

剑指Offer（十六）：合并两个排序的链表

🕒 2017年12月3日 15:22:41 🗨 14 🌡 6,521 °C 📄 编辑



一、前言

本系列文章为《剑指Offer》刷题笔记。

刷题平台：[牛客网](#)

书籍下载：[共享资源](#)

二、题目

输入两个单调递增的链表，输出两个链表合成后的链表，当然我们需要合成后的链表满足单调不减规则。

1、思路

先判断输入的链表是否为空的指针。如果第一个链表为空，则直接返回第二个链表；如果第二个链表为空，则直接返回第一个链表。如果两个链表都是空链表，合并的结果是得到一个空链表。

两个链表都是排序好的，我们只需要从头遍历链表，判断当前指针，哪个链表中的值小，即赋给合并链表指针即可。使用递归就可以轻松实现。

2、代码

C++:

```
1  /*
2  struct ListNode {
3      int val;
4      struct ListNode *next;
5      ListNode(int x) :
6          val(x), next(NULL) {
7          }
8  };*/
9  class Solution {
10 public:
11     ListNode* Merge(ListNode* pHead1, ListNode* pHead2)
12     {
13         //判断指针是否为空
14         if(pHead1 == NULL){
15             return pHead2;
16         }
17         else if(pHead2 == NULL){
18             return pHead1;
19         }
20         ListNode* pMergedHead = NULL;
21         if(pHead1->val < pHead2->val){
22             pMergedHead = pHead1;
23             pMergedHead->next = Merge(pHead1->next, pHead2);
24         }
25         else{
26             pMergedHead = pHead2;
27             pMergedHead->next = Merge(pHead1, pHead2->next);
28         }
29         return pMergedHead;
30     }
31 };
```

Python2.7:

```
1  #-*- coding:utf-8 -*-
```

```
2 # class ListNode:
3 #     def __init__(self, x):
4 #         self.val = x
5 #         self.next = None
6 class Solution:
7     # 返回合并后列表
8     def Merge(self, pHead1, pHead2):
9         # write code here
10        if not pHead1:
11            return pHead2
12        if not pHead2:
13            return pHead1
14        pMergeHead = None
15        if pHead1.val < pHead2.val:
16            pMergeHead = pHead1
17            pMergeHead.next = self.Merge(pHead1.next, pHead2)
18        else:
19            pMergeHead = pHead2
20            pMergeHead.next = self.Merge(pHead1, pHead2.next)
21        return pMergeHead
```



微信公众号

分享技术，乐享生活：微信公众号搜索

「JackCui-AI」关注一个在互联网摸爬滚
打的潜行者。

如果将梦想作为信仰,不放棄地追求下去,一定会梦想成真的。--- 岸本齐史《火影忍者》