

全国计算机等级考试三级数据库考点分析之数据结构算法[3]
PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/221/2021_2022__E5_85_A8_

[E5_9B_BD_E8_AE_A1_E7_c98_221507.htm](https://www.100test.com/kao_ti2020/221/2021_2022__E5_85_A8_) 计算机等级考试训练软件《百宝箱》 考点14树与二叉树之间的转换 1.树转换成二叉树 将一棵树转换成相应的二叉树应包括以下几个步骤：

(1)在兄弟节点之间加条连线 (2)对每一个节点，只保留它与第一个子节点的连线，与其他子节点连线全部抹掉 (3)以树根为轴心，顺时针旋转45°。 2.森林转换成二叉树 如

果 $F = \{T_1, T_2, \dots, T_m\}$ 是森林，则可按如下规则将其转换成一棵二叉树 $B = (\text{root}, LB, RB)$ ：(1)若 F 为空，即 $m = 0$ ，则 B 为树。(2)若 F 非空，即 $m > 0$ ，则 B 的根 root 即为森林中的第一棵树的根 $\text{ROOT}(T_1)$ ； B 的左子树 LB 是从 T_1 中根节点的子树森林 $F_1 = \{T_{11}, T_{12}, \dots, T_{1m}\}$ 转换而成的二叉树；其右子树 RB 是从森林 $F = \{T_2, T_3, \dots, T_m\}$ ：(1)若 B 为空，则 F 为空。(2)若 B 非空，则 F 中第一棵树 T_2 的根 $\text{ROOT}(T_2)$ 即为二叉树 B 的根 root 。 T_2 中根节点的子树森林 F_2 是由 B 的左子树 LB 转换而成的； F 中除 T_2 之外其余树组成的森林 $F = \{T_3, T_4, \dots, T_m\}$ 是由 B 的右子树 RB 转换而成的。

考点15二叉树和树的周游 周游(或称遍历)是树型结构的一种重要运算，是其他运算的基础。周游一棵树就是按一定的次序访问树中所有节点，并且每个节点仅被访问一次的过程。 1.周游二叉树 二叉树的周游主要有以下3种方式。(1)前序法(NLR)。访问根，按前序周游左子树，按前序周游右子树。(2)后序法(LRN)。按后序周游左子树，按后序周游右子树，访问根。(3)对称序法(LNR)。按对称序周游左子树，访问根，按对称序周游右子树。 2.周游树和树林

对树和树林的周游分为按深度优先和按广度优先两种方式进行。按深度优先方式又可分为按先根次序和按后根次序周游两种方式。(1)先根次序周游访问第一棵树的根，按先根次序周游第一棵树的根子树，按先根次序周游其他的树。(2)后根次序周游按后根次序周游第一棵树的子树，访问第一棵树的根，按后根次序周游其他的树。比较一下树与二叉树之间的对应关系，可以看出，按先根次序周游树正好与按前序法周游树对应的二叉树等同，后根次序周游树正好与按对称序法周游对应的二叉树等同。按广度优先方式可以做层次次序周游，首先依次访问层数为0的节点，然后依次访问下一层的节点，直至访问完最后一层的节点。

考点16 二叉树的存储和线索
二叉树的llink—rlink法存储表示 二叉树的存储通常采用链接方式，即每个节点除存储节点自身的信息外再设置两个指针域llink和 rlink，分别指向节点的左子女和右子女。当节点的某个子女为空时，则相应的指针值为空。再加上一个指向树根的指针t，就构成了二叉树的存储表示。这种存储表示法被称为llink - rlink表示法。

2. 线索二叉树 在有n个节点的二叉树的llink - rlink法存储表示中，必定有n+1个空指针域，将这些指针位置利用起来，存储节点在指定周游次序F的前驱、后继节点指针，则得到线索二叉树。

考点17 哈夫曼树 哈夫曼树是树型结构的一种应用二哈夫曼(Huffman)树又称最优树，是一类带权路径长度最短的树，这种树在信息检索中经常用到。所谓路径长度就是从从一个节点到另一个节点所经过的分支总数。树的路径长度是从树的根到每个节点的路径长度之和。完全二叉树就是这种路径长度最短的二叉树。节点的带权路径长度为从该节点到树根之间的路径长度与节点上权的乘

积。树的带权路径长度为树中所有叶子节点的带权路径长度之和，WPL最小的不是完全二叉树，而是权大的叶子离根最近的二叉树。

2.5 “查找”

查找是数据结构中一种很常用的基本运算。查找就是在数据结构中找出满足某种条件的节点。所给的条件可以是关键码字段的值，也可以是非关键码字段的值，本节只考虑基于关键码值的查找。

考点18 顺序查找

顺序查找是线性表的最简单、最基本的查找方法。顺序查找的优点是对线性表节点的逻辑次序无要求，对线性表存储结构也无要求。顺序查找的缺点是速度较慢，平均检索长度与表中节点个数和 n 成正比，查找成功最多需比较 n 次，平均查找长度为 $(n+1)/2$ 次，约为表长的半，查找失败需比较 $n+1$ 次。顺序查找算法的时间复杂度为 $O(n)$ 。

考点19 二分法查找

二分法查找是一种效率比较高的查找方法，在进行二分法查找时，线性表节点必须按关键码值排序，且线性表是以顺序存储方式存储的。二分法查找的优点是比较次数少，查找速度快，平均检索长度小，经过 $\lceil \log_2 n \rceil$ 次比较就可以完成查找过程。缺点是在查找之前要为建立有序表付出代价，同时对有序表的插入和删除都需要平均比较和移动表中的一半元素。一般情况下，二分查找适应于数据相对固定的情况，且二分法查找只适用于线性表的顺序存储。

考点20 分块查找

分块查找又称索引顺序查找，是介于线性查找和二分法查找之间的一种查找方法。它要求把线性表分成若干块，每一块中的节点不必有序，但块与块之间必须排序，不妨设每一块中各节点的关键码都大于前一块的最大关键码值。另外，还要求将各块中的最大关键码值组成一个有序的索引表。满足上述条件的线性表称做分块有序表。它的分块检索的过程分以下两步进行。(I)先

查索引表(可以用线性检索或二分法检索), 确定要找的记录在哪一块。(2)在相应的块中线性检索待查记录。 100Test 下载频道开通, 各类考试题目直接下载。详细请访问 www.100test.com