## 数组

## 线性表

## 树

### 二叉树

#### 性质1

深度为k的树，终端节点最多有2^(k-1)个

#### 性质2

深度为k的树，最多有2^k – 1 个节点

#### 性质3

终端节点有n0个，度为2的节点有n2个，那么：

N0 = n2 + 1

*解答：*

*N = n0 + n1 + n2 （总的节点数=终端数+度为1的节点的数量 +度为 2的节点的数量）*

*N = n1 + 2n2 + 1 （n1代表了n1根线，n2代表了有2n2根线，对应的节点数就是：n1 + 2n2 ，而这种计算方式计算的每根线的下方的节点数，等于说没有计算根节点，所以再+1，就等于总数n了）*

*两者相减，得结果。*

### 遍历二叉树

17.已知二叉树后序遍历序列是dabec，中序遍历序列是debac，它的前序遍历序列是（cedba）

**答案：**

***中序遍历:DEBAC***

***后序遍历:DABEC***

***推导如下：***

***1、从后序可知树根为C,因为最后的节点是树根。***

***2、从中序的规则可知树根在中间,树根的左边是左孩子,右边是右孩子。很明显树根C是没有右孩子,只有左孩子DEBA。***

***中序遍历:DEBA***

***后序遍历:DABE***

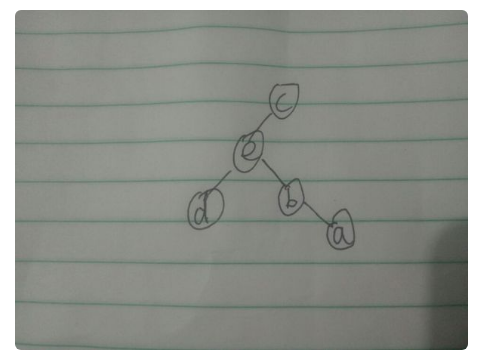
***推出E是左子树的根结点,并且存在左子树D,右子树BA,因为从中序遍历可知E的左边是D,右边是BA***

***中序遍历:BA***

***后序遍历:AB***

***推出B是右子树的根结点,并且存在右子树,但没有左子树,因为从中序遍历可知B只有右子树,没有左子树。***

***还原二叉树如下图：***



18.已知一棵二叉树前序遍历和中序遍历分别为ABDEGCFH和DBGEACHF，则该二叉树的后序遍历为（DGEBHFCA）  
19.若某二叉树的前序遍历访问顺序是abdgcefh，中序遍历访问顺序是dgbaechf，则其后序遍历的结点访问顺序是（gdbehfca）

## 时间复杂度

### 时间复杂度排序

O(1) < O(logn) <O(n) < O(nlog)< n’2 < n’3 < 2’n < n! < n’n