = function(height) {
    let n = height.length;
    let leftMax = Array(n).fill(0);
    let rightMax = Array(n).fill(0);

    leftMax[0] = height[0];
    for (let i = 1; i < n; i++) {
        leftMax[i] = Math.max(height[i], leftMax[i - 1]);
    }

    rightMax[n - 1] = height[n - 1];
    for (let i = n - 2; i >= 0; i--) {
        rightMax[i] = Math.max(height[i], rightMax[i + 1]);
    }

    let water = 0;
    for (let i = 0; i < n; i++) {
        water += Math.min(leftMax[i], rightMax[i]) - height[i];
    }

    return water;
};

// Approach 2: Two Pointers
var trap = function(height) {
    let left = 0, right = height.length - 1;
    let leftMax = 0, rightMax = 0, water = 0;

    while (left < right) {
        if (height[left] < height[right]) {
            if (height[left] >= leftMax) {
                leftMax = height[left];
            } else {
                water += leftMax - height[left];
            }
            left++;
        } else {
            if (height[right] >= rightMax) {
                rightMax = height[right];
            } else {
                water += rightMax - height[right];
            }
            right--;
        }
    }

    return water;
};
```