

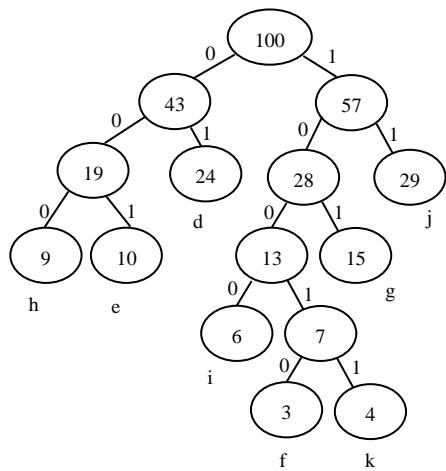
5.1. 某通信系统有八种字符: d、e、f、g、h、i、j、k, 其权值分别为: 0.24、

0.10、0.03、0.15、0.09、0.06、0.29、0.04, 请完成:

- (1) 构造 Huffman 树 (要求所有结点左孩子的权值不大于右孩子的权值);
- (2) 据此设计出各个字符的 Huffman 编码;
- (3) 求出该 Huffman 树的带权路径长度 WPL;
- (4) 并译出下列报文: 111000011011001010011。

解:

(1) Huffman 树: 为表达方便, 权值放大 100



(2) Huffman 编码

字符	Huffman 编码
d	01
e	001
f	10010
g	101
h	000
i	1000
j	11
k	10011

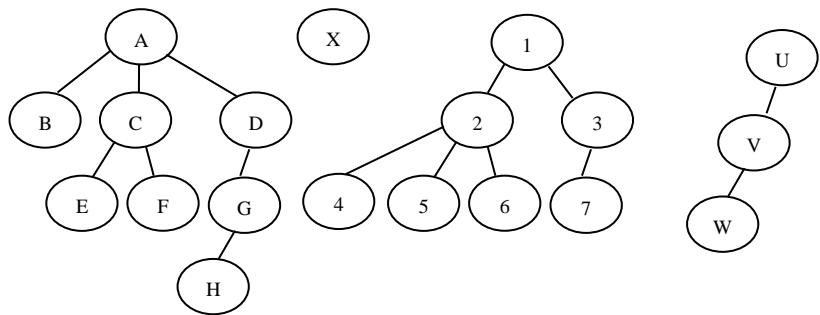
(3) 求带权路径长度 WPL

$$WPL = ((0.03+0.04) \times 5 + 0.06 \times 4 + (0.09+0.10+0.15) \times 3 + (0.24+0.29) \times 2) = 2.67$$

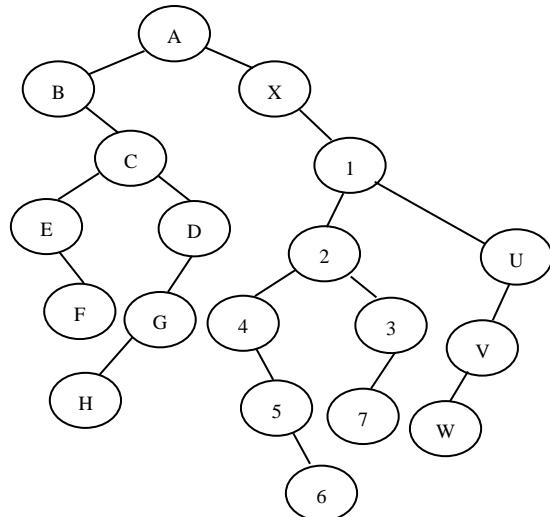
(4) 译报文

报文译码为: 11 1000 01 101 10010 10011
j i d g f k

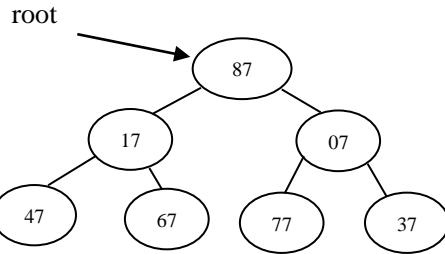
5.2. 将下面的森林 ($F=\{T_1, T_2, T_3, T_4\}$) 转换为对应的二叉树。



解：

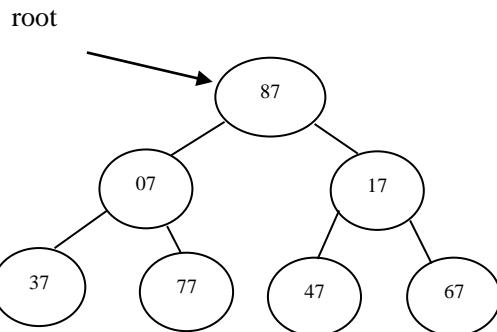


5.3. 以指向左侧二叉树的指针 `root` 作初始值, 执行递归算法 `ShiftTree(root)`,
请指出执行后的二叉树结构图。



```
ShiftTree( T )
{
    if (T == Null) return( 0 );
    TL = T->Lchild;
    ShiftTree( TL );
    TR = T->Rchild;
    if ( ( TL <> Null ) && ( TR <> Null ) && ( TL->data > TR->data ) )
        {
            T->Lchild = TR;
            T->Rchild = TL;
        }
    ShiftTree( TR );
}
```

解：



```
0 ShiftTree( T )
1 {
2     if (T == Null) return( 0 );
3     TL = T->Lchild;
4     ShiftTree( TL );
5     TR = T->Rchild;
6     if ( ( TL <> Null ) && ( TR <> Null ) && ( TL->data > TR->data ) )
7         {
8             T->Lchild = TR;
9             T->Rchild = TL;
10            }
11 }
```