

二叉树

@M了个J

<https://github.com/CoderMJLee>

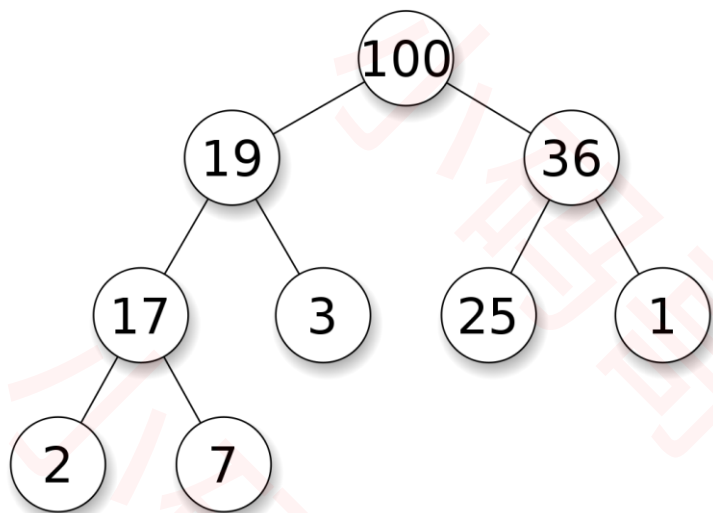
<http://cnblogs.com/mjios>

码拉松

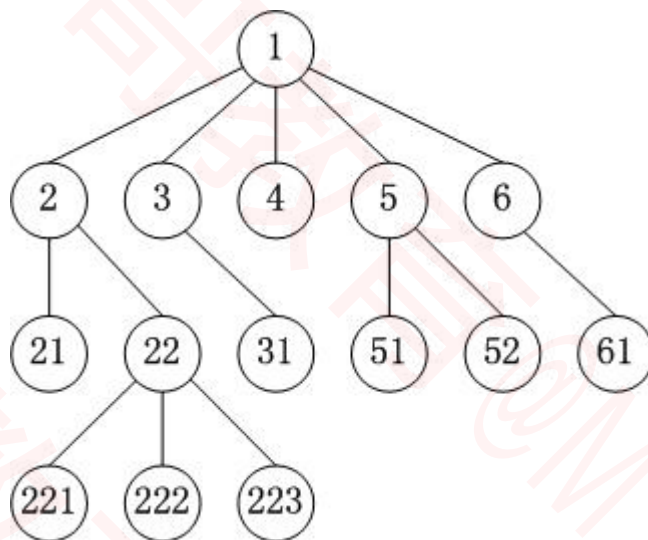


实力IT教育 www.520it.com

树形结构



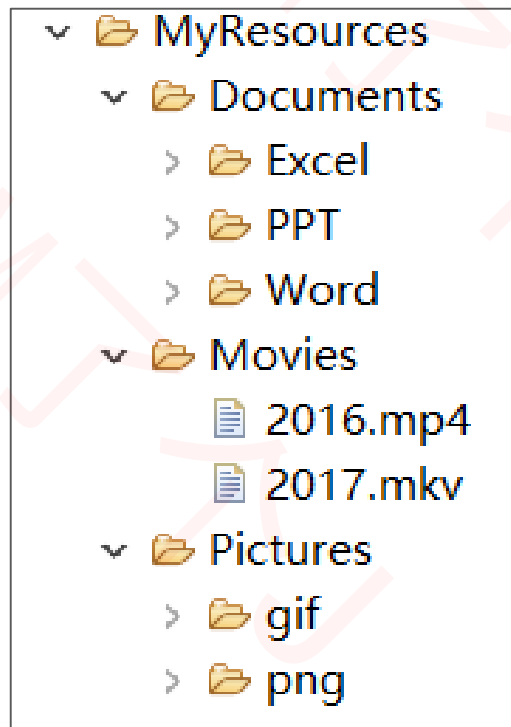
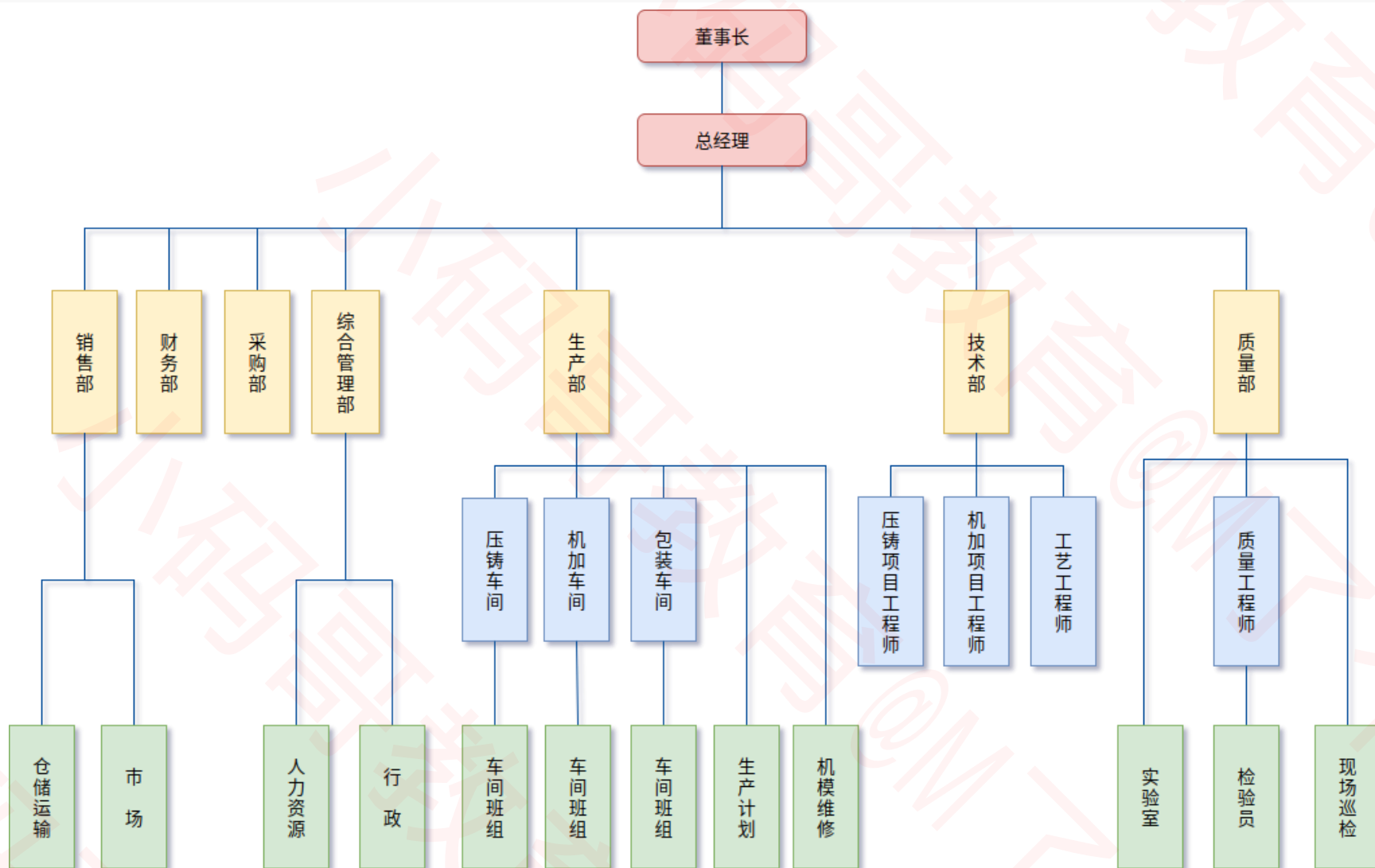
二叉树



多叉树



生活中的树形结构

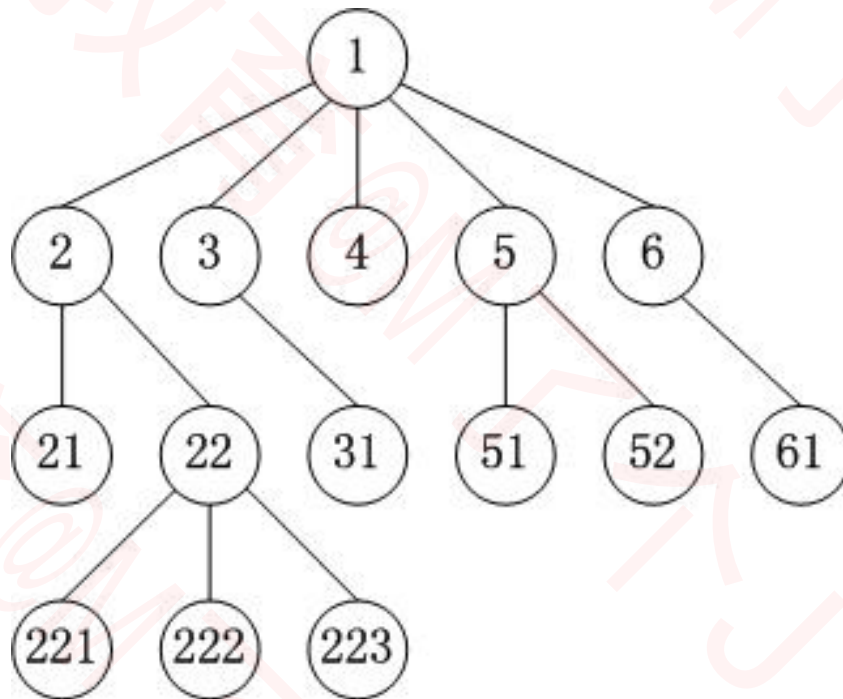


- 使用树形结构可以大大提高效率
- 树形结构是算法面试的重点

树 (Tree) 的基本概念

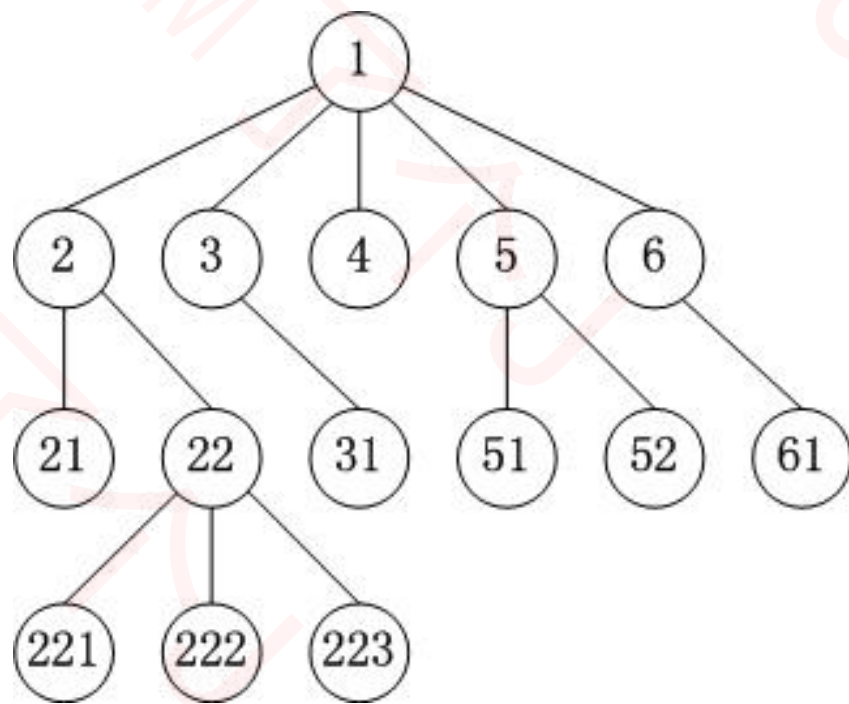
- 节点、根节点、父节点、子节点、兄弟节点
- 一棵树可以没有任何节点，称为**空树**
- 一棵树可以只有1个节点，也就是只有根节点
- 子树、左子树、右子树

- 节点的**度** (degree) : 子树的个数
- 树的**度**: 所有节点度中的最大值
- **叶子节点** (leaf) : 度为0的节点
- **非叶子节点**: 度不为0的节点



树 (Tree) 的基本概念

- **层数** (level) : 根节点在第1层, 根节点的子节点在第2层, 以此类推 (有些教程也从第0层开始计算)
- 节点的**深度** (depth) : 从根节点到当前节点的唯一路径上的节点总数
- 节点的**高度** (height) : 从当前节点到最远叶子节点的路径上的节点总数
- 树的**深度**: 所有节点深度中的最大值
- 树的**高度**: 所有节点高度中的最大值
- 树的**深度** 等于 树的**高度**



有序树、无序树、森林

■ 有序树

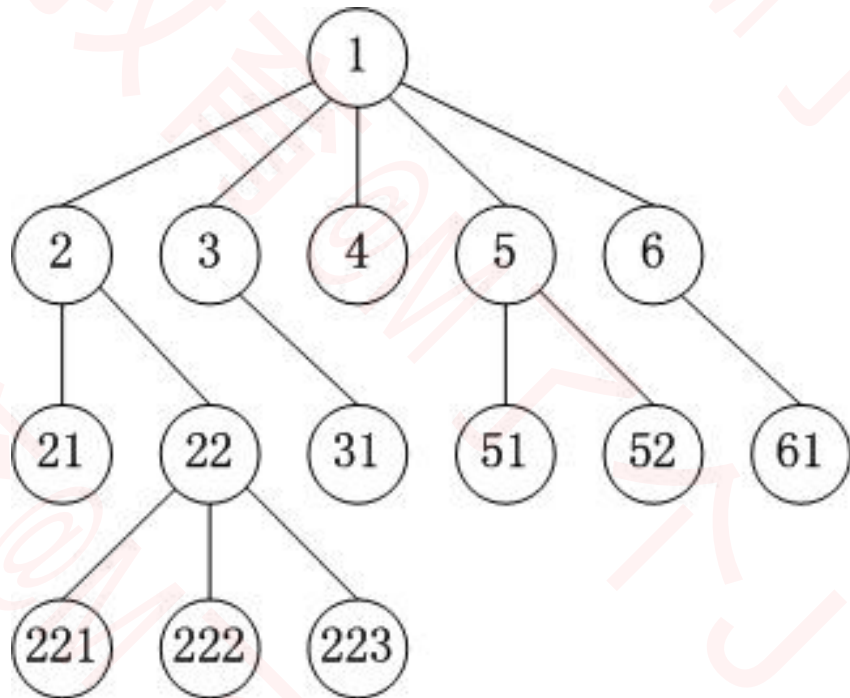
- 树中任意节点的子节点之间有顺序关系

■ 无序树

- 树中任意节点的子节点之间没有顺序关系
- 也称为“自由树”

■ 森林

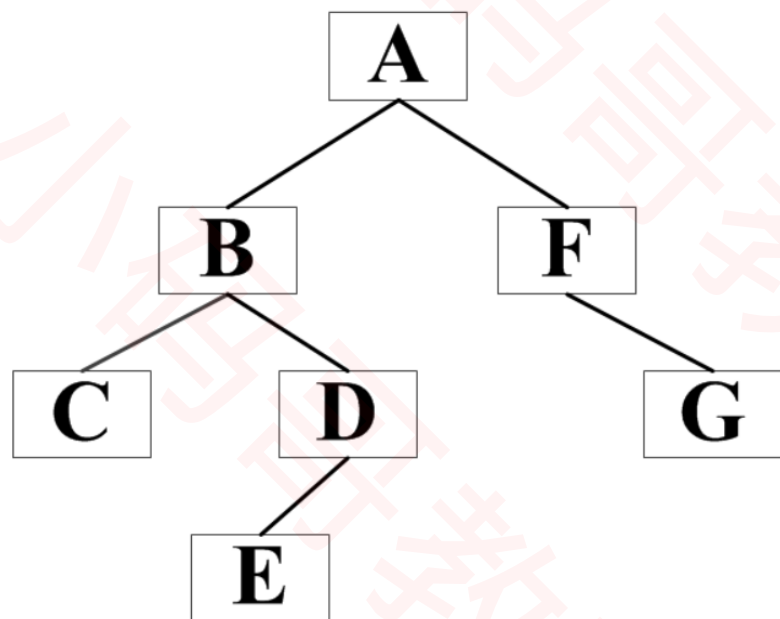
- 由 m ($m \geq 0$) 棵互不相交的树组成的集合



二叉树 (Binary Tree)

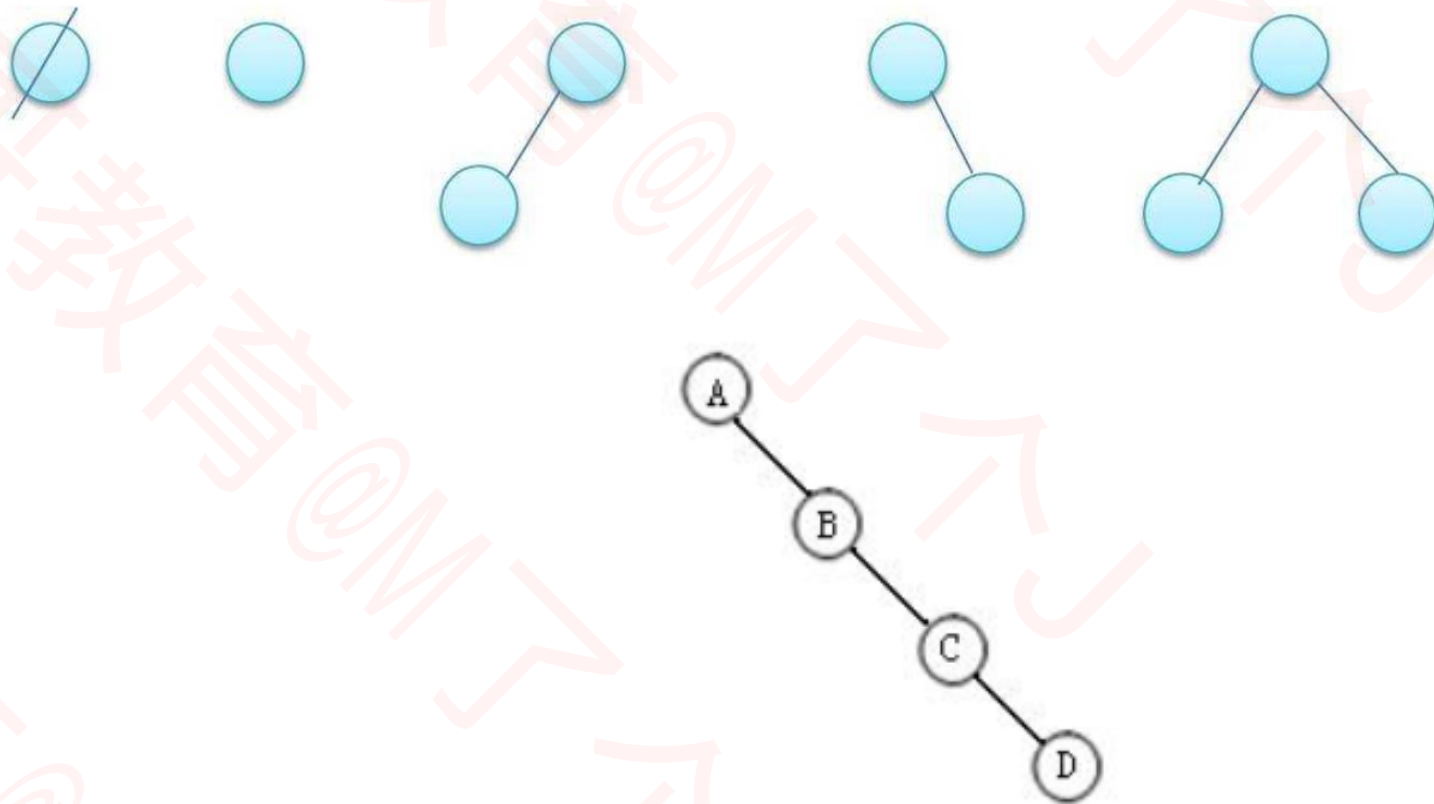
■ 二叉树的特点

- 每个节点的度最大为2 (最多拥有2棵子树)
- 左子树和右子树是有顺序的
- 即使某节点只有一棵子树, 也要区分左右子树



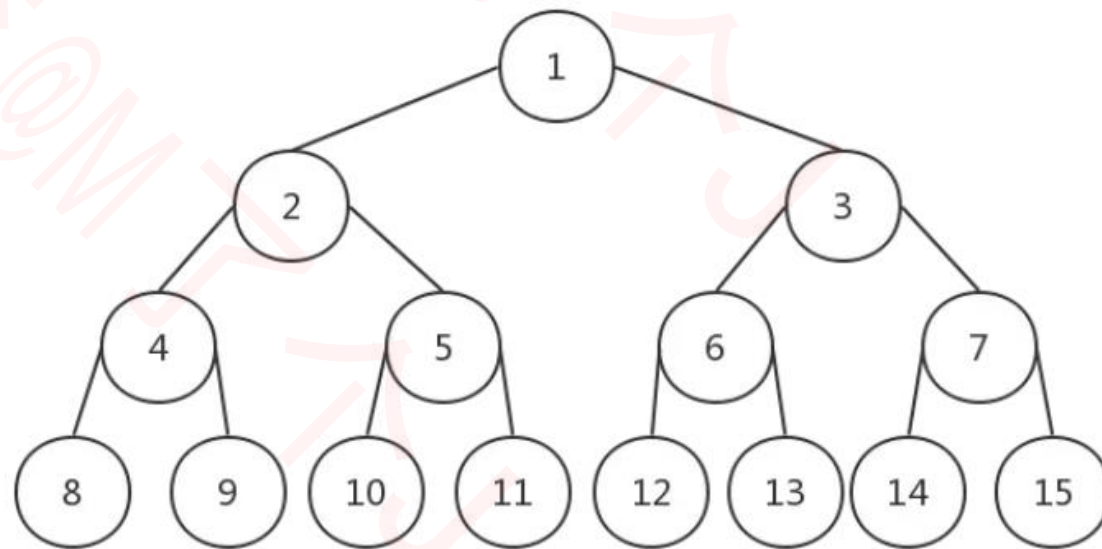
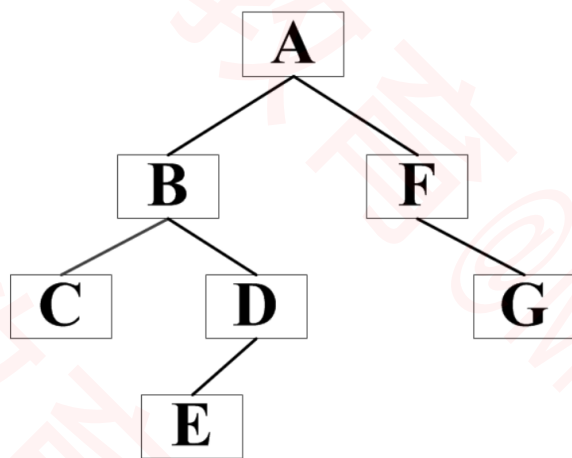
■ 二叉树是有序树 or 无序树?

- 有序树



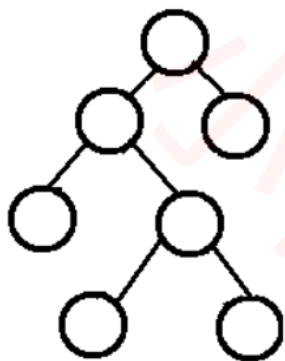
二叉树的性质

- 非空二叉树的第 i 层，最多有 2^{i-1} 个节点 ($i \geq 1$)
- 在高度为 h 的二叉树上最多有 $2^h - 1$ 个结点 ($h \geq 1$)
- 对于任何一棵非空二叉树,如果叶节点个数为 n_0 , 度为2的节点个数为 n_2 , 则有: $n_0 = n_2 + 1$
- 假设度为1的节点个数为 n_1 , 那么二叉树的节点总数 $n = n_0 + n_1 + n_2$
- 二叉树的边数 $T = n_1 + 2 * n_2 = n - 1 = n_0 + n_1 + n_2 - 1$
- 因此 $n_0 = n_2 + 1$

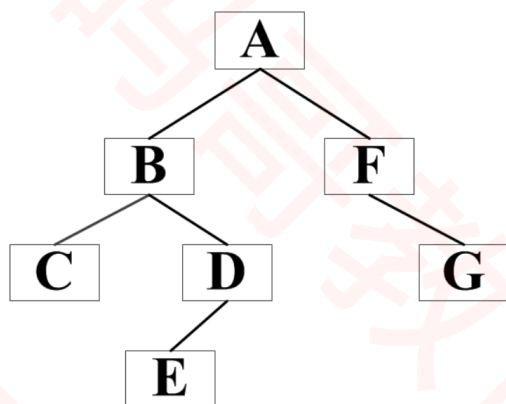


真二叉树 (Proper Binary Tree)

■ 真二叉树：所有非叶子节点的度都为2

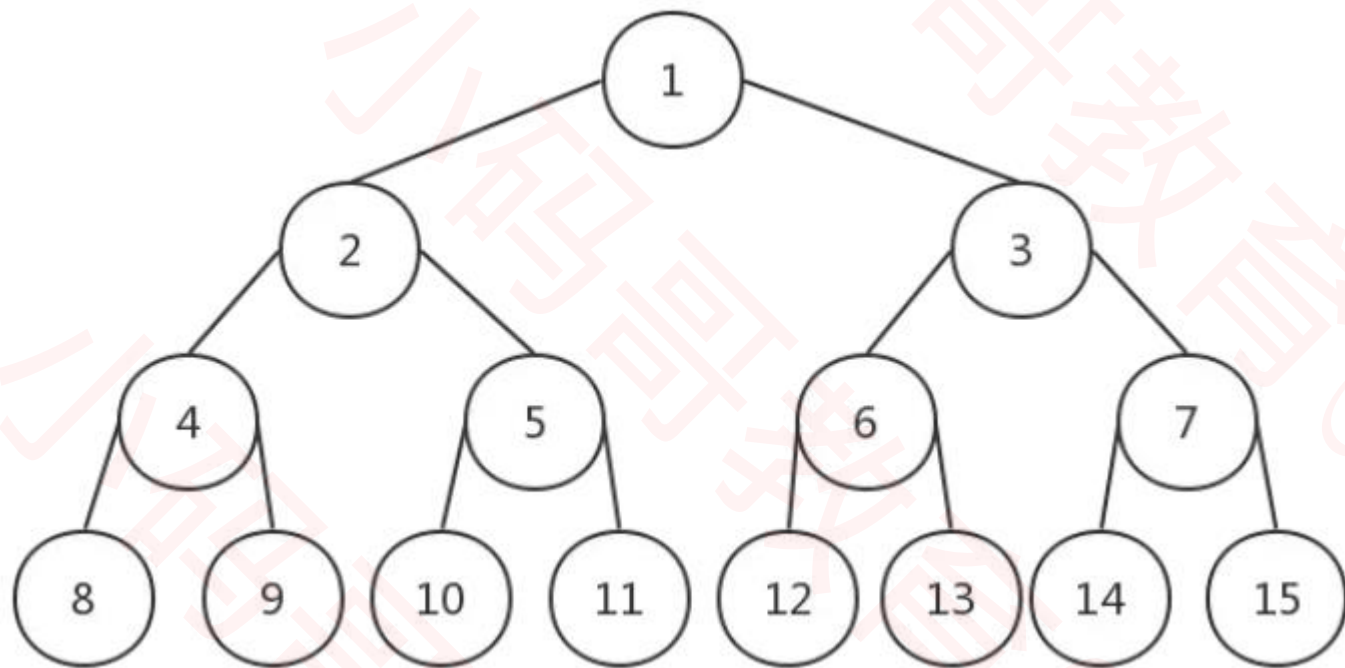


■ 下图不是真二叉树



满二叉树 (Full Binary Tree)

■ 满二叉树：所有非叶子节点的度都为2，且所有的叶子节点都在最后一层



■ 假设满二叉的高度为 h ($h \geq 1$)，那么

□ 第 i 层的节点数量： 2^{i-1}

□ 叶子节点数量： 2^{h-1}

□ 总节点数量 n

✓ $n = 2^h - 1$

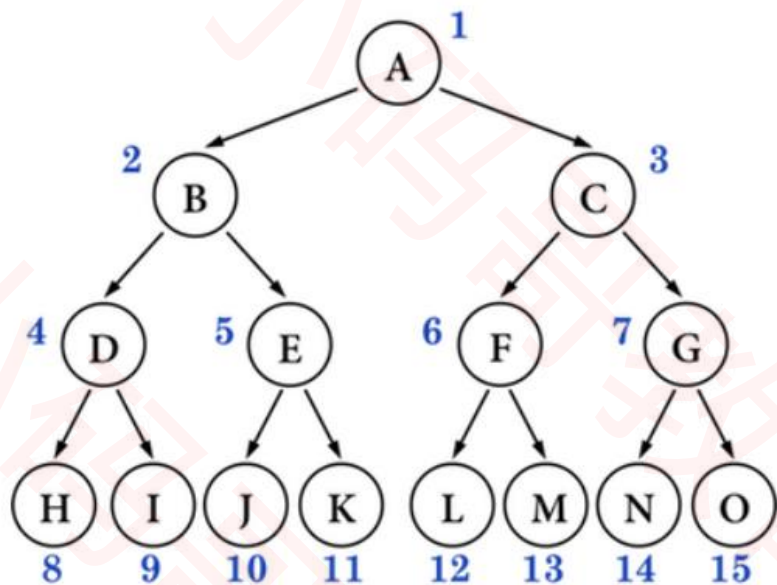
✓ $h = \log_2(n + 1)$

- 在同样高度的二叉树中，满二叉树的叶子节点数量最多、总节点数量最多
- 满二叉树一定是真二叉树，真二叉树不一定是满二叉树

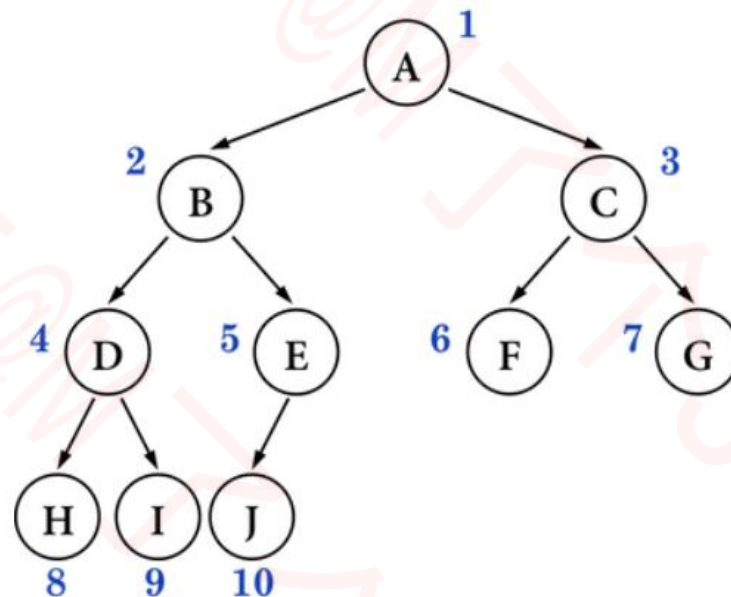
完全二叉树 (Complete Binary Tree)

- **完全二叉树**：高度为 h 、有 n 个结点的二叉树，从左至右、从上到下对节点进行编号，当其每一个结点都与高度为 h 的满二叉树中相同编号的结点一一对应时，称之为完全二叉树

满二叉树



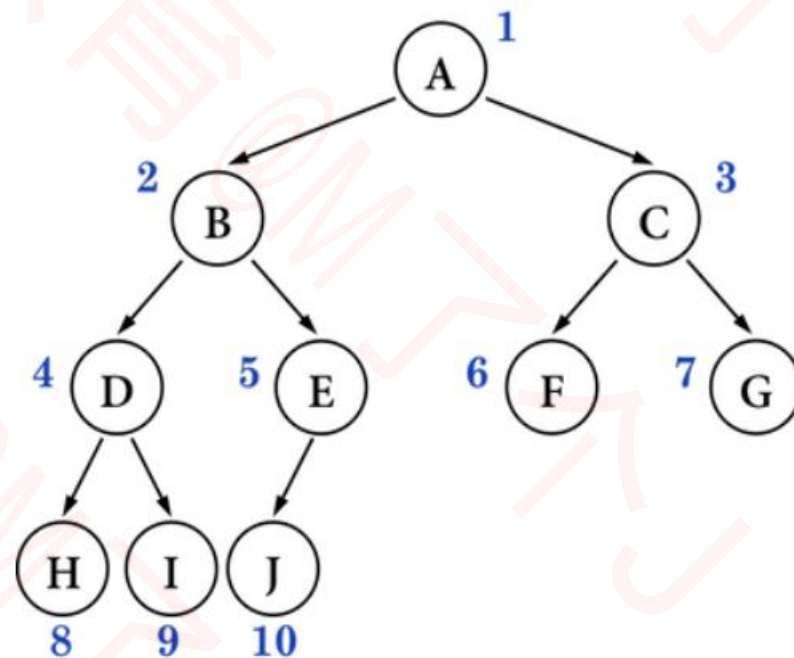
完全二叉树



- 完全二叉树从根结点至倒数第二层是一棵满二叉树，最后一层的叶子结点都靠左对齐
- 满二叉树一定是完全二叉树，完全二叉树不一定是满二叉树

完全二叉树的性质

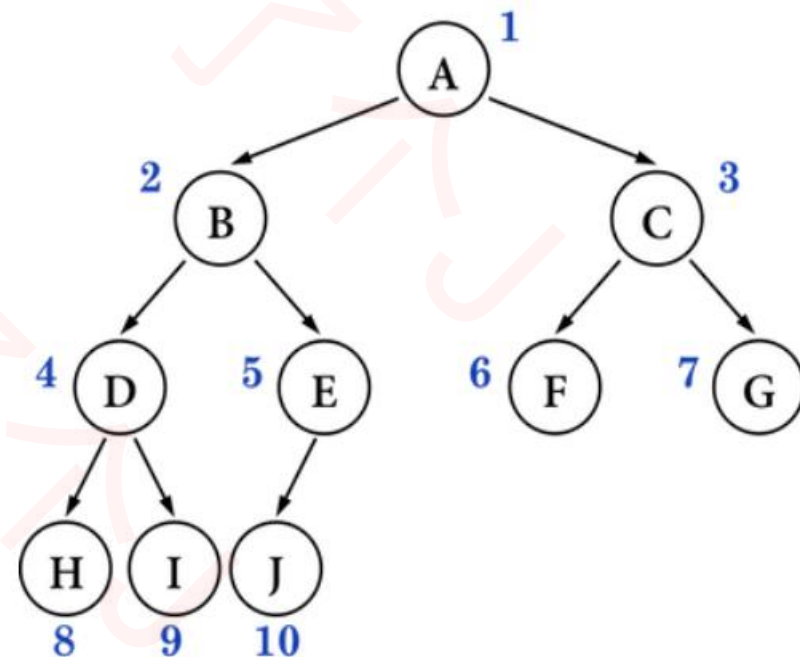
- 叶子节点只会出现在最下面2层
- 度为1的节点只有左子树
- 同样节点数量的二叉树，完全二叉树的高度最小
- 假设完全二叉树的高度为 h ($h \geq 1$)，那么
 - 至少有 2^{h-1} 个节点
 - 最多有 $2^h - 1$ 个节点
 - 总节点数量为 n
 - ✓ $h = \text{floor}(\log_2 n) + 1$
 - ✓ floor是向下取整的意思
 - $2^{h-1} \leq n < 2^h$
 - $h - 1 \leq \log_2 n < h$



完全二叉树

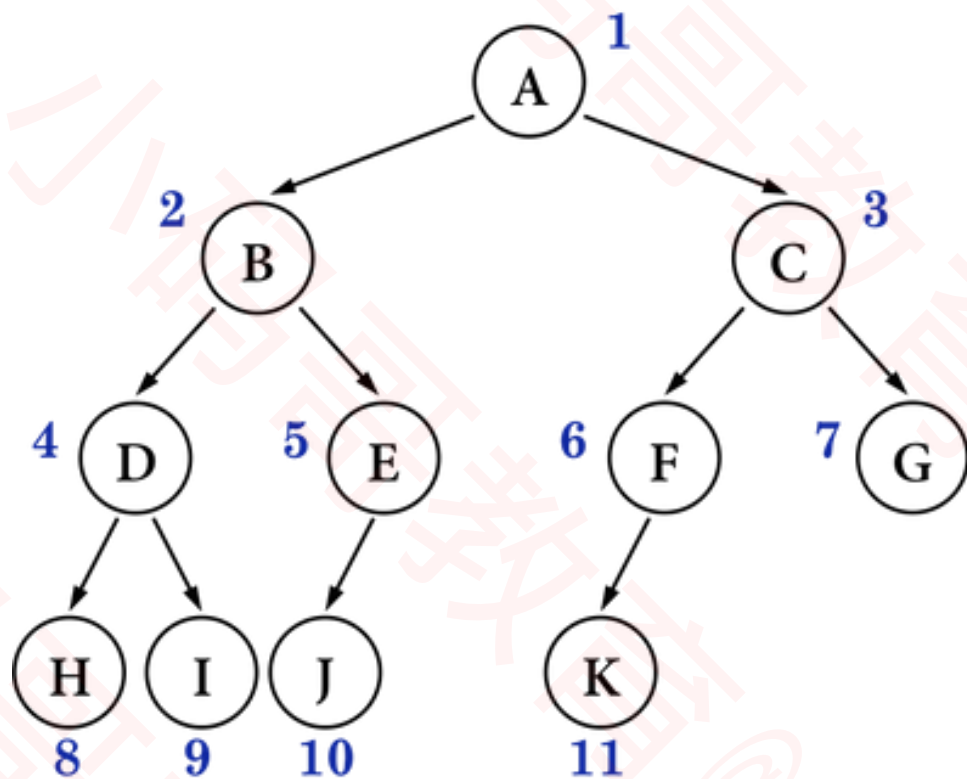
完全二叉树的性质

- 一棵有 n 个节点的完全二叉树 ($n > 0$)，从上到下、从左到右对节点进行编号，对任意第 i 个节点 ($0 < i < n$)
 - 如果 $i = 1$ ，它是根节点
 - 如果 $i > 1$ ，它的父节点编号为 $\text{floor}(i / 2)$
 - 如果 $2i \leq n$ ，它的左子节点编号为 $2i$
 - 如果 $2i > n$ ，它无左子节点
 - 如果 $2i + 1 \leq n$ ，它的右子节点编号为 $2i + 1$
 - 如果 $2i + 1 > n$ ，它无右子节点



完全二叉树

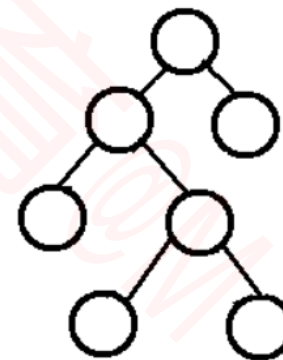
下图不是完全二叉树



国外教材的说法

■ Full Binary Tree: 完满二叉树

□ 所有非叶子节点的度都为2

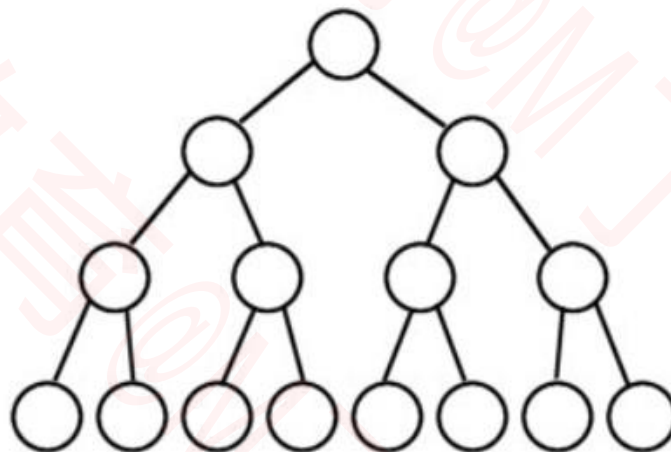


Full Binary Tree

■ Perfect Binary Tree: 完美二叉树

□ 所有非叶子节点的度都为2，且所有的叶子节点都在最后一层

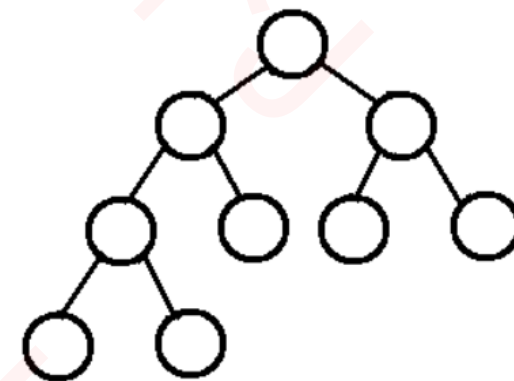
□ 就是国内说的“满二叉树”



Perfect Binary Tree

■ Complete Binary Tree: 完全二叉树

□ 跟国内的定义一样



Complete Binary Tree

练习 - 翻转二叉树

■ <https://leetcode-cn.com/problems/invert-binary-tree/>

输入:



输出:



■ 请分别用递归、迭代（非递归）两种方式实现