

合并两个排序的链表

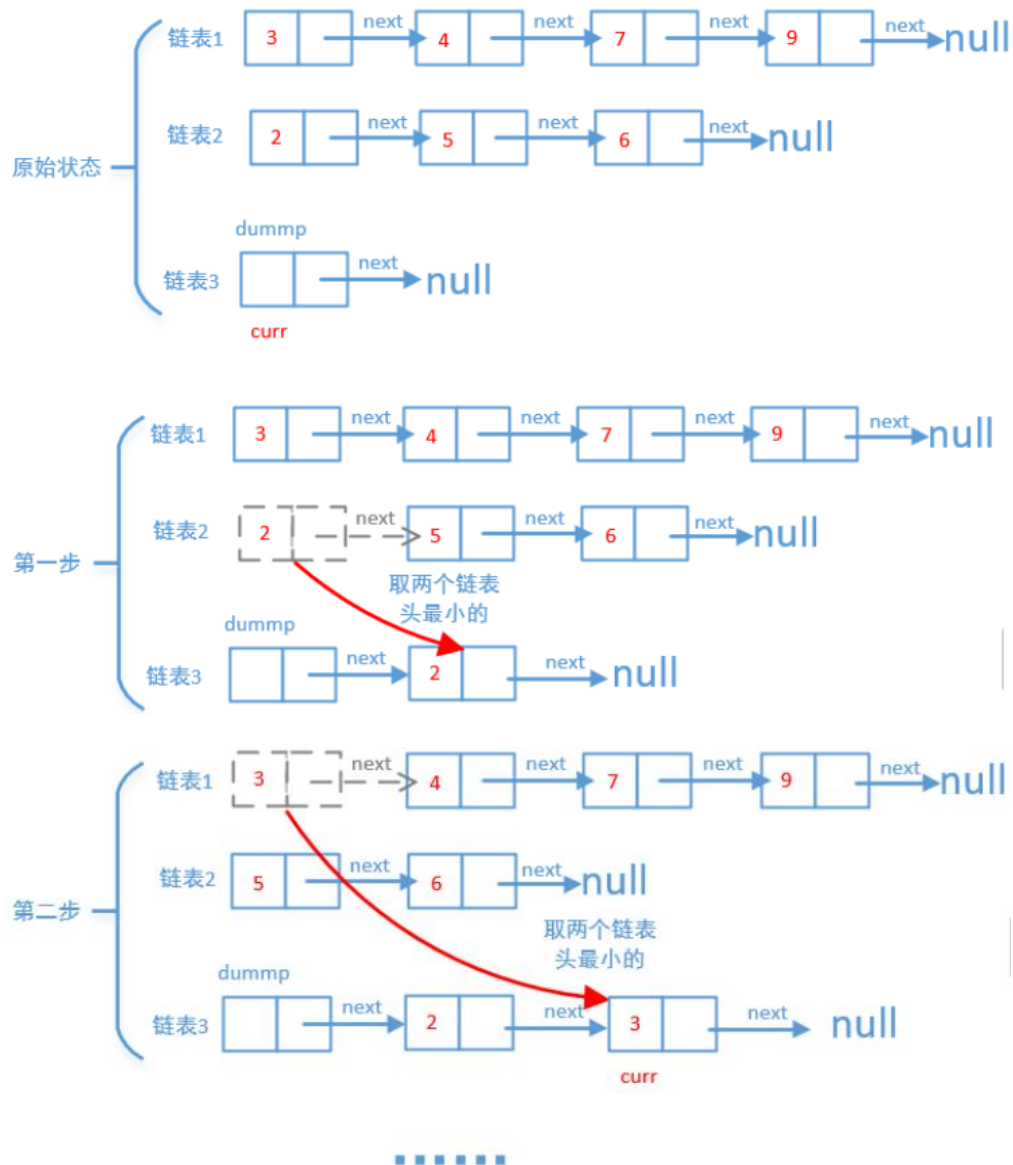
输入两个递增排序的链表，合并这两个链表并使新链表中的节点仍然是递增排序的。

示例 1:

输入：1->2->4, 1->3->4

输出：1->1->2->3->4->4

因为链表是升序的，我们只需要遍历每个链表的头，比较一下哪个小就把哪个链表的头拿出来放到新的链表中，一直这样循环，直到有一个链表为空，然后我们再把另一个不为空的链表挂到新的链表中。



```
/**
 * Definition for singly-linked list.
 * struct ListNode {
 *     int val;
```

```
*     ListNode *next;
*     ListNode(int x) : val(x), next(NULL) {}
* };
*/
class Solution {
public:
    ListNode* mergeTwoLists(ListNode* l1, ListNode* l2) {
        if(l1==nullptr)
            return l2;
        if(l2==nullptr)
            return l1;
        if(l1->val<l2->val)//比较头结点 谁小谁先排，大的排后面
        {
            l1->next=mergeTwoLists(l1->next, l2);
            return l1;
        }
        else
        {
            l2->next=mergeTwoLists(l1, l2->next);
            return l2;
        }
    }
};
```