

江苏大学 2008 年硕士研究生入学考试试题

科目代码: 852

科目名称: 数据结构

考生注意: 答案必须写在答题纸上,写在试卷、草稿纸上无效!

一、单项选择题(每小题 2 分,共 20 分)

- 以下与数据的逻辑结构无关的术语是 ()。
(A) 链表 (B) 队列 (C) 串 (D) 栈
- 对于一个头指针为 head 的带头结点的单链表,判定该表为空表的条件是 ()。
(A) head==NULL (B) head->next==NULL (C) head->next==head (D) head!=NULL
- 以下说法正确的是()。
(A) 队列是线性表的一种特殊存储结构
(B) 队列是先进后出的线性表
(C) 队列可以通过两个栈来模拟实现
(D) 对于插入或删除较为频繁的操作,链队列比循环队列效率更高
- 若一个栈的输入序列为 1,2,3,...,n,输出序列的第一个元素是 i,则第 j 个输出元素是()。
(A) i-j-1 (B) i-j (C) j-i+1 (D) 不确定的
- 串 'abcaabbcbca' 的 next 数组为()。
(A) -1 0 0 0 1 1 0 0 0 1 2 3 (B) -1 0 0 0 1 0 1 0 0 1 2 3
(C) -1 0 0 0 -1 -1 0 2 0 -1 0 0 (D) -1 0 0 0 1 1 2 0 0 1 2 3
- 设有一个二维数组 D[m][n],假设 D[0][0] 存放位置在 600₍₁₀₎,D[3][3] 存放位置在 678₍₁₀₎,每个元素占一个空间,请问 D[2][3] 存放在()位置?(脚注₍₁₀₎表示用 10 进制表示,m>3)。
(A) 658 (B) 648 (C) 653 (D) 633
- 有关二叉树下列说法正确的是()
(A) 一棵二叉树的度可以小于 2 (B) 二叉树的度为 2
(C) 二叉树中至少有一个结点的度为 2 (D) 二叉树中任何一个结点的度都为 2
- 当一个有 n 个顶点的有向图用邻接矩阵 A 表示时,顶点 V_i 的出度是()。
(A) $\sum_{i=1}^n A[i, j]$ (B) $\sum_{j=1}^n A[i, j]$ (C) $\sum_{i=1}^n A[j, i]$ (D) $\sum_{i=1}^n A[i, j] + \sum_{j=1}^n A[j, i]$
- 在所有排序的算法中,关键字比较的次数与记录的初始排列次序无关的是()。
(A) 希尔排序 (B) 起泡排序 (C) 直接插入排序 (D) 简单选择排序
- 双向循环链表中有两个指针域,next 和 priou,分别指向前驱及后继,设 p 指向链表中的一个结点,s 指向一待插入结点,现要求删除 p 所指的结点,则须修改指针()。
(A) p->priou->next=p; p->priou=p->priou->priou;
(B) p->next=p->next->next; p->next->priou=p;
(C) p->next->priou=p->priou; p->priou->next=p->next;
(D) p->priou=p->next->next; p->next=p->priou->priou;

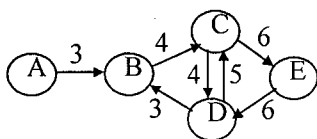
二、填空题(每空 2 分,共 20 分)

- 一个算法的时间复杂度为 $(56 \cdot n^3 + 2008 \cdot n \cdot \log_2 n + 90)/n^2$,其数量级表示为 $O(\quad)$ 。
- 对于一个具有 n 个结点的单链表,在已知地址为 p 的结点后插入一个新结点的时间复杂度为 \quad ,在给定值为 x 的结点后插入一个新结点的时间复杂度为 \quad 。
- 表达式 $a+b \cdot (c+d/e)$ 的后缀表达式为 \quad 。
- 两个字符串相等的充分必要条件是 \quad 。

5. 已知一棵度为 3 的树有 2 个度为 1 的结点,3 个度为 2 的结点,4 个度为 3 的结点,则该树的叶子结点是_____个。
6. 已知广义表 $L=(a,b,(c,d),(e,(f,g)))$,则式子 $\text{Tail}(\text{Head}(\text{Tail}(\text{Tail}(L))))$ 的值是_____。
7. 具有 256 个结点的完全二叉树的深度为_____。
8. 一个有 n 个顶点的无向图最少有_____条边。
9. 具有 6 个关键字的有序表,若查找每个元素的概率相同,则查找成功的折半查找的平均查找长度是_____。
10. 在堆排序、希尔排序、归并排序、快速排序这些排序算法中,稳定的排序算法是_____。

三、应用题(共 60 分)

- 1.(12 分)设 T 是一棵只有度为 0 和度为 2 的结点的二叉树,现若 T 中叶子结点有 6 个。请问:
 - (1) T 树的最大深度 K_{\max} 是多少?最小可能深度 K_{\min} 是多少?(同时请给出任一示例说明)
 - (2) T 树中共有多少个非叶子结点?
 - (3) 另有关键字序列 $F=\{2,3,4,7,8,9\}$ 。请构造一棵哈曼夫树(给出构造过程),并计算该哈曼夫树的带权路径长度 WPL 。
- 2.(16 分)给出一组关键字序列 $F=\{23,13,27,41,19,39,15,21,31,17,29\}$,要求:
 - (1)用链式基数排序法(基数为 10),写出第 1 次分配和收集的结果
 - (2)把该序列调整为最小堆序列,写出建堆过程和堆序列
 - (3)用二路归并排序法,写出前 2 趟的排序结果
 - (4)用希尔排序法,写出增量为 5 的第一趟排序结果
- 3.(16 分)求解下面有向图的有关问题:



- (1)判断此有向图是否有强连通分量?若有请画出该有向图的所有强连通分量。
- (2)画出此有向图的逆邻接表。
- (3)设其顶点 A, B, C, D, E 表示一个县的 5 个村,弧上的权值表示为两村之间的距离,求 A 村到其它各村的最短距离,请给出求解过程。
- (4)从 A 点出发给出该图的所有深度优先遍历序列以及对应的 dfs 生成树。
- 4.(16 分)给定一组关键字序列 $F=\{30,14,15,29,31,17,12,20,11,21\}$,要求:
 - (1)试从空树开始,画出由该关键字序列构成的平衡二叉树,并为每一次的平衡处理指明旋转类型。
 - (2)现采用哈希函数 $H(\text{key})=\text{key} \% 10$,即关键字对 10 取模,用线性探测法解决冲突,设哈希表的大小为 11,即地址范围是 0 到 10,试画出插入上述关键字后的哈希表,并计算出在等概率情况下查找成功时的平均查找长度 ASL 。
 - (3)若上面的哈希表中删除一个记录,应如何操作?为什么?

四、简答题(共 30 分)

- 1.(6 分)线性表有两种存储结构:一是顺序存储结构(简称顺序表),另一是链式存储结构(简称链表)。试问:
 - (1)如果有 n 个线性表同时并存,并且在处理过程中各表的长度会动态变化,线性表的总数也会自动地改变。在此情况下,应选用哪种存储结构?为什么?
 - (2)若线性表的总数基本稳定,且很少进行插入和删除,但要求以最快的速度存取线性表中的元素,那么应采用哪种存储结构?为什么?

- 2.(6分)设 $m \times n$ 阶稀疏矩阵 A 有 t 个非零元素,可用三元组顺序表 L 来压缩存储,试问:非零元素的个数 t 达到什么程度时用三元组顺序表 L 表示 A 才有意义?
- 3.(10分)假设指针 p 指向中序线索二叉树中的某一结点,指针 x 指向待插入的新结点。现要把结点 x 插入到中序线索二叉树中,使之成为结点 p 的右孩子结点,请问如何完成插入(为了解释清楚,请画出相应的中序线索二叉树在插入 x 前后的变化情况)?
- 4.(8分)请解释为什么图没有顺序映像的存储结构,也不直接采用多重链表来表示,而是要根据具体问题要求来设计?常用的图的存储结构有哪些?

五、算法设计题(共 20 分)

- 1.(10分)假设有两个按元素值递减有序排列的单链表 A 和 B ,均以带头结点的单链表作为存储结构,请编写算法将 A 表和 B 表合并成一个按元素值也递减有序的单链表 A ,即实现 $A=A \cup B$ 。
- 2.(10分)试写一算法,将一棵二叉排序树分裂为两棵二叉排序树,使得其中一棵树的所有结点的关键字都小于或等于 x ,另一棵树的任一结点的关键字均大于 x 。(若在你所编写的算法中引入了辅助的存储结构,要简要说明)

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。

//有序单链表的类定义

```
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
};
```

```

//二叉排序树的类定义:
template<class Type> class BinaryTree;
template<class Type> class BinTreeNode//二叉排序树结点类的定义
{friend class BinaryTree<Type>;
private:
    BinTreeNode<Type> *leftChild,*rightChild;
    Type data;
public:
    BinTreeNode():leftChild(NULL),rightChild(NULL){} //构造一个空结点
    BinTreeNode(Type d,BinTreeNode<Type>*lch=NULL,BinTreeNode<Type> *rch=NULL):
        data(d),leftChild(lch),rightChild(rch){} //构造一个非空结点
};//class BinTreeNode
template<class Type> class BinaryTree //二叉排序树的类定义
{private:
    BinTreeNode<Type> *root; //二叉排序树根结点指针
    void destroy(BinTreeNode<Type> *p);//删除以 p 为根的二叉排序树
public:
    BinaryTree():root(NULL) {}//构造函数
    ~BinaryTree(){destroy(root);}//析构函数
};//class BinaryTree

```