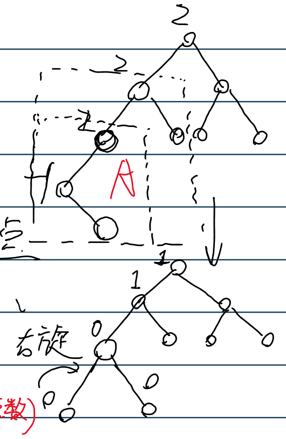
平衡二叉树的失衡情况

Sunday, June 4, 2023 2:30 PM

(1)、二叉树失衡的特点:

1. 新插入的节点,影响其祖先的平衡因子。
2. 当二叉树失衡时,由于只需将最底层失衡节点调为平衡,则上方的平衡因子也会变为平衡,因而需要找到最底层不平衡节点进行调整
3. 插入 ⟹ 影响父节点,在上方某一层开始自下而上的失衡



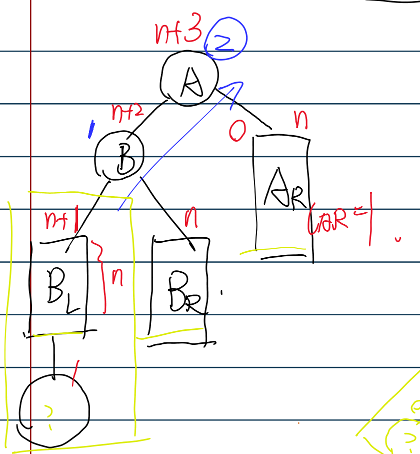
(左子树:直接右旋)

①: LL型失衡: 我们设 n 为节点的层数(最低子节点→该节点共经节点数)

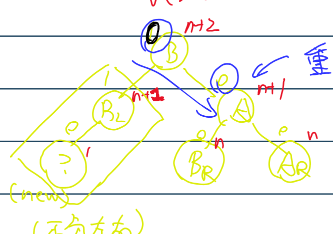
层数 假设最低层失衡结点为A: 当B左子树左节点插入导致失衡:

① 设在如图情况: A_R, B_R, B_L 均为共n层的树。(其最多n层),在 B_L 上加入 new 导致 B_L 变为 n+1)。

另外需要说明: n 可以等于 0,因于 B_R, A_R, B_L 可以同时为 nullptr,操作时仅需整体挪过来即可。



此时,显然将B/以A的右旋转:则变为n+3层。

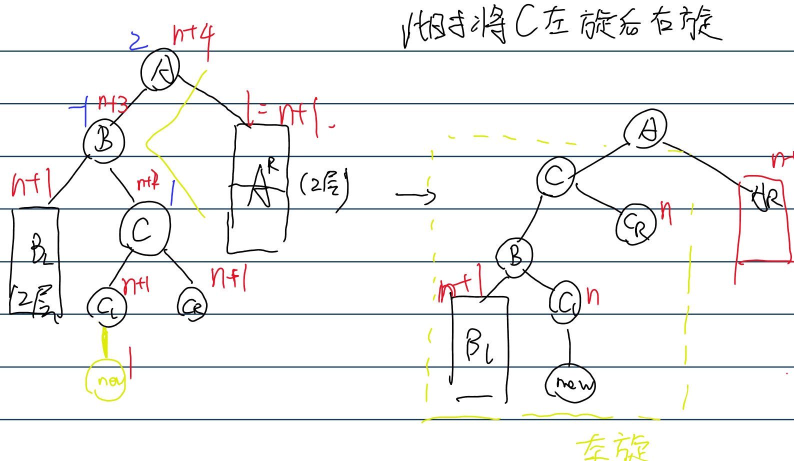


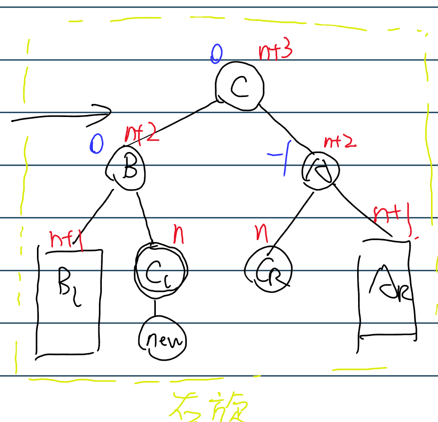
重设对应的平衡因子

(右子树:分两种)

②: LR型失衡: 如图设树原有 n+3 (左n+3,右n+2层),

此时将C左旋后右旋



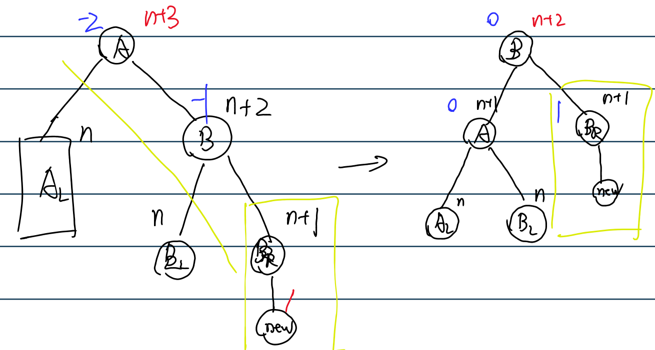


需要说明:如果是在 C_R 下插入结点:仍先左旋后右旋,但旋转之后只需将B的平衡因子改为1,A的平衡因子改为0

(即 2,-1,-1 → 0,1,0) 但是层数的计算方法不变,旋转方式完全相同。

(这也包含了当C本身为插入结点的情况)

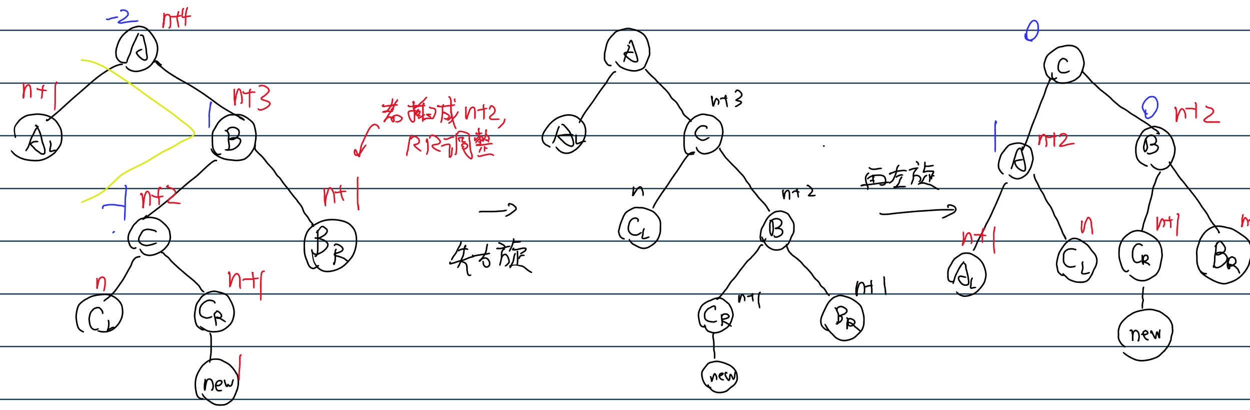
③: RR型失衡: (与LL对称)



④: RL型失衡（与LR型对称）

需要说明的是，我们假设向$B_L(C)$的右子树中进行插入，

（向左树中插入仅权值不同）.



若向 C_L中插入：则：$-2, 1, 1 \rightarrow 0, -1, 0$,

其层数变化情况仍然相同。