**[Remove Nth Node From End of List](https://leetcode.com/problems/remove-nth-node-from-end-of-list/)**

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

**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]

**Constraints:**

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

**Follow up:** Could you do this in one pass?

class Solution {

public:

    ListNode\* removeNthFromEnd(ListNode\* head, int n) {

        ListNode\* dummy = new ListNode(0);

        dummy->next = head;

        ListNode\* first = dummy;

        ListNode\* second = dummy;

        for (int i = 0; i <= n; ++i) {

            first = first->next;

        }

        while (first != nullptr) {

            first = first->next;

            second = second->next;

        }

        ListNode\* temp = second->next;

        second->next = second->next->next;

        delete temp;

        return dummy->next;

    }

};

Link : <https://leetcode.com/problems/remove-nth-node-from-end-of-list/?envType=daily-question&envId=2024-03-03>