# 树

* 1. 完全二叉树

对于有n个节点的完全二叉树，给所有节点按顺序从1开始编号。

对于任何序号为i的节点有：

* 如果i>1,则序号为i的节点的的父节点序号为i/2;
* 如果2i <= n, 则序号为i的节点的的左子节点序号为2i;
* 2i + 1 <= n，则序号为i的节点的的右子节点序号为2i + 1;
  1. 二叉查找树

二叉查找树（Binary Search Tree），（又：[二叉搜索树](http://baike.baidu.com/view/389453.htm" \t "_blank)，二叉排序树）它或者是一棵空树，或者是具有下列性质的[二叉树](http://baike.baidu.com/view/88806.htm" \t "_blank)： 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](http://baike.baidu.com/view/647462.htm" \t "_blank)。

* 1. 平衡二叉树

[平衡二叉搜索树](http://baike.baidu.com/view/2315698.htm)（Balanced Binary Tree，BST）具有以下性质：它是一棵空树或它的左右两个子树的高度差的绝对值不超过1，并且左右两个子树都是一棵平衡二叉树。常用算法有红黑树、AVL、 Treap、伸展树等。在平衡二叉搜索树中，我们可以看到，其高度一般都良好地维持在O（log2n），大大降低了操作的时间复杂度。

* + 1. AVL树

自平衡二叉查找树

* + 1. 红黑树