

Remove Nth Node From End of List

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.

Solution in C++

```
/**
 * 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) {
        vector<ListNode*> e;
        ListNode *p=head;
        while(p) {
            e.push_back(p);
            p=p->next;
        }
        if (n==e.size()) return head->next;
        else if (n==1) e[e.size()-2] -> next = NULL;
        else e[e.size()-n-1]->next = e[e.size()-n]->next;
        return head;
    }
};
```

Solution in Java - Time Complexity: O(L), Space Complexity: O(1)

```
public ListNode removeNthFromEnd(ListNode head, int n) {
    ListNode dummy = new ListNode(0);
    dummy.next = head;
    int length = 0;
    ListNode first = head;
    while (first != null) {
        length++;
        first = first.next;
    }
    length -= n;
    first = dummy;
    while (length > 0) {
        length--;
        first = first.next;
    }
    first.next = first.next.next;
    return dummy.next;
}
```