

## 2021 级数据结构期末试卷回忆版

### 一、选择题

1. 在一个数组中查找一个元素的平均时间复杂度是 ( )。在一个有序的链表中, 插入一个元素, 使链表仍保持有序的平均时间度为 ( )  
A.  $O(n)$                       B.  $O(n\log n)$                       C.  $O(n^2)$                       D.  $O(1)$
2. 经过一趟冒泡排序后的结果
3. 权值序列  $W=\{ \}$ , 哈夫曼树带权路径和
4. 图  $G=\{ \langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 4 \rangle, \langle 1, 4 \rangle \}$ , 下列是拓扑排序的是 ( )  
A. 1234                      B. 2341                      C. 1423                      D. 1243
5. 设一棵  $m$  叉树中有  $N_1$  个度数为 1 的结点,  $N_2$  个度数为 2 的结点,  $\dots\dots$ ,  $N_m$  个度数为  $m$  的结点, 则该树中共有 ( ) 个叶子结点。

$$(A) \sum_{i=1}^m (i-1)N_i \quad (B) \sum_{i=1}^m N_i \quad (C) \sum_{i=2}^m N_i \quad (D) 1 + \sum_{i=2}^m (i-1)N_i$$

6. 对时间消耗为  $(3n + n\log_2 n + n^2 + 8)$ , 它的时间复杂度是 ( )。  
A.  $O(n)$   
B.  $O(n\log n)$   
C.  $O(n^2)$   
D.  $O(\log n)$
7. 图的深度优先遍历
8. 在链表中,  $P$  指向当前结点, 下列哪段代码可以删除  $p$  所在的结点 ( )  
A.  $q = p \rightarrow next; \quad p \rightarrow data = q \rightarrow data; \quad p \rightarrow next = q \rightarrow next; \quad free(q)$   
B.  $q = p \rightarrow next; \quad q \rightarrow data = p \rightarrow data; \quad p \rightarrow next = q \rightarrow next; \quad free(q)$   
C.  $q = p \rightarrow next; \quad p \rightarrow next = q \rightarrow next; \quad free(q)$   
D.  $q = p \rightarrow next; \quad q \rightarrow next = p \rightarrow next; \quad free(q)$

### 二、解答题

1. 一棵 AVL 树  $T$  中结点的关键码均为正整数(从 1 开始取值递加,  $n=1, 2, 3, 4 \dots$ ), 它有下列特点:  
(1) 左子树的高度不高于右子树;  
(2) 删除关键码为  $k_1$  的某个叶结点, 然后再插入关键码  $k_1$ , 得到的 AVL 树与原 AVL 树  $T$  不同;  
(3) 删除  $T$  中关键码为  $k_2$  的非叶结点, 然后再插入关键码  $k_2$ , 得到的 AVL 树与原 AVL 树  $T$  相同;  
(4) 往  $T$  中插入某个关键码  $k_3$ , 然后再删除  $k_3$ , 得到的 AVL 树与原 AVL 树  $T$  不同。  
画出具有上述特点且结点个数最少的一棵 AVL 树。并指出关键码  $k_1$ 、 $k_2$ 、 $k_3$  的值分别是多少?
2. 有长度为 7 的哈希表, 将序列  $\{32, 24, 15, 27, 20, 13\}$  加入哈希表中, 哈希函数为  $H(x) = H \bmod 7$ , 分别画出用线性探测法和链地址法的哈希表, 并计算查找每个元素的平均次数。

### 3. AOV 活动网络

事件的最早时间、最晚时间

活动的最早事件、最晚时间、时间余量

关键路径有哪些、关键活动有哪些

用中文写出以上过程的思路和算法

### 三、算法和编程题

1. 在堆  $\{k_1, k_2, k_3, \dots, k_{n-1}\}$  中插入一个元素  $x$ , 将  $\{k_1, k_2, k_3, \dots, k_{n-1}, x\}$  调整为堆, 用 C++ 或 Java 写出代码, 函数名为 `adjustheap`

2. 写出判断当前二叉树是二叉搜索树的算法;

(1) 用中文详细描述思路;

(2) 用 C++ 或 Java 写出代码;

`boolean isBST(TreeNode* root);`

```
Class TreeNode{
    int keyvalue;
    TreeNode* leftnode;
    TreeNode* rightnode;
}
```