1. 填空

⑴ 树是 n（n≥0）结点的有限集合，在一棵非空树中，有（ 一 ）个根结点，其余的结点分成 m（m＞0）个（ 互不相交 ）的集合，每个集合都是根结点的子树。

⑵ 树中某结点的子树的个数称为该结点的（ 度 ），子树的根结点称为该结点的（ 孩子 ），该结点称为其子树根结点的（ 双亲 ）。

⑶ 一棵二叉树的第 i（i≥1）层最多有（ ）个结点；一棵有 n（n>0）个结点的满二叉树共有（ ）个叶子结点和（ ）个非终端结点。

⑷ 设高度为 h 的二叉树上只有度为 0 和度为 2 的结点，该二叉树的结点数可能达到的最大值是（ ），最小值是（ ）。

⑸ 深度为 k 的二叉树中，所含叶子的个数最多为（ ）。

⑹ 具有 100 个结点的完全二叉树的叶子结点数为（ ）。

⑺ 已知一棵度为 3 的树有 2 个度为 1 的结点，3 个度为 2 的结点，4 个度为 3 的结点。则该树中有（ ）个叶子结点。

⑻ 某二叉树的前序遍历序列是 ABCDEFG，中序遍历序列是 CBDAFGE，则其后序遍历序列是（ ）。

⑼ 在具有 n 个结点的二叉链表中，共有（2n）个指针域，其中（n-1）个指针域用于指向其左右孩子，剩下的（n+1）个指针域则是空的。

（10） 在有 n 个叶子的哈夫曼树中，叶子结点总数为（ ），分支结点总数为（ ）。

2. 选择题

⑴ 如果结点 A 有 3 个兄弟，B 是 A 的双亲，则结点 B 的度是（ D ）。

A 1 B 2 C 3 D 4

⑵ 设二叉树有 n 个结点，则其深度为（D）。



⑶ 二叉树的前序序列和后序序列正好相反，则该二叉树一定是（B）的二叉树。

A 空或只有一个结点 B 高度等于其结点数

C 任一结点无左孩子 D 任一结点无右孩子

（4） 深度为 k 的完全二叉树至少有（B）个结点，至多有（C）个结点。

A 2k-2+1 B 2k-1 C 2k -1 D 2k–1 -1

（5） 一个高度为 h 的满二叉树共有 n 个结点，其中有 m 个叶子结点，则有（D）成立。

A n=h+m B h+m=2n C m=h-1 D n=2m-1

（6） 任何一棵二叉树的叶子结点在前序、中序、后序遍历序列中的相对次序（C）。

A 肯定不发生改变 B 肯定发生改变 C 不能确定 D 有时发生变化

（7）如果 T' 是由有序树 T 转换而来的二叉树，那么 T 中结点的前序序列就是 T' 中结点的（ ）序列，T 中结点的后序序列就是 T' 中结点的（ ）序列。

A 前序 B 中序 C 后序 D 层序

（8）设森林中有 4 棵树，树中结点的个数依次为 n1、n2、n3、n4，则把森林转换成二叉树后，其根结点的右子树上有（ ）个结点，根结点的左子树上有（ ）个结点。

A n1-1 B n1 C n1+n2+n3 D n2+n3+n4

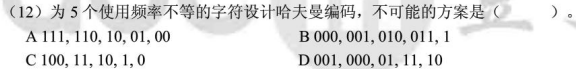
（9）讨论树、森林和二叉树的关系，目的是为了（ ）。

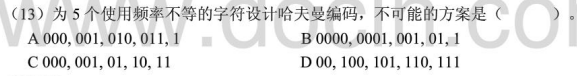
A 借助二叉树上的运算方法去实现对树的一些运算

B 将树、森林按二叉树的存储方式进行存储并利用二叉树的算法解决树的有关问题

C 将树、森林转换成二叉树

D 体现一种技巧，没有什么实际意义





1. 判断题

（1） 在二叉树的前序遍历序列中，任意一个结点均处在其子女的前面。

（2） 二叉树是度为 2 的树。

（3）由树转换成二叉树，其根结点的右子树总是空的。

（4）用一维数组存储二叉树时，总是以前序遍历存储结点。

1. 解答题
2. 
3. 









1. 算法设计题

⑴ 设计算法求二叉树的结点个数。

⑵ 设计算法按前序次序打印二叉树中的叶子结点。

⑶ 设计算法求二叉树的深度。

⑷ 编写算法，要求输出二叉树后序遍历序列的逆序。