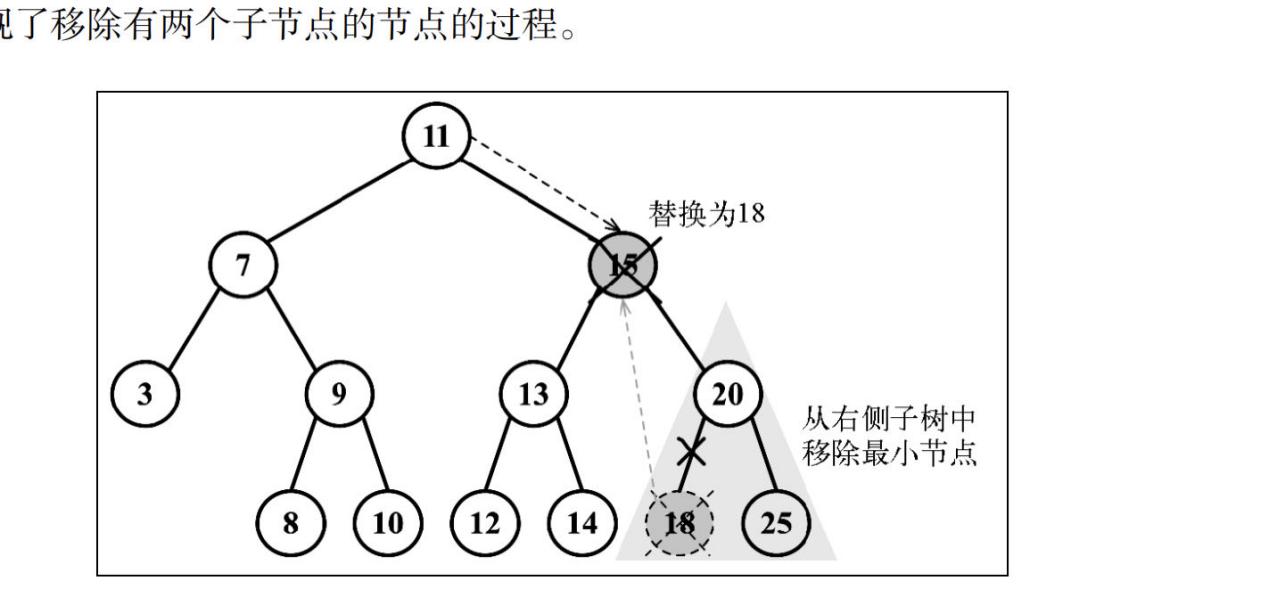
【移除二叉搜索树中两侧都有自节点的节点】对于移除二叉搜索树种有两个子节点的节点的算法来说，你知道为什么要将移除节点右侧子树中的节点值最小的节点找出来，并将它更新到要删除的节点吗？

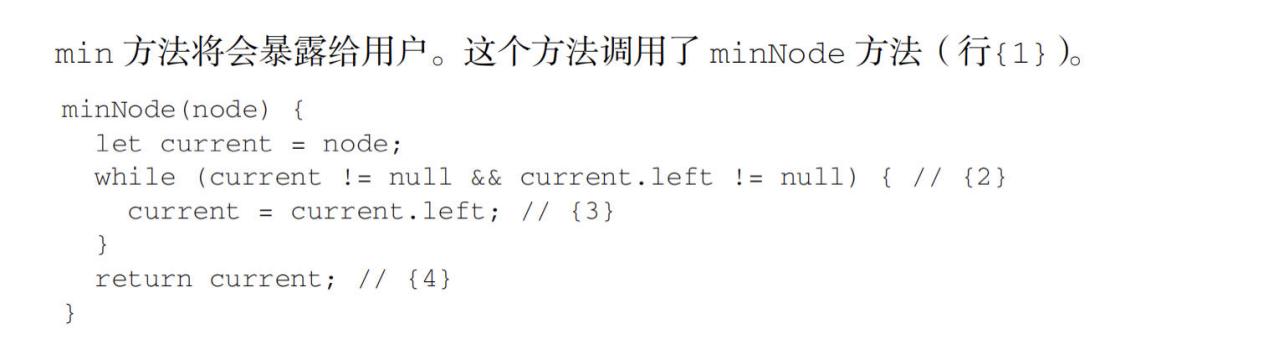


这里其实有几条规律：

1. 二叉搜索树跟节点的左侧节点永远都比根节点小，右侧节点永远都比根节点大。
2. 对于一个只有三层的二叉搜索树来说。右侧节点的左侧节点，它的值就是大于根节点，而又小于其它右侧节点值得节点。

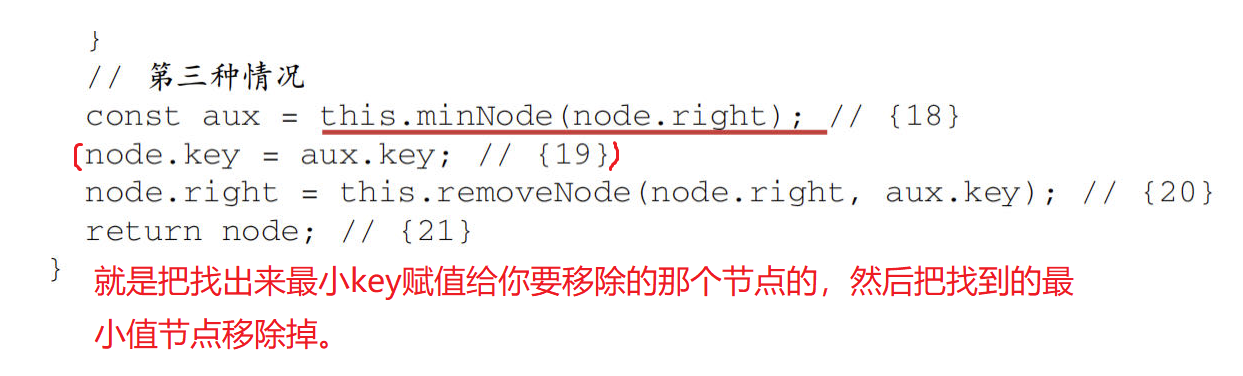
【移除二叉搜索树中两侧都有自节点的节点】你能尝试完成移除一个二叉搜索树中一个两侧都有节点的节点的算法吗？

首先移除这个节点，需要找到它右侧子树中的最小节点。所以我们要有一个寻找二叉树中最小值得方法，这个方法我们先前写过，这里我帮你回忆一下：



其实就是递归的去寻找节点的左侧节点，直到找不到为止。

接下来我们来欣赏一下，删除一个有两侧节点的节点的算法：



什么是AVL自平衡树？

AVL 树是一种自平衡二叉搜索树，意思是任何一个节点左右两侧子树的高度之差最多为 1。

最后我在来补充一下为什么会需要AVL自平衡树：

