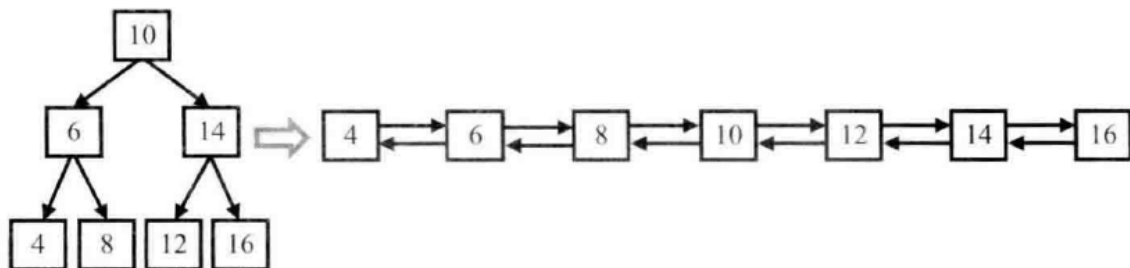


面试题 36：二叉搜索树与双向链表

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节点，只能调整树中节点指针的指向。比如，输入图 4.15 中左边的二叉搜索树，则输出转换之后的排序双向链表。二叉树节点的定义如下：

```
struct BinaryTreeNode
{
    int                m_nValue;
    BinaryTreeNode*    m_pLeft;
    BinaryTreeNode*    m_pRight;
};
```



解

递归转换左子树和右子树，将根节点与左子树和右子树分别相连

```
TreeNode* convert(TreeNode* root)
{
    if(!root)
        return nullptr;
    if(root && !root->left && !root->right)
    {
        return root;
    }
    TreeNode* left=convert(root->left);
    TreeNode* p=left;
    while(p && p->right)
    {
        p=p->right;
    }
    if(left)
    {
        root->left=p;
        p->right=root;
    }
}
```

```
}
TreeNode* right=convert(root->right);
if(right)
{
    root->right=right;
    right->left=root;
}
return left==nullptr?root:left;
}
```