

# 赫夫曼树(Huffman Tree)

---

## 一、 基本介绍

- 1) 给定  $n$  个权值作为  $n$  个叶子结点，构造一棵二叉树，若该树的带权路径长度 (wpl) 达到最小，称这样的二叉树为最优二叉树，也称为哈夫曼树 (Huffman Tree)，还有的书翻译为霍夫曼树。
- 2) 赫夫曼树是带权路径长度最短的树，权值较大的结点离根较近

## 二、 赫夫曼树几个重要概念和举例说明

- 1) 路径和路径长度：在一棵树中，从一个结点往下可以达到的孩子或孙子结点之间的通路，称为路径。通路中分支的数目称为路径长度。若规定根结点的层数为 1，则从根结点到第  $L$  层结点的路径长度为  $L-1$
- 2) 结点的权及带权路径长度：若将树中结点赋给一个有着某种含义的数值，则这个数值称为该结点的权。结点的带权路径长度为：从根结点到该结点之间的路径长度与该结点的权的乘积
- 3) 树的带权路径长度 (WPL)：树的带权路径长度规定为所有叶子结点的带权路径长度之和，记为 WPL (weighted pathlength)，权值越大的结点离根结点越近的二叉树才是最优二叉树。

公式中， $W_k$  为第  $k$  个叶子结点的权值； $L_k$  为该结点的路径长度。

$$WPL = \sum_{k=1}^n w_k l_k$$

4) WPL最小的就是赫夫曼树

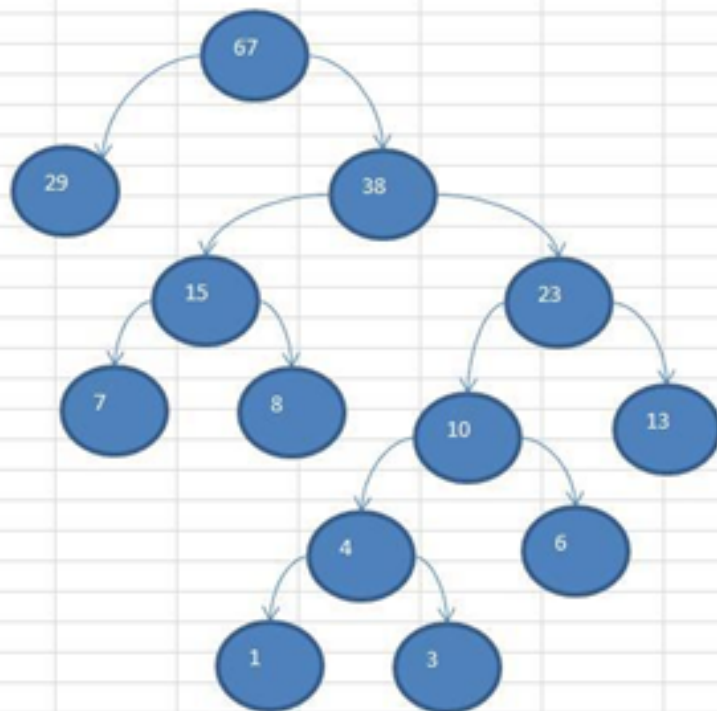
### 三、构成赫夫曼树的步骤：

1) 从小到大进行排序，将每一个数据，每个数据都是一个节点，每个节点可以看成是一颗最简单的二叉树

2) 取出根节点权值最小的两颗二叉树

3) 组成一颗新的二叉树，该新的二叉树的根节点的权值是前面两颗二叉树根节点权值的和

4) 再将这颗新的二叉树，以根节点的权值大小再次排序，不断重复 1-2-3-4 的步骤，直到数列中，所有的数据都被处理，就得到一颗赫夫曼树



67, 29,