# 21. Merge Two Sorted Lists

Merge two sorted linked lists and return it as a new list. The new list should be made by splicing together the nodes of the first two lists.、

Example:

Input: 1->2->4, 1->3->4

Output: 1->1->2->3->4->4

## 递归算法：

通过这个题目发现自己对递归的思想掌握还是很肤浅。

递归算法的使用注意以下两点就可以：

1. 递归方法的具体内涵透彻；如**计算n的阶乘，只要用n乘以n-1的阶乘即可**。
2. 递归的终止条件。递归求阶乘，终止条件就是n=0或1的时候。

整合List利用递归解决的思路分析：

由于两个链表已经排好顺序，且整合后的链表也需要有顺序的。

递归终止条件：只要任何一个链表结束了，那就终止了。

**递归方法：整合l1和l2,分成两种情况，一是l1开头，然后整合l1.next和l2，二是l2开头，整合l1和l2.next**。这两种情况的条件就是l1和l2的大小关系。

/\*\*

\* Definition for singly-linked list.

\* public class ListNode {

\* int val;

\* ListNode next;

\* ListNode(int x) { val = x; }

\* }

\*/

class Solution {

public ListNode mergeTwoLists(ListNode l1, ListNode l2) {

if(l1 == null) return l2;

if(l2 == null) return l1;

**if(l1.val < l2.val){**

**l1.next = mergeTwoLists(l1.next,l2);**

**return l1;**

**}else{**

**l2.next = mergeTwoLists(l1,l2.next);**

**return l2;**

**}**

}

}

解释思路：整合之后的list也是需要按照顺序的。

这是已经排好序的链表，比较特殊。

因此，如果l1的值小于l2的值，则需要比较11的下一个值是否还小于l2的值，依次类推，需要找到那个恰好大于l1的对应l2的值。

如：1-2-4-9；

5-6-7；

这样的话对于l1的1-2-4是不需要改变的。

举例说明：

l1 为 1；l2 为2；

此时：结果为1-2；

l1 为 1-2，l2为3；

结果为1-2-3.