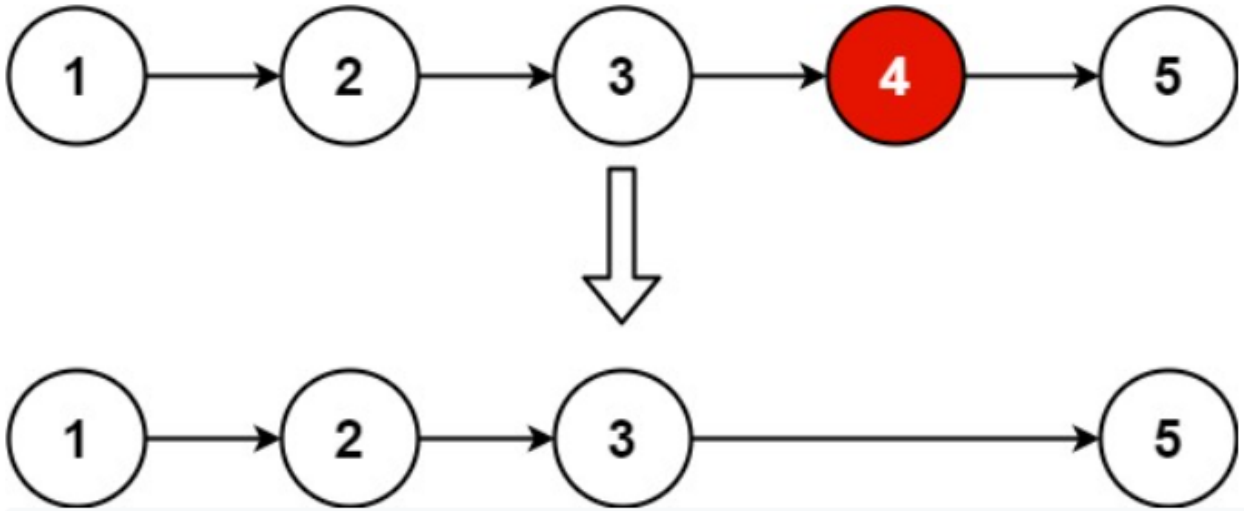
```
19 | Medium | Remove Nth Node From End of List | Linked List

Given the head of a linked list, remove the nth node from the end of the list
and return its head.

Constraints:
The number of nodes in the list is sz.
1 <= sz <= 30
0 <= Node.val <= 100
1 <= n <= sz
```

Example 1:




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

Example 2:

```
Input: head = [1], n = 1
Output: []
```

Example 3:

```
Input: head = [1,2], n = 1
Output: [1]
```

```cpp
// to get the length of the LL
int getLength(ListNode *head) {
    ListNode* t = head;
    int count = 0;
    while(t) {
        count++;
        t = t->next;
    }
    return count;
}
```

```cpp
ListNode* removeNthFromEnd(ListNode* head, int n) {
    ListNode *p1 = head;

    // if we need to delete first node.
    if(getLength(head) == n) {
        return head->next;
    }

    while(n>1) {
        p1 = p1->next;
        n--;
    }

    ListNode* slow = head, *fast = p1->next;
    while(fast != NULL && fast->next != NULL) {
        slow = slow->next;
        fast = fast->next;
    }

    ListNode* del = slow->next;
    slow->next = slow->next->next;
    delete del;
    return head;
}
```

#100daysofDSA





/rvislive

Rakesh Vishwakarma