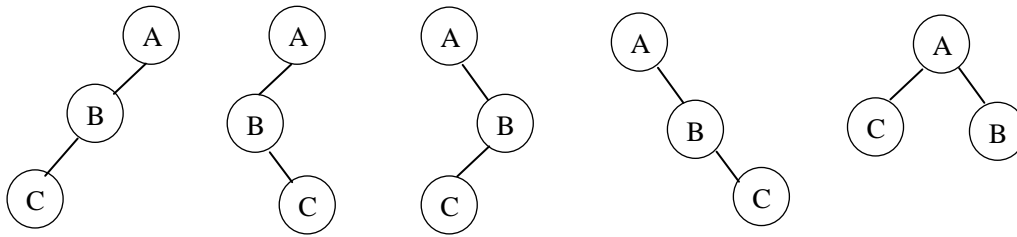


☆树与二叉树

1. 对于三个结点 A、B 和 C，共可构成多少种不同的树？



$$A_3^3 \times 5 = 30 \text{ 种}$$

2. 如果一棵树有 n_1 个度为 1 的结点， n_2 个度为 2 的结点， \dots n_m 个度为 m 的结点，则共有多少个叶结点？

解答：设有 n_0 个叶子节点

$$\text{总结点数 } n = n_0 + n_1 + n_2 + \dots + n_m$$

$$\text{总分支数 } e = n - 1 = n_0 + n_1 + n_2 + \dots + n_m - 1$$

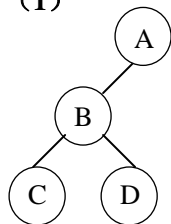
$$= m \cdot n_m + (m-1) \cdot n_{m-1} + \dots + 2 \cdot n_2 + n_1$$

$$n_0 = \left(\sum_{i=2}^m (i-1)n_i \right) + 1$$

则有

3. 分别按前、中序和后序遍历算法，列出下图各二叉树的结点序列。

(1)

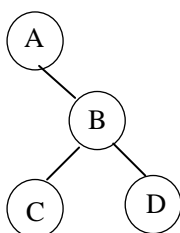


前序：A B C D

中序：C B D A

后序：C D B A

(2)

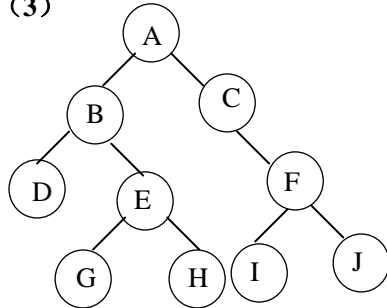


前序: A B C D

中序: A C B D

后序: C D B A

(3)



前序: A B D E G H C F I J

中序: D B G E H A C I F J

后序: D G H E B I J F C A

3. 写出一个将二叉树中每个结点的左、右孩子交换的算法

swaptree(T)。

算法:

```
void swaptree(T)//交换所有结点的左右子树
{
    T->lchild<->T->rchild; //交换左右子树
    if(T->lchild) swaptree(T->lchild);
    if(T->rchild) swaptree(T->rchild); //左右子树再分别交换各自的左右子树
}
```

5.编写按层次遍历二叉树的算法，同层结点按从左至右的次序进行。

思路: 利用一个队列，首先将根（头指针）入队列，以后若队列不空则取队头元素 p，如果 p 不空，则访问之，然后将其左右子树入队列，如此循环直到队列为空。

算法:

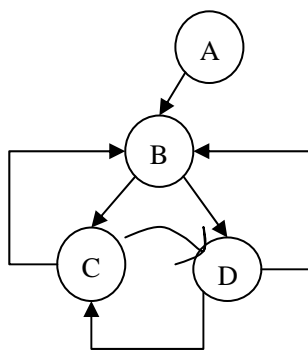
```
void LevelOrder ( BinTree bt )
{
    // 队列初始化为空
    InitQueue( Q );
    // 根入队列
    EnQueue( Q, bt );
    // 队列不空则继续遍历
```

```

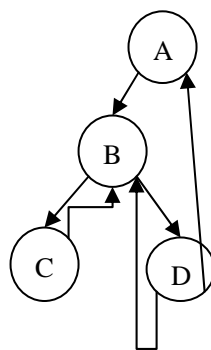
while ( ! QueueEmpty(Q) ) {
DeQueue( Q, p );
if ( p!=NULL ) {
visit ( p->data );
    // 左、右子树入队列
EnQueue( Q, p->lchild );
EnQueue( Q, p->rchild );
    }
}
}

```

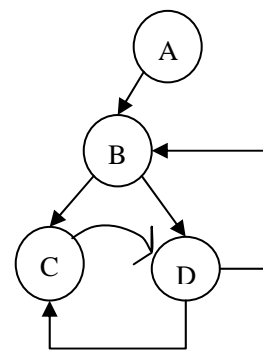
6.分别画出第 3 题中各二叉树的前序线索树、中序线索树和后序线索树。



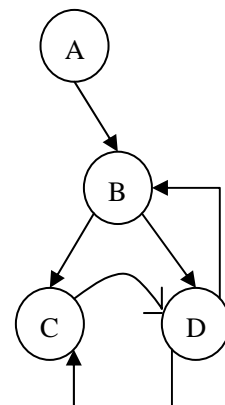
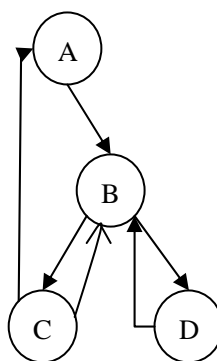
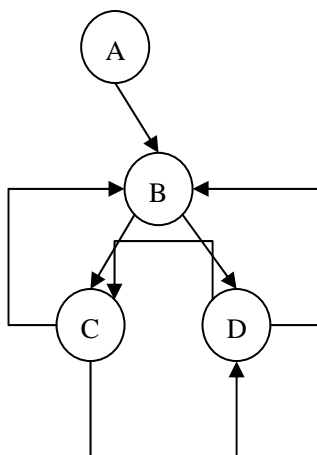
前



中



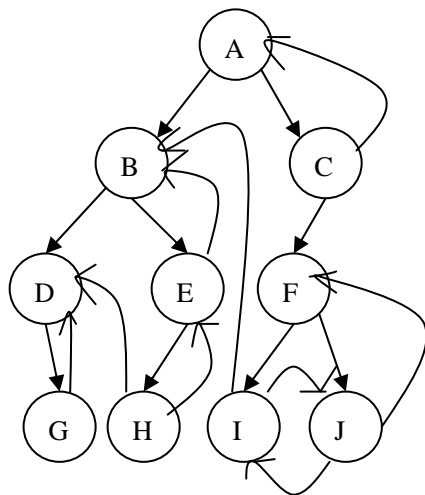
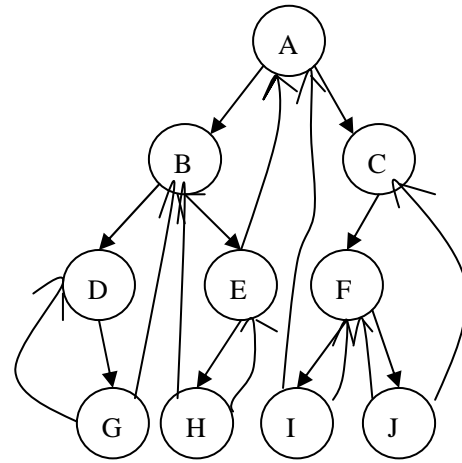
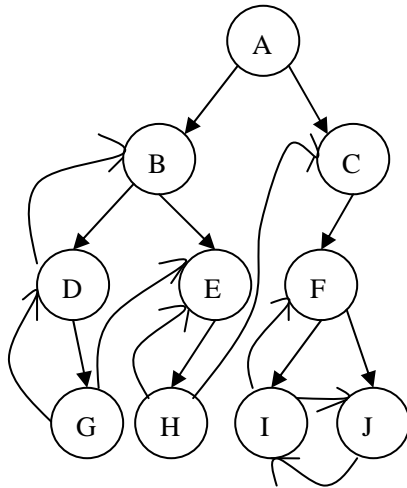
后



前

中

后



7.把如下的各树转换成二叉树，或各二叉树转换成相应的树

PPT 里没看到图，故略，该题很简单的

9.试以二叉链表作为存储结构，写出计算二叉树中叶结点数目的算法。

算法：

intLeafCount_BiTree(Bitree T)//求二叉树中叶子结点的数目

{

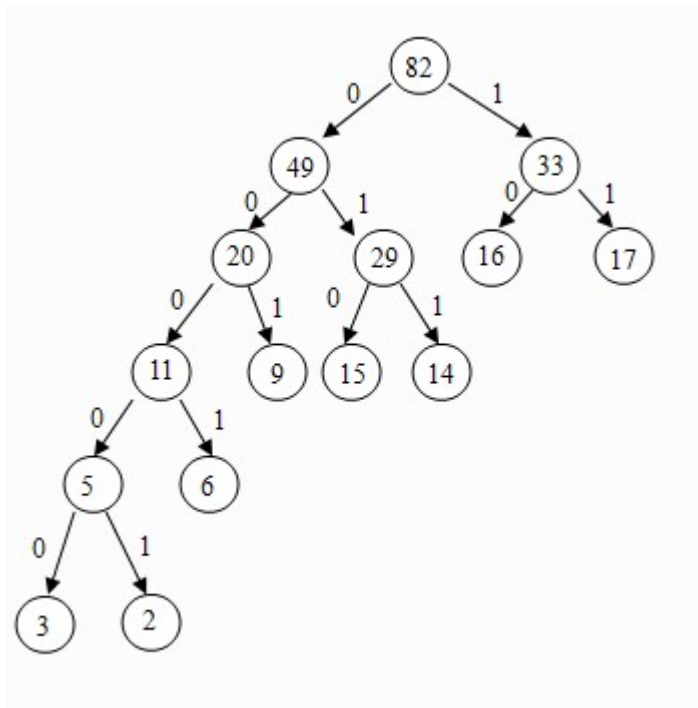
```

if(!T) return 0; //空树没有叶子
else if(!T->lchild&&!T->rchild) return 1; //叶子结点
    else return Leaf_Count(T->lchild)+Leaf_Count(T->rchild); //左子树的叶子数加上
    右子树的叶子数
} //LeafCount_BiTree

```

10. 给定权值集合{15,3,14,2,6,9,16,17}，构造响应的哈夫曼树，并计算它的带权路径长度。

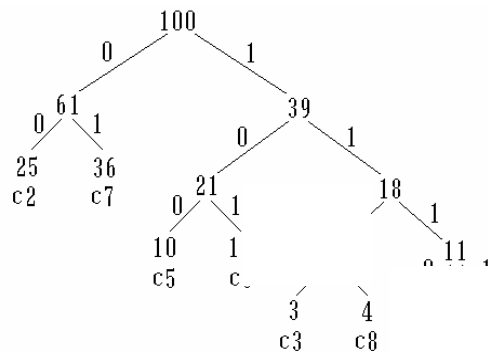
解答： 带权路径长度= $3 \times 5 + 2 \times 5 + 6 \times 4 + 9 \times 3 + 15 \times 3 + 14 \times 3 + 16 \times 2 + 17 \times 2$
=249



11. 假定用于通讯的电文仅由 8 个字母 a、b、c、d、e、f、g、h 组成，

个字母在电文中出现的频率分别为 5,25,3,6,10,11,36,4。试为这 8 个字母设计不等长 huffman 编码。

已知字母集 {a, b, c, d, e, f, g, h}
次数 {5, 25, 3, 6, 10, 11, 36, 4}



则 Huffman 编码为

a	b	c	d	e	f	g	h
1110	00	1100	1111	100	101	01	1101

电文总码数为 $4 * 5 + 2 * 25 + 4 * 3 + 4 * 6 + 3 * 10 + 3 * 11 + 2 * 36 + 4 * 4$
 $= 257$