

1 对修改后的 remove 函数的阐述和分析

对于修改后的 remove 函数，首先检查节点是否为空。然后递归查找要删除的节点，小于当前元素就递归到左子树，反之就递归到右子树。如果当前节点有两个子节点，那么通过 detachMin 函数找到右子树中的最小节点，替换当前节点并且删除。如果当前节点只有一个子节点或者没有子节点，用非空子节点替换当前节点。然后我用辅助函数 balance 来保持 AVL 树的平衡。对于 balance 树，我的设计思路是先检查节点是否为空，然后检查并修正左子树的平衡，如果左子树高度大于右子树，那么就利用 rotateWithLeftChild 函数进行单旋转，否则就利用 doubleWithLeftChild 进行双旋转，然后同理检查并且修正右子树，最后用 updateHeight 函数更新节点高度。

2 测试的结果

测试结果一切正常。详细的结果见上文的测试程序的设计思路部分。