

江西师范大学硕士研究生入学考试初试科目
考 试 大 纲

科目代码、名称: 863 数据结构与程序设计

适用专业: 081200 计算机科学与技术

一、考试形式与试卷结构

(一) 试卷满分 及 考试时间

本试卷满分为 150 分, 考试时间为 180 分钟。

(二) 答题方式

答题方式为闭卷、笔试。

试卷由试题和答题纸组成; 答案必须写在答题纸相应的位置上。

(三) 试卷题型结构

1. 单项选择题: 10 小题, 每小题 2 分, 共 20 分
2. 填空题: 10 小题, 每小题 2 分, 共 20 分
3. 程序填空与程序分析题: 4 小题, 每小题 6 分, 共 24 分
4. 解答题: 4 小题, 第 1 题 10 分, 共 40 分
5. 算法与程序设计题: 3 小题, 第 1、2 小题每小题 14 分, 第 3 小题 18 分, 共 46 分

二、考查目标 (复习要求)

全日制攻读硕士学位研究生入学考试数据结构与程序设计科目考试内容包括《数据结构》课程主要内容, 要求考生系统掌握相关学科的基本知识、基础理论和基本方法, 并能运用相关理论和方法分析、解决程序设计中的实际问题。

三、考查范围或考试内容概要

第一章 概论

1. 数据结构的基本概念与术语
2. 算法与算法分析

第二章 线性表及其顺序存储

1. 线性表
2. 顺序表及其应用
3. 栈的概念及其应用
4. 队列的概念及其应用

第三章 线性表及其链式存储

1. 链式存储
2. 单链表

3. 带头结点的单链表及其应用
4. 循环单链表与双链表
5. 链式栈与链式队列

第四章 字符串、数据与特殊矩阵

1. 字符串及模式匹配
2. 特殊矩阵的压缩存储
3. 稀疏矩阵

第五章 递归

1. 递归的基本概念与递归程序设计
2. 递归程序设计执行过程的分析
3. 递归程序到非递归程序的转换

第六章 树

1. 树的概念
2. 树的存储结构
3. 树的遍历

第七章 二叉树

1. 二叉树的基本概念
2. 二叉树的存储结构
3. 二叉树的遍历（递归与非递归）
4. 穿线二叉树的基本概念与构造
5. 树、森林和二叉树的转换

第八章 图

1. 图的基本概念
2. 图的存储结构（邻接矩阵法、邻接表法）
3. 图的遍历
4. 生成树与最小生成树
5. 最短路径
6. 拓扑排序
7. 关键路径

第九章 检索

1. 检索的基本概念
2. 线性表的检索
3. 二叉排序树
4. 平衡二叉排序树
5. Huffman 树
6. B-树

7. 散列表的检索
8. 查找算法的分析及应用

第十章 排序

1. 排序的基本概念
2. 插入排序（直接插入排序、折半插入排序、希尔排序）
3. 选择排序（简单选择排序、堆排序）
4. 交换排序（冒泡排序、快速排序）
5. 二路归并排序（merge sort）
6. 基数排序
7. 各种内部排序算法的比较
8. 内部排序算法的应用

参考教材或主要参考书:

1. 《数据结构》(C 语言版) 第二版, 李云清, 杨庆红, 揭安全 编著, 人民邮电出版社, ISBN:978-7-115-20703-6

四、样卷

江西师范大学硕士研究生入学考试试题（样卷）

专业：081200 计算机科学与技术 科目：数据结构与程序设计

注：考生答题时，请写在考点下发的答题纸上，写在本试题纸或其它答题纸上的一律无效！

（本试题共计 页）

一、选择题（每小题 2 分，共 20 分）

1、若语句 S 的执行时间为 $O(1)$ ，那么下列程序段的时间复杂度为：（ ）

```
for (i=0; i<n; i++)
    for (j=0; j<=i; j++)
        S;
```

(A) $O(n)$ (B) $O(n^2)$ (C) $O(n \log n)$ (D) $O(n \cdot i)$

2、在一个单链表中，若 p 所指结点不是最后结点，在 p 之后插入 s 所指结点，则执行的语句序列是（ ）。

(A) $s \rightarrow next = p; p \rightarrow next = s;$ (B) $p \rightarrow next = s; s \rightarrow next = p;$
(C) $s \rightarrow next = p \rightarrow next; p = s;$ (D) $s \rightarrow next = p \rightarrow next; p \rightarrow next = s;$

3、设高度为 h 的二叉树上只有度为 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（ ）（注：只含有一个根结点的树高为 1）

(A) $2h$ (B) $2h-1$ (C) $2h+1$ (D) $h+1$

4、一组记录的关键码为 (46, 79, 56, 38, 40, 84)，则利用堆排序的方法建立的初始堆为（ ）。

(A) 84, 79, 56, 38, 40, 46 (B) 79, 46, 56, 38, 40, 80
(C) 84, 79, 56, 46, 40, 38 (D) 84, 56, 79, 40, 46, 38

5、设二叉排序树中关键字由 1 至 1000 的整数构成，现要检索关键字为 363 的结点，下述关键字序列哪一个不可能是二叉排序树上搜索到的序列（ ）。

(A) 2, 252, 401, 398, 330, 344, 397, 363
(B) 924, 220, 911, 244, 898, 258, 362, 363
(C) 952, 202, 911, 240, 912, 245, 363
(D) 2, 399, 387, 219, 266, 382, 381, 278, 363

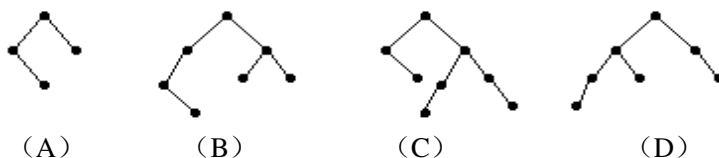
6、已知某完全二叉树采用顺序存储结构，结点数据信息 A、B、C、D、E、F、G、H，顺序存放在数组的前 8 个单元，则该完全二叉树的后序遍历序列为（ ）。

(A) HDEBFGCA (B) HEDBFGCA (C) HDEBAFGC (D) HDEFGBCA

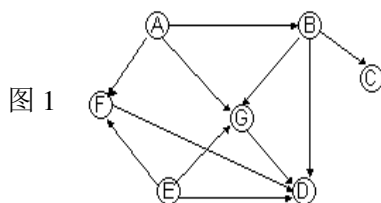
7、稀疏矩阵一般的压缩存储方法有（ ）两种。

(A) 二维数组和三维数组 (B) 三元组和散列表
(C) 三元组和十字链表 (D) 散列表和十字链表

8、下列哪棵树不是 AVL 树（ ）。



9、设有一个有向图如图 1 所示，请指出下列哪个序列不是该图的拓扑排序序列（ ）。



- (A) EAFBGDC (B) AEBCGFD (C) ABCGEFD (D) EABGFCD

10、判断一个有向图是否存在回路除了可以利用拓扑排序方法外，还可以利用（ ）。

- (A) 单源最短路 Dijkstra 算法 (B) 所有顶点对最短路 Floyd 算法
(C) 广度优先遍历算法 (D) 深度优先遍历算法

二、填空题（每小题 2 分，共 20 分）

1、数据的存储结构一般分为顺序存储、链式存储、_____和_____四种方式。

2、中缀表达式 $(A + B) * D + E / (F + A * D) + C$ 对应的后缀表达式是_____。

3、KMP 算法是解决模式匹配的有效算法，设数组下标从 0 从始，则模式串“babbabbab”对应的 next[] 值是_____。

4、一棵具有 46 个叶结点的完全二叉树，最多有_____个结点。

5、若 p 已指向了二叉树 t 的树根，则让 p 指向这棵二叉树的中序首点（中序遍历下的第一个结点）可以用以下的语句实现：_____。

6、编号为 1, 2, 3, 4 的四列火车通过一个栈式的列车调度站，可能得到的调度结果有_____种。

7、_____排序方法采用的是二分法的思想，_____排序方法其数据结构的组织采用的是完全二叉树结构。

8、若函数 `int max(int a[], int n)` 用于求长度为 n ($n > 0$) 的整型组 a 中的最大值，其递归定义可以表示为：_____。

9、若一棵度为 7 的树有 8 个度为 1 的结点，有 7 个度为 2 的结点，有 6 个度为 3 的结点，有 5 个度为 4 的结点，有 4 个度为 5 的结点，有 3 个度为 6 的结点，有 2 个度为 7 的结点，该树一共有_____个叶结点。

10、散列函数_____以为自变量，函数值作为结点的_____。

三、程序填空与程序分析题（每小题 6 分，共 24 分）

1、函数 `frequency()` 的功能统计字符串 s 中各个不同字符出现的频度，数组

a 记录 s 中不同的字符，数组 c 中记录每一种字符的出现次数，k 返回不同字符的总数，请将程序补充完整。

```
void frequency(char s[], char a[], int c[], int *k)
{
    int i, j, len = strlen(s);
    if ( !len )          /*原字符串为空*/
    {
        printf("The string is empty.\n ");    *k = 0;
        return;
    }
    else                /*原字符串不为空*/
    {
        a[0] = s[0];    c[0] = 1;    *k = 1;    /*处理第一个字符*/
        for ( i = 1; i < len; i++ )
            ①                /*数组 c 初始化*/
        for ( i = 1; i < len; i++ )
        {
            /*对 s 中剩余字符进行检测*/
            j = 0;
            while ( j < *k && a[j] != s[i] ) /*检查 s[i] 是否已在 a[]
中*/
                ②
            if ( j == *k )                /*s[i] 不在 a 中*/
            {
                a[*k] = ③
                c[*k] = ④
                (*k) = ⑤
            }
            else ⑥                /*s[i] 已在 a 中*/
            }
        }
    }
}
```

2、设带头结点的单链表的存储结构定义如下：

```
typedef char datatype;
typedef struct node{
    datatype data;
    struct node * next;
} listnode;
typedef listnode* linklist;
```

函数 deletelist() 的功能是将带头结点的单链表 head 中的重复结点删除（即 data 域相同的结点只保留第一个结点），请将程序补充完整。

```
linklist deletelist(linklist head)
{
    linklist p, pre, s;
    p = head->next;
    while (p)
```

```

    {
        pre=p;
        s=p->next;
        while (s!=NULL)
            {while (s&& s->data!=p->data) /*从结点 p 后找与其相
同的结点*/
            { ⑦
            s=s->next;
            }

            if (s!=NULL) /*找到了与 p 相同的结点 s*/
                { pre->next=s->next;
                  ⑧
                }
                ⑨
            }

        p=p->next;
    }
    return head;
}

```

3、二叉排序树链式存储结构定义如下：

```

typedef int datatype;
typedef struct node
{
    datatype key; /*结点值*/
    struct node *lchild,*rchild; /*左、右孩子指针*/
} bsnode;
typedef bsnode *bstree;

```

函数 insertbstree () 的功能是在二叉排序树 t 中插入一个新结点 x (若树中存在相同结点则不做插入操作)，请将其补充完整。

```

void insertbstree(bstree *t,datatype x)
{
    bstree f,p;
    ⑩
    while (p) /*查找插入位置*/
    {
        if (x==p->key) ⑪ /* 若二叉排序树 t 中已有 key，则
无需插入*/
        {
            f=p; /* f 用于保存新结点的最终插入位
置 */
            p=(x<p->key)? ⑫
        }
        p=(bstree) malloc(sizeof(bsnode)); /*生成待插入的新结点*/
        p->key=x;
        ⑬
    }
}

```

```

        if (*t==NULL)      (14)          /*原树为空*/
        else
            if (x<f->key)
                (15)
            else    f->rchild=p;
    }

```

4、阅读下面的递归程序，说明这个函数的功能是什么？如果数组的大小为 n ，则该程序的时间复杂度是多少？设数组 a 的初始值是 21, 30, 52, 35, 69, 70, 90, 61, 78, 99，则执行 $\text{change1}(a, 0, 9)$ 后，数组 a 的内容是什么？

```

void change1(int a[], int low, int high)
{
    int i, j, t;
    i=low; j=high;
    if (i<j)
    {
        while (i<j && a[i]%2!=0)    i++;
        while (i<j && a[j]%2==0)    j--;
        if (i!=j)
        {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
        }
        change1(a, i+1, j-1);
    }
}

```

四、解答题（每小题 10 分，共 40 分）

1、一棵前序序列为 1, 2, 3, 4 的二叉树，其中序序列可能是 4, 1, 2, 3 吗？设一棵二叉树的前序序列为 1, 2, 3, 4, 5, 6, 7, 8, 9，其中序序列为 2, 3, 1, 5, 4, 7, 8, 6, 9，试画出该二叉树。

2、假定用于通信的电文仅由 8 个字母 $c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8$ 组成，各字母在电文中出现的频率分别为 5, 25, 3, 6, 10, 11, 36, 4。试构造哈夫曼树，并为这 8 个字母设计不等长 Huffman 编码。

3、顺序表的快速排序为何采用由外向内来回比较法，是否可以从同一个方向扫描？采用带头结点的单链表存储的线性表是否可以快速排序？对初始序列 (50, 20, 79, 24, 49, 84, 3, 99, 12) 以 50 作为“枢轴”进行第一次划分后的结果是什么？

4、 给定无向网如下图 2 所示，请采用 prim 算法用图示描述求解该图的最小生成树的过程。（初始入选点为 A，每选取一条边画一个图）

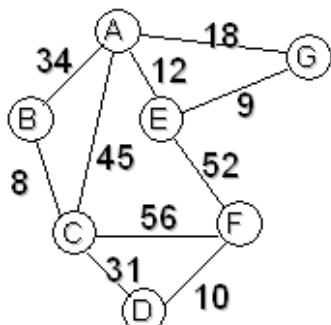


图 2

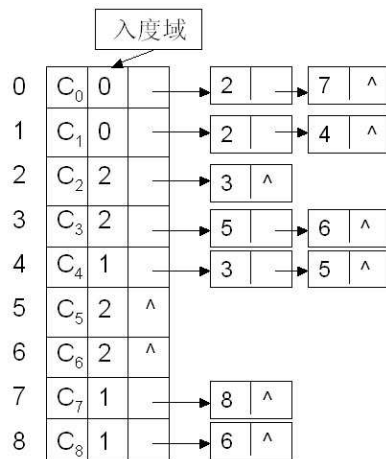


图 3

五、算法与程序设计题（第 1、2 题每小题 14 分，第 3 小题 18 分，共 46 分）

答题要求：

- ①描述算法的基本设计思想；
- ②给出每个算法所需的数据结构定义；
- ③根据设计思想和实现步骤，采用用 C 语言写出对应的算法程序，关键之处请给出简要注释。

1、设一带头结点的单链表的头指针为 head，链表的记录中包含着整数类型的 info 域，试采用直接插入法将此链表的记录按照 info 递增的次序进行就地排序。

2、二叉树采用二叉链表存储结构，t 为其根结点，分别用递归与非递归方法编写函数，返回二叉树的前序尾点地址（前序遍历下的最后一个结点）。

3、AOV 网采用带入度域的邻接表存储结构如图 3 所示。请用 C 语言定义这种结构，编写程序建立一个有向图的这种存储结构（出边表），并输出图中入度为 0 的顶点。