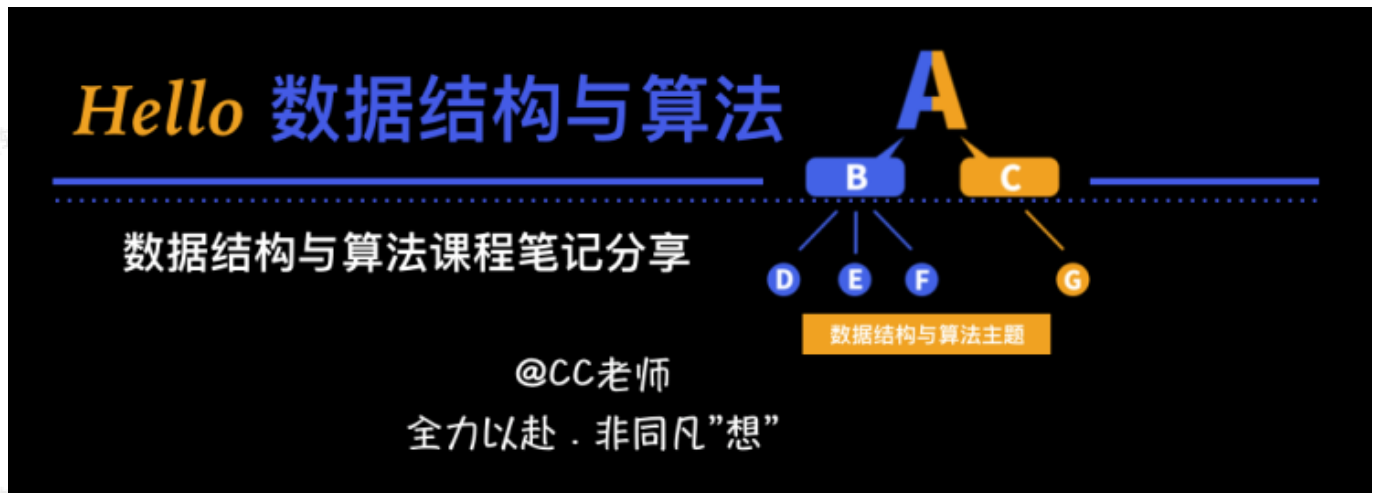


# 010--数据结构与算法 [二叉树初探]



## 一. 课程安排:

- 课程日期: 2020 年 4 月 24 日 第 10 次课程 (共9次课程)
- 授课老师: CC 老师
- 研发老师: CC 老师
- 课程时长: 2小时
- 课程主题: 二叉树初探
- 课程时间安排:
  - 上课: 20:00 – 21:00
  - 休息: 21:00 – 21:10
  - 上课: 21:10 – 22:00

### • 课程作业:

- 博客: 总结二叉树顺序存储下的实现

### • 课程内容安排:

- 了解树与二叉树相关定义
- 二叉树顺序存储的定义与相关操作的实现
- 二叉树顺序存储下的前序/后序/中序/层序遍历
- 二叉树链式存储下相关操作以及遍历实现

## 二. 课程内容笔记

1. 结点: 树中的一个独立单元. 包含一个数据元素及若干指向其他子树的分支. 例如, A,B,C,D等都是结点;

2. 结点的度: 结点拥有的子树数称谓结点的度. 例如A的度是3, C的度为1, D的度为3, F的度为0.
3. 树的度: 树的度是树内各结点度的最大值,例如,上图中的应该是3;
4. 叶子: 度为0的结点称谓叶子或终端结点. 例如,K,J,F,G,M,I,J 都是树的叶子.
5. 非终端结点: 度不为0的结点成为非终端结点或分支结点. 除了根结点以外,非终端结点也称为内部结点;
6. 双亲和孩子: 结点的子树的根称为该结点的孩子, 相应地, 该结点称为孩子的双亲. 例如,B的双亲为A, B的孩子有E和F.
7. 兄弟: 同一个双亲的孩子之间称为兄弟结点, 例如H,I和J互为兄弟;
8. 祖先: 从根到该结点所经历的分支上的所有结点, 例如, M的祖先为A,D,H.
9. 子孙: 以某结点为根的子树中的任一结点都称为该结点的子树. 例如,B的子孙为E,F.
10. 层次: 结点的层次从根开始定义起, 根为第一层, 根的孩子为第二层. 树中任一层次等于双亲结点的层次加1.
11. 堂兄弟: 双亲在同一层的解答互为堂兄弟. 例如,结点G与E,F,H,i,J 互为堂兄弟.
12. 有序树和无序树: 如果将树的结点的各子树看成从左到右是有次序的(即不能互换)则称为该树为有序树, 否则是无序树. 在有序树中最左边的子树的根称为第一个孩子,最右边的称为最后一个孩子. 什么叫有序树,就类似在家谱中第一房太太,到第五房太太以及孩子是有顺序的.这样存在顺序关系叫有序树.
13. 节点的高度: 节点到叶子节点的最长路径(边数)
14. 节点的深度: 根结点到这个结点所经历的边的个数
15. 节点的层数: 节点的深度-1
16. 树的高度: 根结点的高度.

**二叉树(Binary Tree) 是 $n$  ( $n \geq 0$ )个结点所构成的集合. 它或为空树( $n=0$ ),对于非空树T:**

1. 有且仅有一个称之为根结点
2. 除了根结点以外的其余结点分为2个互不相交的子集 $T_1, T_2$ . 分别称为T的左子树和右子树,且 $T_1$ 和 $T_2$ 本身都是二叉树.

**二叉树的特性:**

1. 二叉树每个结点至多只有2颗子树(二叉树中不存在度大于2的结点). 所以二叉树中不存在大于2的结点.  
**注意:** 不是只有2个子树,而是最多只有. 如果二叉树中没有子树或者只有一颗树是可以的.
2. 二叉树的子树有左右之分,其次序不能任意颠倒.类似:就像人的双手,双脚.有顺序之分
3. 即使只有一棵树,也需要区分是左子树还是右子树. 类似: 就像你在生活中,摔伤了手.伤的是左手还是右手,对你的生活影响都是完全不同的.

对一颗具有 $n$ 个结点的二叉树按层序编号,如果编号为 $i$  ( $1 \leq i \leq n$ )的结点与同样深度的满二叉树中编号为 $i$ 的结点二叉树中位置完全相同. 则这颗二叉树称为完全二叉树.

1. 首先"完全" 和 "满" 的差异, 满二叉树一定是一个完全二叉树不一定是满的.
2. 完全二叉树的所有结点和同样深度的满二叉树,它们按照层序编号相同的结点一一对应. 这里有一个关键词是**按层序编号**.

1. 叶子结点只能出现在最下两层
2. 最下层的叶子一定集中在左部连接
3. 倒数第二层,若有叶子节点,一定都在右部连续位置
4. 如果结点度为1, 则该结点只有左孩子, 既不存在只有右子树的情况
5. 同样结点数的二叉树, 完全二叉树的深度最小;