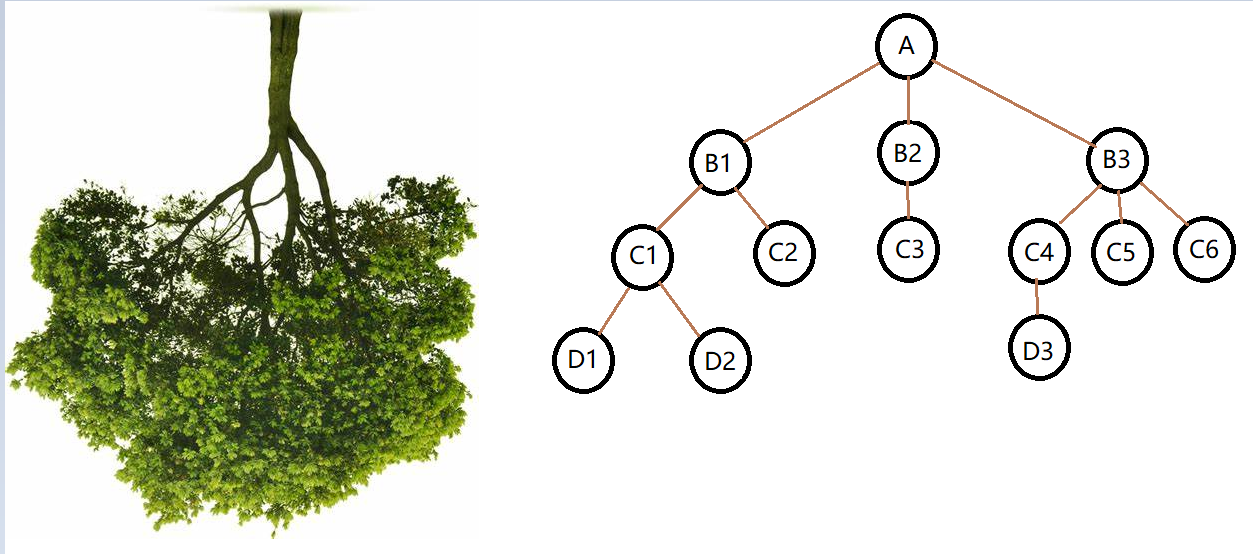
# 二叉树



## 树结构常见概念

度：结点拥有的子树的数量被称作结点的度。

叶子结点：度为零的结点被称作叶子节点，也叫终端结点。

分支结点：树中除了叶子结点以外的其他结点被称作分支结点，也叫非终端结点。

根结点：根结点是特殊的分支结点，根结点没有父结点。

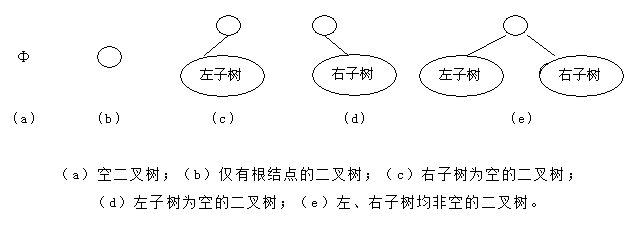
树的度：树内部各结点度的最大值被称作树的度。

树的高度：树中结点的最大层次被称为树的高度，也叫树的深度。

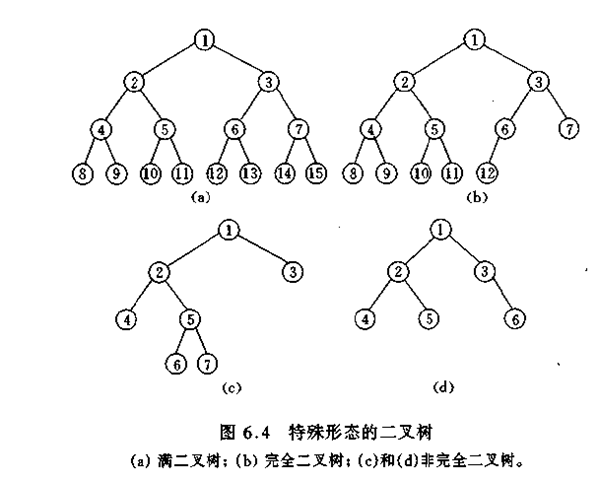
有序树：如果将树中结点的子树看作从左到右有序（即不可交换），则称该树是有序树。

## 二叉树定义

二叉树是一种特殊的有序树，树中的结点的度数不大于2，即树中的结点最多只有两棵子树。二叉树的5种形式如下图所示：



## 特殊二叉树



a：满二叉树 b：完全二叉树 c、d：普通二叉树

满度二叉树和完全二叉树是两种特殊形态的二叉树：若二叉树的高度为h，且树的总结点数为2h - 1,则称该二叉树为满二叉树，也称满度二叉树，该树的特点是，每一层的节点数都达到最大值。对于深度为K的，有n个结点的二叉树，当且仅当其每一个结点都与深度为K的满二叉树中编号从1至n的结点一一对应时称之为完全二叉树。

## 二叉树的性质

**性质1二叉树第i层最多有2i-1个结点（i>=1）**

等比数列通项公式：an = a1·qn-1

**性质2**  **高度为h的二叉树最多有2h -1个结点（h>=1）**

等比数列求和公式：Sn = a1(1-qn)/(1-q)

**性质3**  **对任何一棵二叉树，n0 = n2 + 1（n0、n1、n2分别表示树中度为0、1、2的结点的数量）**

（1）结点总数：

    树的结点总数n = n0 + n1 + n2

（2）分支总数：

树的分支总数 = n - 1 = n1 + 2n2  => n = n1 + 2n2 + 1（树中除了根结点，其他所有结点都有一个分支进入，所以树的分支总数等于结点总数减1；又因为所有的分支都是由度为1和度为2的结点射出，所有树的分支总数又等于n1 + 2n2）

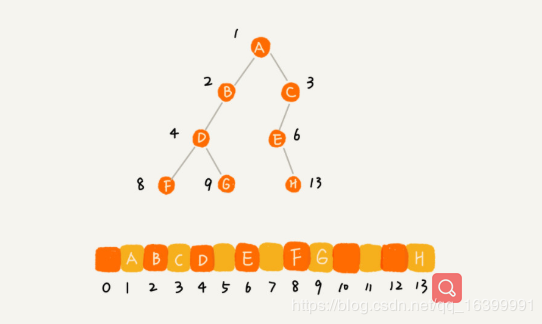
（3）由（1）（2）得，n0 = n2 + 1

**性质4** **完全二叉树中编号为i的结点，其左孩子编号为2i，右孩子编号为2i+1，父结点编号为i/2**

## 实现方式

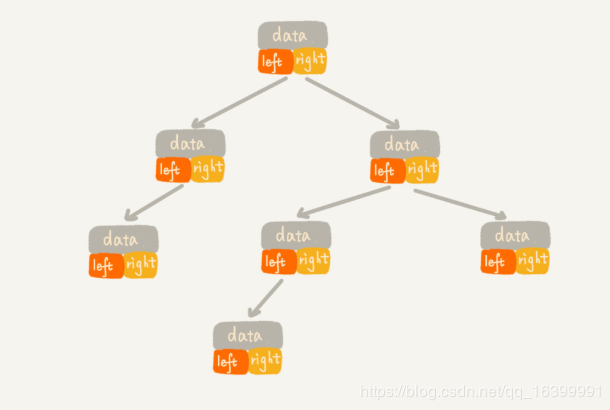
### 数组实现

利用二叉树的性质“完全二叉树中编号为i的结点，其左孩子编号为2i，右孩子编号为2i+1，父结点编号为i/2”，可以通过下标方便地访问结点的子结点和父结点。



数组实现方式可以快速的访问结点的父节点和子结点，但对于非完全二叉树，使用数组实现将浪费大量的存储空间，若二叉树是完全二叉树，使用数组实现是比较节省空间的，因为不需要使用额外的空间存储指针。

### 链式实现



## 遍历方式

