**2012年全国硕士研究生统一入学考试自命题试题**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

学科与专业名称：计算机技术，软件工程

考试科目代码与名称：830 数据结构

**一. 选择题(每题2分，共30分)**

1．队列操作的原则是（ **A** ）。

A. 先进先出 B. 后进先出 C. 只能进行插入 D. 只能进行删除

2. 一个栈的进栈序列是a, b, c, d, e, 则栈的不可能的输出序列是（ **C** ）。

A. edcba B. decba C. dceab D. abcde

3. 采用顺序查找法查找长度为n的线性表时,每个元素的平均查找长度为 （ **C**  ）。

A. n B. n/2 C.(n+1)/2 D.(n-1)/2

4. 线性表的链接实现有利于(  **B** )运算。

A. 读表元素 B.插入 C. 查找 D. 定位

5. 设单链表中指针p指着结点A,若要删除A之后的结点(若存在),则需要修改指针的操作为(  **A** )。

A. p->next=p->next->next B. p=p->next

C. p=p->next->next D. p->next=p

6. 在内部排序中,排序时不稳定的有(  **C**  )。

A. 插入排序 B. 冒泡排序 C. 快速排序 D. 归并排序

7. 在AOE网中，完成工程的最短时间是（  **A**  ）。

A．从源点到汇点的最长路径的长度 B．从源点到汇点的最短路径的长度

C．最长的回路的长度 D．最短的回路的长度

8．以下（  **D** ） 方法所用辅助存储空间最大。

A． 堆排序 B． 希尔排序 C．快速排序 D．归并排序

9．具有8个顶点的无向图至少应有（  **C**  ）条边才能确保是一个连通图。

A．5 B．6 C．7 D．8

10. 对具有n个结点的有序表中折半查找时，其时间复杂度是（ **B** ）。

A．O(nlog2n） B．O(log2n） C．O(n） D．O(n2）

11．如果希望对平衡二叉树遍历的结果是升序的，应采用（  **B**  ）遍历方法。

A．先序 B．中序 C．后序 D．层次

12. 稀疏矩阵一般的压缩存储方法有两种,即:（ **C**  ）。

A. 二维数组和三维数组 B. 三元组和散列

C. 三元组和十字链表 D. 散列和十字链表

13. 循环队列中是否可以插入下一个元素 （ **D**  ）。

A. 与曾经进行过多少次插入操作有关.

B. 只与队尾指针的值有关,与队头指针的值无关.

C. 只与数组大小有关,与队首指针和队尾指针的值无关

D. 与队头指针和队尾指针的值有关.

14. 在线索化二叉树中，T所指结点没有左子树的充要条件是（  **B** ）。

A．T->left=NULL B．T->ltag=1

C．t->ltag=1且t->left=Null D．以上都不对

15. 以下说法中不正确的是（  **D**  ）。

A．无向图中的极大连通子图称为连通分量

B．连通图的广度优先搜索中一般要采用队列来暂存刚访问过的顶点

C．图的深度优先搜索中一般要采用栈来暂存刚访问过的顶点

D．有向图的遍历不可采用广度优先搜索方法

**二．填空题(每题2分，共20分)**

1．一组记录（50，40，95，20，15，70，60，45，80）进行冒泡排序时，第一趟需进行相邻记录的交换的次数为  **7** 。

2．数据结构按逻辑结构可分为两大类，它们分别**线性结构和非线性结构** 。

3．由n个权值构成的哈夫曼树共有 **2\*n-1** 个结点。

4．在散列表(hash)查找中，评判一个散列函数优劣的两个主要条件是： **出现冲突的概率**

和 **遇到冲突后的处理方法** 。

5．单链表中设置头结点的作用是 **使得不需要对表头元素的操作进行特殊化处理，空表和非空表插入元素时的处理得到统一**  。

6．一棵深度为k的满二叉树的结点总数为 ，一棵深度为k的完全二叉树的结点总数的最小值为  。

7．一个无向图有n个顶点和e条边，则所有顶点的度的和为 **2\*e**  。

8．在二叉链表中判断某指针p所指结点为叶子结点的条件是 **p->lchild==NULL&&p->rchild==NULL**  。

9．堆栈是一种操作受限的线性表，它只能在线性表的 栈顶 进行插入和删除操作，对栈的访问是按照 **先进后出** 的原则进行的。

10．若某记录序列的关键字序列是（235，346，021，558，256），用链式基数排序方法排序，第一次收集的结果是  **(021,235,346,256,558)**  。

**三．判断题（每题1分，共10分，正确的选t，错误的选f）**

1．如果T2是由树T1转换而来的二叉树,那T1中结点的先序就是T2中结点的先序。（**√** ）

2．在一个有向图的邻接表或逆邻接表中，如果某个顶点的链表为空，则该顶点的度一定为零。（ **×**  ）

3．线性表中的每一个元素都有一个前驱和后继元素。（ **×** ）

4．按中序遍历一颗二叉排序树所得到的中序遍历序列f是一个递增序列。（ **√** ）

5．若网中有几条关键路径，提高一条关键路径上的活动的速度，不能导致整个工程缩短工期。 （ **×** ）

6．一颗满二叉树同时又是一颗平衡树。（ **√** ）

7．数据结构是研究数据的物理结构、逻辑结构以及它们之间的相互关系。（ **√** ）

8. 拓扑排序是一种内部排序的算法。( **×** )

9.已知一颗树的先序序列和后序序列，一定能构造出该树。（ **×** ）

10．n阶对称矩阵可压缩存储到n2/2个元的空间中。( **×** )

**四. 简答题（50分）**

1. 给定关键字序列 T=(65，57，45，39，12，98，86，35），采用快速排序算法，以第一个元素为枢轴，对该序列由小到大排序，并写出具体排序过程。 (8分)

**答案：**

**第一趟：35，57，45，39，12，65，86，98**

**第二趟：12，35，45，39，57，65，86，98**

**第三趟：12，35，39，45，57，65，86，98**

1. 简述下列算法的功能。（6分）

void Process(LinkList &L, int x, int y) { // L线性表的元素递增有序排列

LinkList p=L, q, s;

if ((p->next) && (x<=y))

{ while (p->next && p->next->data<=x) p=p->next;

If (p->next) return ERROR;

q=p->next;

while (q->next && q->next->data<y)

{ s=q; q=q->next; free(s); }

p->next=q->next;

free(q);

}

}

**答：删除L线性表中元素大于x且小于y的结点。**

1. 使用克鲁斯卡尔算法构造出图1所示的图G的一棵最小生成树（要求写出构造过程）。（10分）

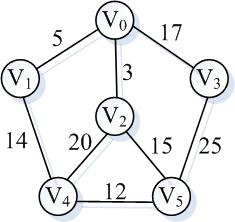
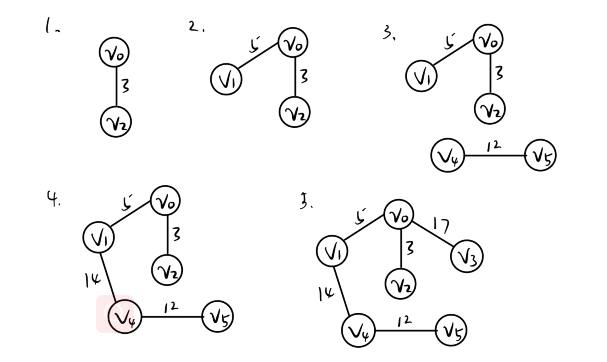


图1



1. 已知一个图如图2所示，若从顶点a出发，按深度优先搜索法进行遍历，写出可能得到的一种顶点序列；按广度优先搜索法进行遍历，写出可能得到的一种顶点序列。 (4分)

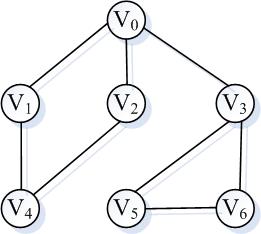


图2

**答：深度优先搜索序列：**

**广度优先搜索序列：**

1. 给定图3所示带权有向图及其邻接矩阵，利用Floyd算法，求每一对顶点之间的最短路径及其路径长度（要求写出求解过程）。 （12分）

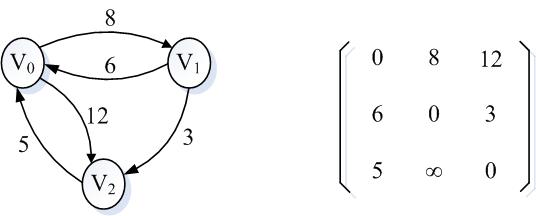
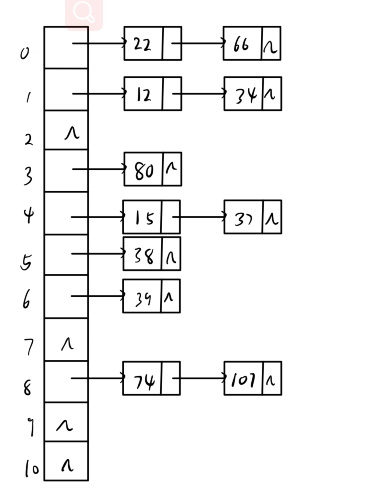


图3

**答：**

1. **由V0到V1的距离为8，V1到V2的距离为3，可知V0通过V1再到V2的距离为11，故更新邻接矩阵V0到V2的距离为11.**
2. **由V2到V0的距离为5，V0到V1的距离为8，可知V2通过V0再到V2的距离为13，故更新邻接矩阵V2到V1的距离为13.**

6．给出一组关键字的序列为{ 12，15，34，37，39，22，38，66，74，80，107 }，假设哈希函数为Hash(key)=key mod 11，画出按照链地址法处理冲突构造所得的哈希表，并在记录的查找概率相等的前提下，计算成功查找的平均查找长度。（10分）

**答：**

**ASL(成功)=(1+2+1+2+1+1+2+1+1+1+2)/11=15/11**

**五．算法填空，（每空2分,共16分）**

1. 下面的算法将元素e加入队列Q中，请在           处填上适当内容，使其成为一个完整算法。

typedef struct QNode **{**

QElemType data;

structQNode **\***next;

**}** QNode, **\***QueuePtr;

typedef struct {

QueuePtr front; // 队头指针

QueuePtr rear; // 队尾指针

} LinkQueue, \* LinkQueuePtr;

Boolean EnQueue (LinkQueuePtr Q, QElemType e) { //元素e加入到队列Q中

p = **(QNode \*)malloc(sizeof(QNode)** ;

if (!p) return FALSE;

p->data = e;

p->next = **NULL** ;

**Q->rear->next**  = p;

Q->rear =  **p** ;

return TRUE;

}

1. 下面是先序遍历二叉树的算法非递归算法，请在           处填上适当内容，使其成为一个完整算法。

typedef struct BiTNode { // 结点结构

TElemType data;

struct BiTNode \*lchild, \*rchild; // 左右孩子指针

} BiTNode, \*BiTree;

void PreOrderTraverse(BiTree ,Status(\*Visit)(TElemType)) {

//采用二叉链表存储结构,Visit是对结点操作的应用函数

InitStack(S);

BiTree p=T;

while( **p!=NULL && !Empty(S)**  ) {

if (p) { Visit(p->data);

**Push(S,p)**  ;

p=p->lchild;

}

else {  **Pop(S,p)** ;

p= **p->rchild**  ;

}

}

}

**六．编写算法（24）**

1．试编写统计二叉树中叶子结点个数的算法。（10分）

**typedef struct BiTNode { // 结点结构**

**TElemType data;**

**struct BiTNode \*lchild, \*rchild; // 左右孩子指针**

**} BiTNode, \*BiTree;**

**int solve(BiTNode \*p){**

**if(!p) return 0;**

**else if(p->lchild==NULL && p->rchild==NULL) return 1;**

**else return solve(p->lchild)+solve(p->rchild);**

**}**

2．设计一个图的数组表示存储结构，并编写采用数组表示法构造一个无向网的算法。（14分）

**#define MAXN 550 //顶点数上限**

**#define INF 0x3f3f3f3f //初始化距离无穷大**

**int G[MAXN][MAXN]; //邻接矩阵**

**int get(int n){**

**for(int i = 0; i < n; i++){**

**for(int j = 0; j < n; j++){**

**G[i][j] = INF; //初始化顶点间的距离**

**}**

**}**

**while(scanf("%d%d%d", &u, &v, &w)!=EOF){**

**if(u < 0 || u >= n || v < 0 || v >= n) return -1;//若读入不符合要求的顶点则返回报错码-1**

**G[u][v] = G[v][u] = w;**

**}**

**return 1;**

**}**