Delete Node in a Linked List

错误理解以为给的是链表的起点，导致提交错误代码如下：

class Solution {

public:

void deleteNode(ListNode\* node) {

ListNode \*p,\*q;

if(node->next->next!=NULL)

{

p=node->next->next;

if(p->val==3)

{

q=p->next;

node->next->next=q;

free(p);

}

}

}

};

后经过网上得到的分析：题意即让我们删除链表的一个节点，但是却与我们熟知的一般情况不同，它没有给我们链表的起点，只给了我们一个要删的节点。

不太一样在于：我们之前要删除一个节点的方法是要有其前一个节点的位置，然后将其前一个节点的next连向要删节点的下一个，然后delete掉要删的节点即可。这道题的处理方法是先把当前节点的值用下一个节点的值覆盖了，然后我们删除下一个节点即可，代码如下：

void deleteNode(struct ListNode\* node) {

struct ListNode\* p;

if(node==NULL) return;

if(node->next==NULL) node=NULL; //这里应该同下处理，但也已通过

if(node->next!=NULL)

{

p=node->next;

node->val=p->val;

node->next=p->next;

free(p);

}

}

class Solution {

public:

void deleteNode(ListNode\* node) {

if(node==NULL||node->next==NULL){

return;

}

//这里题意说所给的需要删除的结点不会是最后一个，所以排除node->next==NULL

if(node->next!=NULL){

node->val=node->next->val;

node->next=node->next->next;

}

}

};