19. Remove Nth Node From End of List

Medium

Given a linked list, remove the *n*-th node from the end of list and return its head.

**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.

**Follow up:**

Could you do this in one pass?

/\*\*

\* 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) {

if(head==NULL) return NULL;

ListNode\* dummy = (struct ListNode\*)malloc(sizeof(struct ListNode));

dummy->next=head;

dummy->val=0;

ListNode\* front=dummy, \*tail=dummy;

while(front!=NULL&&n>0){

front=front->next;

n--;

}if(front==NULL) return NULL;

while(front->next!=NULL){

front=front->next;

tail=tail->next;

}

tail->next=tail->next->next;

return dummy->next;

}

};

Success

[Details](https://leetcode.com/submissions/detail/205152350/)

Runtime: 4 ms, faster than 100.00% of C++ online submissions for Remove Nth Node From End of List.

Memory Usage: 774.1 KB, less than 59.01% of C++ online submissions for Remove Nth Node From End of List.