

实验七 huffman 与树

一、 实验目的

- 1、熟悉 huffman 树;
- 2、熟悉 huffman 编码与解码;
- 2、二叉树的深入理解与应用。

二、 定义数据结构

huffman 树

```
typedef struct {  
    WeightType weight;  
    int parent,lchild,rchild;  
}HTNode,*HuffmanTree;  
  
typedef HTNode *HuffTree
```

三、 实验内容

- 1、根据给定的权值数组创建哈夫曼树并进行编码与解码.

//建树

```
void HuffmanTree(HuffTree &HT, WeightType *w, int n);
```

//从哈夫曼树 HT 上求得叶子节点的哈夫曼编码并存入数组 HC 中，
并打印出每个叶子节点对应的编码

```
void HuffmanCoding(HuffTree HT,char * * &HC,int n);
```

//哈夫曼译码，对已编码的字符数组进行译码，并显示结果

```
void HuffmanDecoding(HuffTree HT, int n);
```

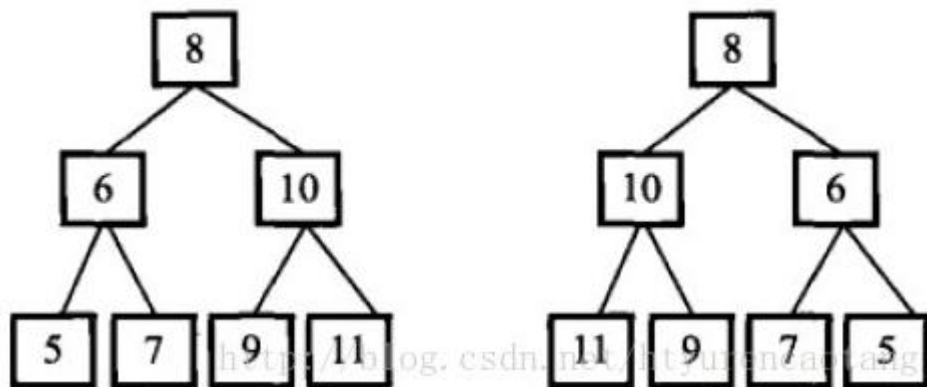
测试： 6 种字符 A,B,C,D,E,F 在报文中出现的次数分别是：

4,3,7,5,1,6。

对 110111101 进行解码，结果为 ABF

2、求二叉树的镜像

例如：

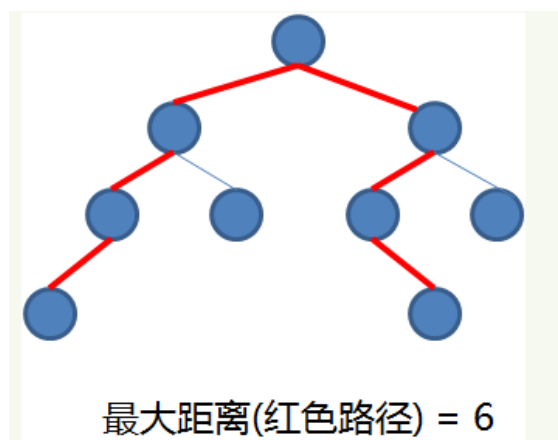


测试数据（先序序列）8 6 5 ## 7 ## 10 9 ## 11 ##（#代表空结点）

输出结果（先序序列）8 10 11 ## 9 ## 6 7 ## 5 ##（可不带#号）

3、（附加题）求二叉树中相距最远的两个节点之间的距离。

例如：



注意会存在最大距离不通过根节点，仅位于左子树或者右子树的情况。

测试数据（先序序列） 6 4 2 3 ##### 5 1 ## 7 ##（#代表空结点），输出 **5**。（非图中树对应数据）