Array -6

14 November 2025 17:19

Stock Buy & Sell

{7, 1, 5, 3, 6, 4}, n = 6

Initially:

max-profit = 0
best-buy = 7 (bb)

i = 0

7 ≯ 7

best_buy = min(7, 7); => 7

| 7 | 1 | 5 | 3 | 6 | 4 | (bb ↓ at 7, i ↑ at 7)

i = 1

1 ≯ 7

best buy = min(1, 7) => 1

| 7 | 1 | 5 | 3 | 6 | 4 | (best buy ↓ at 1, i ↑ at 1)

i = 2

5 > 1

max-profit = max(0, 5-1) => 4

best_buy = min(5, 1) => 1

| 7 | 1 | 5 | 3 | 6 | 4 | (bb ↓ at 1, i ↑ at 5)

i = 3

3 > 1

max-profit = max(4, 3-1) => 4

best_buy = min(1, 3) => 1

| 7 | 1 | 5 | 3 | 6 | 4 | (bb ↓ at 1, i ↑ at 6)

i = 4

6 > 4

max-profit = max(4, 6-1) = 5

best_buy = min(1, 6) => 1

. . .

i = 5

4 ≯ 4

best_buy = 1

loop ends => Return max-profit => 5

i = 5

| 7 | 1 | 5 | 3 | 6 | 4 | (bb ↓ at 1, i ↓ at 4)

② Container with most water

{1, 8, 6, 2, 5, 4, 8, 3, 7}

Initially:

left = 0, right = n-1
width = 0, height = 0
curr_area = 0, max_area = 0

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | n = 9 (l ↑ at 1, r ↑ at 7)

while (l < r) → loop starts
width = abs(8 − 0) = 8
height = min(7, 1) = 1
curr_area = 8 × 1 = 8
max_area = max(0, 8) = 8
h[left] < h[right] => l++

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l ↑ at 8, r ↑ at 7)

width = abs(8 − 1) = 7
height = min(8, 7) = 7
curr_area = 7 × 7 = 49
max_area = max(8, 49) = 49
h[l] > h[r] => r--

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l ↑ at 8, r ↑ at 3)

width = abs(7 − 1) = 6
height = min(8, 3) = 3
curr_area = 6 × 3 = 18
max_area = max(49, 18) = 49
h[l] > h[r] => r--

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l ↑ at 8, r ↑ at 8)

width = abs(6 − 1) = 5
height = min(8, 8) = 8
curr_area = 5 × 8 = 40
max_area = max(49, 40) = 49
h[l] ≮ h[r] => r--

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l ↑ at 8, r ↑ at 4)

width = abs(5 − 1) = 4
height = min(8, 4) = 4
curr_area = 4 × 4 = 16
max_area = max(49, 16) = 49
h[l] > h[r] => r--

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l ↓ at 8, r ↓ at 5)

. . .

| 1 | 8 | 6 | 2 | 5 | 4 | 8 | 3 | 7 | (l, r ↓ at 8)

loop end => l = r => Return max area => **49**