

## 平衡二叉树

当且仅当两个子树的高度差不超过1时，这个树是平衡二叉树。（同时是排序二叉树）

## 满二叉树

一棵深度为k，且有 $2^k - 1$ 个节点的二叉树 特点：每一层上的结点数都是最大结点数

## 完全二叉树

若设二叉树的高度为h，除第h层外，其它各层(1~h-1)的结点数都达到最大个数，第h层从右向左连续缺若干结点，这就是完全二叉树。

## 二叉查找树

(Binary Search Tree)，（又：二叉搜索树，二叉排序树）

若它的左子树不空，则左子树上所有结点的值均小于它的根结点的值；若它的右子树不空，则右子树上所有结点的值均大于它的根结点的值；它的左、右子树也分别为二叉排序树。

## 最优二叉树（哈夫曼树）

通过刚才的步骤，我们可以得出构造赫夫曼树的赫夫曼算法描述。

1. 根据给定的  $n$  个权值  $\{w_1, w_2, \dots, w_n\}$  构成  $n$  棵二叉树的集合  $F = \{T_1, T_2, \dots, T_n\}$ ，其中每棵二叉树  $T_i$  中只有一个带权为  $w_i$  根结点，其左右子树均为空。
2. 在  $F$  中选取两棵根结点的权值最小的树作为左右子树构造一棵新的二叉树，且置新的二叉树的根结点的权值为其左右子树上根结点的权值之和。
3. 在  $F$  中删除这两棵树，同时将新得到的二叉树加入  $F$  中。
4. 重复 2 和 3 步骤，直到  $F$  只含一棵树为止。这棵树便是赫夫曼树。

# 红黑树

红黑树的五个性质：

- 1) 每个结点要么是红的，要么是黑的。
- 2) 根结点是黑的。
- 3) 每个叶结点，即空结点 (NIL) 是黑的。
- 4) 如果一个结点是红的，那么它的俩个儿子都是黑的。
- 5) 对每个结点，从该结点到其子孙结点的所有路径上包含相同数目的黑结点。

