

Delete Middle node of linked list

```
class Solution {
public:
    int size(ListNode* head) {
        int cnt = 0;
        ListNode* temp = head;
        while (temp != NULL) {
            cnt++;
            temp = temp->next;
        }
        return cnt;
    }

    ListNode* deleteMiddle(ListNode* head) {
        if (!head || !head->next) {
            return NULL;
        }

        int n = size(head);
        int middle = n / 2;

        ListNode* temp = head;
        for (int i = 0; i < middle - 1; i++) {
            temp = temp->next;
        }
        temp->next = temp->next->next;

        return head;
    }
};
```

The screenshot displays a coding platform interface with the following components:

- Problem List:** Shows the current problem and navigation options.
- Submission Status:** Indicates "Accepted" with 70 / 70 testcases passed. The user "12ananya" submitted at Feb 13, 2025 21:43.
- Performance Metrics:**
 - Runtime:** 7 ms, Beats 12.12%.
 - Memory:** 312.06 MB, Beats 55.29%.
- Graph:** A bar chart showing the distribution of runtime performance across different test cases.
- Code Editor:** Displays the C++ code for the solution, including the `size` and `deleteMiddle` methods.
- Testcase Results:** Shows "Accepted" with a runtime of 0 ms for the current test case.