

2020/06/09_数据结构_第8课_二叉平衡树_左双旋、右双旋

笔记本： 数据结构

创建时间： 2020/6/10 星期三 10:36

作者： ileemi

标签： 二叉平衡树_左双旋、右双旋

- [二叉搜索树](#)
- [二叉平衡树](#)
 - [二叉平衡树的实现原理](#)
 - [平衡二叉树的失衡调整](#)
 - [左单旋](#)
 - [右单旋](#)
 - [右双旋（先左旋，再右旋 --LR）](#)
 - [左双旋（先右旋，再左旋 --RL）](#)
- [二叉平衡树各操作的时间复杂度](#)

二叉搜索树

各个操作的时间复杂度：

- 插入 --> 对数阶 ($\log(n)$)
- 删除 --> 对数阶 ($\log(n)$)
- 查询 --> 对数阶 ($\log(n)$)
- 修改 --> (先修改再插入 $2\log(n)$) 对数阶 ($\log(n)$)

二叉搜索树的时间复杂度不稳定，当出现左斜，右斜树时，时间复杂度和链表一样。

二叉平衡树

为了避免二叉搜索树变成 "链表"，出现了平衡二叉树，让树的结构看起来尽量 "均衡"，左右子树的结点数量尽量一样多。

性质：

- 可以是空树
- 不是空树时，任何一个结点的左子树与右子树都是平衡二叉树，树中的任意结点，其左子树和右子树的高度差的绝对值不超过1。

二叉平衡树的实现原理

二叉树平衡构建的基本思想就是在构建二叉排序树的过程中，每当插入一个结点时，先检查是否因插入而破坏了树的平衡性，若破坏了树的平衡性，需要找出最小不平衡

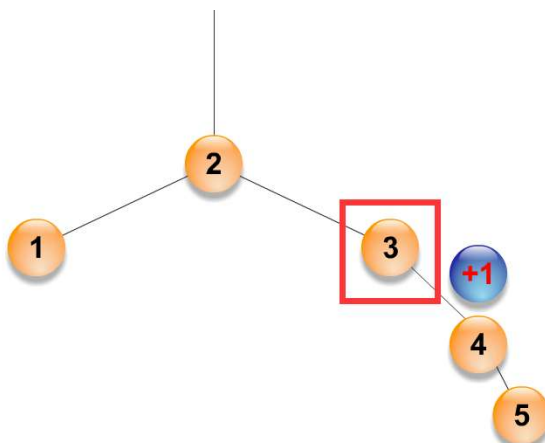
树（距离插入点最近的，且平衡因子（左子树的高度减去右子树的高度）的绝对值大于1的结点为根的子树）。在保持二叉排序树特性的前提下，调整最小不平衡树中各个结点之间的链接关系，进行相应的旋转，使之成为新的平衡子树。

平衡二叉树的失衡调整

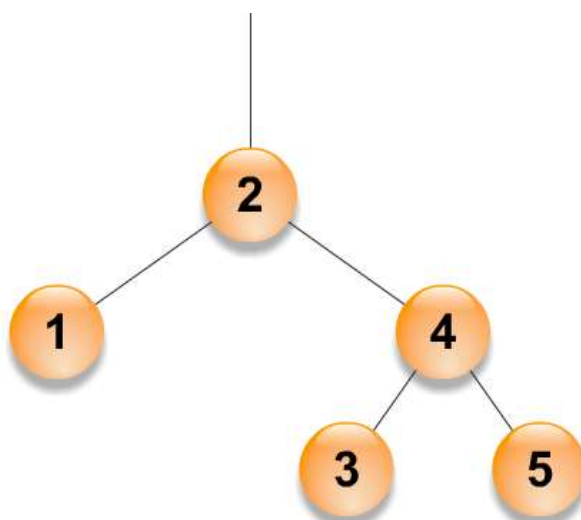
平衡二叉树的失衡调整主要是通过旋转最小不平衡树来实现的。

最小不平衡树： 在新插入的结点向上查找，以第一个平衡因子的绝对值超过1的结点为根的子树称为最小不平衡子树。

树的失衡，可能是由多棵子树同时失衡的，当出现这种情况，只需要调整最小不平衡子树，就可以将不平衡的树调整为平衡的树。



如上图，根结点 2 的平衡因子为：2，根结点2的右孩子 3 的平衡因子为：2，这个时候，整棵树同时存在两棵不平衡子树，这里以 3 为根的子树就是最小不平衡子树，此时只需要将最小不平衡子树进行左旋，这个整数就可以恢复平衡。



左单旋

左孩子的高度减去右孩子的高度差值小于-1时，进行左旋转。

左单旋，将不平衡的节点从右往左拉，其不平衡节点的右孩子变成了父节点，并把其右孩子的左节点出让给降级后节点的右子节点。

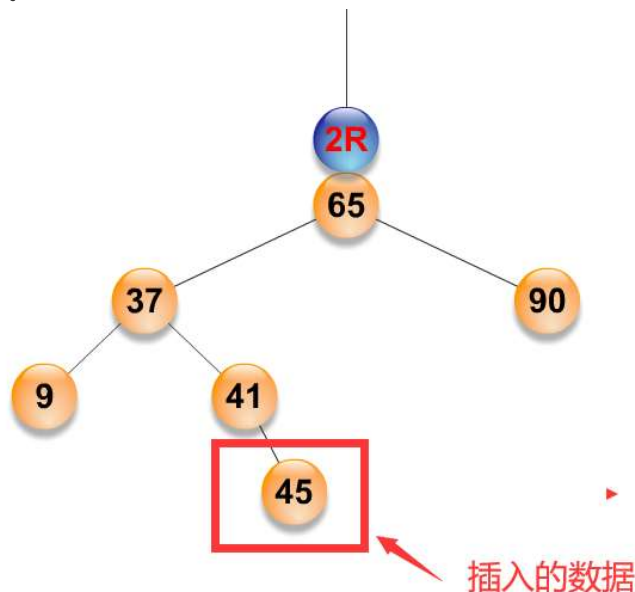
右单旋

右单旋，将不平衡的节点从左往右拉，其不平衡节点的左孩子变成了父节点，并把其左孩子右左节点出让给降级后节点的左子节点。

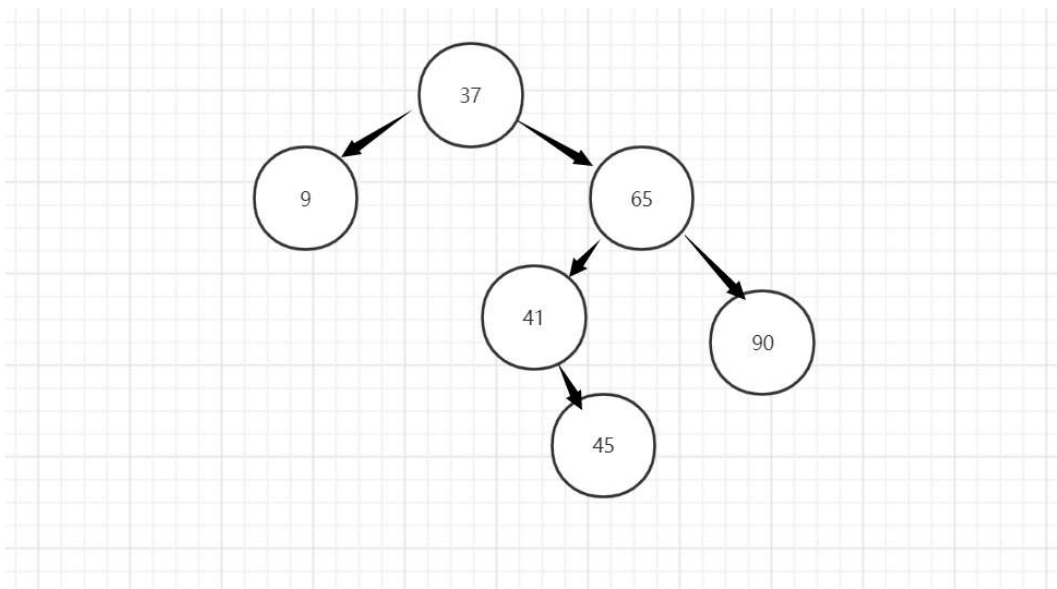
左旋就是往左变换，右旋就是往右变换。不管是左旋还是右旋，旋转的目的都是将节点多的一支 出让节点 给另一个节点少的一支。

右双旋（先左旋，再右旋 --LR）

在原来平衡的二叉树上，其结点的左子树的右子树下有新的结点插入，导致结点的左右子树高度差大于1，如下图，根结点 65，其左子树 37 的右子树 41，该结点 41 的右孩子（或者左孩子）插入了结点 45，导致根结点 65 的左子树高度为 2，右子树高度为 0，差值为2。



这种情况进行一次旋转是不行的，例如：以结点 65 进行右旋，旋转后，其依然不平衡，如下图：



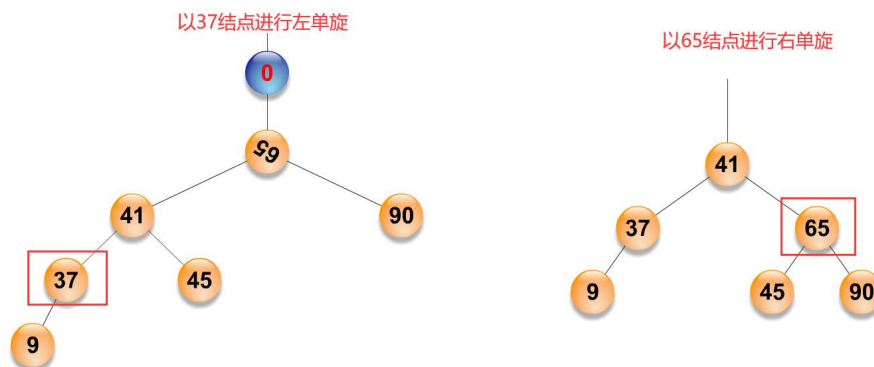
对于上图所示，其是左双旋情况，和上面的右双旋互为镜像，右单旋和左单旋同样是互为镜像关系。

右单旋和左单旋一样，只需要对其不平衡结点进行旋转一次，就可以将不平衡树调整为平衡状态，而左右旋和右左旋需要进行两次旋转才能将不平衡树调整为平衡。

左右旋旋转时机：

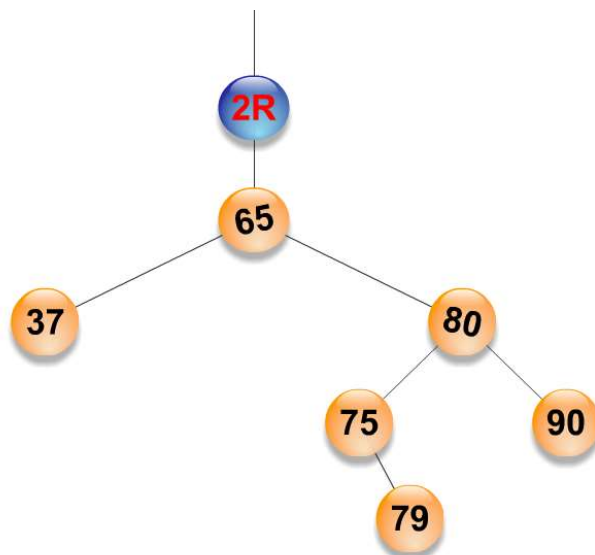
左右旋的旋转方式，需要从不平衡结点找其左孩子的两个结点高度，不平衡结点左孩子的左节点高度低于右节点高度，就需要先将不平衡结点的左孩子进行左单旋，使其旋转后，左节点的高度大于右节点的高度，这个是偶，就可以对不平衡节点进行右单旋。

按照上面的例子，其旋转过程如下图所示：

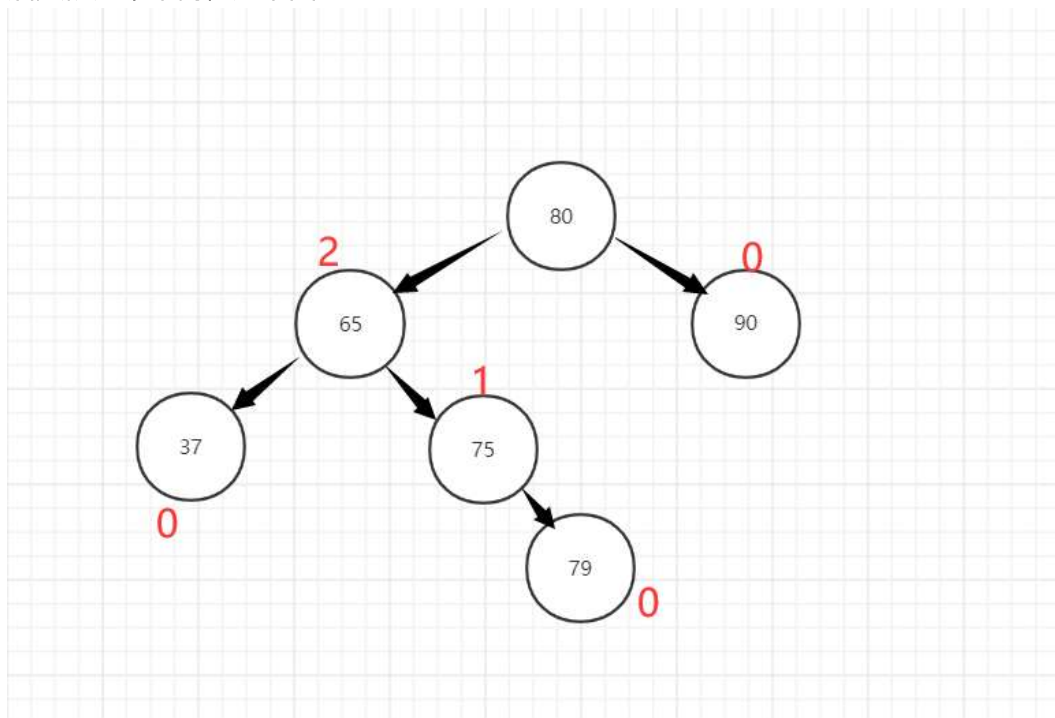


左双旋（先右旋，再左旋 --RL）

在原来平衡的二叉树上，其结点的右子树的左子树下有新的结点插入，导致结点的左右子树高度差大于1，如下图，根结点 65，其右子树 80 的左子树 75，该结点 75 的右孩子（或左孩子）插入了结点 79，导致根结点 65 的左子树高度为 0，右子树高度为 2，差值为2，如下图。



同上述的左右旋一样，这种情况旋转一次是不行的，如果以节点 65 进行左旋，旋转后其依然不平衡，如下图：

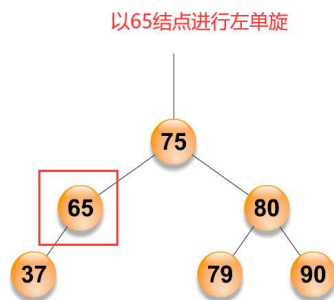
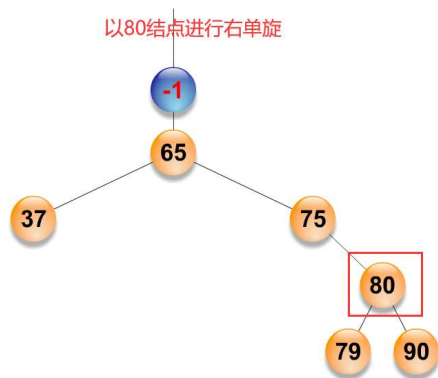


对此，依然要进行两次旋转

右左旋旋转时机：

右左旋的旋转方式，需要从不平衡结点找其右孩子的两个结点高度，不平衡结点右孩子的左结点高度高于右结点高度，就需要先将不平衡结点的右孩子进行右单旋，使其旋转后，左结点的高度小于右结点的高度，这个是偶，就可以对不平衡结点进行左单旋。

按照上面的例子，其旋转过程如下图所示：



二叉平衡树各操作的时间复杂度

- 插入 --> 对数阶 ($\log(n)$)
- 删除 --> 对数阶 ($\log(n)$)
- 查询 --> 对数阶 ($\log(n)$)
- 修改 --> (先修改再插入 $2\log(n)$) 对数阶 ($\log(n)$)