

数据结构习题

2023年11月26日 21:15

一、绪论练习

- (3) **数据结构**是一个二元组 $\text{Data_Structure} = (D, R)$ ，其中，D 是数据元素的有限集，R 是 D 上关系的有限集。

7. 线性表的顺序存储结构是一种 (**A**) 的存储结构，线性表的链式存储结构式一种 (**B**) 存储结构。

- A. 随机存取 B. 顺序存取
C. 索引存取 D. 散列存取

二、线性表

1. 线性表的逻辑顺序和存储顺序总是一致的 (×)

- 解析: 只有顺序存储情况下才一致, 链式存储因为是随机选择物理存储单元, 所以不一致。

(3) **插入运算**: 在第 i 个位置上插入 x, 从 a_n 到 a_i 都要向下移动一个位置, 共移动 $n-i+1$ 个元素。

- 等概率情况下, 平均移动数据元素的次数: $E_{in} = \sum_{i=1}^{n+1} p_i (n-i+1) = \frac{1}{n+1} \sum_{i=1}^{n+1} (n-i+1) = \frac{n}{2}$

例: 已知表 L

(1) 表首插入 S 结点的语句序列是 6, 3。

(2) 表尾插入 S 结点的语句序列是 2, 9, 1, 7

- ① $P \rightarrow \text{next} = S;$ ② $P = L;$ ③ $L = S;$
④ $P \rightarrow \text{next} = S \rightarrow \text{next};$ ⑤ $S \rightarrow \text{next} = P \rightarrow \text{next};$
⑥ $S \rightarrow \text{next} = L;$ ⑦ $S \rightarrow \text{next} = \text{NULL};$

注意: 在表头插入之后需要更新表头, 在表尾插入之后需要将表尾置空

三、栈和队列

16. 用不带头结点的单链表存储队列, 其队头指针指向队头结点, 对尾指针指向队尾结点, 则在进行出队操作时 (**D**)。

- A. 仅修改队头指针 B. 仅修改队尾指针 当队列之中只有一个元素时
C. 队头、队尾指针都要修改 D. 队头、队尾指针都可能要修改 删除时需要修改队尾指针

7. 中缀表达式 $A-(B+C/D)*E$ 的后缀形式是 (**D**)。

- A. $AB-C+D/E*$ B. $ABC+D/E*$
C. $ABCD/E*+-$ D. $ABCD/+E*-$

- 9. 在一个循环队列中, 队首指针指向队首元素的 前一个位置。
队首指针指向实际队首还是队首前空位, 这是要看事先约定的。一般有两种方案: 方案一、将队首指针指向实际队首, 队尾指针指向队尾的下一个空位; 方案二、将队首指针指向队首前一个空位, 队尾指针指向实际队尾。(此处应该就是指方案二)
 - 在循环队列中, 为了避免判空与判满时出现歧义, 其中之一要指向空位, 而不是两个都指向实际的首尾。

四、树

8.在一棵有 n 个结点的二叉树中，若度为 2 的结点数为 n_2 ，度为一的结点数为 n_1 ，度为 0 的结点数为 n_0 ，则树的最大高度为 (E)，其叶结点数为 G；树的最小高度为 B，其叶结点数为 (G)；若采用链表存储结构，则有 (D) 个空链域。

- A. $n/2$ B. $\lceil \log_2 n \rceil$ C. $\log_2 n$ D. n
E. $n_0 + n_1 + n_2$ F. $n_1 + n_2$ G. $n_2 + 1$ H. 1
I. $n + 1$ J. n_1 K. n_2 L. $n_1 + 1$

- 最大高度时树为单支树，应该为结点数。（感觉D,E都可以）
- 对于二叉树来说根据度数相等可以列出方程 $n_0 + n_1 + n_2 - 1 = 0 \cdot n_0 + 1 \cdot n_1 + 2 \cdot n_2 \implies n_0 = n_2 + 1$ （第二空感觉G,H均可）
- 完全二叉树时高度最小

10. 设高度为 h 的二叉树中只有度为 0 和度为 2 的结点，则此类二叉树中所包含的结点数至少为 B，至多为 F。

- A. $2h$ B. $2h - 1$ C. $2h + 1$ D. $h + 1$ E. 2^{h-1} F. $2^h - 1$ G. $2^{h+1} + 1$ H. 2^{h+1}

各括号顺序选择 B,F。对于只有度为 0 和度为 2 的结点的二叉树，在高度 h 固定时，让每一层结点达到最少，就可使二叉树的结点总数达到最少，此时，除第 1 层一个结点之外，其他 $h-1$ 层各有 2 个结点，所以二叉树至少有 $2(h-1) + 1 = 2h - 1$ 个结点。反之，在高度 h 固定时，让每一层结点达到最多，就可使二叉树的结点总数达到最多，这就是满二叉树情形，结点个数可达 $2^h - 1$ 。

13. 一棵有 124 个叶结点的完全二叉树，最多有 () 个结点。

- A. 247 B. 248 C. 249 D. 250 E. 251

- 一棵124个叶结点的完全二叉树，假设 n_0 为叶子结点数， n_1 为度为1结点数， n_2 为度为2结点数，则有总结点数为 $n_0 + n_1 + n_2$ ；而 $n_2 = n_0 - 1 = 123$ ；且完全二叉树中度为1的结点只能为一个或0个，所以总结点数为 $124 + 1 + 123 = 248$ 个

任何一棵二叉树的叶结点在先序，中序和后序遍历中的相对次序 (A)。

- A. 不发生改变 B. 发生改变 C. 不能确定 D. 以上都不对

- 相对次序发生变化的都是子树的根。无论是根左右、左根右、左右根，对于遍历来说都是先左后右

线索二叉树是一种物理结构（因为带了线索，暗含了只能通过指针存储等，所以是存储结构即物理结构）

22. 若二叉树采用二叉链表作存储结构，要交换其所有分支结点左右子树的位置，利用 () 遍历方法最合适。 C

- A. 前序 B. 中序 C. 后序 D. 层次

23. 欲实现任意二叉树的后序遍历的非递归算法而不必使用栈结构，最佳方案是二叉树采用

- (A) 存储结构。 三叉链表中多存了parent指针，指向父结点

- A. 三叉链表 B. 广义表 C. 二叉链表 D. 顺序

树	森林	二叉树
先根遍历	前序遍历	前序遍历
后根遍历	中序遍历	中序遍历

对于树的后根遍历来说可以用二叉树的中序遍历来实现

PS: 有序树指孩子之间有顺序之分

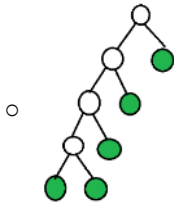
35. 若只考虑有序树的情形，则具有 7 个结点的不同形态的树共有 (A) 种。

- A. 132 B. 154 C. 429 D. 前三者均不正确

- 具有 n 个结点不同形态的树 和具有 $n-1$ 个结点不同形态的二叉树数量相同。因此本题转换为求 6 个结点不同形态的二叉树的数目。求二叉树的数目可以使用 卡克兰数 求解。下边为卡特兰数的公式。

$$f(n) = \frac{(2n)!}{n!(n+1)!}$$

- 设T是哈夫曼二叉树，具有5，个叶结点树T的高度最高可以是5



5. 一棵深度为 h 的满 k 叉树有如下性质：第 h 层上的节点都是叶子节点，其余各层上的每个节点都有 k 棵非空子树。

- 如果按层次顺序从 1 开始对全部结点编号，则第 i 层上的结点数目是____；编号为 n 的结点的双亲结点（若存在）的编号是____；编号为 n 的结点的第 i 个孩子结点（若存在）的编号是____，编号为 n 的结点有右兄弟的条件是____,其右兄弟的编号是____。
 - k^{i-1} , $\left\lceil \frac{n-2+k}{k} \right\rceil$, $(n-1)k+i+1$, $i \neq nk+1$, $n+1$
- 11. N 个结点的完全二叉树，按从上到下的，从左到右给结点顺序编号，则编号最大的非叶结点编号为 $\lfloor n/2 \rfloor$ ，编号最小的叶结点为 $\lfloor (n+1)/2 \rfloor$ 。
 - 结点数 $n=n_0+n_1+n_2=2*n_0-1+n_1 \Rightarrow n_0=(n+1-n_1)/2$, 又 $n_1=0$ 或 1 , 所以最多有 $(n+1)/2$ 个叶子结点, 最少有 $n/2$ 个。

- 设 F 是一个森林， B 是由 F 变换得的二叉树。若 F 中有 n 个非终端结点，则 B 中右指针域为空的结点有 $n+1$

1. 题目非终端结点为 n , 那么假设森林总结点为 m , 终端结点（即叶子结点）为 $m-n$, 总指针域就是 $2*m$ 。

2. 根据二叉树的特性，可知，在 m 个结点的二叉链表中，有 $m+1$ 个空指针域。

终端结点转化为二叉树后，该结点没有左孩子，左指针域就为空，即 $m-n$ 。

3. 所以右指针域为空的个数为： $m+1-(m-n)=n+1$ 。

○ 右指针域为空个数 = 空指针域个数 - 左指针域为空个数

五、图

14. 用 DFS 遍历一个无环有向图，并在 DFS 算法退栈返回时，打印出相应的顶点，则输出的顶点序列是 **A**。
- 退栈时先进后出**
- A.逆拓扑有序的 B.拓扑有序的 C.无序的

正确的 AOE 网而言，必须是 **C**，AOE 中，某边权值应当是 **D** 权值为零的边表示 **B**

- (1) A.完全图 B.哈密尔顿图 C.无环图 D.强连通图
- (2) A.实数 B.正整数 C.正数 D.非负数
- (3) A.为决策而增加的活动 B.为计算方便而增加的活动
- C.表示活动间的时间顺序关系 D.该活动为关键活动

六、排序

1. 下述几种排序方法中，平均查找长度最小的是 (**C**)。
- A. 插入排序 B. 选择排序 C. 快速排序 D. 归并排序

类别	排序方法	时间复杂度			空间复杂度	稳定性
		最好情况	最坏情况	平均情况	辅助存储	
插入排序	直接插入排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
	希尔排序	$O(n)$	$O(n^2)$	$\sim O(n^{1.3})$	$O(1)$	不稳定
交换排序	冒泡排序	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
	快速排序	$O(n \log n)$	$O(n^2)$	$O(n \log n)$	$O(n \log n)$	不稳定
选择排序	直接选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
	堆排序	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(1)$	不稳定
归并排序		$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	稳定
基数排序 k:待排元素的维数, m为基数的个数		$O(n+m)$	$O(k*(n+m))$	$O(k*(n+m))$	$O(n+m)$	稳定

- $O(n \log n)$:快速排序,归并排序,堆排序
- $O(n^2)$:直接插入,简单选择排序,冒泡排序
- $O(n)$:基数排序
- 基数排序的排序速度与数据的初始排列顺序无关
- 33. 在含有 n 个元素的小根堆（堆顶元素最小）中，关键字最大的记录可能存储在 $[n/2]+3$ 位置上。
 - 小根堆中最大的数一定是放在叶子节点上，堆本身是个完全二叉树，完全二叉树的叶子节点的位置大于 $[n/2]$ (这是个选择题选项忘记加进去)
- 6. n 个记录的冒泡排序算法所需最大移动次数为 $3n(n-1)/2$ ，最小移动次数为 0 。 感觉不会这样考
- 在内部排序中,平均比较次数最少的是快速排序,要求附加的内存容量最大的是归并排序,排序时不稳定的有希尔排序,快速排序,直接选择,堆排序.

考试题目:

- 在索引查找中，查找表中有 3069 个元素，共平均分为 31 块，如果索引表采用顺序查找，则成功查找的平均查找长度是 66。
 - 分成31块,每个块有 $3069/31=99$ 个元素,查找块的平均需要 $(31+1)/2=16$;块内查找元素需要 $(99+1)/2=50$,所以一共为 $50+16=66$
- 下述哪一条是顺序存储结构的优点 A。
 - A. 存储密度大 B. 插入运算方便
 - C. 删除运算方便 D. 方便各种逻辑结构的存储表示