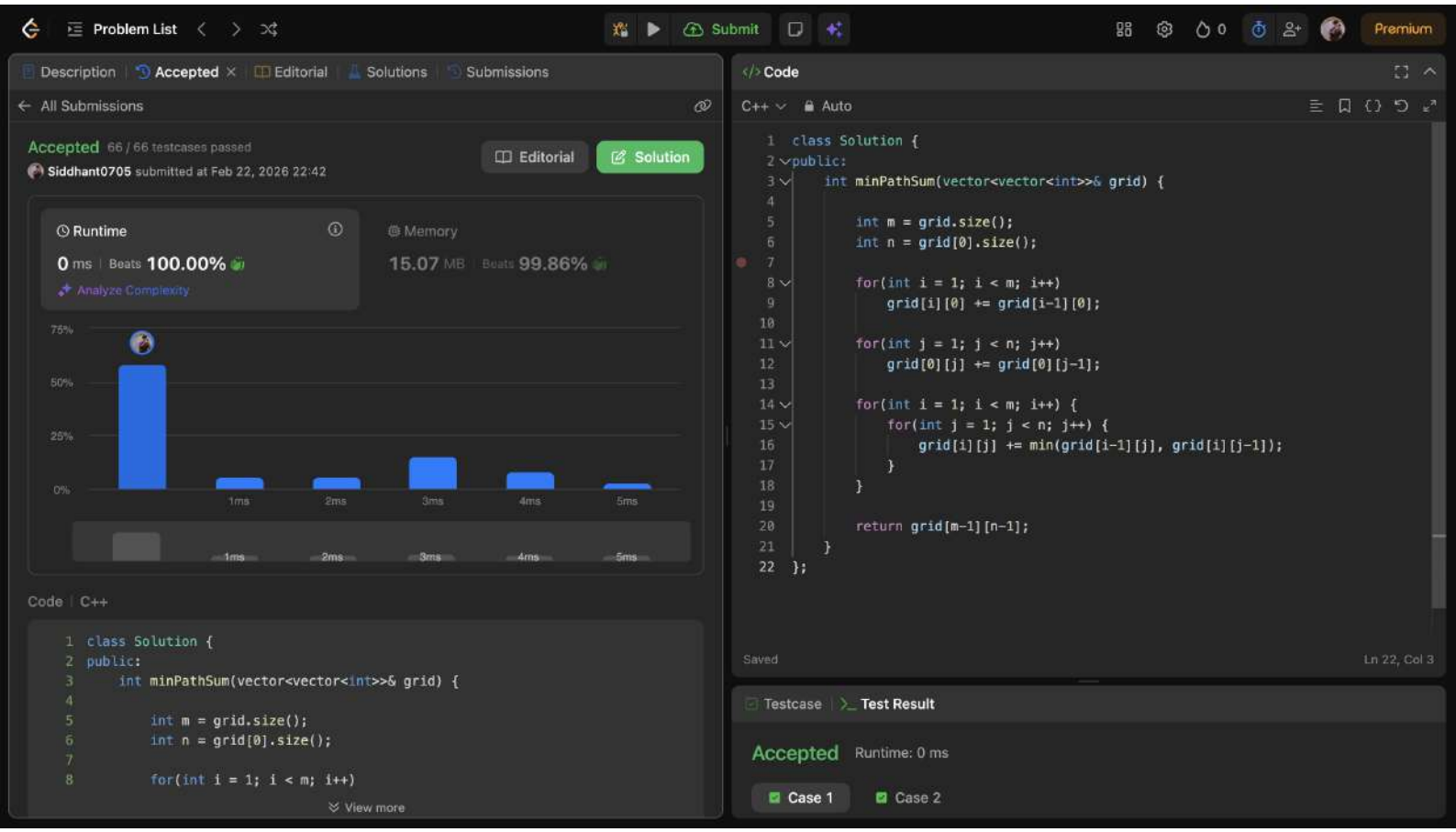
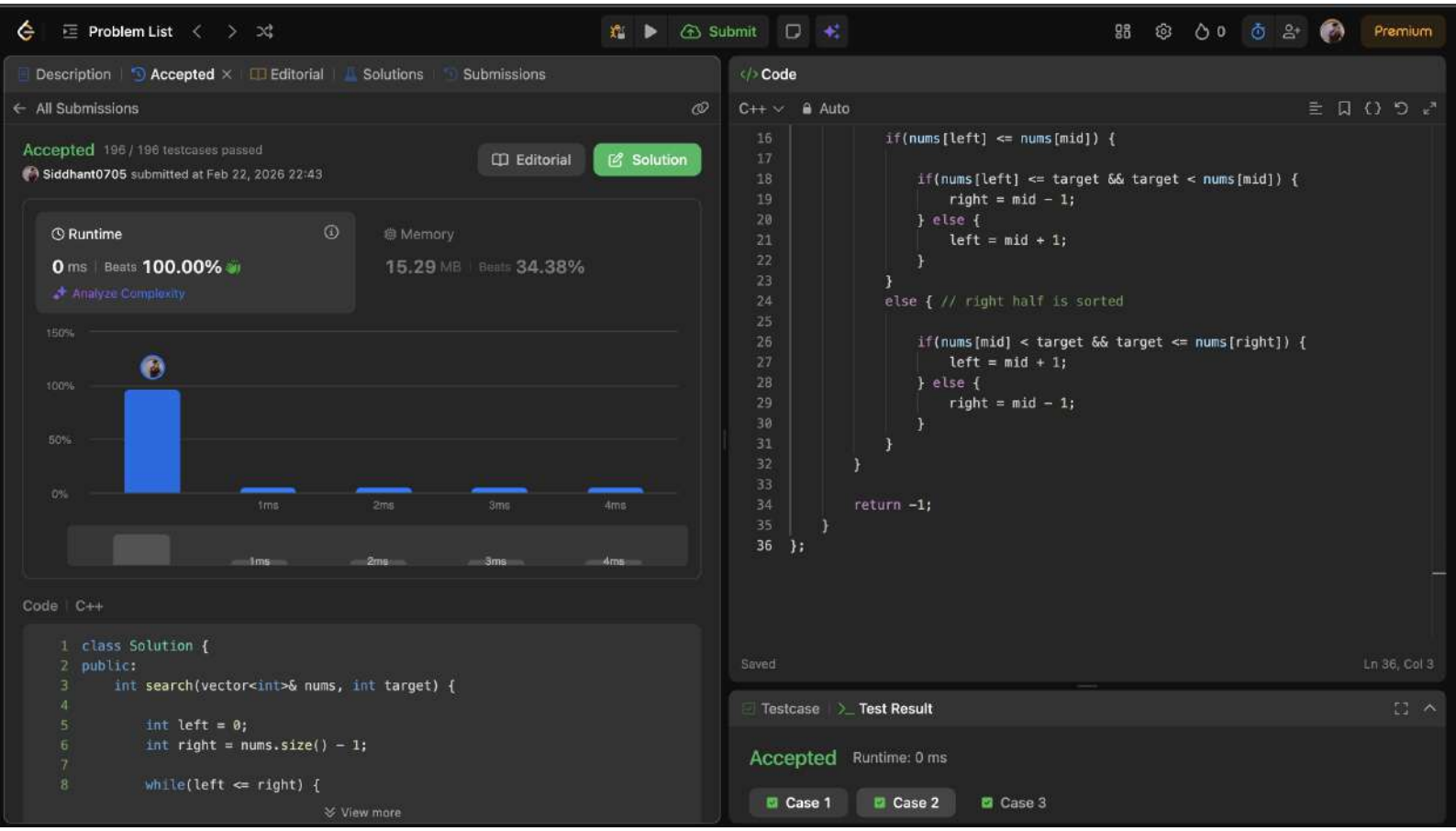

Problem List
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 66 / 66 testcases passed
Siddhant0705 submitted at Feb 22, 2026 22:42
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
15.07 MB Beats 99.86%
75%
50%
25%
0%
1ms
2ms
3ms
4ms
5ms
Code | C++
class Solution {
public:
int minPathSum(vector<vector<int>>& grid) {
int m = grid.size();
int n = grid[0].size();
for(int i = 1; i < m; i++)
View more
Code
C++
Auto
class Solution {
public:
int minPathSum(vector<vector<int>>& grid) {
int m = grid.size();
int n = grid[0].size();
for(int i = 1; i < m; i++)
grid[i][0] += grid[i-1][0];
for(int j = 1; j < n; j++)
grid[0][j] += grid[0][j-1];
for(int i = 1; i < m; i++) {
for(int j = 1; j < n; j++) {
grid[i][j] += min(grid[i-1][j], grid[i][j-1]);
}
}
return grid[m-1][n-1];
}
};
Saved
Ln 22, Col 3
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2


Problem List
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 196 / 196 testcases passed
Siddhant0705 submitted at Feb 22, 2026 22:43
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
15.29 MB | Beats 34.38%
150%
100%
50%
0%
1ms
2ms
3ms
4ms
Code | C++
class Solution {
public:
int search(vector<int>& nums, int target) {
int left = 0;
int right = nums.size() - 1;
while(left <= right) {
View more
Code
C++
Auto
if(nums[left] <= nums[mid]) {
if(nums[left] <= target && target < nums[mid]) {
right = mid - 1;
} else {
left = mid + 1;
}
}
else { // right half is sorted
if(nums[mid] < target && target <= nums[right]) {
left = mid + 1;
} else {
right = mid - 1;
}
}
}
return -1;
}
};
Saved
Ln 36, Col 3
Testcase
Test Result
Accepted
Runtime: 0 ms
Case 1
Case 2
Case 3