19. Remove Nth Node From End of List ★

Question | Editorial Solution

My Submissions (/problems/remove-nth-node-from-end-of-list/submissions/)

Total Accepted: **131231** Total Submissions: **421530** Difficulty: **Easy**

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://leetcode.com/discuss/questions/oj/remove-nth-node-from-end-of-list)

Pick One (/problems/random-one-question/)

C++ </>

```
1  /**
2   * Definition for singly-linked list.
3   * struct ListNode {
4   *     int val;
5   *     ListNode *next;
6   *     ListNode(int x) : val(x), next(NULL) {}
7   * };
8   */
9  class Solution {
10 public:
11     ListNode* removeNthFromEnd(ListNode* head, int n) {
12         if (n <= 0) return head;
13         ListNode* prev = head, *end = head;
14         for (int i = 0; i < n; ++i) {
15             if(end) end = end->next;
16         }
17         if (end) {
18             while (end->next) {
```

Send Feedback (mailto:admin@leetcode.com?subject=Feedback)

```
19                  prev = prev->next;
20                  end = end->next;
21              }
22              ListNode* remove = prev->next;
23              // relink
24              prev->next = remove->next;
25              delete remove;
26          }
27          else {
28              // 1->2->3  n = 3
29              head = prev->next;
30              delete prev;
31          }
32          return head;
33      }
```

Notes

Custom Testcase

Run Code | Submit Solution