

The background features several overlapping circles in various shades of blue, ranging from light cyan to a deep, vibrant blue. The circles are positioned on the left side of the frame, creating a modern, abstract design.

软件设计师

--数据结构基础

高级项目经理 任铄

QQ: 1530841586

第一章 数据结构基础

- 1.1 线性表
- 1.2 树和二叉树
- 1.3 图
- 1.4 排序
- 1.5 查找

一、树

树是 $n(n \geq 0)$ 个结点的有限集合， $n=0$ 时称为空树，在任一非空树中

- 有且仅有一个称为根的结点。
- 其余的结点可分为 $m(m \geq 0)$ 个互不相交的子集 T_1, T_2, \dots, T_m ，其中每个子集本身又是一棵树，并称其为根结点的子树。

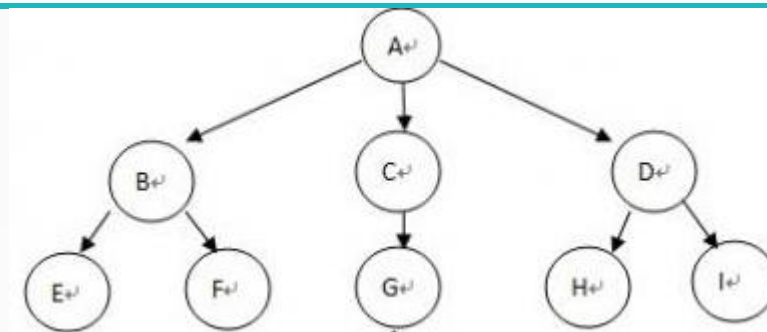
高级项目经理 任铄

QQ: 1530841586

向上人生路!

1、树的基本概念

- 双亲和孩子
- 兄弟：具有相同双亲的结点互为兄弟。
- 结点的度：一个结点的子树的个数记为该结点的度。
- 树的度：树中各结点的度的最大值
- 叶子结点：也称为终端结点，指度为零的结点。
- 内部结点：度不为零的结点称为分支结点或非终端结点。除根结点之外，分支结点也称为内部结点。

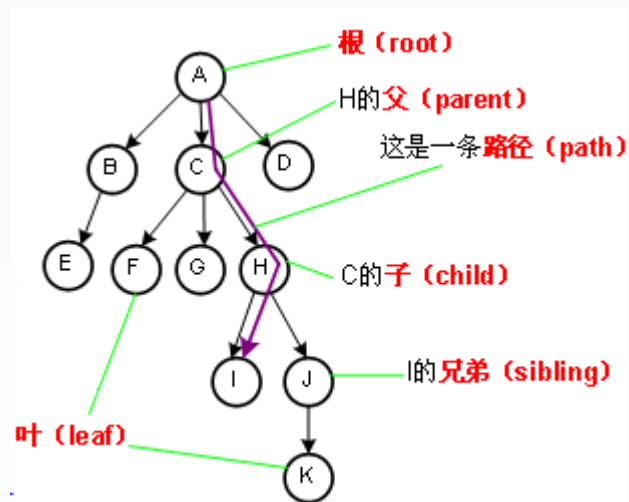


高级项目经理 任铄

QQ: 1530841586

向上人生路!

- 结点的层次：根为第一层，根的孩子为第二层，依此类推。
- 树的高度：一棵树的最大层次数记为树的高度（或深度）。
- 有序（无序）树：若将树中的结点的各子树看成是从左到右具有次序的，即不能交换，则称该树为有序树，否则称为无序树。
- 森林：是 $m(m \geq 0)$ 棵互不相交的树的集合



高级项目经理 任铄
QQ: 1530841586

2、树的存储结构

- 标准存储结构
 - ✓ 结点的数据
 - ✓ 指向子结点的指针
- 带逆存储结构
 - ✓ 结点的数据
 - ✓ 指向子结点的指针
 - ✓ 指向其父结点的指针

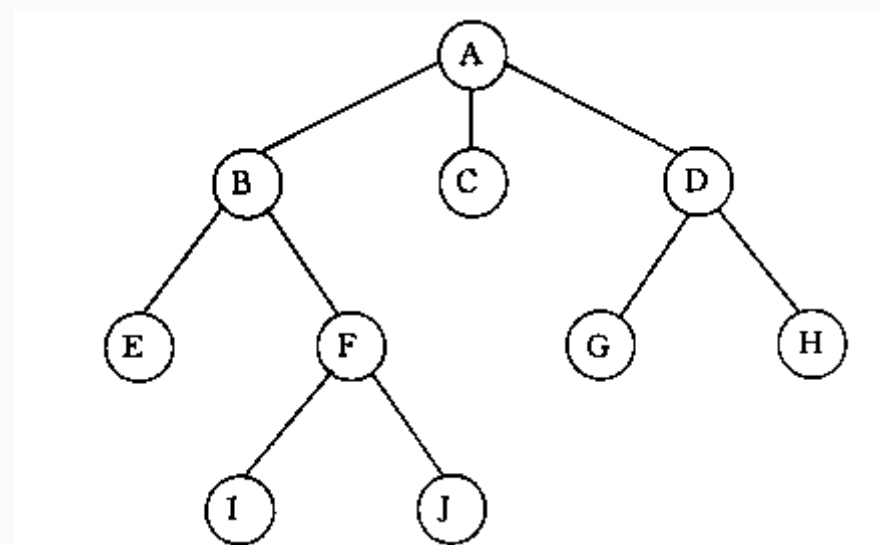
高级项目经理 任铄

QQ: 1530841586

3、树的遍历

遍历是指对树中所有结点信息的访问，即依次对树中每个结点访问一次且仅访问一次。

- 前序遍历A B E F I J C D G H
- 后序遍历E I J F B C G H D A
- 层次遍历A B C D E F G H I J



高级项目经理 任铄

QQ: 1530841586

高级项目经理 任铄

QQ: 1530841586

二、二叉树

二叉树(BinaryTree)是 $n(n \geq 0)$ 个结点的有限集合，它或者是空树($n=0$)，或者是由一个根结点及两棵互不相交的、分别称为左子树和右子树的二叉树所组成。

二叉树与树的区别：

- 二叉树的结点的最大度为2，而树中不限制结点的度。
- 二叉树的结点的子树要区分左子树和右子树

1、二叉树的性质

- (1) 二叉树第 i 层上的结点数目最多为 2^{i-1} ($i \geq 1$)。
- (2) 深度为 k 的二叉树至多有 $2^k - 1$ 个结点 ($k \geq 1$)。
- (3) 在任意一棵二叉树中，若终端结点数为 n_0 ，度为2的结点数为 n_2 ，则 $n_0 = n_2 + 1$ 。
- (4) 具有 n 个结点的完全二叉树的深度为 $\lfloor \log_2 n \rfloor + 1$ 。

高级项目经理 任铄

QQ: 1530841586

(5)对一棵有 n 个结点的完全二叉树的结点按层次自左至右进行编号，则对任一结点 i 有：

- 若 $i=1$ ，则结点 i 是二叉树的根，无双亲，若 $i>1$ ，则其双亲为 $\lfloor i/2 \rfloor$ 。
- 若 $2i>n$ ，则结点 i 无左孩子，否则其左孩子为 $2i$ 。
- 若 $2i+1>n$ ，则结点 i 无右孩子，否则其右孩子为 $2i+1$ 。

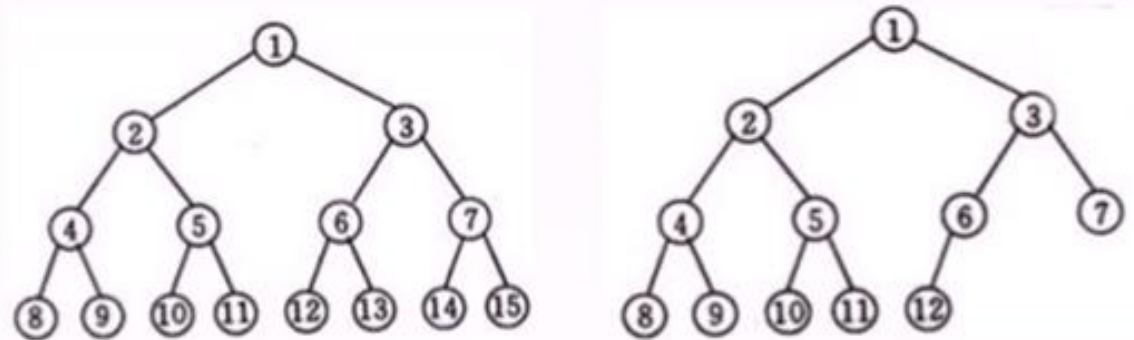
高级项目经理 任铄

QQ: 1530841586

向上人生路!

若深度为 k 的二叉树有 2^k-1 个结点，则称其为**满二叉树**。

深度为 k 、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 的满二叉树编号从1至 n 的结点一一对应时，称之为**完全二叉树**。



高级项目经理 任铄
QQ: 1530841586

2、二叉树的存储结构

高级项目经理 任铄

QQ: 1530841586

(1)顺序存储结构

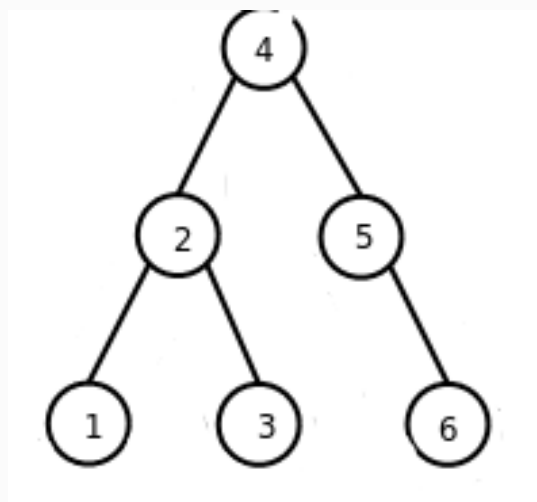
对完全二叉树既简单又节省空间，而对于一般二叉树则不适用。

(2)链式存储结构

由于二叉树中结点包含有数据元素、左子树根、右子树根及双亲等信息，因此可以用三叉链表或二叉链来存储二叉树。链表的头指针指向二叉树的根结点。

3、二叉树的遍历

- 前序遍历 4 2 1 3 5 6
- 中序遍历 1 2 3 4 5 6
- 后序遍历 1 3 2 6 5 4



高级项目经理 任铄
QQ: 1530841586

通过二叉树的两种序列，可唯一确定这棵树，并得出另一序列。
例：已知二叉树前序遍历是ABHFDECKG，中序遍历序列是
HBDFAEKCG，它的后序遍历序列是(B)。

- A、 BHFDECKGA B、 HDFBKGCEA
C、 HFDBKCGEA D、 ABEHFCDKG

高级项目经理 任铄
QQ: 1530841586

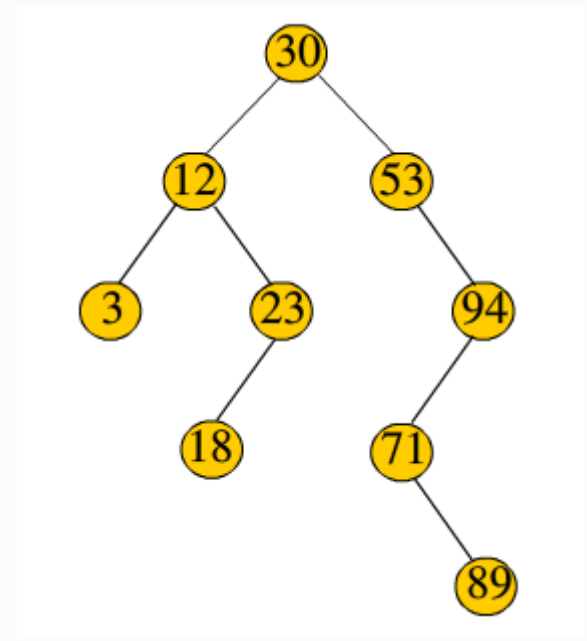
三、二叉排序树

又称为二叉查找树，定义：或者是一棵空树，或者是具有下列性质的二叉树：

(1) 若左子树不空，则左子树上所有结点的值均小于它的根结点的值；

(2) 若右子树不空，则右子树上所有结点的值均大于或等于它的根结点的值；

(3) 左、右子树也分别为二叉排序树；

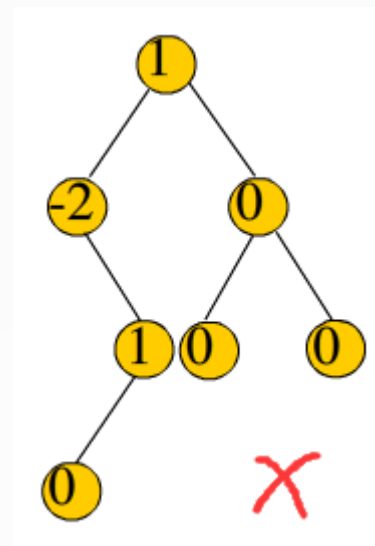
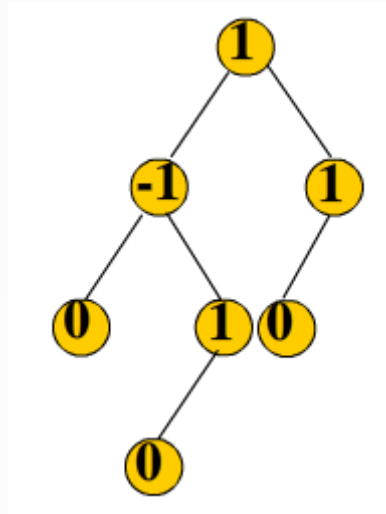


高级项目经理 任铄

QQ: 1530841586

四、平衡二叉树

又被称为AVL树，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。



五、线索树

n 个结点的二叉链表中含有 $n+1$ ($2n-(n-1)=n+1$)个空指针域。利用二叉链表中的空指针域，存放指向结点在某种遍历次序下的前趋和后继结点的指针（这种附加的指针称为“线索”）。

高级项目经理 任铄

QQ: 1530841586

六、最优二叉树

给定 n 个权值作为 n 的叶子结点，构造一棵二叉树，若带权路径长度达到最小，称这样的二叉树为最优二叉树，也称为哈夫曼树(Huffman tree)。哈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。

高级项目经理 任铄

QQ: 1530841586

可以通过下列渠道沟通联系：

- 1、QQ:1530841586
- 2、QQ群：164955673