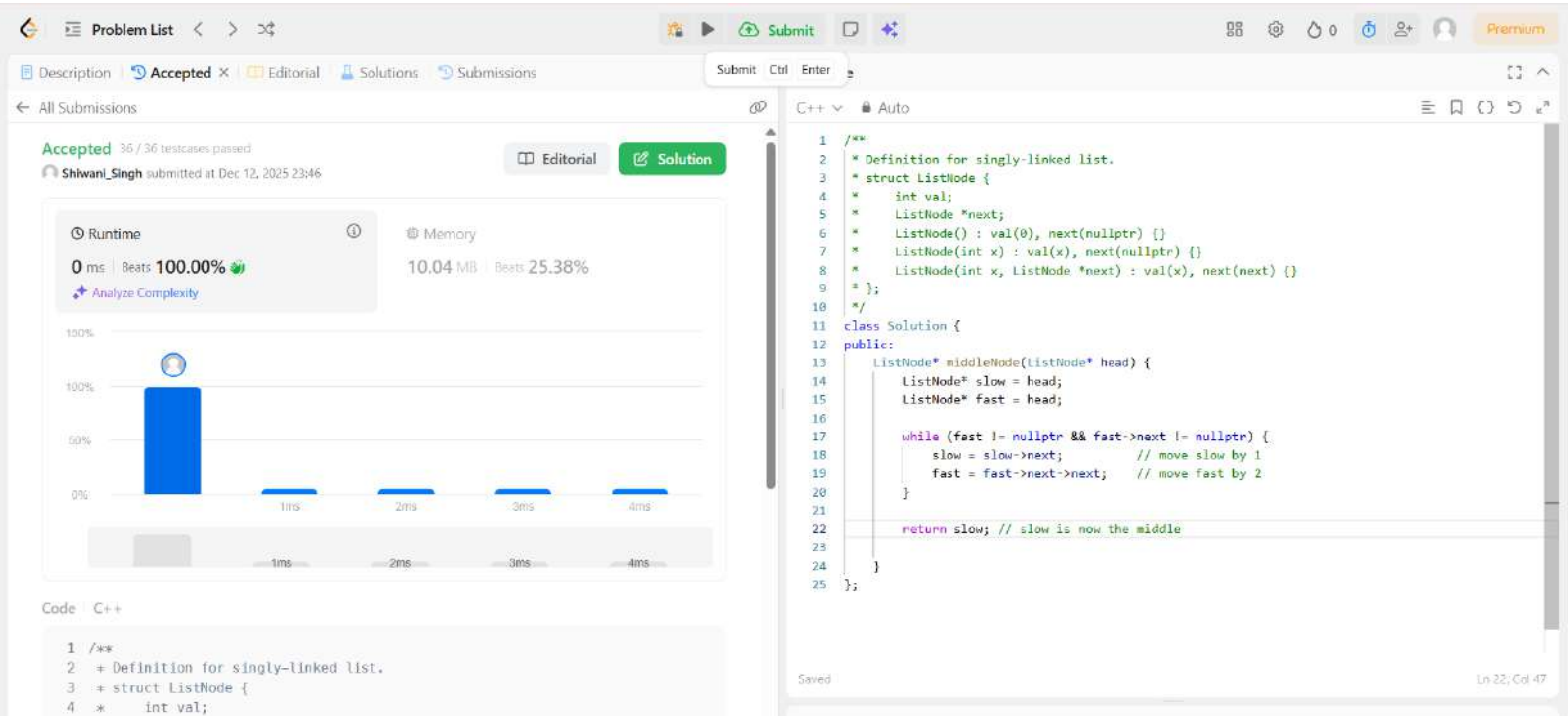
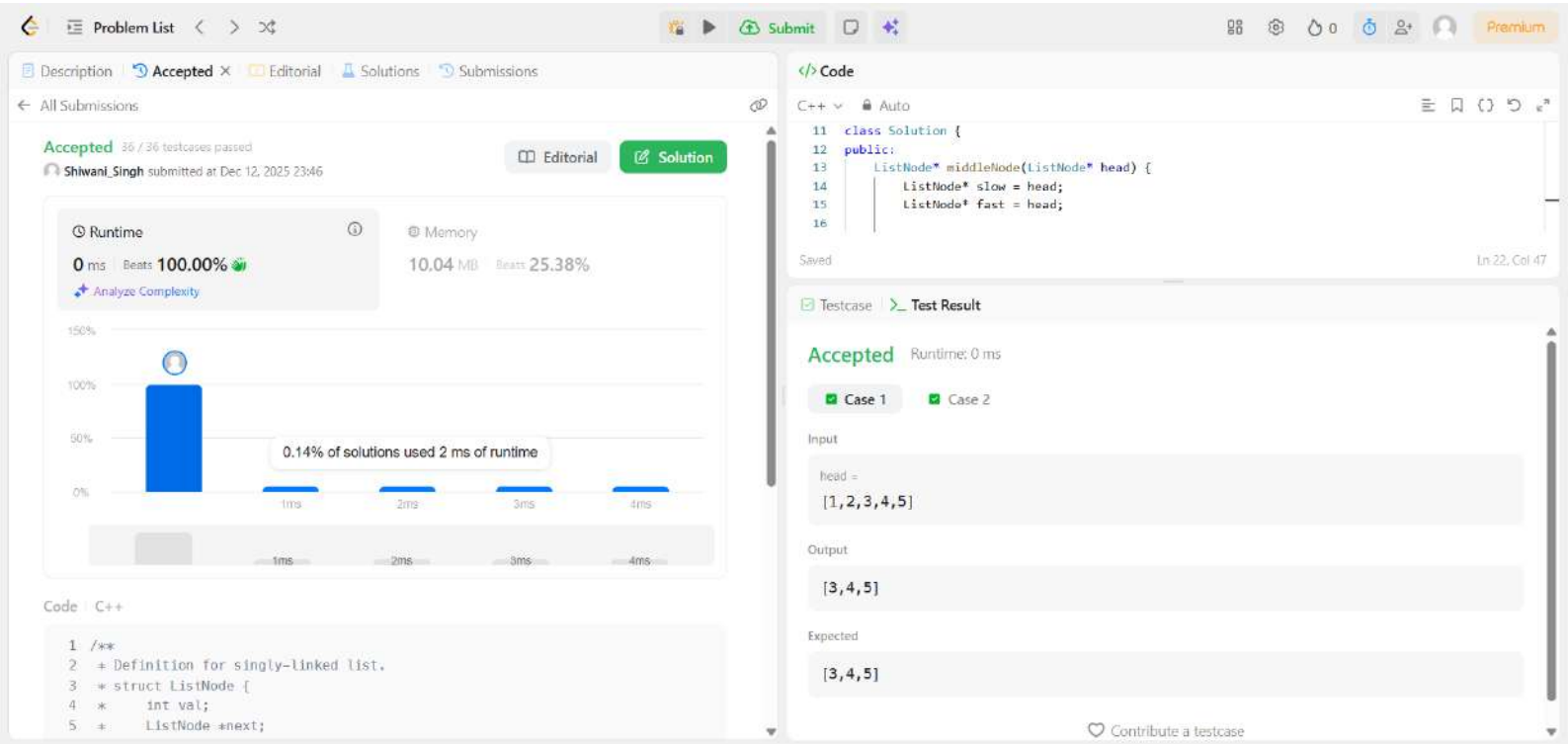

Problem List
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
Submit Ctrl Enter
All Submissions
C++
Auto
Accepted 36 / 36 testcases passed
Shiwani_Singh submitted at Dec 12, 2025 23:46
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
10.04 MB | Beats 25.38%
100%
50%
0%
1ms
2ms
3ms
4ms
Code C++
/**
* Definition for singly-linked list.
* struct ListNode {
* int val;
* ListNode *next;
* ListNode() : val(0), next(nullptr) {}
* ListNode(int x) : val(x), next(nullptr) {}
* ListNode(int x, ListNode *next) : val(x), next(next) {}
* };
*/
class Solution {
public:
ListNode* middleNode(ListNode* head) {
ListNode* slow = head;
ListNode* fast = head;
while (fast != nullptr && fast->next != nullptr) {
slow = slow->next; // move slow by 1
fast = fast->next->next; // move fast by 2
}
return slow; // slow is now the middle
}
};
Saved
Ln 22, Col 47


Problem List
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 36 / 36 testcases passed
Shiwani_Singh submitted at Dec 12, 2025 23:46
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
10.04 MB Beats 25.38%
150%
100%
50%
0%
0.14% of solutions used 2 ms of runtime
1ms
2ms
3ms
4ms
Code | C++
1 /**
2 * Definition for singly-linked list.
3 * struct ListNode {
4 * int val;
5 * ListNode *next;
Code
C++
Auto
11 class Solution {
12 public:
13 ListNode* middleNode(ListNode* head) {
14 ListNode* slow = head;
15 ListNode* fast = head;
16
Saved
Ln 22, Col 47
Testcase | Test Result
Accepted Runtime: 0 ms
Case 1
Case 2
Input
head =
[1,2,3,4,5]
Output
[3,4,5]
Expected
[3,4,5]
Contribute a testcase