# 树结构

# 树和二叉树

树的特点：非线性结构，一个直接前驱，但可能有多个直接后继(1:n)

树的定义：由一个或多个(n≥0)结点组成的有限集合，在任何一颗非空树T中：

1. 有且仅有一个结点称为根(root)
2. 当n>1时，其余的结点分为m(m≥0)个互不相交的有限集合T1,T2...Tm。每个集合本身又是树，被称为这个根的子树.

注：树的定义具有递归性，即树中还有树.

树的表式法：

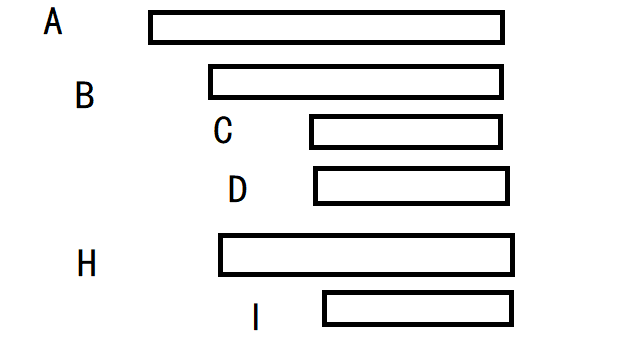
图形表示法

嵌套集合表示法：（画圆圈）

广义表表示法：根作为由字数森林组成的表的名字写在表的左边(括号表示)

示例：A(B(E(K,L),F),(C(G),D(H(M),I,J)

凹入表示法（目录表示法）



1. 示例术语

根--根结点（没有前驱）

有序树---结点各子树从左至右有序，不能互换（左为第一）

无序树---结点各子树可互换位置

双亲：上层的那个结点（直接前驱）

孩子：下层结点的子树的根（直接后继）

兄弟：同一双亲下的同一层结点（孩子间互称兄弟）

堂兄弟：双亲位于同一层的结点（但并非同一双亲)

祖先：从根到该结点所经分支的所有结点

子孙：该结点下层子树的任一结点

结点：树中的数据元素

结点的度：结点拥有的子树的数目（有几个直接后继就是几）

结点的层次：从根到该结点的层数（根结点算第一层）

叶子：度为0的结点（终端结点）

分支结点：度不为0 的点（非终端结点）

树的度：所有结点度中的最大值（Max{各结点的度}）

树的深度(或高度)：所有结点中最大的层数（Max{各结点的层次}）

森林：n(n>0)个互不相交的树的集合

删除树的根结点就成了森林。反之，给n棵独立的树增加一个结点，将n棵作为新增结点的树，树就成了森林

二叉树

二叉树的结构最简单，规律性最强

可以证明，所有的树都能转化为唯一对应的二叉树，不失一般性

1. 二叉树的定义：

基本特征：

1. 每个结点最多只有两棵子树（不存在大于2的结点）
2. 左子树和右子树次序不能颠倒（有序树）

性质1：在二叉树的第i层上至多有2i-1 个结点（i>0)

性质2：深度为k的二叉树至多有2k-1个结点(k>0)

性质3：对于任何一棵二叉树，若度为2的结点数有n2个，叶子数结点数为n0，则n0=n2+1

证明性质3：

全部结点数：n=n0+n1+n2（叶子数+度为1的结点数+度为2 的结点数）

（出根结点外，每个结点必有一个直接前驱，即一个分支）

二叉树中全部结点数n=B+1(总分支数+根结点）

总分支数：B=n1+2n2

## 满二叉树

二叉树中每层都是满的

定义:深度为且有2k-1个结点的二叉树,可以是完全二叉树

特点：每一层的结点数都是最大结点数。可以对满二叉树的结点纪进行连续编号

完全二叉树：

除最后一层外，其余层都是满的.

深度为k，有n个结点的二叉树，当且仅当每一个结点都与深度为k的满二叉树编号从1至n的结点一一所对应时，称为完全二叉树.

特点：（1）叶子结点只可能在层次最大的两层上面出现.

1. 对任一结点，若其右分支下的子孙的最大层次为h，则其左分支下的子孙的最大层数必为 h或h+1

性质4：具有n个结点的完全二叉树的深度为[log2n]+1 []表示向下取整

遍历二叉树

1. 先序遍历
2. 中序遍历
3. 后序遍历

# 二叉树的存储结构

1. 顺序存储结构：用一组地址连续的存储单元依次自上而下，自左至右存储二叉树上的结点单元
2. 仅适合于完全二叉树
3. 对于非完全二叉树：将各层空缺处全部补上‘虚结点’，其内容为0

## 链式存储结构

二叉树表中包含2个指针域，一般从根节点开始存储

为了便于找到结点的双亲，可再增加一个双亲域指针，将二叉表变为三叉表

遍历二叉树

1. 先序遍历：根结点，左子树，右子树
2. 中序遍历：左子树，根结点，右子树
3. 后序遍历：左子树，右子树，根结点

讨论：若已知先序序列（或后序序列）和中序序列，能否恢复出对应的二叉树

前序，中序，唯一

前序，后序，不唯一

# 树和森林

树的存储方式：（1）双亲表示法 （2）孩子表示法 （3)孩子兄弟表示法

1. 用双亲表示法存储

方法：用一组连续空间来存储树的结点，同时在每个结点中附设一个指示器，指示其双亲结点在链表中的位置。

1. 用孩子存储法来表示

方法：将每个结点的孩子排列起来，形成一个带表头（装父亲结点）的线性表（n个结点要设立n个链表),再将n个表头用数组存放起来，这样就形成一个混合结构

1. 孩子兄弟表示法

用二叉链表来存储树，但链表的两个指针域含义不同

# 哈夫曼树(最优二叉树)

路径：由一个结点到另一个结点间的分支所构成

路径长度：路径上的分支数目

树的路径长度：从根到每一结点长度之和

带权路径长度：结点到根的路径长度与结点上权值的乘积，公式:WPL=∑

树的带权路径长度：书中所有叶子结点的带权路径长度之和

哈夫曼树：带权路径长度最小的树

1. 构造哈夫曼树的基本步骤

一颗有n个结点的哈夫曼树共有2n-1个结点

## 哈夫曼编码

1. 统计频度
2. 用频度构造哈夫曼树
3. 约定：左分支为0，右分支为1,叶结点为对应字符
4. 从根到叶子路径的0或1的序列即为字符的哈夫曼编码