

**北方工业大学**  
**2000 年硕士学位研究生入学考试试题**

考试科目: **数据结构**

适用专业: **计算机应用**

说 明:

要求: 算法用类 C 或类 PASCAL 语言描述, 在题目之前注明自己所用语言。算法中若利用栈或队列操作, 可直接用其基本操作, 但书写必须规范。

一、简要回答下列各题 (1..9 题每题 3 分, 10 题 5 分)

- 1、对于直接插入排序、简单选择排序、快速排序、冒泡排序和归并排序这几种排序方法中, 哪个算法有可能出现这样情况: 在最后一趟排序开始之前, 所有元素都不在其最终位置上? 哪个算法在最好情况下时间复杂度为  $O(n)$ ? 哪个算法要求辅助空间最大?
- 2、用单链表表示一个长度为  $n$  的队列, 若只设一个指向链表头结点的指针, 那么入队列和出队列的时间复杂度为多少?
- 3、设有一个链栈, 结点结构为 

data	next
------	------

, 栈顶指针为  $Ls$  ( $Ls \neq \text{NULL}$ ), 写出将元素  $e$  入栈的主要操作步骤。
- 4、循环队列用数组  $A[0..m-1]$  存放其元素值, 已知队列的头尾指针分别是  $\text{front}$  和  $\text{rear}$ , 则当前队列的长度为多少?
- 5、已知某二叉树的后序序列为  $\text{dabec}$ , 中序序列为  $\text{debac}$ , 它的先序序列是什么?
- 6、从概念上讲, 树与二叉树是两种不同的数据结构, 那么将树转化为二叉树的基本目的是什么?
- 7、什么是线索二叉树? 一棵具有  $n$  个结点的二叉树中共有多少个线索?
- 8、有一份电文中共使用五个字符:  $a$ 、 $b$ 、 $c$ 、 $d$ 、 $e$ , 它们的出现频率依次为 4、7、5、2、9, 给出每个字符的 Huffman 编码。



9、按二分查找法对一个长度为 12 的有序表进行查找，在表内各元素等概率的情况下，查找成功所需的平均查找长度为多少？

10、设散列表的长度为 9，散列函数为  $H(x) = i / 3$  ( $/$  为整除)，其中  $i$  为键值  $x$  中第一个字母在字母表中的序号，若键值的输入序列为：Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec，用链地址法处理冲突。

(1) 构造散列表。

(2) 求出在等概率情况下，查找成功时的平均查找长度。

二、(本题 8 分)

设计一个高效算法，求出二叉排序树  $T$  中数据域值最大的结点的指针。

树的结点结构为 

lchild	data	rchild
--------	------	--------

三、(本题 8 分)

设计一个算法，求中序线索树  $T$  中指针  $P$  ( $P \neq \text{NULL}$ ) 所指结点的后继结点的指针。

线索树的结点结构为 

lchild	ltag	data	rtag	rchild
--------	------	------	------	--------

$ltag$   $rtag$  为 0 时表示子树，为 1 时表示线索

四、(本题 10 分)

设计一个算法，将给定的带头单链表  $L$  逆置。即给定链表所表示的线性表为  $(a_1, a_2, \dots, a_{n-1}, a_n)$ ，逆置后所表示的线性表为  $(a_n, a_{n-1}, \dots, a_2, a_1)$ 。

五、(本题 12 分)

一棵树  $T$  以二叉链表（孩子—兄弟）表示，其结点结构为：

```
typedef struct CSNode {
    ElemType data;
    Struct CSNode *firstchild, *nextsibling;
} CSNode, *CSTree;
```

设计一个算法按层次遍历树中各个结点。

六、(本题 15 分)

设计一个算法，判断有向图  $G$  (图中有  $n$  个顶点) 是否是一棵以  $V_0$  为根的有向树。



## 七、(本题 15 分)

表插入排序是对直接插入排序算法的一种改进, 写出表插入排序中求有序链表的算法。结构定义如下:

```
#define SIZE 1000 // 静态链表容量
typedef struct {
    KeyType key; // 待排关键字
    int next; // 指针项
} SLNode; // 表结点类型
typedef struct {
    SLNode r[SIZE]; // 0 号单元为表头结点
    int length; // 链表当前长度
} SLinkListType;
```