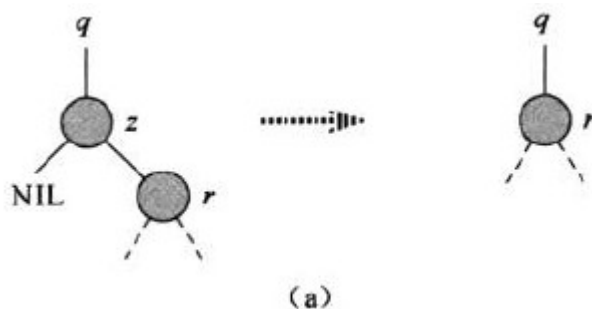


二叉搜索树

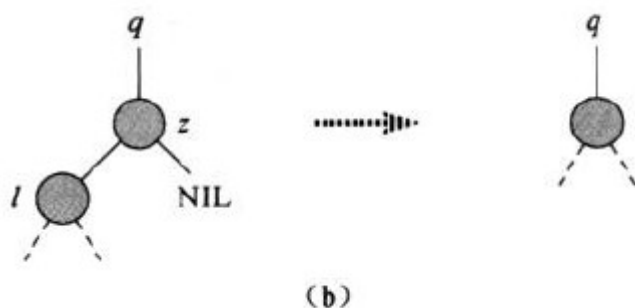
删除操作

从一颗二叉搜索树 T 中删除一个节点 z 的整个策略分为三种情况（其中以最后一种情况最为棘手）：

1. 如果 z 没有孩子节点，那么只是将其删除，并修改 z 的父节点，用 NIL 作为孩子来替换 z
2. 如果 z 只有一个孩子，那么将这个孩子提升到树中 z 的位置上，并修改 z 的父节点，用 z 的孩子来替换 z
 - 有右孩子，没左孩子

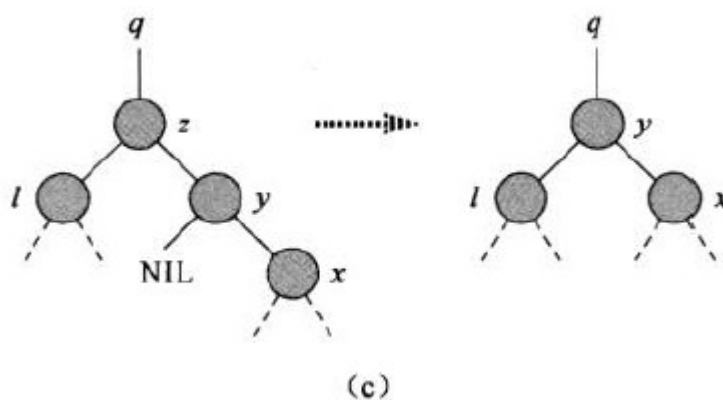


- 有左孩子，没右孩子

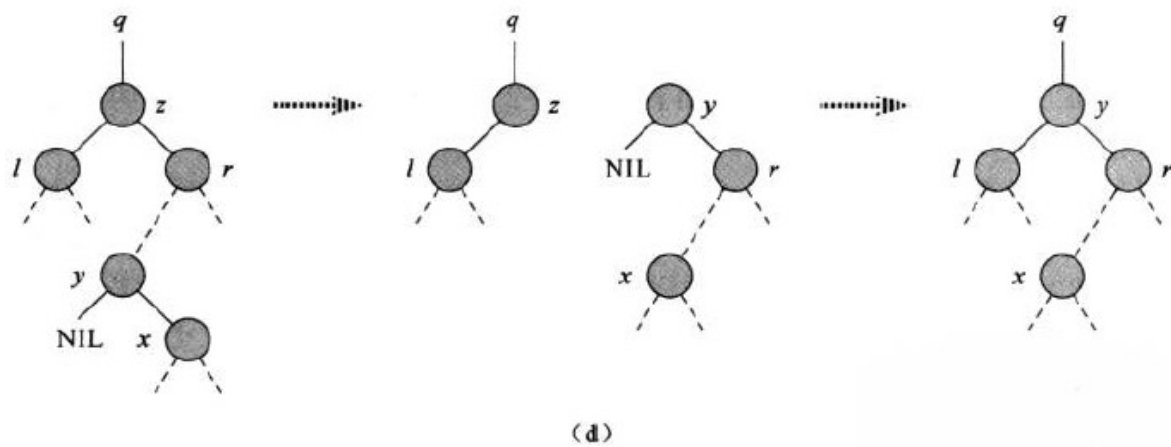


3. 如果 z 有两个孩子，那么找 z 的后继 y （中序遍历的直接后继），并让 y 占据树中 z 的位置。 z 原来的右子树部分成为 y 的新的右子树， z 的左子树成为 y 的新的左子树。这种情况最为棘手，还涉及到 y 原来的子节点和 y 原来父节点相连的问题

- y 是 z 的右孩子



- o y位于z的右子树上，但不是其右孩子



实现

详见作业