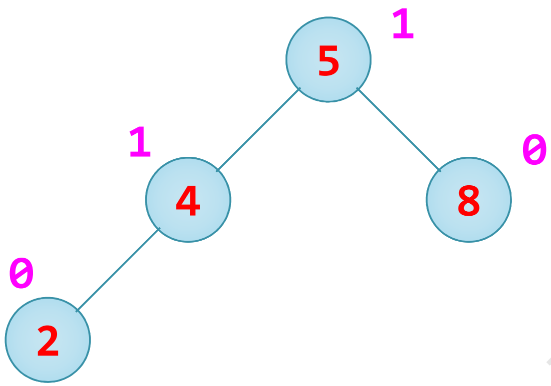
9.2.2 平衡二叉树

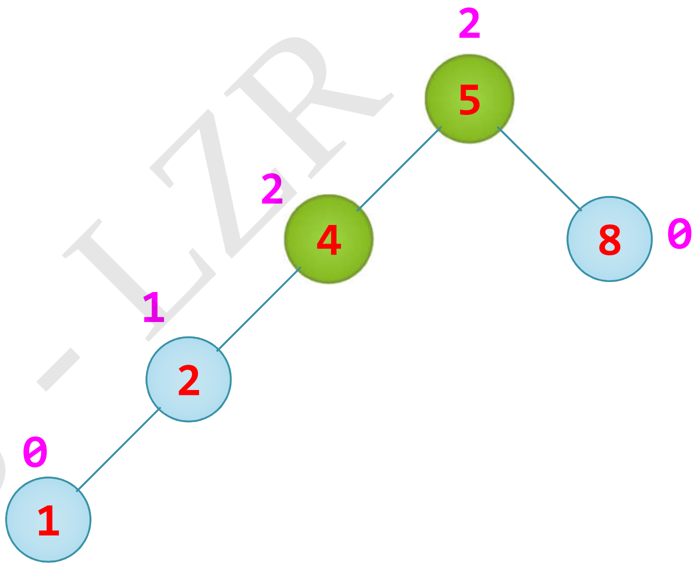
定义：平衡二叉树（AVL）或者是一棵空树；或者是具有如下特性的二叉树：

- （1）二叉排序树中任何一个结点的左子树和右子树高度之差的绝对值不超过1；
- （2）它的左、右子树也分别都是平衡二叉树。

- 二叉树中每个结点设置一个平衡因子bf（balancd factor）域。
- 每个结点的平衡因子是该结点左子树的高度减去右子树的高度。
- 从平衡因子的角度可以说，若一棵二叉树中所有结点的平衡因子的绝对值小于或等于1，即平衡因子取值为1、0或-1，则该二叉树称为平衡二叉树。



平衡二叉树



非平衡二叉树

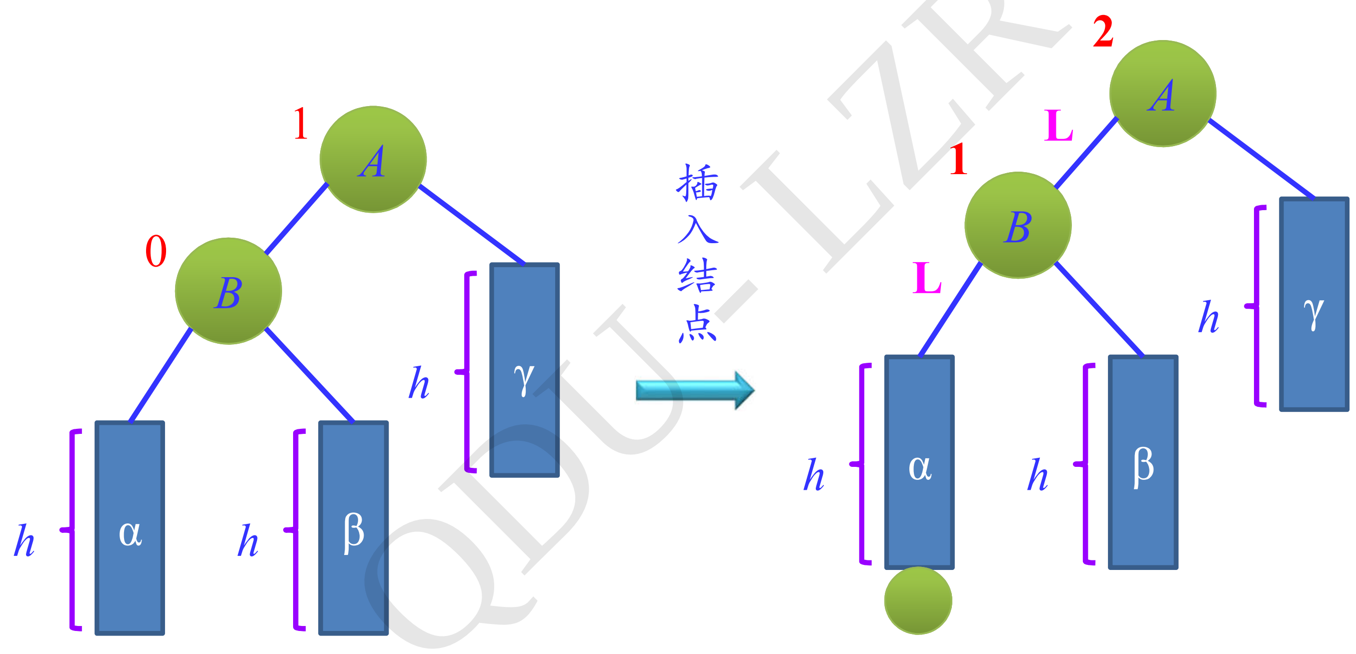
- 我们希望由任何初始序列构成的二叉排序树都是**AVL**树。

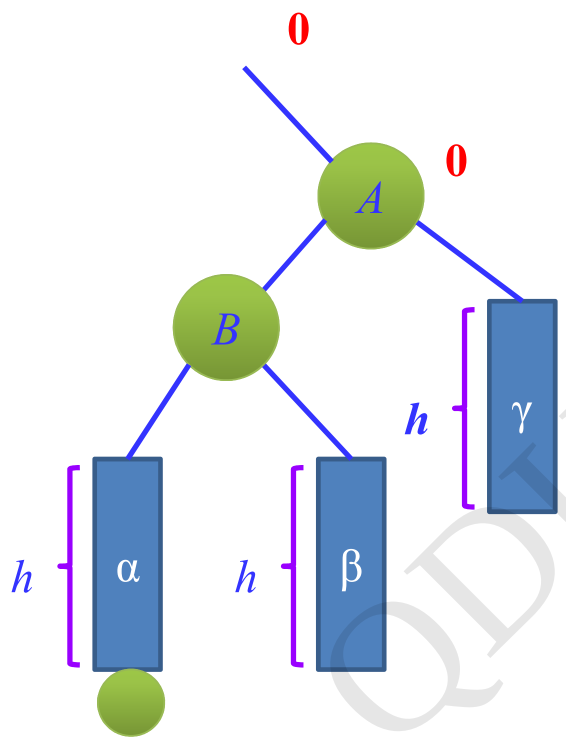
> 因为**AVL**树上任何结点的左右子树的深度之差都不超过**1**，则可以证明它的深度和$\log_2 n$是同数量级的（其中n为结点个数）。由此，它的平均查找长度也和$\log_2 n$同数量级。

平衡化旋转

- 如果在一棵AVL树中插入一个新结点，造成了不平衡。必须调整树的结构，解决方法是调整(旋转)，使之平衡化。
- 平衡化旋转有两类：
  - 单旋转 (LL旋转和RR旋转)
  - 双旋转 (LR旋转和RL旋转)
- 每插入一个新结点时，AVL树中相关结点的平衡状态会发生改变。因此，在插入一个新结点后，需要<u>从插入位置沿通向根的路径回溯</u>，<u>检查各结点的平衡因子</u>。

（1）LL型调整



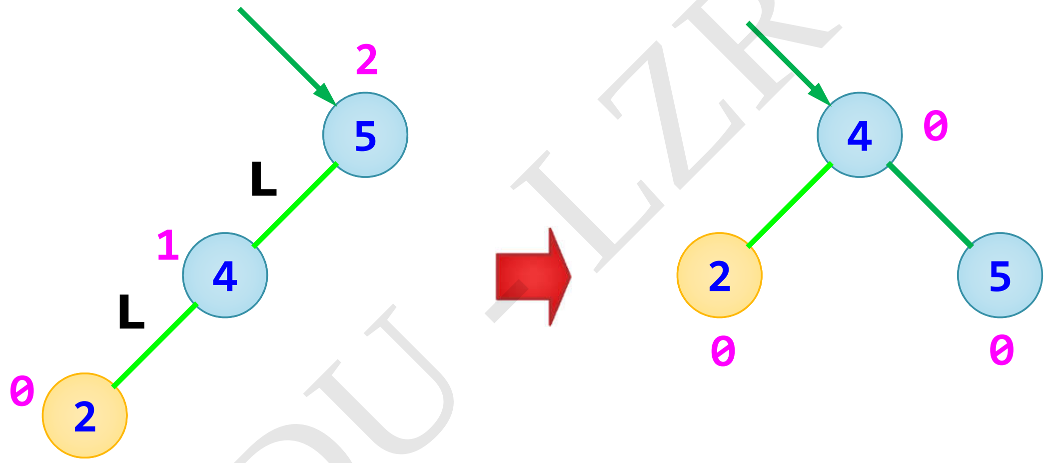


LL调整后的结果

LL型调整过程：

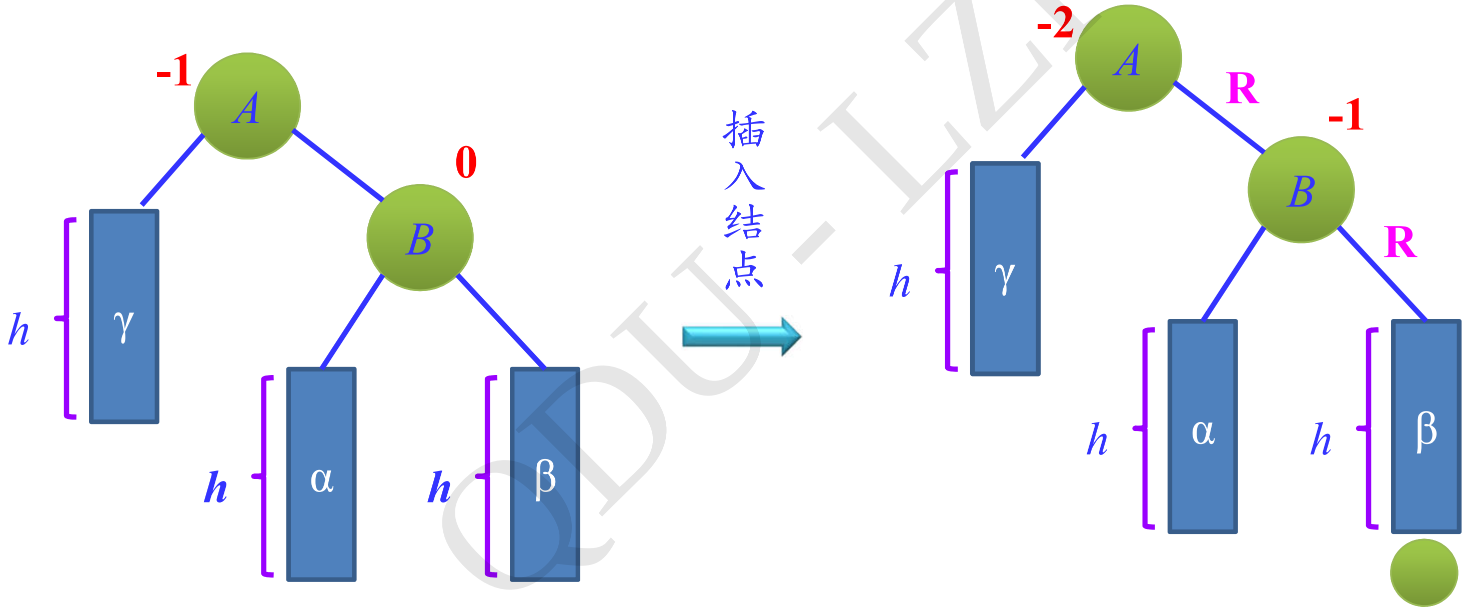
- B结点带左子树α一起上升
- A结点成为B的右孩子
- 原来B结点的右子树β作为A的左子树

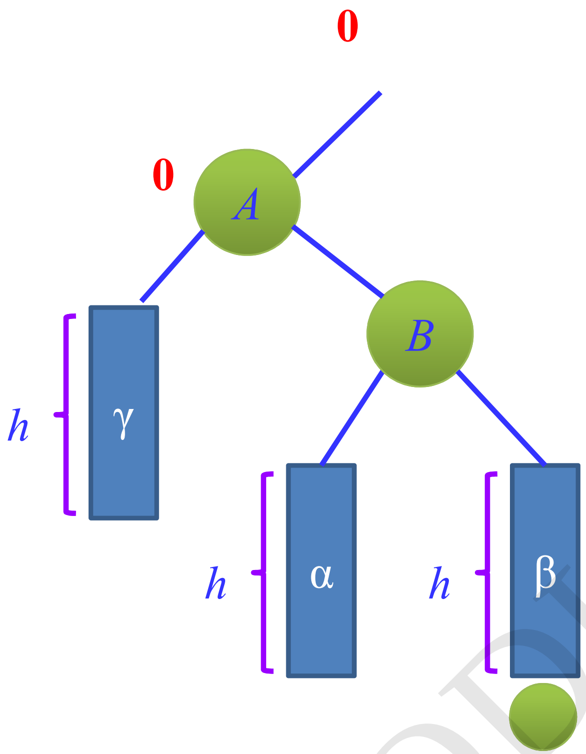
LL调整实例



插入关键字2

（2）RR型调整



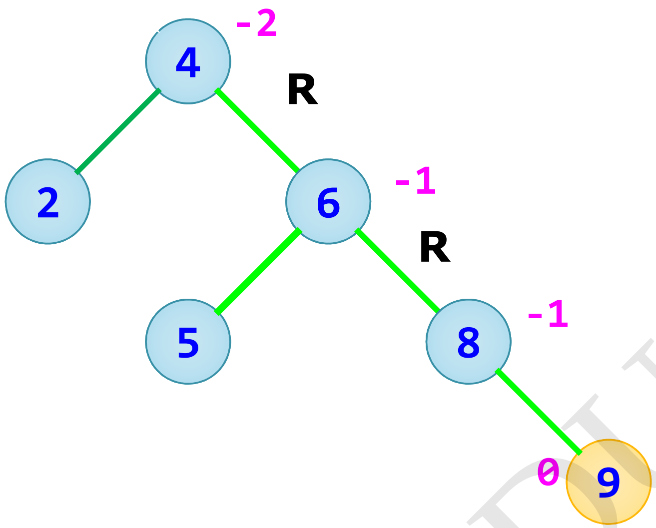


RR调整后的结果

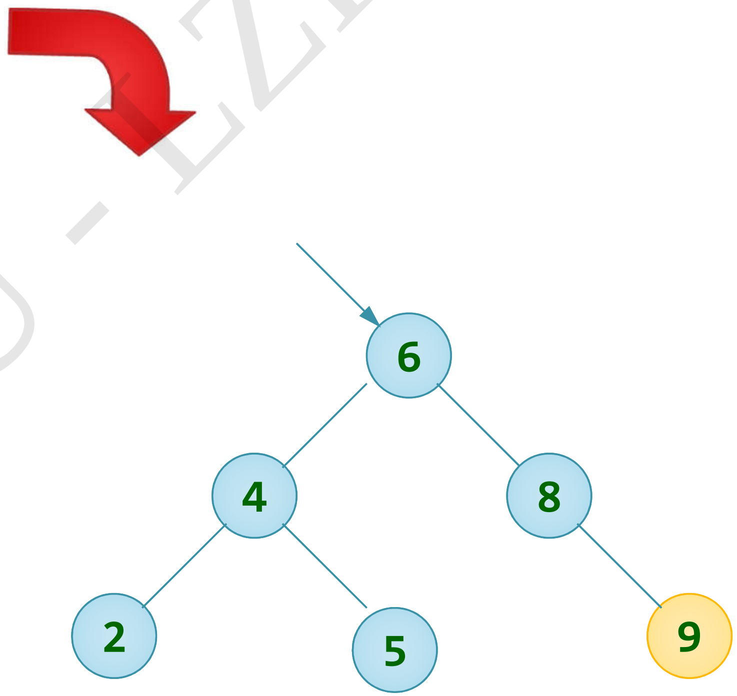
RR型调整过程:

- B结点带右子树β一起上升
- A结点成为B的左孩子
- 原来B结点的左子树α作为A的右子树

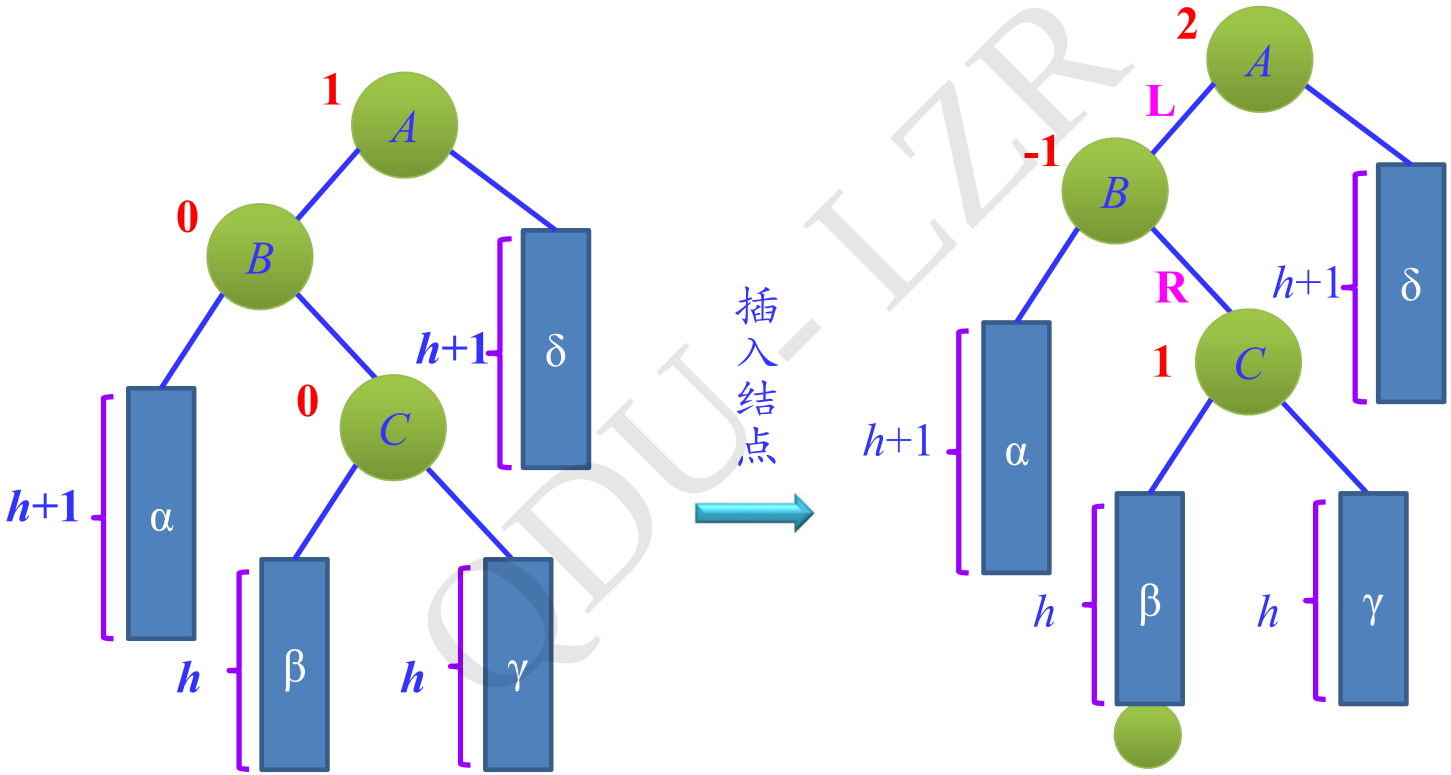
RR调整实例

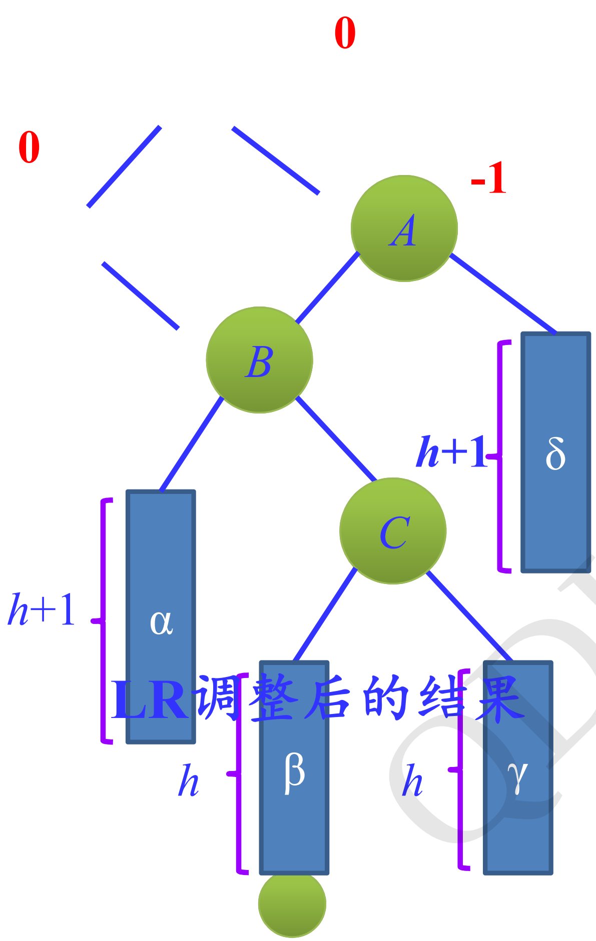


插入关键字 9



（3）LR型调整

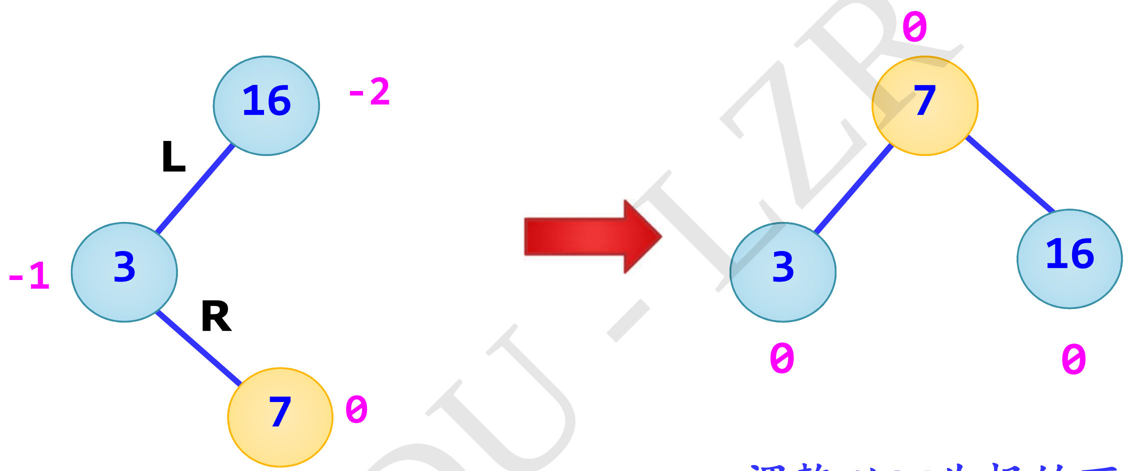




LR型调整过程:

- C结点穿过A、B结点上升
- B结点成为C的左孩子，A结点成为C的右孩子
- 原来C结点的左子树β 作为B的右子树；原来C结点的右子树γ 作为A的左子树

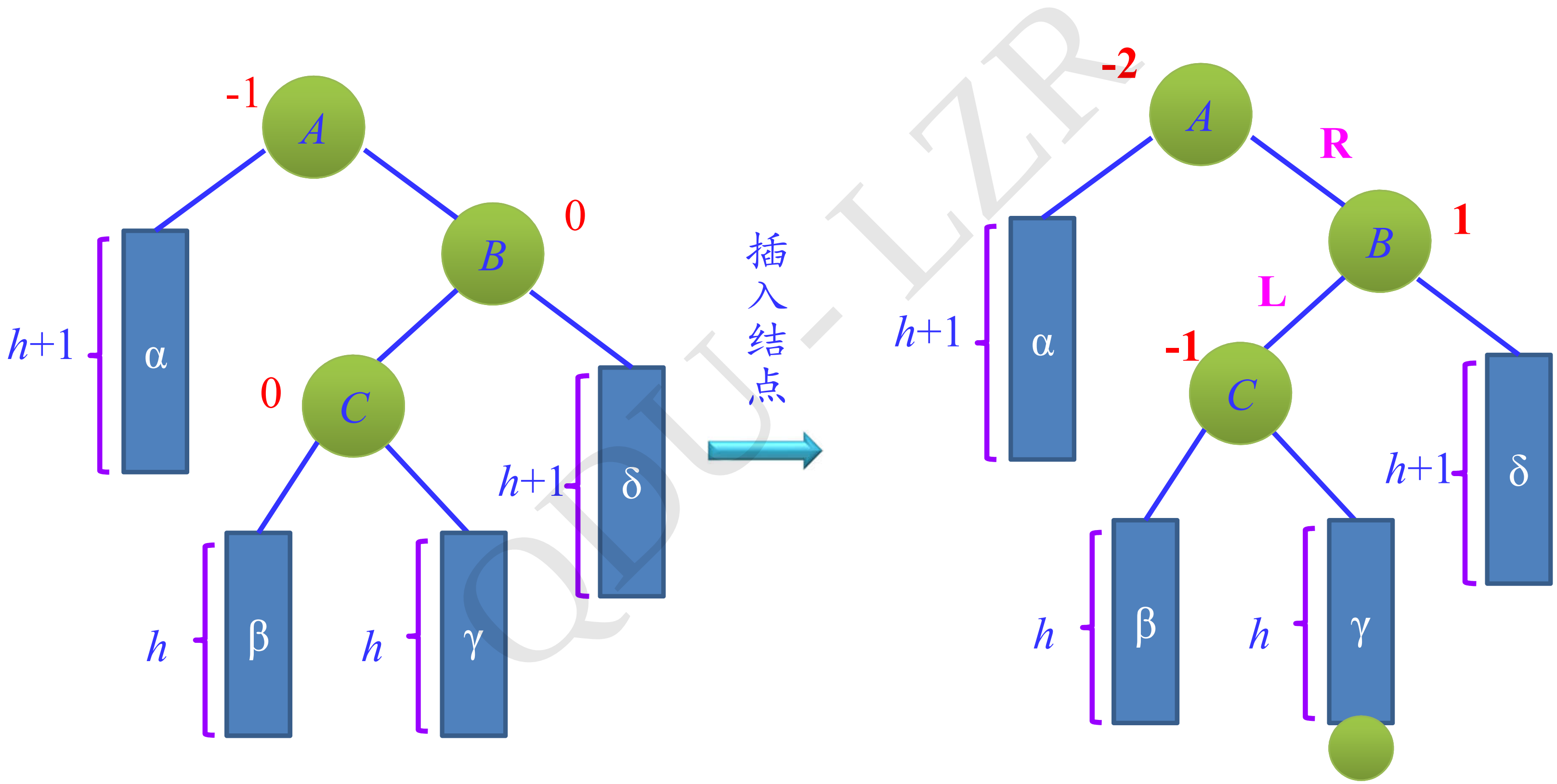
LR调整实例

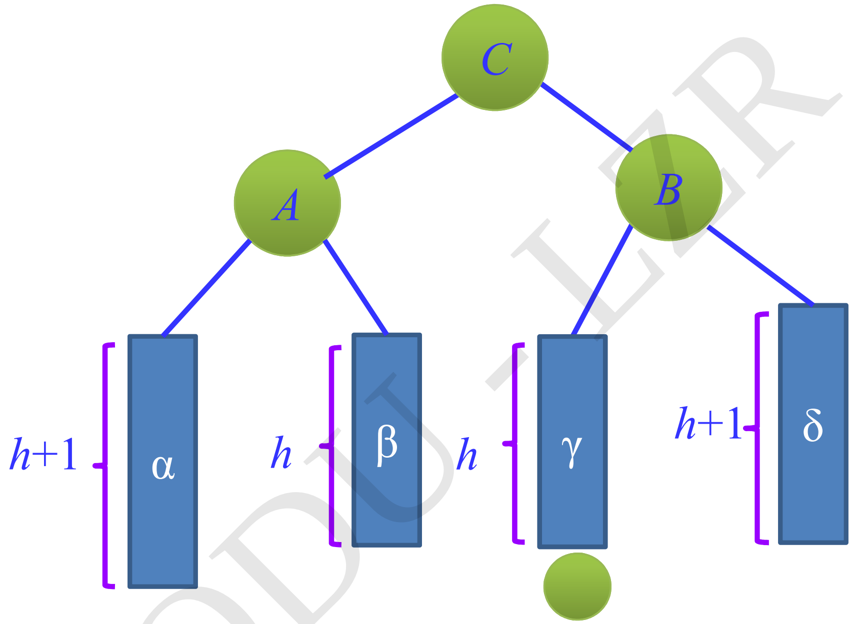


插入7

调整以16为根的不平衡子树

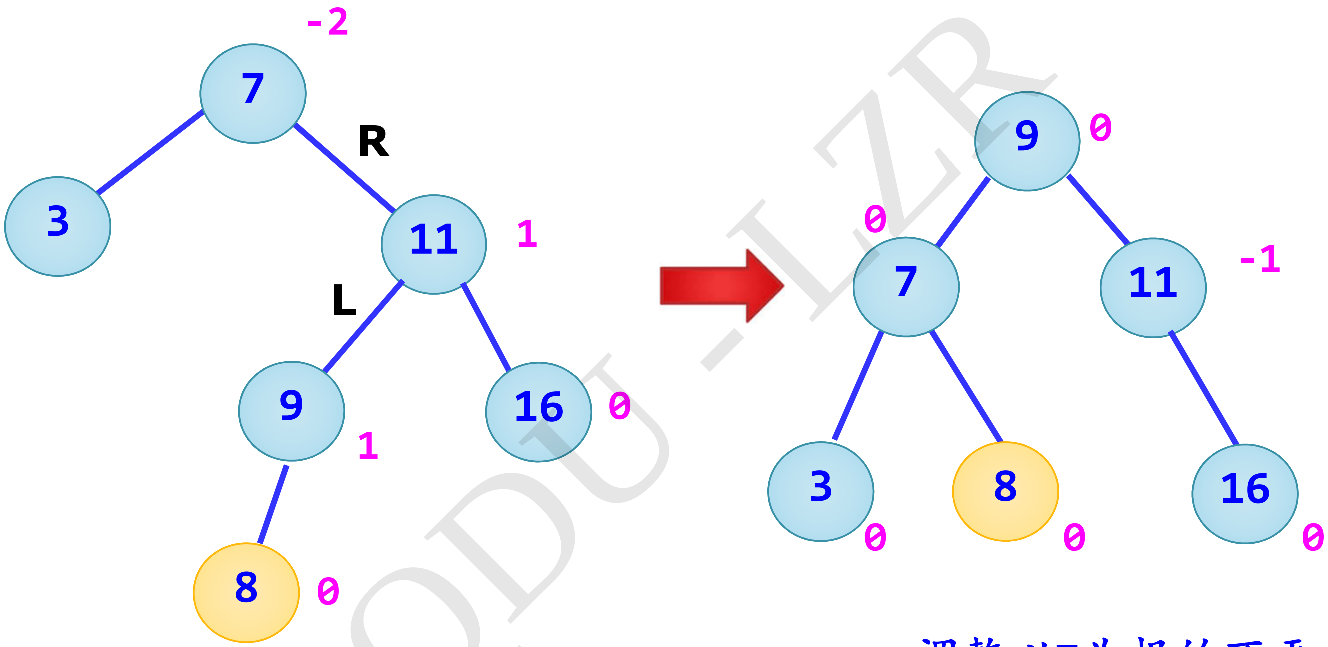
（4）RL型调整





C
A
B
h+1
α
h
β
h
γ
h+1
δ

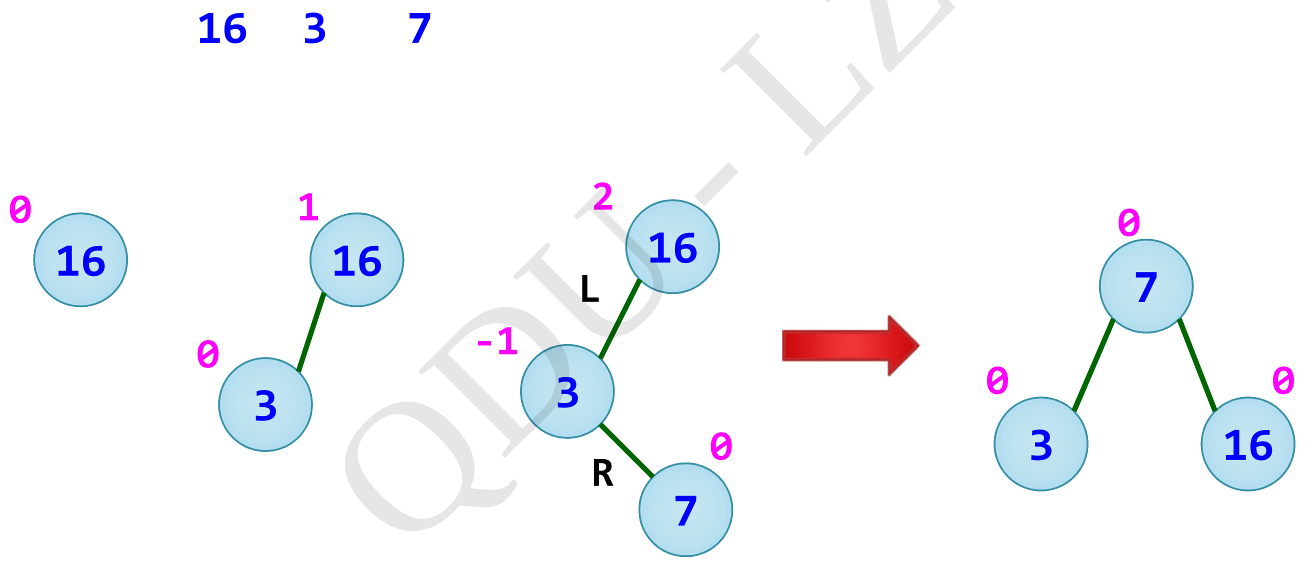
RL调整实例



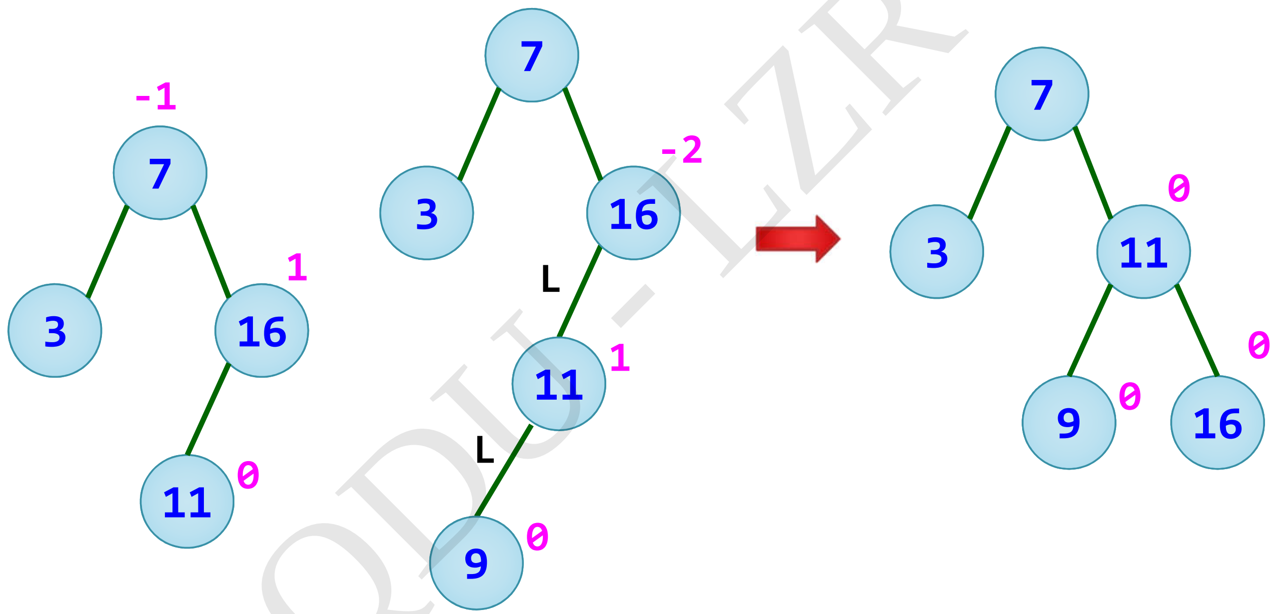
插入8

调整以7为根的不平衡子树

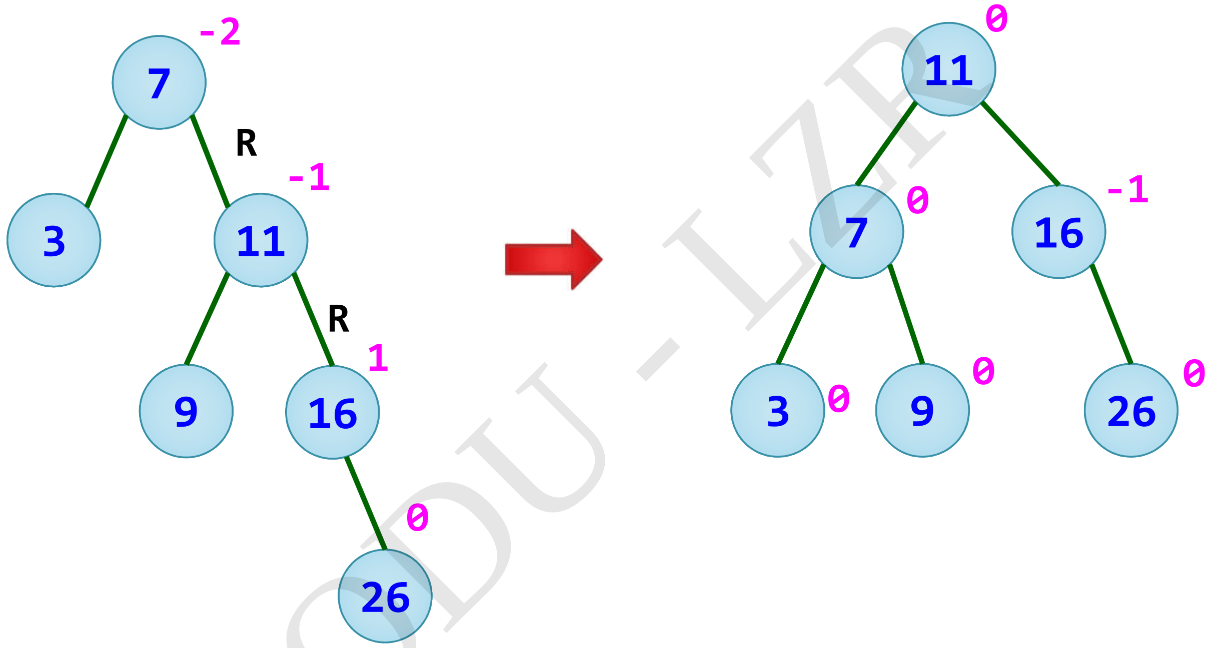
【示例-1】输入关键字序列（16，3，7，11，9，26，18，14，15），给出构造一棵平衡二叉排序树的步骤。



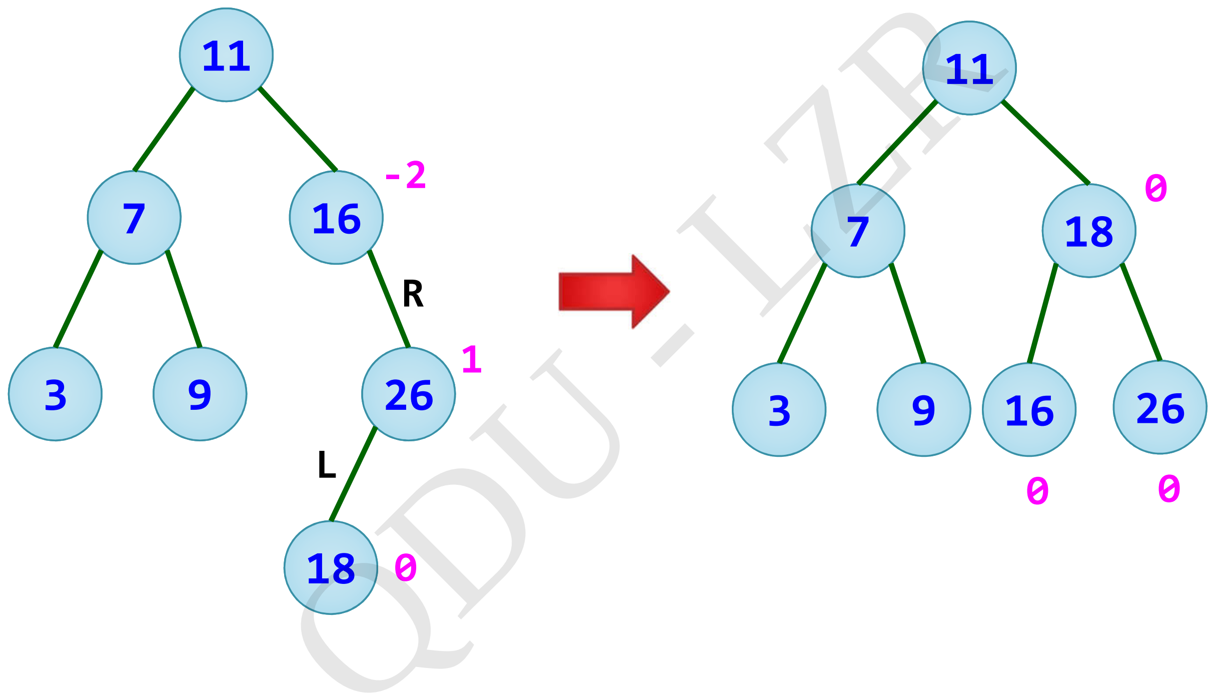
11 9

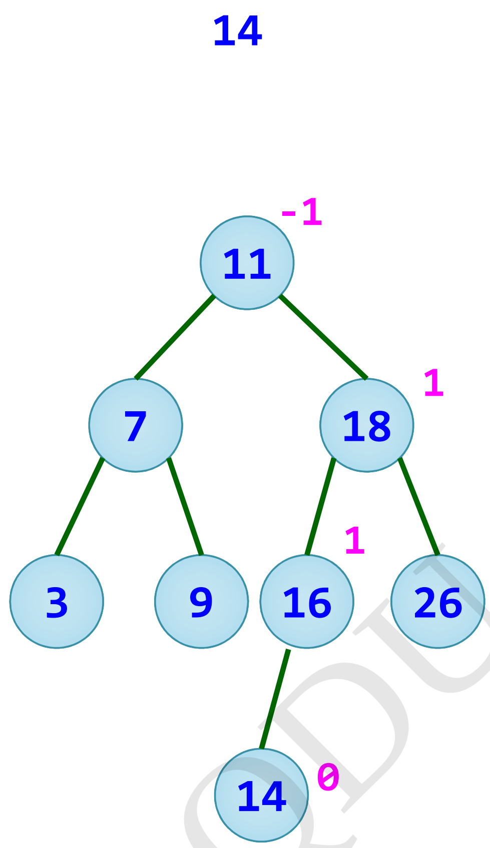


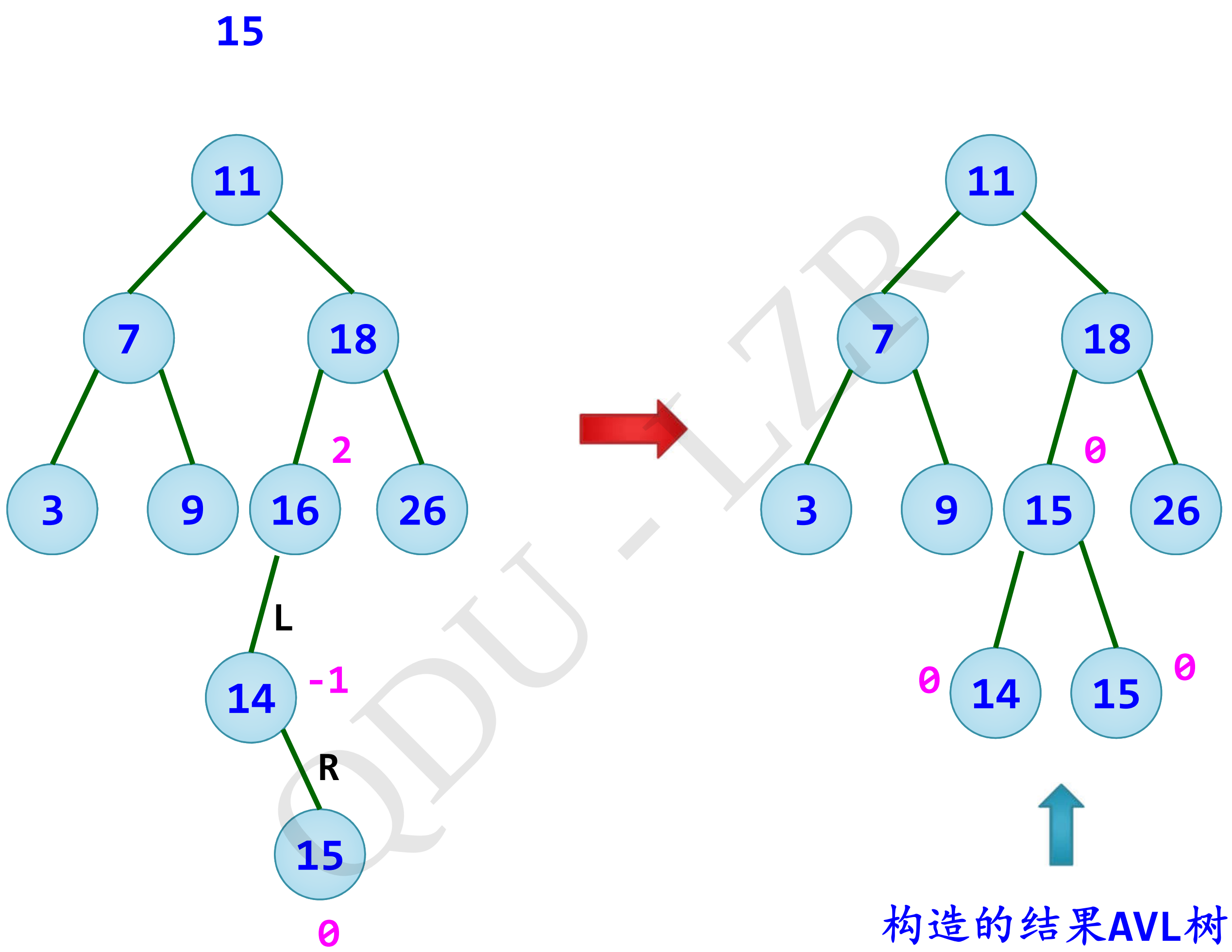
26



18

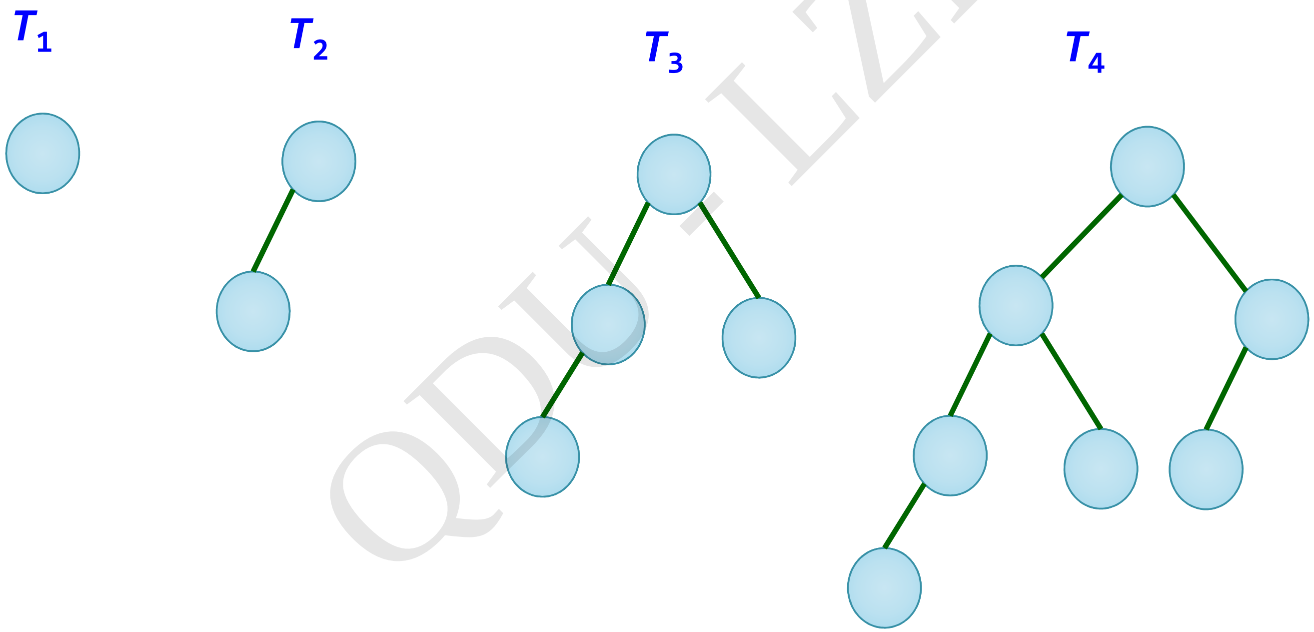



14
-1
11
7
18
1
3
9
16
1
26
14
0


15
11
7
18
3
9
16
26
2
L
14
-1
R
15
0
11
7
18
3
9
15
26
0
14
15
0
0
构造的结果AVL树

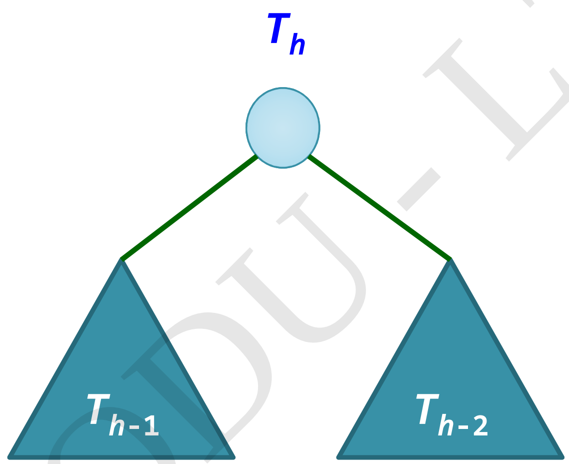
- 在平衡二叉树上进行查找的过程和在二叉排序树上进行查找的过程完全相同。
- 在最坏的情况下，普通二叉排序树的查找长度为O(*n*)。那么，平衡二叉树的情况又是怎样的呢？下面分析平衡二叉树的高度*h*和结点个数*n*之间的关系。

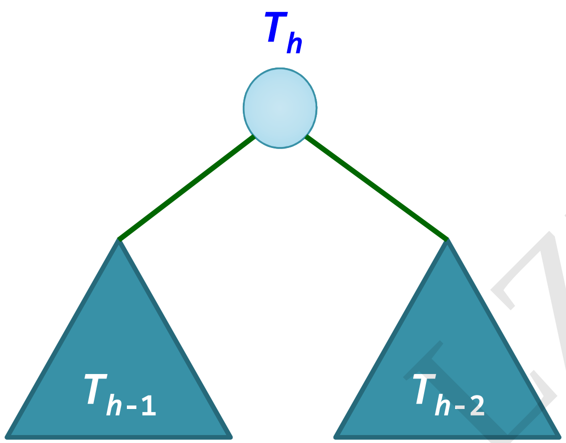
构造一系列的平衡二叉树T_1，T_2，T_3，…，其中，T_h（h=1，2，3，…）是高度为h且结点数尽可能少的平衡二叉树，如下图所示的T_1，T_2，T_3和T_4。



结点个数n最少的平衡二叉树

■ 构造T_h，先分别构造T_{h-1}和T_{h-2}，使T_h以T_{h-1}和T_{h-2}作为其根结点的左、右子树。





通过计算上述平衡二叉树中的结点个数，来建立高度与结点个数之间的关系。设$N(h)$为T_h的结点数，从图中可以看出有下列关系成立：

$N(1)=1$，$N(2)=2$，$N(h)=N(h-1)+N(h-2)+1$

当$h>1$时，此关系类似于定义Fibonacci数的关系：

$$F(1)=1,\ F(2)=1,\ F(h)=F(h-1)+F(h-2)$$

通过检查两个序列的前几项就可发现两者之间的对应关系：

$$N(h)=F(h+2)-1$$

由于Fibonacci数满足渐近公式：$F(h)=\frac{1}{\sqrt{5}}\varphi^{h}$

其中，$\varphi=\frac{1+\sqrt{5}}{2}$

故由此可得近似公式：$N(h)=\frac{1}{\sqrt{5}}\varphi^{h+2}-1\approx 2^{h}-1$

即：$h\approx\log_2(N(h)+1)$。

结论：含有n个结点的平衡二叉树的平均查找长度为$O(\log_2 n)$。

—— END ——