一、题目说明

这个题目是19. Remove Nth Node From End of List，不言自明。删除链表倒数第n个元素。难度是Medium！

二、我的解答

链表很熟悉了，直接写代码。

性能如下：

```
Runtime: 8 ms, faster than 35.76% of C++ online submissions for Remove Nth Node From End of List.
Memory Usage: 8.8 MB, less than 5.26% of C++ online submissions for Remove Nth Node From End of List.
```

```
#include<iostream>
using namespace std;

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;
            if(n<0) return NULL;
            int cur = n;
            ListNode*p = head;
            ListNode* nTh = p;
            while(cur>0 && nTh!=NULL){
                nTh = nTh->next;
                cur--;
            }
            //n超过链表长度
            if(nTh==NULL && cur>0) return head;
            //删除第1个元素
            if(nTh==NULL && cur==0){
                ListNode * t = p->next;
                if(t!=NULL){
                    head = p->next;
                    delete p;
                    return head;
                }else{
                    delete p;
                    return NULL;
                }
            }

            while(p!=NULL && nTh!=NULL && nTh->next!=NULL){
```

```
                p=p->next;
                nTh = nTh->next;
            }
            if(p!=NULL){
                ListNode * tmp = p->next;
                if(p->next !=NULL){
                    p->next = tmp->next;
                }

                delete tmp;
            }
            return head;
        }
};
int main(){
    Solution s;
    ListNode dummy(0);
    ListNode *p;
    int i = 5;
    while(i>0){
        ListNode *tmp = new ListNode(i);
        tmp->next = dummy.next;
        dummy.next = tmp;
        i--;
    }
    p = dummy.next;
    while(p!=NULL){
        cout<<p->val<<" ";
        p=p->next;
    }
    cout<<endl;

    ListNode*r = s.removeNthFromEnd(dummy.next,2);
    p = r;
    while(p!=NULL){
        cout<<p->val<<" ";
        p=p->next;
    }
    cout<<endl;

    return 0;
}
```

三、改进

删除一个变量，性能大幅提高：

```
Runtime: 4 ms, faster than 88.76% of C++ online submissions for Remove Nth Node From End of List.
Memory Usage: 8.8 MB, less than 5.26% of C++ online submissions for Remove Nth Node From End of List.
```

改进后代码如下：

```
#include<iostream>
```

```
using namespace std;

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;
            if(n<0) return NULL;
            int cur = n;
            ListNode*p = head;
            ListNode* nTh = p;
            while(cur>0 && nTh!=NULL){
                nTh = nTh->next;
                cur--;
            }
            //n超过链表长度
            if(nTh==NULL && cur>0) return head;
            //删除第1个元素
            if(nTh==NULL && cur==0){
                if(p->next!=NULL){
                    head = p->next;
                    delete p;
                    return head;
                }else{
                    delete p;
                    return NULL;
                }
            }

            while(p!=NULL && nTh!=NULL && nTh->next!=NULL){
                p=p->next;
                nTh = nTh->next;
            }
            if(p!=NULL){
                ListNode * tmp = p->next;
                if(p->next !=NULL){
                    p->next = tmp->next;
                }

                delete tmp;
            }
            return head;
        }
};
int main(){
    Solution s;
    ListNode dummy(0);
    ListNode *p;
    int i = 5;
    while(i>0){
        ListNode *tmp = new ListNode(i);
        tmp->next = dummy.next;
        dummy.next = tmp;
```

```
        i--;
    }
    p = dummy.next;
    while(p!=NULL){
        cout<<p->val<<" ";
        p=p->next;
    }
    cout<<endl;

    ListNode*r = s.removeNthFromEnd(dummy.next,2);
    p = r;
    while(p!=NULL){
        cout<<p->val<<" ";
        p=p->next;
    }
    cout<<endl;

    return 0;
}
```

再次改进:

```
class Solution{
    public:
        ListNode * removeNthFromEnd(ListNode* head,int n){
            if(head==NULL) return NULL;
            if(n<0) return NULL;
            int len = 0;
            ListNode*p = head;
            while(p!=NULL){
                p = p->next;
                len++;
            }
            //n超过链表长度
            if(len<n) return head;
            //删除第1个元素
            if(len==n){
                head = head->next;
                return head;
            }

            int t = len -n -1;
            p=head;
            while(t-->0){
                p=p->next;
            }
            p->next = p->next->next;

            return head;
        }
};
```