

# 武汉大学

## 二00八年招收硕士研究生入学考试试题

考试科目及代码：            数据结构 (代码：821)

适用专业： 计算机应用技术、计算机软件与理论、计算机系统架构

说明：1. 答题内容写在答题纸上，写在试卷或草稿纸上一律无效。考完后试题随答题纸交回。

2. 考试时间3小时，总分值150分。

### 一、 填空题 (每空1分，共20分)

- 1、数据结构中评价算法好坏的两个重要指标是            和           。
- 2、在具有n个元素的循环队列中，队满时共有            个元素。
- 3、带有头结点Head的单链表为空的条件是           。
- 4、对长度为n的顺序表在等概率的情况下顺序查找成功的平均查找长度为           ，查找失败的平均查找长度为           。
- 5、广义表(a, (a, b), d, e, ((I, j), k)) 的长度是           。
- 6、F是T1, T2, T3三棵树组成的森林，F对应的二叉树为B，已知T1, T2, T3的结点数分别为n1, n2和n3则二叉树B的左子树中有            个结点，右子树中有            个结点。
- 7、一颗含有101个结点的完全二叉树存储在数组A[1..101]中，对 $1 \leq k \leq 101$ ，若A[k]为叶子结点，则k的最小值是           。
- 8、一棵二叉树高度为h，所有结点的度或为0，或为2，则这棵二叉树最少有            结点。
- 9、N个顶点的连通图用邻接矩阵表示时，该矩阵至少有            个非零元素。
- 10、求图的最小生成树有两种算法，            算法适合于求稀疏图的最小生成树。
- 11、AOV网中，结点表示           ，弧表示           。
- 12、在各种查找法中，平均查找长度与结点个数n无关的查找方法是           。
- 13、用二分法检索表(a1, a2, a3, ..., a17)，需要比较2次才能找到的元素是           。
- 14、快速排序的时间复杂度是           。
- 15、在作进栈运算时应先判别栈是否           ，在作退栈运算时应先判别栈是否           。

### 二、 简答题 (20分)

- 1、(6分) 设a, b, c三个元素的进栈次序是a, b, c，符号PUSH与POP分别表示对栈进行一次进栈操作与一次出栈操作。
  - (1) 请分别写出所有可能的出栈序列以及获得该出栈序列的操作序列；
  - (2) 指出不可能出现的出栈序列。

准考证号：

报考学科、专业：

姓名：

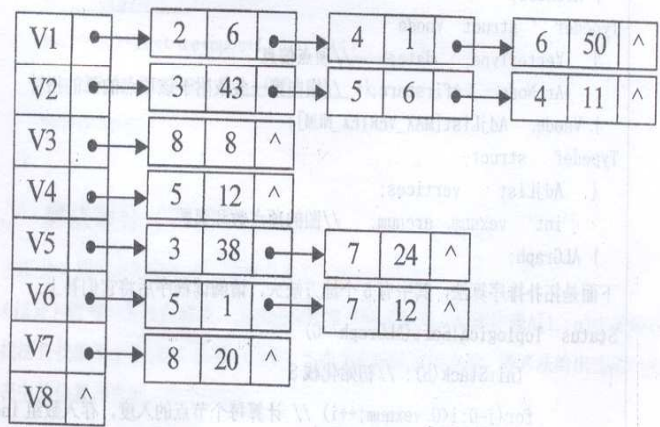
密封线内不要写题

2、(8分) 对于一个有向图,除了进行拓扑排序,还可以采用什么方法判断图中是否存在回路?请简述判断原则。

3、(6分) 有  $n$  个结点的二叉树进行后序遍历,若在算法中使用栈,则栈中最多时有多少项,最少时有多少项?并简单说明理由。

三、综合题 (56分)

- 1、(10分) 证明:对于任意满二叉树,其分枝数  $B=2(n_0-1)$ ,其中  $n_0$  为叶子结点数。
- 2、(8分) 已知一个通信系统中使用的字符为 a, b, c, d, e, f, g, h, 它们的频率分别为 15, 3, 8, 9, 27, 12, 16, 4, 请根据哈夫曼算法构造一棵哈夫曼树,并写出各个字符的编码。
- 3、(8分) 已知关键字序列  $B = \{18, 75, 60, 43, 54, 90, 46, 31, 58, 73, 15, 34\}$ , 采用哈希函数  $H(K) = K \% 13$  实现散列存储,并以线性探测法解决冲突,试写出散列表,并计算平均查找长度。
- 4、(20分) 下图是带权有向图 G 的邻接表表示法,其中出边表中的每个结点均含有三个字段,依次为边的另一个顶点在顶点表中的序号、边的权值和指向下一个边结点的指针。求:
  - (1) 每个顶点的入/出度;
  - (2) 邻接矩阵;
  - (3) 以结点 V1 出发深度遍历图 G 所得的结点序列;
  - (4) 以结点 V1 出发广度遍历图 G 所得的结点序列;
  - (5) 从结点 V1 到结点 V8 的最短路径;
  - (6) 从结点 V1 到结点 V8 的关键路径;
  - (7) 写出至少 2 个拓扑序列。



5. (10分) 排序有很多方法, 设一数组中原有数据如下: 15, 13, 20, 18, 12, 60。下面是一组由不同排序方法进行一遍排序后的结果。在空格中填写对应的排序方法。

- (1) ( ) 排序一遍排序后的结果为: 12, 13, 15, 18, 20, 60
- (2) ( ) 排序一遍排序后的结果为: 13, 15, 18, 12, 20, 60
- (3) ( ) 排序一遍排序后的结果为: 13, 15, 20, 18, 12, 60
- (4) ( ) 排序一遍排序后的结果为: 12, 13, 20, 18, 15, 60
- (5) ( ) 排序一遍排序后的结果为: 13, 15, 18, 20, 12, 60

#### 四、 算法填空 (每个空 2 分, 共 30 分)

1、假设用带头结点的单循环链表表示队列, 队尾结点指针为 R, 下面为非空队列出队的算法, 请将算法填写完整。

```
Void OutQueue (LinkList *R, ElemType &e)
{
    LinkList p;
    (1) _____: e=p->data; R->next->next=(2) _____;
    if( (3) _____ ) R=R->next;
    free(p);
}
```

2、下面给出了有向图的数据结构定义:

```
#define MAX_VERTEX_NUM 20
typedef struct ArcNode
{
    int adjvex; //该弧所指向的顶点位置
    struct ArcNode *nextarc; //指向下一条弧的指针
} ArcNode;
typedef struct Vnode
{
    VertexType data; //顶点信息
    ArcNode *firstarc; //指向第一条依附于该顶点的弧的指针
} Vnode, AdjList[MAX_VERTEX_NUM];
typedef struct
{
    AdjList vertices;
    int vexnum, arcnum; //图的顶点数和弧数
} ALGraph;
```

下面是拓扑排序算法, 其中有 5 个地方缺失, 请阅读程序后将它们补上。

```
Status TopologicalSort(ALGraph G)
{
    InitStack(S); //初始化栈 S
    for(i=0; i<G.vexnum; ++i) // 计算每个节点的入度, 存入数组 indegree
    {
        (4) _____; p=G.vertices[i].firstarc;
        while(p) { indegree[i]++; (5) _____; }
    }
}
```

```

        if( (6) ) push(S, i); //入栈
    }
    (7) ;
    while(!StckEmpty(S)) // 判断栈是否为空
    {
        pop(S, i); //出栈
        printf(i, G.vertices[i].data); ++count;
        for( (8) ; p ; p=p->nextarc)
        {
            k=p->adivex; if( (9) ) push(S, k);
        }
        if( (10) ) return ERROR; // 该图存在环
        else return OK;
    }
    
```

3、下面程序的作用是实现由已知某二叉树的前序遍历和中序遍历序列，生成一棵用二叉链表表示的二叉树并打印出后序遍历序列，请写出程序所缺的语句。

提示：二叉树中结点数据域为 data（字符型），左孩子右孩子指针域分别为 left 和 right。

```

TNODE *restore(char *ppos, char *ipos, int n)
// ppos 和 ipos 分别存放二叉树的前序遍历和中序遍历序列，n 表示结点数
{
    TNODE *ptr; char *rpos; int k;
    if(n<=0) return NULL;
    ptr->data= (11) ;
    for( (12) ; rpos<ipos+n;rpos++) if(*rpos==*ppos) break;
    k= (13) ;
    ptr->left=restore(ppos+1, (14) , k);
    ptr->right=restore( (15) +k, rpos+1, n-1-k);
    return ptr;
}
    
```

五、 算法设计 (24分)

- 1、(12分) 编写递归算法求二叉树中度为 1 的结点个数。
- 2、(12分) 请写一非递归算法，该算法在按值严格递增排列的顺序表 A[1..n] 中采用折半查找法查找值不小于 item 的最小元素。若表中存在这样的元素，则算法给出该最小元素在表中的位置，否则，给出信息 0。