

LEET CODE 4(Rotate list)

SOURCE CODE:

The screenshot shows the LeetCode editor interface for problem 61, "Rotate List". The left pane displays the problem description, which asks to rotate a linked list to the right by k places. It includes two examples: Example 1 with a list [1, 2, 3, 4, 5] rotated by 2 to become [4, 5, 1, 2, 3], and Example 2 with a list [0, 1, 2] rotated by 3 to become [0, 1, 2]. The right pane shows the C++ code for the solution, which uses a two-step process: first, it finds the length of the list and calculates $k = k \% \text{length}$; second, it uses a fast-slow pointer technique to find the new head and tail of the rotated list.

```
1 /**  
2  * Definition for singly-linked list.  
3  * struct ListNode {  
4  *     int val;  
5  *     struct ListNode *next;  
6  * };  
7  */  
8 struct ListNode* rotateRight(struct ListNode* head, int k) {  
9     if (head == NULL || k == 0) {  
10        return head;  
11    }  
12    struct ListNode* current = head;  
13    int length = 1;  
14    while (current->next != NULL) {  
15        current = current->next;  
16        length++;  
17    }  
18    k = k % length;  
19    if (k == 0) {  
20        return head;  
21    }  
22    current = head;  
23    for (int i = 1; i < length - k; i++) {  
24        current = current->next;  
25    }  
26    struct ListNode* newHead = current->next;  
27    current->next = NULL;  
28    current = newHead;  
29    while (current->next != NULL) {  
30        current = current->next;  
31    }  
32    current->next = head;  
33    return newHead;  
34 }
```

OUTPUT:

CASE-1

This screenshot shows the test results for Case 1 of the "61. Rotate List" problem. The input is a linked list with values [1, 2, 3, 4, 5] and $k = 2$. The output is [4, 5, 1, 2, 3], which matches the expected result. The test case is marked as "Accepted" with a runtime of 0 ms. The interface also includes a "Contribute a testcase" button.

Accepted Runtime: 0 ms

Case 1 Case 2

Input

head =
[1, 2, 3, 4, 5]

k =
2

Output

[4, 5, 1, 2, 3]

Expected

[4, 5, 1, 2, 3]

Contribute a testcase

CASE-2

Problem List

DescriptionEditorialSolutionsSubmissions

61. Rotate List

MediumTopicsCompanies

Given the `head` of a linked list, rotate the list to the right by `k` places.

Example 1:

1 → 2 → 3 → 4 → 5

rotate 1 5 → 1 → 2 → 3 → 4

rotate 2 4 → 5 → 1 → 2 → 3

Input: head = [1,2,3,4,5], k = 2
Output: [4,5,1,2,3]

Example 2:

0 → 1 → 2

rotate 1 2 → 0 → 1

rotate 2 1 → 2 → 0

rotate 3 0 → 1 → 2

Code

TestcaseTest Result

Accepted Runtime: 0 ms

Case 1Case 2

Input

head =
[0,1,2]

k =
4

Output

[2,0,1]

Expected

[2,0,1]

Contribute a testcase