

剑指Offer（二十六）：二叉搜索树与双向链表

🕒 2017年12月14日 11:28:48 🔒 15 🌡 6,071 °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

输入一棵二叉搜索树，将该二叉搜索树转换成一个排序的**双向链表**。要求不能创建任何新的结点，只能调整树中结点指针的指向。

1、思路

举例说明：

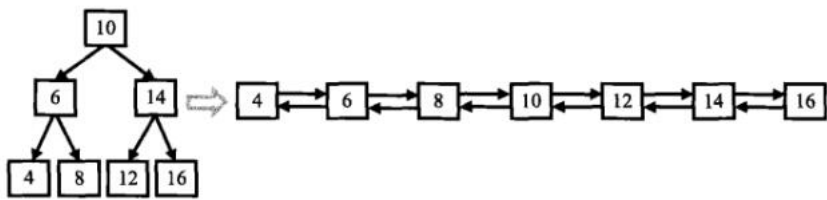
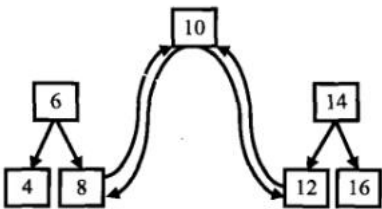


图 4.12 一棵二叉搜索树及转换之后的排序双向链表

二叉搜索树如上图所示，我们将其转换为配需双向链表。

根据二叉搜索树的特点：左结点的值<根结点的值<右结点的值，我们不难发现，使用**二叉树**的中序遍历出来的数据的数序，就是排序的顺序。因此，首先，确定了二叉搜索树的遍历方法。

接下来，我们看下图，我们可以把树分成三个部分：值为10的结点、根结点为6的左子树、根结点为14的右子树。根据排序双向链表的定义，值为10的结点将和它的左子树的最大一个结点链接起来，同时它还将和右子树最小的结点链接起来。



按照中序遍历的顺序，当我们遍历到根结点时，它的左子树已经转换成一个排序的好的双向链表了，并且处在链表中最后一个的结点是当前值最大的结点。我们把值为8的结点和根结点链接起来，10就成了最后一个结点，接着我们就去遍历右子树，并把根结点和右子树中最小的结点链接起来。

2、代码

C++:

```
1  /*
2  struct TreeNode {
3      int val;
4      struct TreeNode *left;
5      struct TreeNode *right;
6      TreeNode(int x) :
7          val(x), left(NULL), right(NULL) {
8      }
9  };*/
10 class Solution {
11 public:
12     TreeNode* Convert(TreeNode* pRootOfTree)
13     {
14         //用于记录双向链表尾结点
15         TreeNode* pLastNodeInList = NULL;
16
17         //开始转换结点
18         ConvertNode(pRootOfTree, &pLastNodeInList);
19
20         //pLastNodeInList指向双向链表的尾结点，我们需要重新回头结点
21         TreeNode* pHeadOfList = pLastNodeInList;
22         while(pHeadOfList != NULL && pHeadOfList->left != NULL){
23             pHeadOfList = pHeadOfList->left;
24         }
25         return pHeadOfList;
26     }
27
28     void ConvertNode(TreeNode* pNode, TreeNode** pLastNodeInList){
29         //叶结点直接返回
30         if(pNode == NULL){
31             return;
32         }
33         TreeNode* pCurrent = pNode;
34         //递归左子树
35         if(pCurrent->left != NULL)
36             ConvertNode(pCurrent->left, pLastNodeInList);
37
38         //左指针
39         pCurrent->left = *pLastNodeInList;
40         //右指针
41         if(*pLastNodeInList != NULL){
42             (*pLastNodeInList)->right = pCurrent;
43         }
44         //更新双向链表尾结点
45         *pLastNodeInList = pCurrent;
46         //递归右子树
47         if(pCurrent->right != NULL){
48             ConvertNode(pCurrent->right, pLastNodeInList);
49         }
50     }
51 };
```

C+



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。

人在为了保护最重要的人的时候，会变得非常坚强。--- 白《火影忍者》