

# 江苏大学

## 2012 年硕士研究生入学考试初试试题 ( A 卷 )

科目代码: 851 科目名称: 数据结构 满分: 150 分

注意: ①认真阅读答题纸上的注意事项; ②所有答案必须写在答题纸上, 写在本试题纸或草稿纸上均无效;

③本试题纸须随答题纸一起装入试题袋中交回!

### 一、单项选择题(每小题 1 分,共 10 分)

1. 线性表 $(a_1, a_2, \dots, a_n)$ 采用静态链表存储时,访问第  $i$  位置元素的时间复杂度为( )。  
(A)  $O(n^2)$  (B)  $O(n)$  (C)  $O(i)$  (D)  $O(1)$
2. 设深度为  $k$  的二叉树上只有度为 0 和度为 2 的结点,则这类二叉树上所含结点总数最少( )个。  
(A)  $k+1$  (B)  $2k$  (C)  $2k-1$  (D)  $2k+1$
3. 设顺序存储的线性表共有 123 个元素,按分块查找的要求等分成 3 块。若对索引表采用顺序查找来确定块,并在确定的块中进行顺序查找,则在查找概率相等的情况下,分块查找成功时的平均查找长度为( )。  
(A) 21 (B) 23 (C) 41 (D) 62
4. 如下陈述中正确的是( )。  
(A) 串是一种特殊的线性表 (B) 串的长度必须大于零  
(C) 串中元素只能是字母 (D) 空串是由空格组成的串
5. 与数据元素本身的形式、内容、相对位置、个数无关的是数据的( )。  
(A) 存储结构 (B) 存储实现 (C) 逻辑结构 (D) 运算实现
6. 除了( ),其它任何指针都不能在算法中作为常量出现。  
(A) 头指针 (B) 尾指针 (C) 指针型变量 (D) 空指针
7. 稀疏矩阵的压缩存储方法是只存储( )。  
(A) 非零元素 (B) 三元组  $(i, j, a_{ij})$  (C)  $a_{ij}$  (D)  $i, j$
8. 设  $\text{rear}$  是指向非空带头结点的循环单链表的尾指针,  $\text{delete}$  是释放结点的空间,则删除表中第一个元素结点的操作可表示为( )。  
(A)  $p=\text{rear}; \text{rear}=\text{rear} \rightarrow \text{next}; \text{delete } p;$   
(B)  $\text{rear}=\text{rear} \rightarrow \text{next}; \text{delete } \text{rear};$   
(C)  $p=\text{rear} \rightarrow \text{next} \rightarrow \text{next}; \text{rear} \rightarrow \text{next} \rightarrow \text{next}=p \rightarrow \text{next}; \text{delete } p;$   
(D)  $\text{rear}=\text{rear} \rightarrow \text{next} \rightarrow \text{next}; \text{delete } \text{rear};$
9. 对广义表  $L = (a, (b, c))$ , 进行  $\text{Tail}(L)$  操作后的结果为( )。  
(A)  $c$  (B)  $b, c$  (C)  $(b, c)$  (D)  $((b, c))$
10. 若元素  $a, b, c, d, e, f$  依次进栈,允许进栈、退栈操作交替进行,但不允许连续三次进行退栈工作,则不可能得到的出栈序列是( )。  
(A)  $dcebf a$  (B)  $cbdaef$  (C)  $bcaefd$  (D)  $afedcb$

### 二、填空题(每小题 2 分,共 10 分)

1. 两个串是相等的是指\_\_\_\_\_。
2. 设  $A$  是  $n \times n$  的对称矩阵,每个元素长度为  $K$  个字节,将  $A$  的对角线及对角线上方的元素以列为主的次序存放在内存首地址为  $BA$  开始的一片地址连续的内存空间  $B$  中,则对  $A$  中任一元素  $a_{ij}(1 \leq i, j \leq n, \text{且 } i \leq j)$  在  $B$  中的存储地址为\_\_\_\_\_。
3. 具有  $n$  个结点的二叉排序树在最差情况下的平均查找长度(在等概率而且查找成功情况下)是\_\_\_\_\_。
4. 根据数据元素之间关系的不同,数据结构可分为\_\_\_\_\_结构和\_\_\_\_\_结构两大类。这两大类结构

还可细分为四类基本结构,即集合、线性结构、树形结构、网状结构(或图形结构)。

5. 设循环队列用下标范围是 0 到 99 的数组 Q 存放元素值,队列的头指针为 20,指向队首元素的前一个位置,尾指针为 10,指向队尾元素位置,则队列 Q 中当前所含元素个数为\_\_\_\_\_。

### 三、应用题(共 90 分)

- 1.(18 分)已知森林的中序序列为:JBMLFCAEDHKIG,后序序列为:BJMCFLEHIKGDA。要求:

- (1) 画出该森林。
- (2) 画出该森林中第一棵树的带双亲域的孩子链表。
- (3) 画出该森林所对应的二叉树的左子树的先序前驱线索二叉链表。

- 2.(6 分) 设某密码电文由 5 个字母(A,B,C,D,E)组成,每个字母在电文中的出现频率分别是(7,2,6,3,10),试为这 5 个字母设计相应的哈夫曼编码,要求写出构造过程。

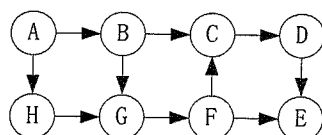
3. (14 分) 带权连通图的邻接矩阵如下图所示, 其顶点 A, B, C, D, E, F, G 的编号依次为 1, 2, 3, 4, 5, 6, 7, 要求:

- (1) 基于图的邻接矩阵从顶点 A 出发, 求它的 dfs 遍历序列和 dfs 生成树。
- (2) 基于图的邻接矩阵从顶点 E 出发, 求它的 bfs 遍历序列和 bfs 生成树。
- (3) 根据普利姆(Prim)算法, 请画出从 A 点出发的最小生成树以及构造过程。

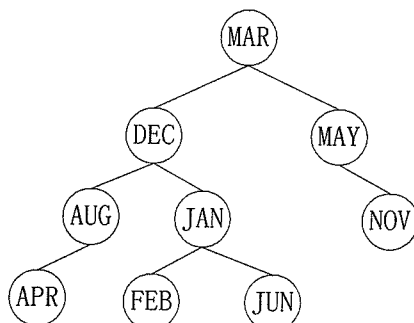
	1	2	3	4	5	6	7
1	0	5	4	2	6	$\infty$	$\infty$
2	5	0	$\infty$	$\infty$	$\infty$	$\infty$	3
3	4	$\infty$	0	$\infty$	1	$\infty$	$\infty$
4	2	$\infty$	$\infty$	0	$\infty$	3	$\infty$
5	6	$\infty$	1	$\infty$	0	5	$\infty$
6	$\infty$	$\infty$	$\infty$	3	5	0	1
7	$\infty$	3	$\infty$	$\infty$	$\infty$	1	0

4. (7 分)有一 AOV 网如下图所示, 要求:

- (1) 写出对该 AOV 网进行拓扑排序时的所有拓扑有序序列。
- (2) 画出邻接表。



- 5.(6 分)已知一棵平衡二叉树如图所示, 其中各结点间大小关系(中序次序)按字典序排列, 请画出插入结点 JUL 后的平衡二叉树, 并说明采用了哪种类型的旋转方式, 并标出插入后平衡二叉树中各结点的平衡因子。



- 6.(15 分)已知关键字序列 F={19, 24, 10, 17, 15, 38, 18, 40}, 哈希函数为  $H(\text{Key}) = \text{Key} \bmod 7$ , 即关键字对 7

取模,其中 Key 为关键字,哈希地址空间为 0~9。要求:

- (1) 计算出每一个关键字的哈希函数值。
- (2) 画出相应的哈希表,发生冲突时,采用二次探测再散列法解决。探测的增量序列为  $d_i = 1^2, -1^2, 2^2, -2^2, 3^2, \dots, \pm k^2, k \leq 5$ 。
- (3) 计算出装填因子。
- (4) 给出查找每一个关键字的比较次数。
- (5) 求出等概率下查找成功时的平均查找长度 ASL。

7. (24 分) 已知关键字序列  $F = \{42, 70, 53, 65, 24, 56, 48, 92, 86, 33\}$ , 要求:

- (1) 把它调整为一个小顶堆,给出建堆过程。并求建堆时总的比较次数是多少?
- (2) 若采用链式基数排序方法排序,请写出第一趟“分配”之后各队列的状态和第一趟“收集”之后的关键字序列。
- (3) 画出按关键字从大到小为序的步长为 3 的第一趟希尔排序结果。

#### 四、简答题(共 20 分)

- 1.(6 分)请叙述头指针和头结点的概念和作用。
- 2.(8 分)试求有  $n$  个叶结点的非满的完全二叉树的高度,请给出推导过程。
- 3.(6 分)顺序查找,二分查找,哈希(散列)查找的时间分别为  $O(n), O(\log_2 n), O(1)$ 。既然有了高效的查找方法,为什么低效的方法还不放弃?

#### 五、算法设计题(共 20 分)

- 1.(10 分) 设  $L$  是带头结点的非空单链表的头指针,数据元素类型为整型,而且表中元素值各不相同。请编写算法删除单链表上最大值的元素,同时返回被删除元素的值。
- 2.(10 分)试基于图的广度优先搜索策略写一算法,判断以邻接表方式存储的有向图中是否存在由顶点  $V_i$  至顶点  $V_j$  的路径( $i \neq j$ )。

注意:

- (1) 可用(类)Pascal 语言或(类)C 语言或 C++语言描述你的算法;
- (2) 请简要描述你的算法思想;
- (3) 若你的算法是(类)Pascal 或(类)C 语言编写,则请给出相应的存储结构描述;
- (4) 若你的算法是用 C++语言描述,则可参考使用以下给出的相关存储结构的类定义,算法中可以使用类中已列出的成员函数。若在你的算法中使用了未列出的成员函数,则要写出该成员函数的完整算法描述。
- (5) 若在你的算法中使用了未列出的其他的辅助存储结构与算法,也请给出说明。

//单链表的类定义

```
template <class type> class linklist; //单链表前视声明
```

```
template <class type> class node{//单链表结点类
```

```
    friend class linklist <type>; //定义单链表类 linklist <type>为结点类的友元
```

```
    private:
```

```
        node <type> *next; //链指针域
```

```
    public:
```

```
        type data; //数据域
```

```
        node (node <type> *pNext = NULL) {next = pnext;}//构造函数,用于构造头结点
```

```
};
```

```
template <class type> class linklist{ //单链表类定义
```

```
    private:
```

```
        node <type> *head; //指向头结点的头指针
```

```
    public:
```

```

linklist ( ){ head = new node <type> ( ); head->next=NULL; }//构造函数
~linklist ( );//析构函数
};
//图的邻接表的类定义:
const int MaxVertexes = 20; //最大的顶点数
template <class vertexType, class arcType> class Graph;
template < class arcType> struct ArcNode { //定义弧结点
friend class Graph <class vertexType, class arcType>;
    int adjvex; //弧头顶点序号
    arcType weight; //弧上的信息(权)
    ArcNode<arcType> *nextarc; //指向下一条弧结点的指针
    ArcNode ( ) { } //构造函数
    ArcNode( int v , arcType w ) : adjvex( v ) , weight( w ) , next( NULL ){ } //构造函数
}
template < class arcType , class vertexType > struct VertexNode {
// 定义顶点结点
friend class Graph <class vertexType, class arcType>;
    vertexType data; //顶点的信息
    ArcNode<arcType> *firstarc; //指向以该顶点为弧尾的第一个弧结点的指针
}
template <class vertexType, class arcType> Graph{ //定义图
private:
    VertexNode < arcType , vertexType > * VertexesTable; //顶点表
    int CurrentNumVertexes; //当前的顶点数
    int CurrentNumArcs; //当前的边(或弧)数
    int GetVertexPos( const vertexType &v ); // 取顶点 v 在数组中的位置
public:
    Graph : CurrentNumVertexes (0), CurrentNumArcs (0){}; //构造函数
    Graph ( vertexType v[ ] , int num = MaxVertexes ); //构造函数
    ~Graph ( ); //析构函数
}

```