# 二叉搜索树与双向链表

## 题目

[牛客网](https://www.nowcoder.com/practice/947f6eb80d944a84850b0538bf0ec3a5?tpId=13&tqId=11179&rp=1&ru=%2Fta%2Fcoding-interviews&qru=%2Fta%2Fcoding-interviews%2Fquestion-ranking&tPage=2)

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。

## 解题思路

1. 由于 BST 的特性，采用中序遍历正好符合排序
2. 要考虑 root 节点要与 左节点的最大值连接，与右节点的最小值连接
3. 增加一个已排序链表的指针，指向最后一个已排序节点

public TreeNode Convert(TreeNode pRootOfTree) {
 if (pRootOfTree == null) {
 return null;
 }

 TreeNode[] nodeList = {new TreeNode(-1)};

 ConvertToLink(pRootOfTree, nodeList);

 TreeNode cursor = pRootOfTree;
 while (cursor.left != null) {
 cursor = cursor.left;
 }

 cursor.right.left = null;
 return cursor.right;
}

private void ConvertToLink(TreeNode root, TreeNode[] nodeList) {
 if (root == null) {
 return;
 }

 ConvertToLink(root.left, nodeList);

 root.left = nodeList[0];
 nodeList[0].right = root;
 nodeList[0] = root;

 ConvertToLink(root.right, nodeList);
}