1.

出度等于入度(根节点入度为 0)

$2n_2 + 3n_3 + 4n_4 + \cdots + \mathrm{m}n_m = \mathrm{n} + n_2 + n_3 + n_4 + \cdots + n_m - 1$

移项得

$n_2 + 2n_3 + 3n_4 + \cdots + (\mathrm{m}-1)n_m = \mathrm{n} - 1$

即

$n = n_2 + 2n_3 + 3n_4 + \cdots + (\mathrm{m}-1)n_m + 1$

2.

完全二叉树叶节点数小于满二叉树叶节点数

$n < 2^{h-1}$

取对数得

$\log_2 n < h - 1$

向上取整得

$\lceil \log_2 n \rceil = h - 1$

移相得

$h = \lceil \log_2 n \rceil + 1$

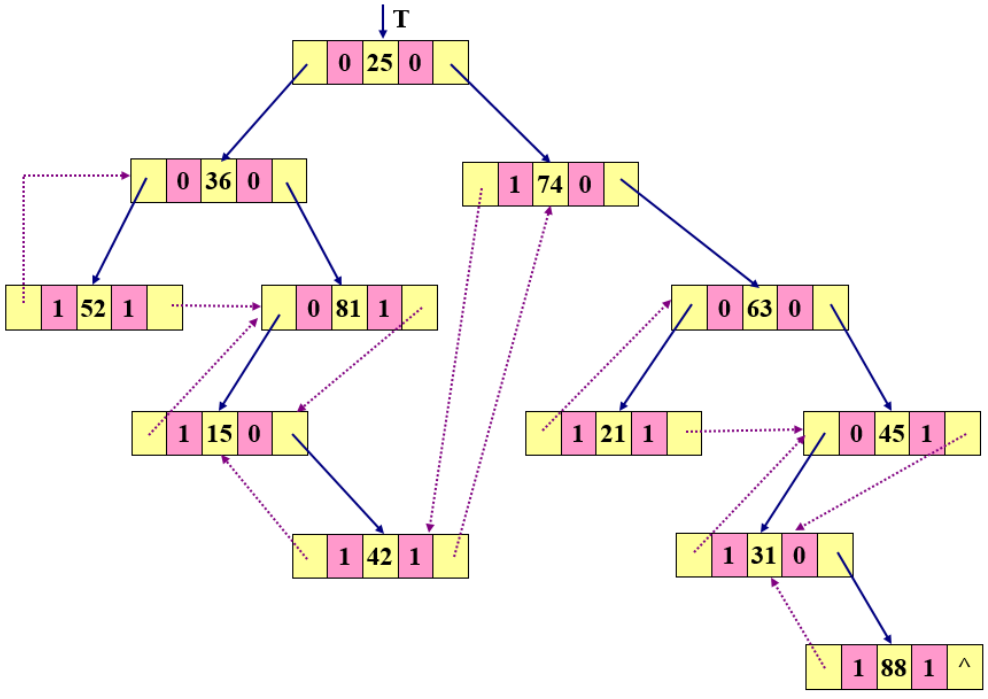
3.

(1) DLR: 25 36 52 81 15 42 74 63 21 45 31 88

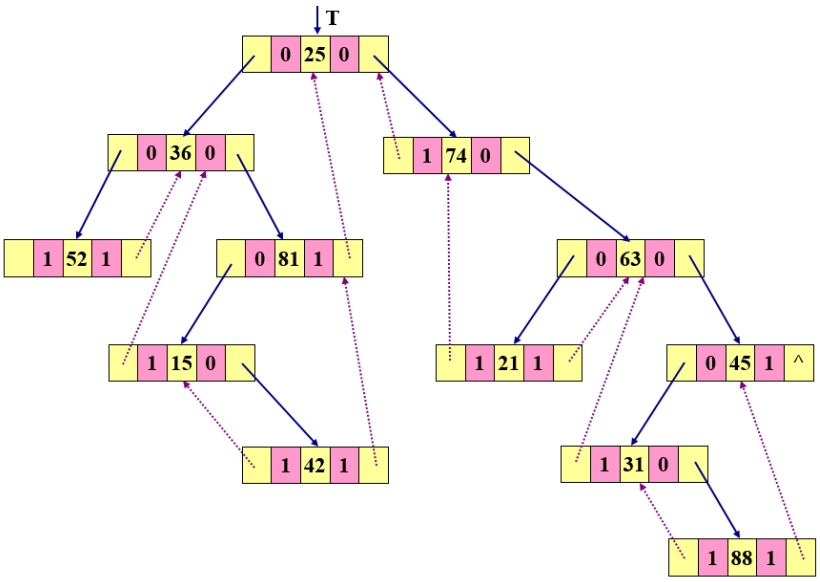
LDR: 52 36 15 42 81 25 74 21 63 31 88 45

LRD: 52 42 15 81 36 21 88 31 45 63 74 25

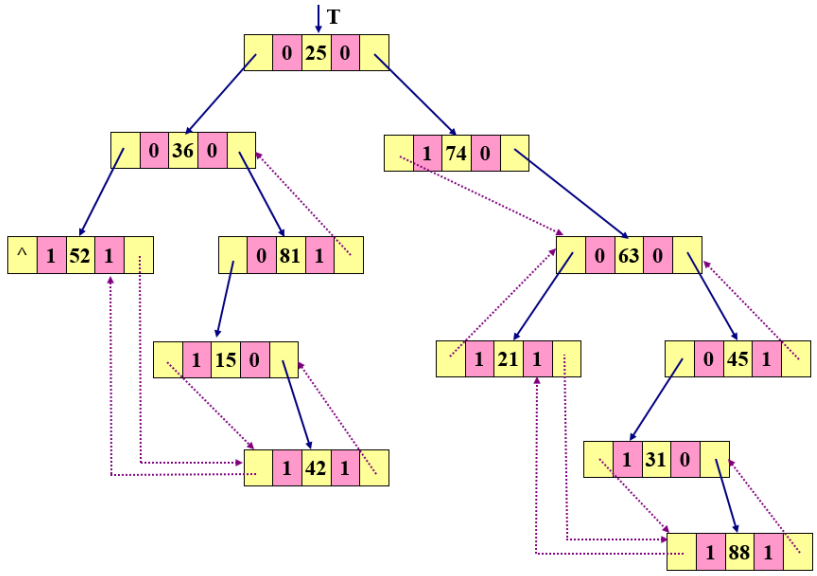
(2) 前序：



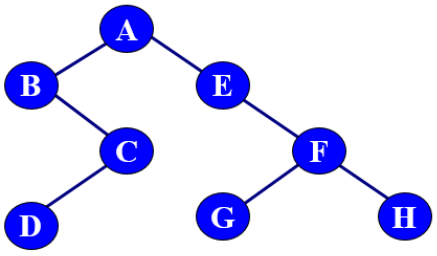
中序：


T
0 25 0
0 36 0
1 74 0
1 52 1
0 81 1
0 63 0
1 15 0
1 21 1
0 45 1 ^
1 42 1
1 31 0
1 88 1

后序：


T
0 25 0
0 36 0
1 74 0
^ 1 52 1
0 81 1
0 63 0
1 15 0
1 21 1
0 45 1
1 42 1
1 31 0
1 88 1

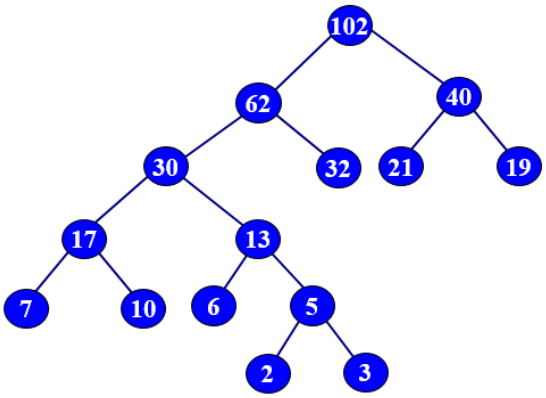
4.


A
B
E
C
F
D
G
H

5.

```
BTptr Find (BTptr BT) {
        BTptr p;
        // 没有右子树
        if (BT->Rtag==1)
                return NULL;
        p = BT->Rchild;
        // 寻找右子树的后序遍历的第一个节点
        while(p->Ltag == 0 || p->Rtag == 0)
                p = p->Ltag == 0 ? p->Lchild : (p->Rtag == 0 ? p->Rchild : NULL);
        return p;
}
```

6.



WPL = 102 + 62 + 30 + 17 + 13 + 5 + 40 = 269