

平衡二叉查找树（2）

胡船长

初航，我带你；远航，靠自己。

1、顺序表与链表 (3)	6、平衡二叉查找树 (4)
2、栈与队列 (3)	7、堆与优先队列 (2)
3、树与二叉树 (2)	8、森林和并查集 (2)
4、图的存储与遍历 (3)	9、图论算法入门 (2)
5、排序与查找 (4)	10、字符串匹配算法 (2)

第一重点
第二重点
第三重点
其他

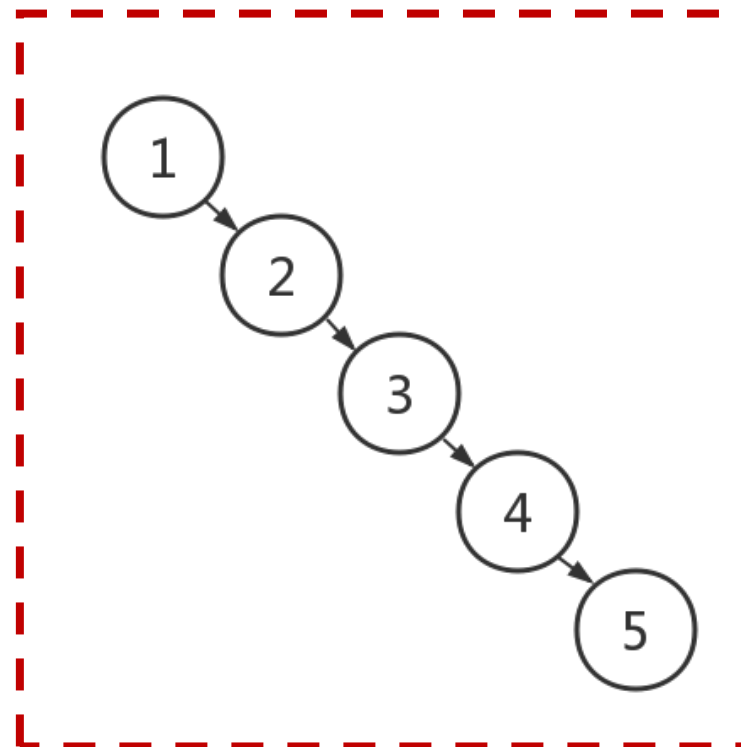
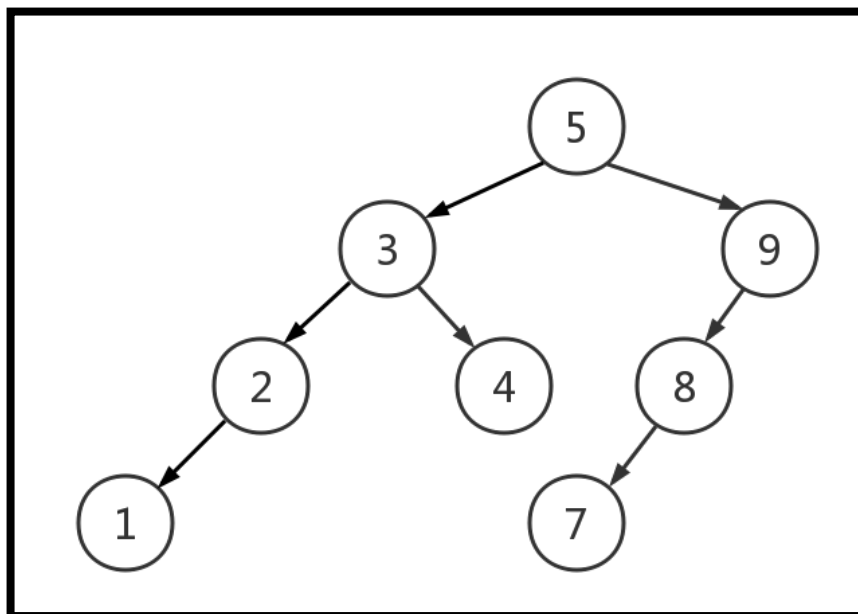
【回顾】随堂练习-1

请按照如下顺序插入数字，画出对应的二叉搜索树

1 : [5 9 8 3 2 4 1 7]

2 : [1 2 3 4 5]

【回顾】随堂练习-1



AVL 树

名称：AVL 树

发明者：

G.M. **A**delson-**V**elsky

E.M. **L**andis

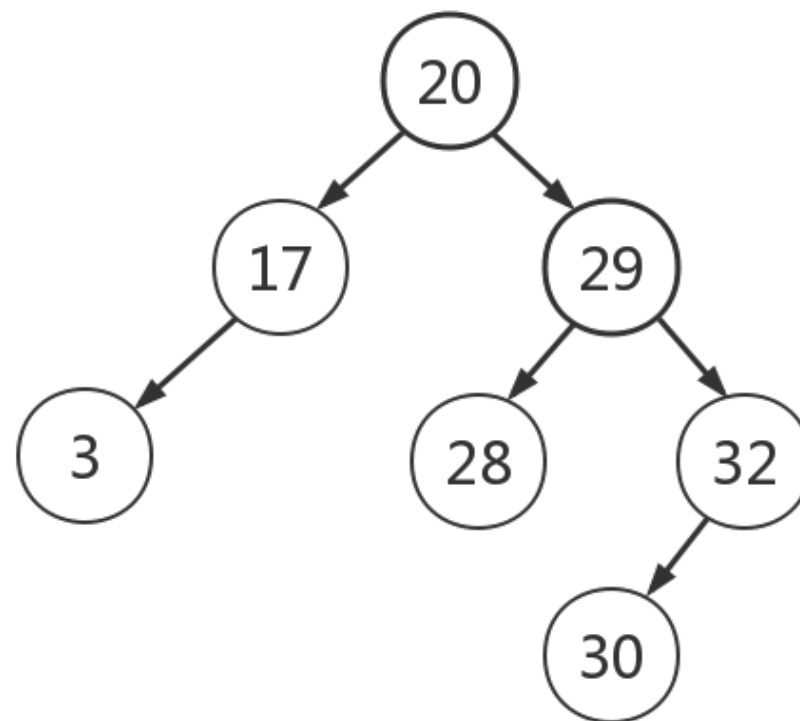
年代：1962 年 (55 岁)

性质：

$$| H(\text{left}) - H(\text{right}) | \leq 1$$

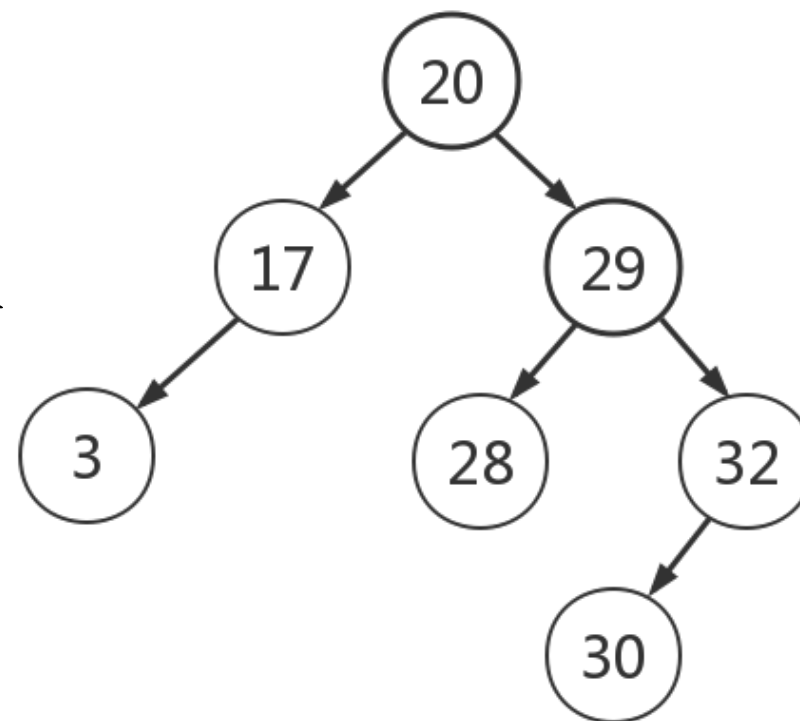
优点：

由于对每个节点的左右子树的树高做了限制，所以整棵树不会退化成一个链表



思考：

- 1、高度为 H 的 BS 树，所包含的节点数量在什么范围之内？
- 2、高度为 H 的 AVL 树，所包含的节点数量在什么范围之内？

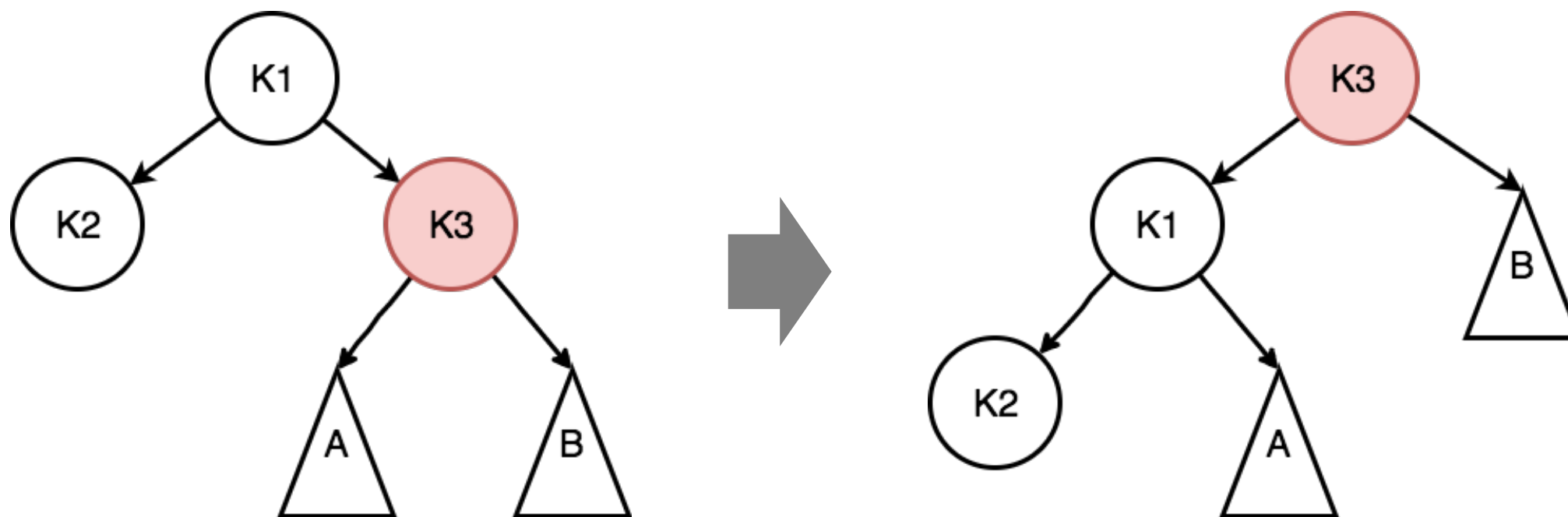


思考：

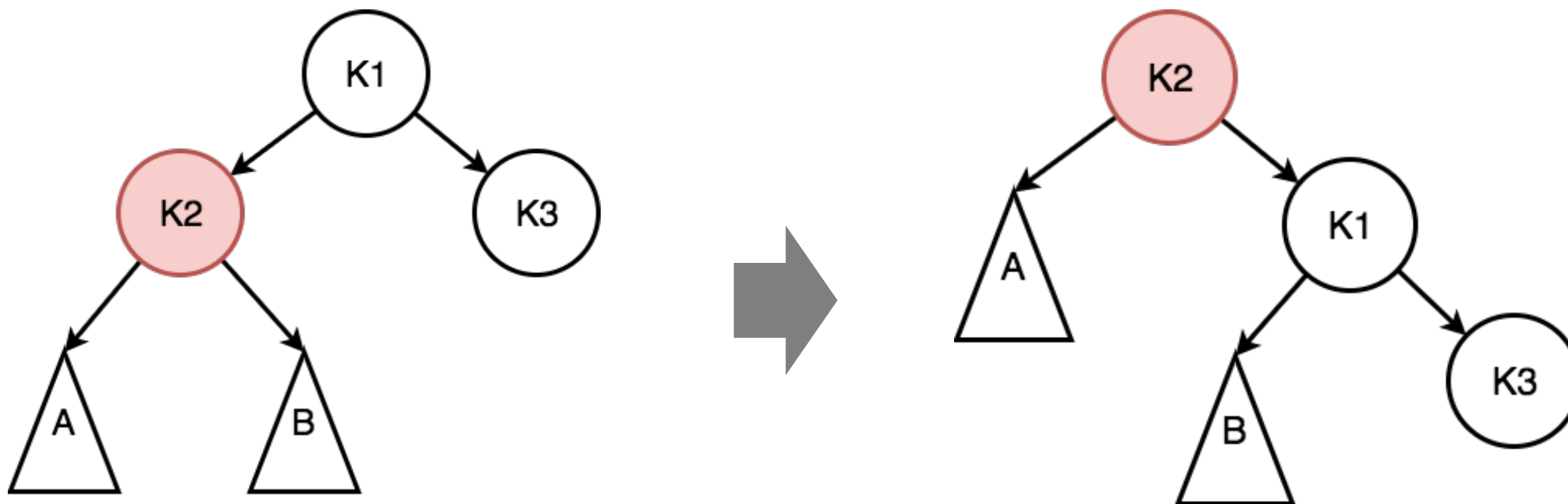
$$1、H \leq \text{SIZE}(H) \leq 2^H - 1$$

$$2、\text{low}(H - 2) + \text{low}(H - 1) + 1 \leq \text{SIZE}(H) \leq 2^H - 1$$
$$\text{low}(1) = 1, \text{low}(2) = 2, \text{low}(3) = 4, \text{low}(4) = 7 \dots$$

AVL 树-左旋

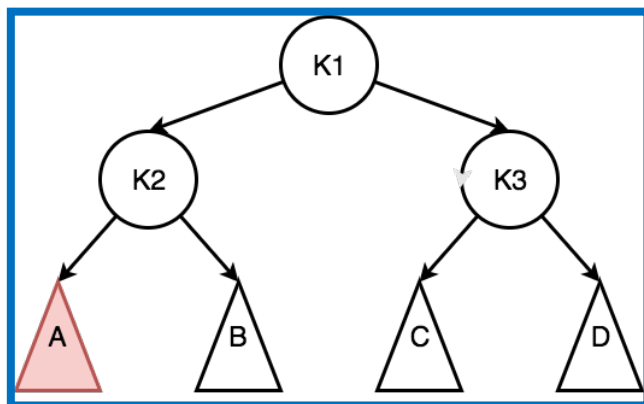


AVL 树-右旋

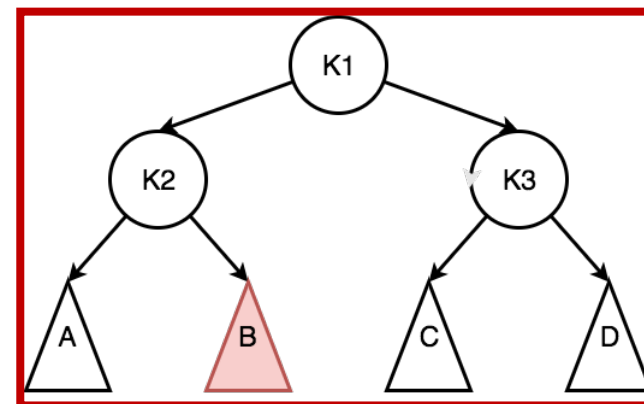


AVL 树-失衡类型

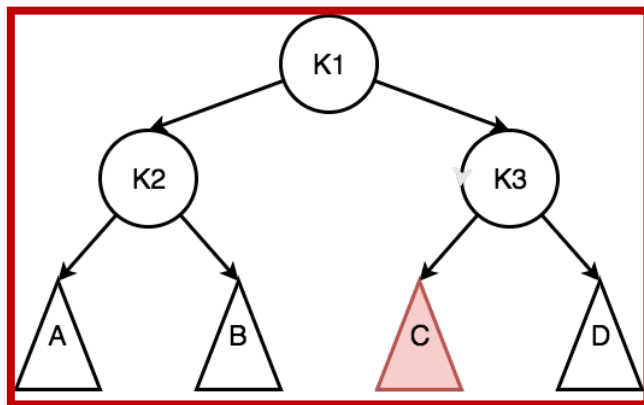
LL 型



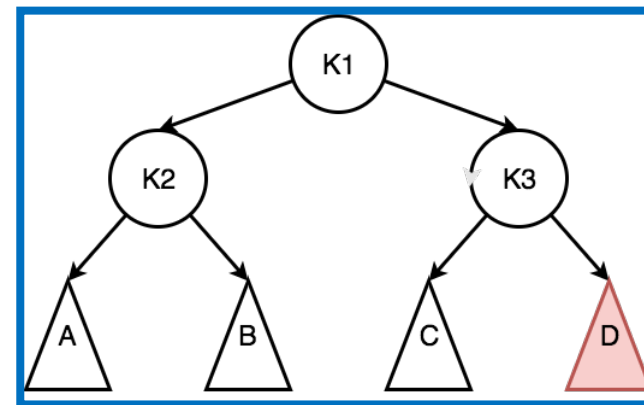
LR 型



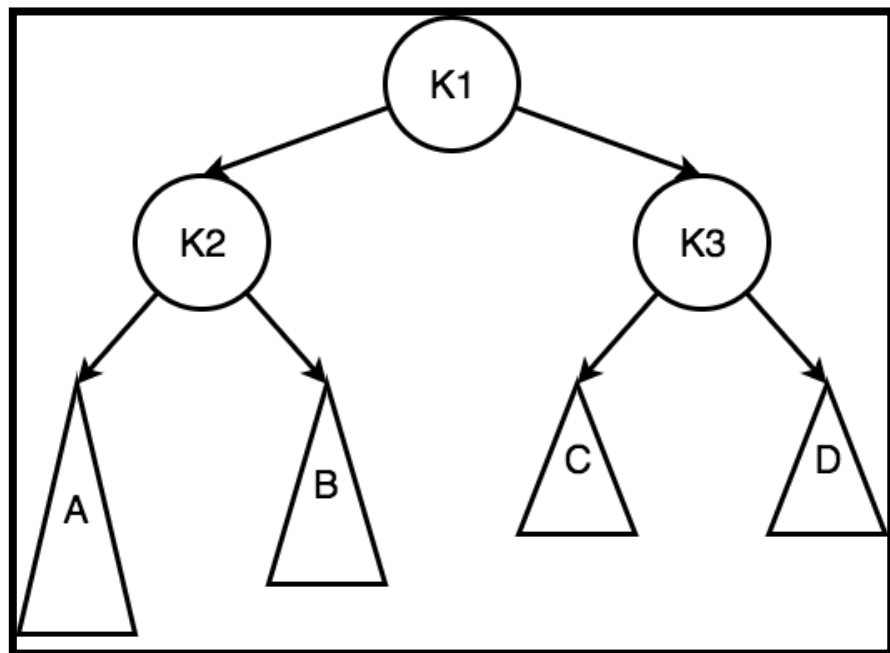
RL 型



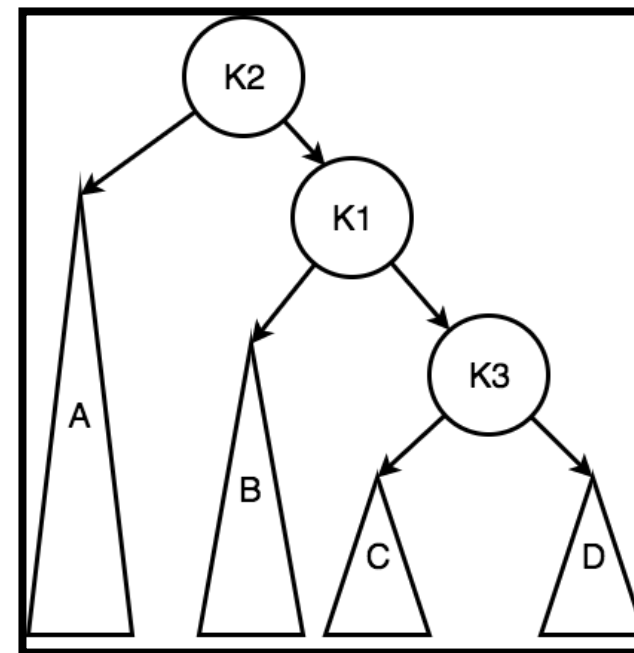
RR 型



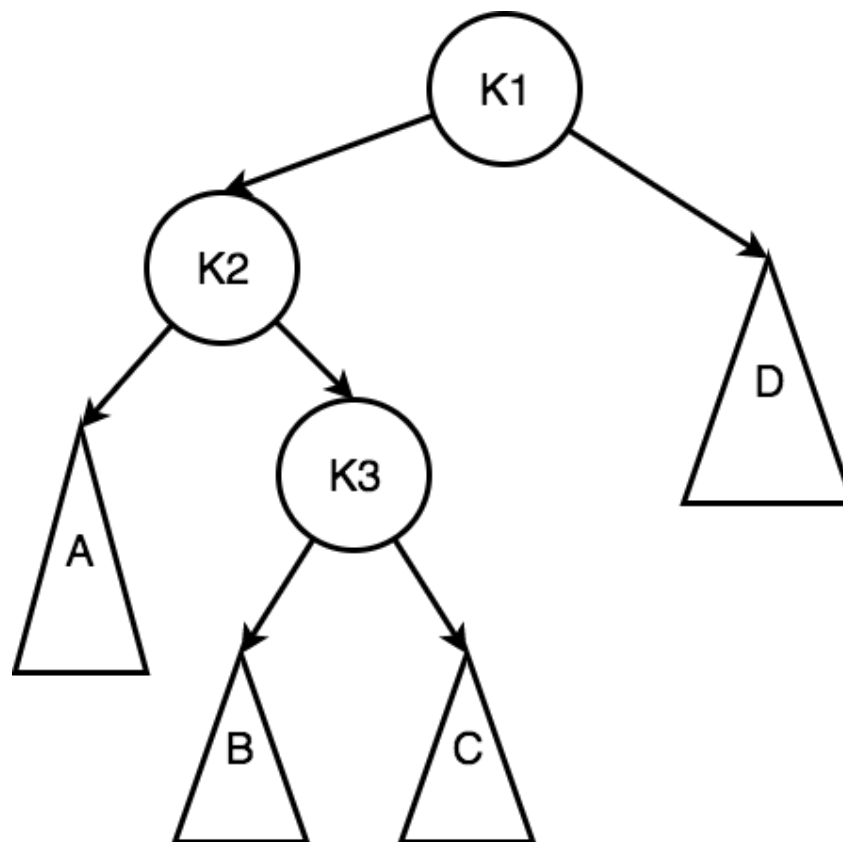
AVL 树-LL 型



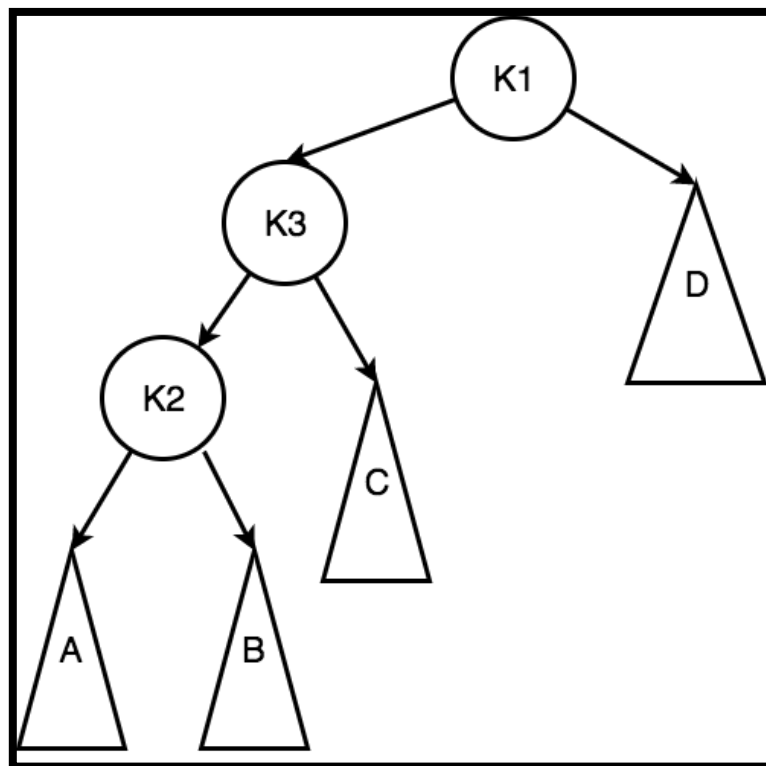
右旋



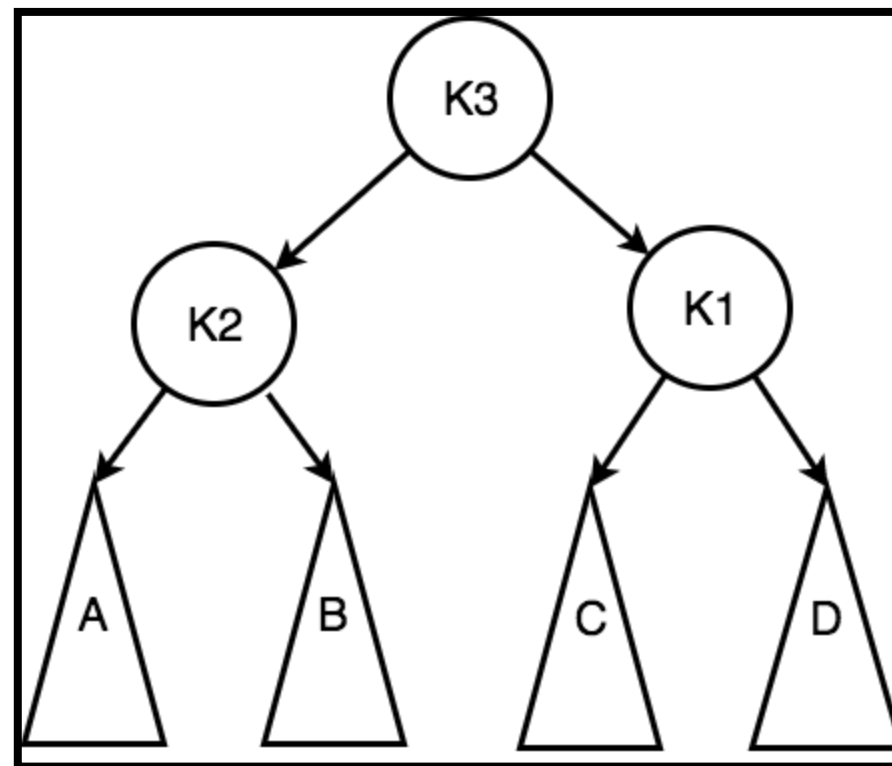
AVL 树-LR 型



AVL 树-LR 型-先左旋



右旋



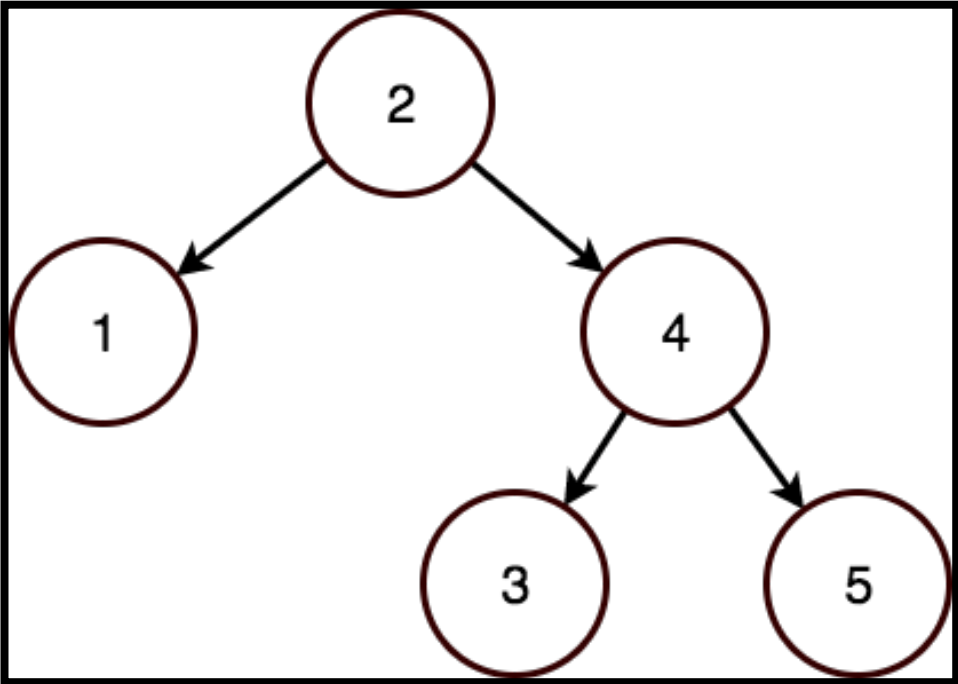
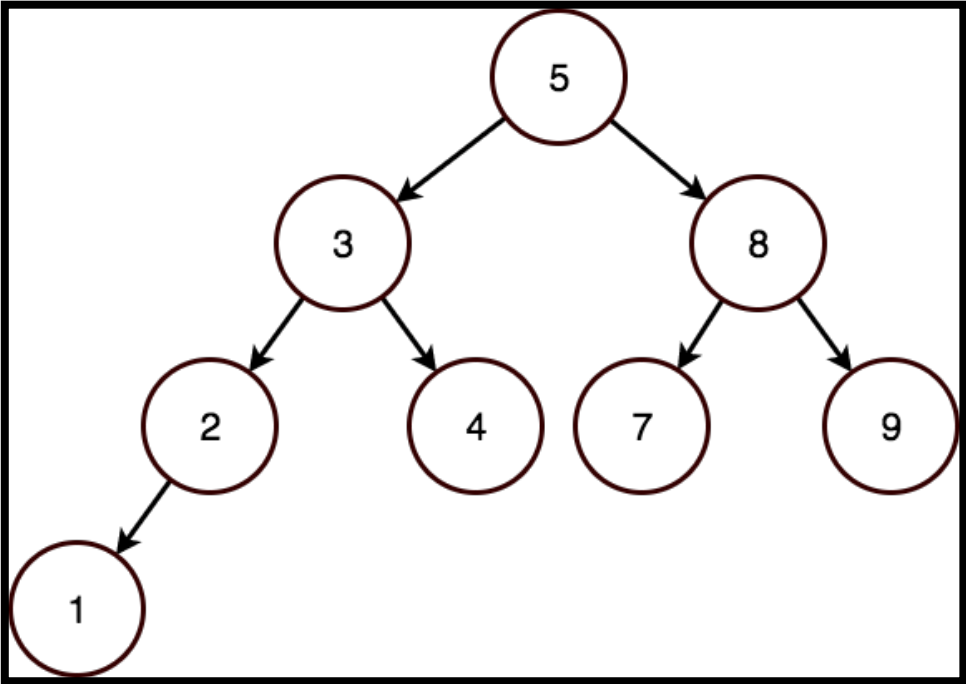
随堂练习-2

请按照如下顺序插入数字，画出对应的 AVL 树

1 : [5 9 8 3 2 4 1 7]

2 : [1 2 3 4 5]

随堂练习-2



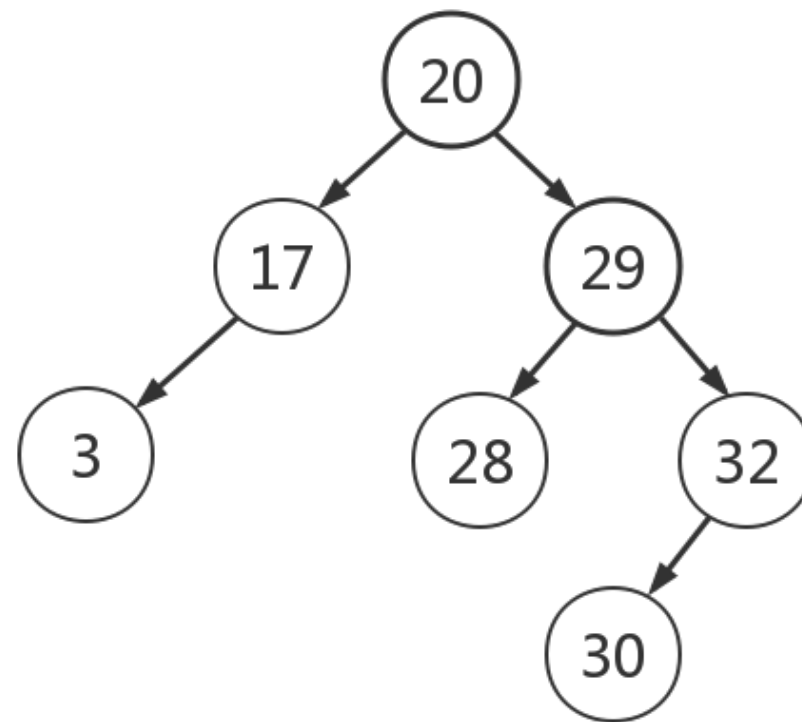
名称: Size Balanced 树

性质:

```
Size(left) >= Size(right->left)
Size(left) >= Size(right->right)
Size(right) >= Size(left->left)
Size(right) >= Size(left->right)
```

优点:

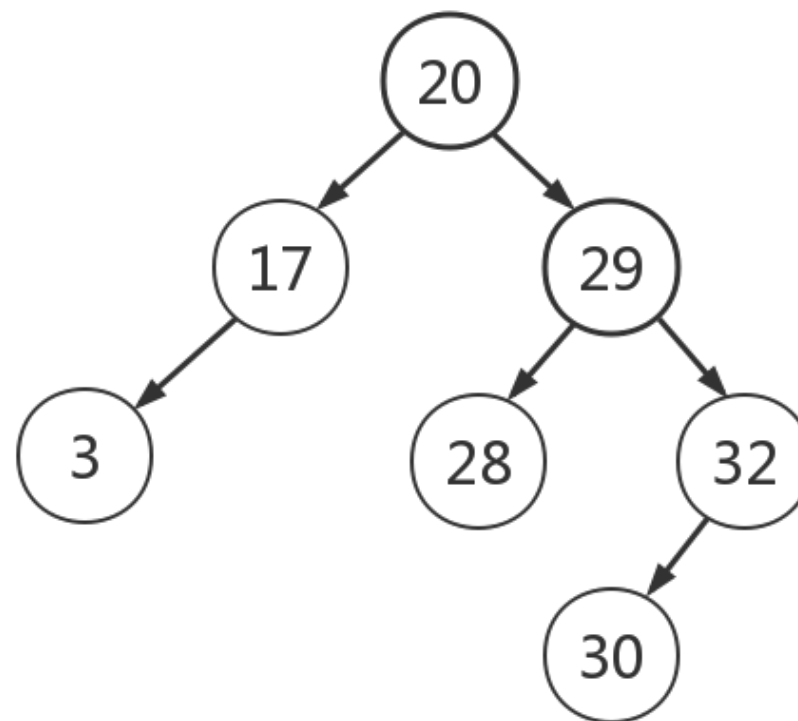
由于对每个节点的左右子树的节点数量做了限制，所以整棵树不会退化成链表



思考：

1、高度为 H 的 BS 树，所包含的节点数量在什么范围之内？

2、高度为 H 的 SB 树，所包含的节点数量在什么范围之内？



请自行推导 SB Tree 的插入以及删除算法

课后练习题

Leetcode 110

Leetcode 669