Description | Accepted × | Editorial | Solutions | Submissions

← All Submissions

Accepted 65 / 65 testcases passed

Siddhant0705 submitted at Feb 20, 2026 21:20

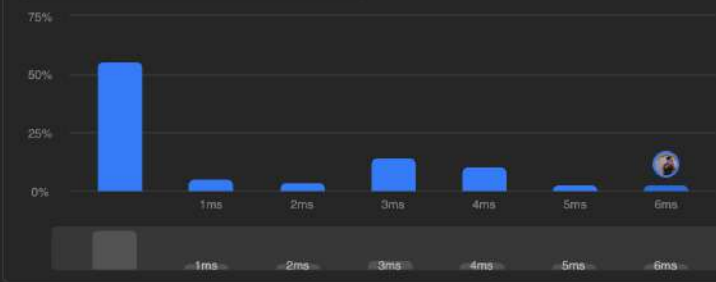Editorial | Solution

Runtime		Memory
6 ms	Beats **8.67%**	62.79 MB	Beats **97.18%**
Analyze Complexity		

75%
50%
25%
0%
1ms   2ms   3ms   4ms   5ms   6ms

Code | C++

```cpp
1  class Solution {
2  public:
3      int maxArea(vector<int>& height) {
4
5          int left = 0;
6          int right = height.size() - 1;
7          int maxWater = 0;
8
```

View more

</> Code   C++ ∨ 🔒 Auto

```cpp
2  public:
3      int maxArea(vector<int>& height) {
4
5          int left = 0;
6          int right = height.size() - 1;
7          int maxWater = 0;
8
9          while(left < right) {
10
11             int h = min(height[left], height[right]);
12             int width = right - left;
13             int area = h * width;
14
15             maxWater = max(maxWater, area);
16
17             if(height[left] < height[right])
18                 left++;
19             else
20                 right--;
21         }
22
23         return maxWater;
24     }
25 };
```

Saved   Ln 25, Col 3

☑ Testcase | >_ **Test Result**

Accepted Runtime: 0 ms

☑ Case 1   ☑ Case 2

Submit

Premium

Description | Accepted × | Editorial | Solutions | Submissions

</> Code

← All Submissions

C++ ⌄ 🔒 Auto

Accepted 150 / 150 testcases passed

Siddhant0705 submitted at Feb 20, 2026 21:21

Editorial   Solution
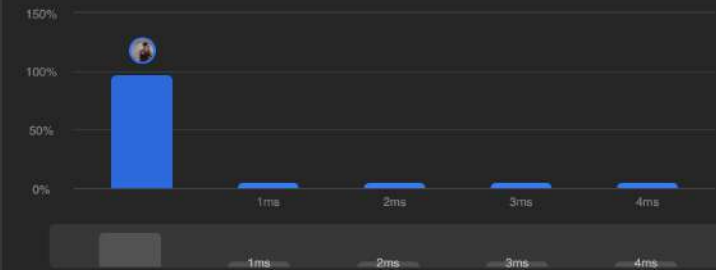
🕐 Runtime   ⓘ   @ Memory

0 ms | Beats **100.00%** 🖐️   **14.22** MB | Beats **10.68%**

✦ Analyze Complexity

150%

100%

50%

0%

1ms   2ms   3ms   4ms

1ms   2ms   3ms   4ms

Code | C++

```cpp
1  class Solution {
2  public:
3      int findMin(vector<int>& nums) {
4
5          int left = 0;
6          int right = nums.size() - 1;
7
8          while(left < right) {
```

≫ View more

```cpp
1  class Solution {
2  public:
3      int findMin(vector<int>& nums) {
4
5          int left = 0;
6          int right = nums.size() - 1;
7
8          while(left < right) {
9
10             int mid = left + (right - left) / 2;
11
12             if(nums[mid] > nums[right]) {
13                 left = mid + 1;
14             }
15             else {
16                 right = mid;
17             }
18         }
19
20         return nums[left];
21     }
22 };
```

Saved   Ln 22, Col 3

☑ Testcase | >_ **Test Result**

Accepted Runtime: 0 ms

☑ Case 1   ☑ Case 2   ☑ Case 3