描述：

将两个排序链表合并为一个新的排序链表

样例2:

输入: list1 = 1->3->8->11->15->null, list2 = 2->null

输出: 1->2->3->8->11->15->null

因为是两个链表，又是已经排好序的。所以第一时间的想法就是用归并排序的想法来解决这道题。因为归并排序是对已经排好序的两个数组进行操作，然后再让两根指针指向开头，再依次比较辅助数组的元素后就可以解决这个排序问题了。因为链表不一定是长度相等的，再则链表也不能盲目套用归并排序的代码，我就稍微做了一些修改。这道题我觉得首先要了解归并排序，之后问题就不大了。

代码如下：

ListNode \* mergeTwoLists(ListNode \* l1, ListNode \* l2) {

// write your code here

ListNode\* head = new ListNode(0);

ListNode\* a = head;

while (l1 != nullptr && l2 != nullptr) {

if (l1->val < l2->val)

{

a->next = l1;

l1 = l1->next;

a = a->next;

}

else {

a->next = l2;

l2 = l2->next;

a = a->next;

}

}

if (l1 == nullptr)

a->next = l2;

else if (l2 == nullptr)

a->next = l1;

return head->next;

}

以下是通过测试的截图：

