19. Remove Nth Node From End of List

Add to List ▾

Question | Editorial Solution | My Submissions (/problems/remove-nth-node-from-end-of-list/submissions/)

Total Accepted: **155329** Total Submissions: **482413** Difficulty: **Easy** Contributors: **Admin**

Notes

Given a linked list, remove the n^{th} node from the end of list and return its head.

For example,

```
Given linked list: 1->2->3->4->5, and n = 2.

After removing the second node from the end, the linked list becomes 1->2->3->5.
```

Note:

Given *n* will always be valid.

Try to do this in one pass.

Subscribe (/subscribe/) to see which companies asked this question

Show Tags

Have you met this question in a real interview? Yes No

Discuss (https://discuss.leetcode.com/category/27) | Top Solutions | Pick One (/problems/random-one-question/)

C++ ▾

```cpp
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
 *     ListNode *next;
 *     ListNode(int x) : val(x), next(NULL) {}
 * };
 */
class Solution {
public:
    ListNode* removeNthFromEnd(ListNode* head, int n) {
        int length = 0;
        if(head==NULL) return NULL;
        ListNode* slow = new ListNode(0);
        ListNode* prehead = slow;
        slow->next = head;
        ListNode* fast = head;
        int i=0;
        while(i<n&&fast!=NULL){
            fast=fast->next;
            i++;
        }
        if(i<n)
            return head;
        while(fast!=NULL){
            slow = slow->next;
            fast = fast->next;
        }
        slow->next = slow->next->next;
        return prehead->next;

    }
};
```

Custom Testcase □

Contribute Testcase

Run Code | Submit Solution