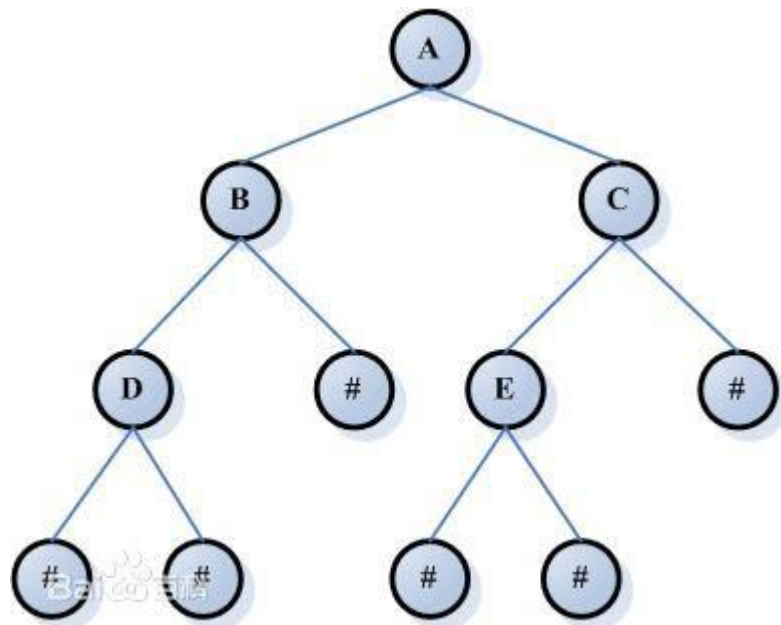


## 什么是二叉树

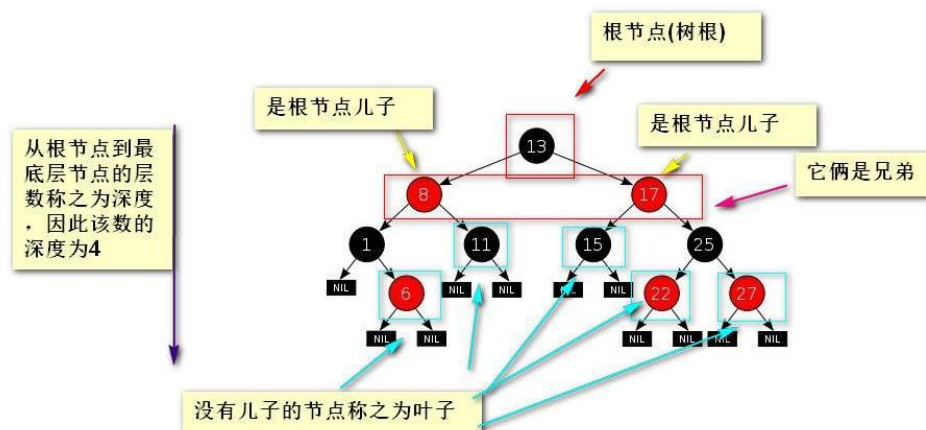
二叉树是树的一种，每个节点最多可具有两个子树，即结点的度最大为 2（结点度：结点拥有的子树数）。

例：



二叉树.png

树的一些概念

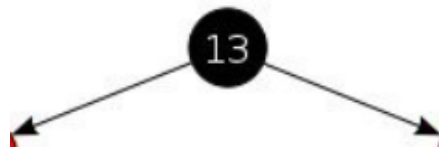


二叉树就是每个节点不能多于有两个儿子，上面的图就是一颗二叉树，而且还是一种特殊的二叉树：二叉查找树(binary search tree)。

- 定义：当前根节点的左边全部比根节点小，当前根节点的右边全部比根节点大。
  - 可以看出，这对我们找一个数是非常方便快捷的

一棵树至少会有一个节点(根节点)

树由节点组成，每个节点的数据结构是这样的：

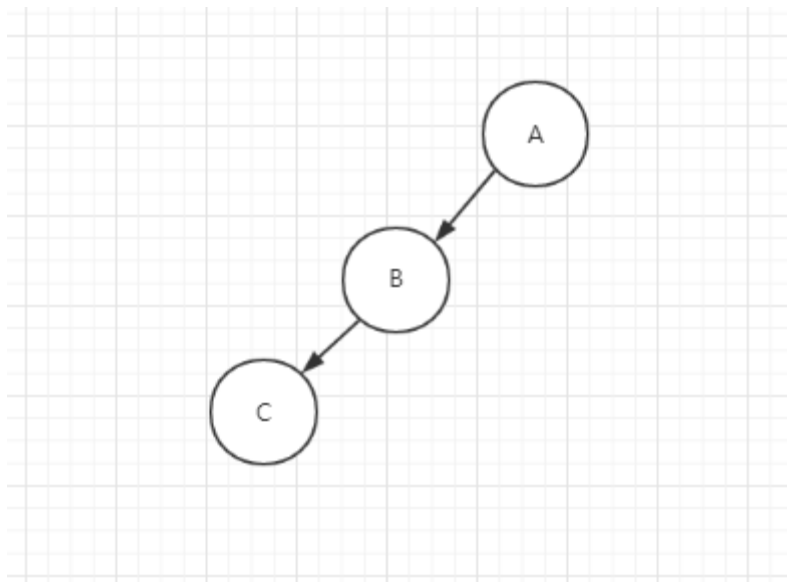


因此，我们定义树的时候往往是\*\*->定义节点->节点连接起来就成了树\*\*，而节点的定义就是：一个数据、两个指针(如果有节点就指向节点、没有节点就指向 null)

## 2. 二叉树的种类

### 2.1 斜树

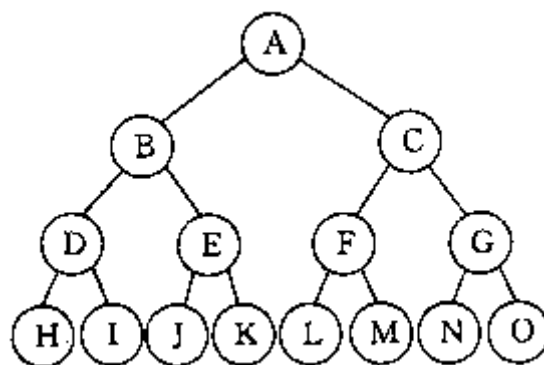
所有结点都只有左子树，或者右子树。



左斜树.png

### 2.2 满二叉树

所有的分支节点都具有左右节点。



满二叉树.png

## 2.3 完全二叉树

若设二叉树的深度为  $h$ ，除第  $h$  层外，其它各层 ( $1 \sim h-1$ ) 的结点数都达到最大个数，第  $h$  层所有的结点都连续集中在最左边，这就是完全二叉树。

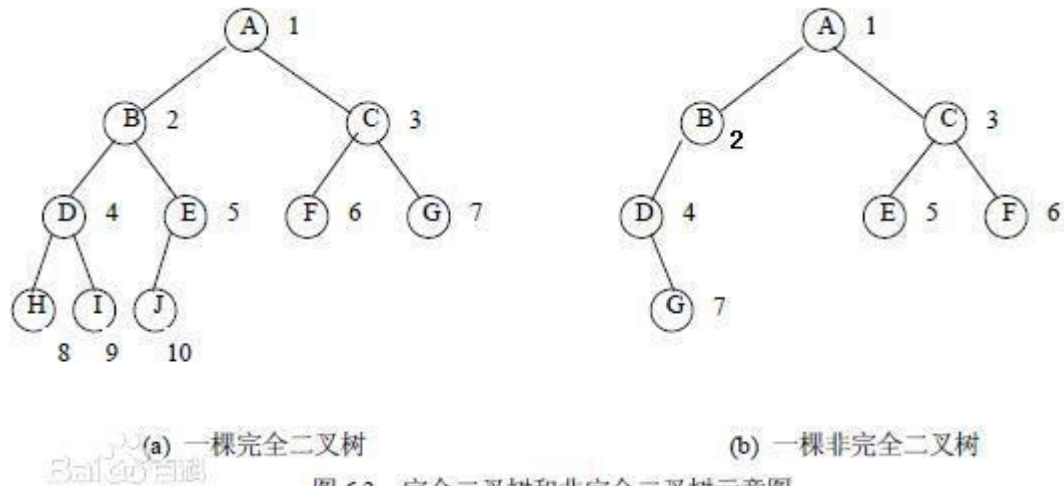


图 6.3 完全二叉树和非完全二叉树示意图

完全二叉树与非完全二叉树.png

## 3. 二叉树的一些性质

二叉树第  $i$  层上的结点数最多为  $2^{i-1}$  ( $i \geq 1$ )

深度为  $h$  的二叉树至多有  $2^h - 1$  个结点 ( $h \geq 1$ )

包含  $n$  个结点的二叉树的高度至少为  $\log_2(n+1)$

在任意一棵二叉树中，若终端结点的个数为  $n_0$ ，度为 2 的结点数为  $n_2$ ，则  $n_0 = n_2 + 1$

## 4. 二叉树的遍历方式

二叉树的遍历方式，一般分为先序遍历，中序遍历，后序遍历。

- 先序遍历
  - 先访问根节点，然后访问左节点，最后访问右节点(根->左->右)
- 中序遍历
  - 先访问左节点，然后访问根节点，最后访问右节点(左->根->右)
- 后序遍历
  - 先访问左节点，然后访问右节点，最后访问根节点(左->右->根)

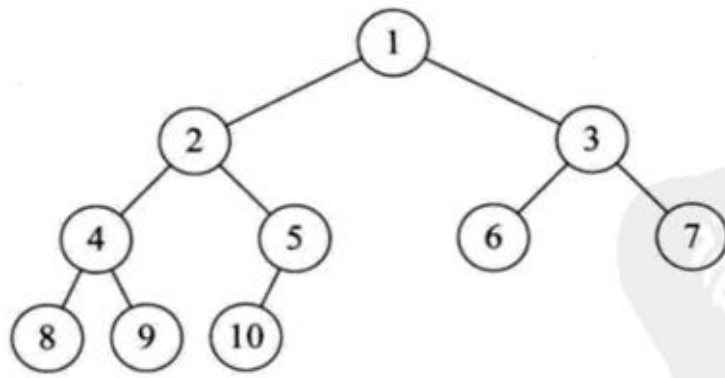


image.png

先序遍历(根-左-右):1-2-4-8-9-5-10-3-6-7

中序遍历:(左-根-右):8-4-9-2-10-5-1-6-3-7

后序遍历(左-右-根):8-9-4-10-5-2-6-7-3-1