Problem List

Submit

Premium

Description | Accepted | Editorial | Solutions | Submissions

All Submissions

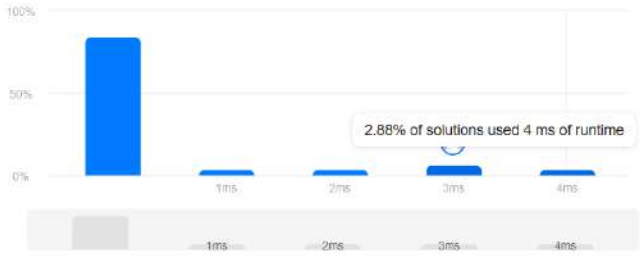
Runtime

3 ms | Beats 10.94%

Analyze Complexity

Memory

20.20 MB | Beats 42.78%



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) {}
```

View more

Code

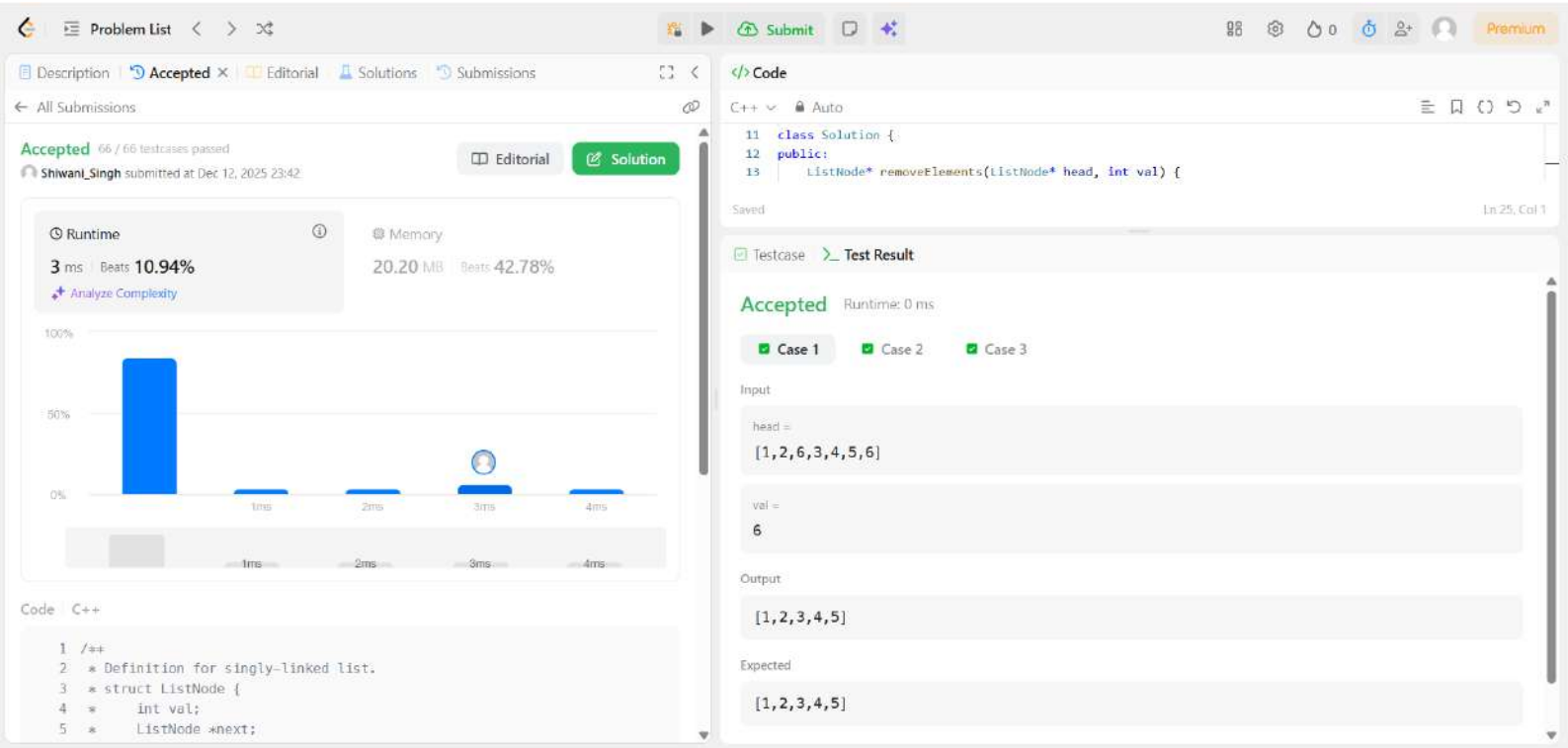
C++ | Auto

```
/**
 * 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* removeElements(ListNode* head, int val) {
        ListNode* dummy = new ListNode(0);
        dummy->next = head;

        ListNode* curr = dummy;

        while (curr->next != nullptr) {
            if (curr->next->val == val) {
                ListNode* temp = curr->next;
                curr->next = curr->next->next;
                delete temp;
            } else {
                curr = curr->next;
            }
        }
        return dummy->next;
    }
};
```

Saved

Ln 25, Col 1

Testcase | Test Result


Problem List
Submit
Premium
Description
Accepted
Editorial
Solutions
Submissions
All Submissions
Accepted 66 / 66 testcases passed
Shiwani_Singh submitted at Dec 12, 2025 23:42
Editorial
Solution
Runtime
3 ms | Beats 10.94%
Analyze Complexity
Memory
20.20 MB Beats 42.78%
100%
50%
0%
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* removeElements(ListNode* head, int val) {
Saved
Ln 25, Col 1
Testcase
Test Result
Accepted Runtime: 0 ms
Case 1
Case 2
Case 3
Input
head =
[1,2,6,3,4,5,6]
val =
6
Output
[1,2,3,4,5]
Expected
[1,2,3,4,5]