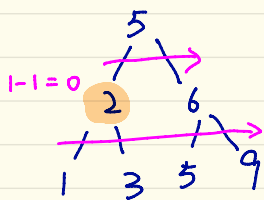


## ① 判断是否 Balanced tree

"Height of a node"



每个  $\rightarrow$  代表一层

● 代表目标节点。

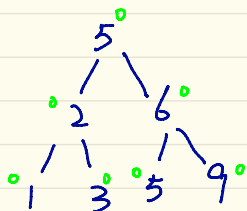
看他的左/右边分别多少层

左正右负，两数相加

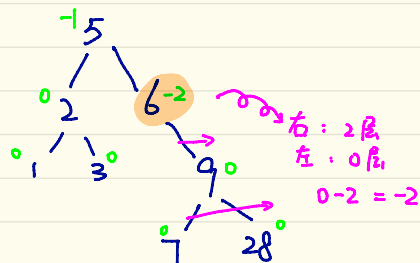
得出 ● 的数值

★ 如果有任何一个 ● 的数值不是 -1 / 0 / 1

则该树不是 balanced tree



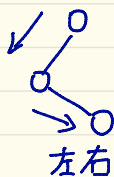
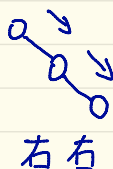
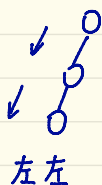
平衡树例子



不平衡树例子

## ② 旋转前：观察树

不平衡的情况只有 4 种

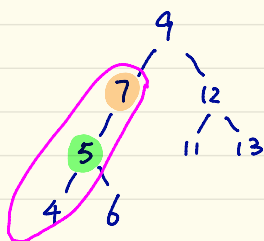


### ③ 旋转方法

情况：方法

口诀：左左：右右，RR  
 右右：左左，LR  
 左右：左右，LRR  
 右左：右左，RLR

RR: Right Rotation



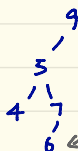
只看最后3个

“左左 → 右右” → RR

① 永远是转 ● 后面的第一个点 ●

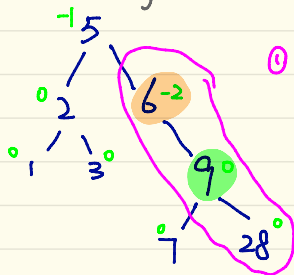
② 下子传上左 ● 的另一个子节点，传给上面当子节点。

③



永远在左，因为不可能比7大。

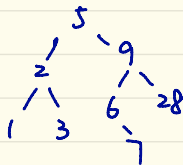
LR: Left Rotation



① 右右左左 → LL

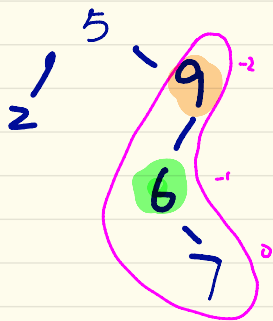
② 下子传上右 (不可能比 ● 小)

③



LRR

Left - Right - Rotation

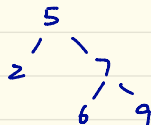


① 左右左右 → LRR

② 转 ● 后的一个点, ● 转成左左的形式



③ 再用RR的右法转



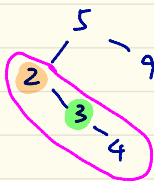
RLR

Right - Left - Rotation



① 右左右左 → RLR

② 转 ● 点.



③ 右右左左

