

LAB PROGRAM 4B-MIDDLE OF THE LL

```
/*
 * Definition for singly-linked list.
 *
 * struct ListNode {
 *     int val;
 *     struct ListNode *next;
 * };
 */
struct ListNode* middleNode(struct ListNode* head) {
    int count=1;
    struct ListNode* temp=head;
    struct ListNode* middle=head;
    while(temp->next!=NULL)
    {
        temp=temp->next;
        count++;
    }
    int pos;
    if(count%2==0)
    {
        pos=(count/2)+1;
    }
    else
    {
        pos=(count+1)/2;
    }
    for(int i=1;i<=pos-1;i++)
    {
        middle=middle->next;
    }
    return middle;
}
```

}

Screenshot of a web browser showing the LeetCode problem "Middle of the Linked List". The status bar indicates "Accepted" with 36/36 testcases passed, submitted at Nov 24, 2025 03:34. The runtime is 0 ms (Beats 100.00%) and memory usage is 8.76 MB (Beats 11.90%). The code editor shows the following Java solution:

```
class Solution {
    public ListNode middleNode(ListNode head) {
        if (head == null || head.next == null) {
            return head;
        }
        ListNode slow = head;
        ListNode fast = head;
        while (fast != null && fast.next != null) {
            slow = slow.next;
            fast = fast.next.next;
        }
        return slow;
    }
}
```

The "Test Result" section shows "Accepted" with runtime 0 ms. It includes two cases: Case 1 (Input: head = [1,2,3,4,5], Output: [3,4,5]) and Case 2 (Input: head = [1,2,3,4,5,6], Output: [4,5,6]). The "Expected" output for both cases is [3,4,5] and [4,5,6] respectively. A "Contribute a testcase" button is present.

Screenshot of a web browser showing the LeetCode problem "Middle of the Linked List". The status bar indicates "Accepted" with 36/36 testcases passed, submitted at Nov 24, 2025 03:34. The runtime is 0 ms (Beats 100.00%) and memory usage is 8.76 MB (Beats 11.90%). The code editor shows the same Java solution as the previous screenshot.

The "Test Result" section shows "Accepted" with runtime 0 ms. It includes two cases: Case 1 (Input: head = [1,2,3,4,5], Output: [3,4,5]) and Case 2 (Input: head = [1,2,3,4,5,6], Output: [4,5,6]). The "Expected" output for both cases is [3,4,5] and [4,5,6] respectively. A "Contribute a testcase" button is present.