

# 武汉科技学院

## 2007 年招收硕士学位研究生试卷

科目代码 406

科目名称 数据结构 (A 卷)

考试时间 2007 年 1 月 21 日下午

报考专业 计算机应用技术

- 1、试题内容不得超过画线范围，试题必须打印，图表清晰，标注准确。
- 2、试题之间不留空格。
- 3、答案请写在答题纸上，在此试卷上答题无效。

| 题号 | 一 | 二 | 三 | 四 | 五 | 六 | 七 | 八 | 九 | 十 | 十一 | 得分 |
|----|---|---|---|---|---|---|---|---|---|---|----|----|
| 得分 |   |   |   |   |   |   |   |   |   |   |    |    |

本试卷总分 150 分，考试时间 3 小时。

### 一、单项选择题（每题 2 分，共 50 分）

- 1、数据元素之间存在一对一关系的数据结构是\_\_\_\_\_。  
A. 集合      B. 线性结构      C. 树形结构      D. 图状结构
- 2、下面程序段的时间复杂度为\_\_\_\_\_。  

```
for(i=0; i<n; i++)
    for(j=0; j<m; j++)
        A[i][j]=0;
```

A.  $O(m)$       B.  $O(n)$       C.  $O(m*n)$       D.  $O(m/n)$
- 3、带头结点的单链表 head 为空的判断条件是\_\_\_\_\_。  
A. head == NULL      B. head != NULL  
C. head->next == head      D. head->next == NULL
- 4、在单链表中，指针 p 指向结点 A，若要删除 A 之后的结点（存在），则指针的操作方式为\_\_\_\_\_。  
A. p->next = p->next->next;      B. p = p->next;  
C. p = p->next->next;      D. p->next = p;

- 5、在双向循环链表中，在 p 所指的结点之后插入指针 f 所指的新结点，其操作步骤是\_\_\_\_\_。
- A.  $p \rightarrow next = f$ ;  $f \rightarrow prior = p$ ;  $p \rightarrow next \rightarrow prior = f$ ;  $f \rightarrow next = p \rightarrow next$ ;  
 B.  $p \rightarrow next = f$ ;  $p \rightarrow next \rightarrow prior = f$ ;  $f \rightarrow prior = p$ ;  $f \rightarrow next = p \rightarrow next$ ;  
 C.  $f \rightarrow prior = p$ ;  $f \rightarrow next = p \rightarrow next$ ;  $p \rightarrow next = f$ ;  $p \rightarrow next \rightarrow prior = f$ ;  
 D.  $f \rightarrow prior = p$ ;  $f \rightarrow next = p \rightarrow next$ ;  $p \rightarrow next \rightarrow prior = f$ ;  $p \rightarrow next = f$ ;
- 6、设有一个栈，元素依次进栈的顺序为 a, b, c, d, e。下面\_\_\_\_\_是不可能的出栈序列。
- A. a, b, c, d, e      B. b, c, d, e, a  
 C. e, a, b, c, d      D. e, d, c, b, a
- 7、设 C 语言数组 Data[m] 作为循环队列的存储空间，front 为队头指针，rear 为队尾指针，则执行出队操作的语句是\_\_\_\_\_。
- A.  $front = front + 1$ ;      B.  $front = (front + 1) \% m$ ;  
 C.  $rear = rear + 1$ ;      D.  $rear = (rear + 1) \% m$ ;
- 8、一个队列的入队顺序是 a, b, c, d, 则出队顺序是\_\_\_\_\_。
- A. a, b, c, d      B. b, c, d, a  
 C. d, c, b, a      D. c, d, a, b
- 9、下列四个序列中，哪一个是堆\_\_\_\_\_。
- A. 75,65,30,15,25,45,20,10      B. 75,65,45,10,30,25,20,15  
 C. 75,45,65,30,15,25,20,10      D. 75,45,65,10,25,30,20,15
- 10、已知串 S="teacher", T="ache", 则执行子串定位函数 Index(S, T, 2), 后的函数返回值是\_\_\_\_\_。
- A. 2      B. 6      C. 0      D. 3
- 11、设 S="ABC" 则 StrLength 为求串长的函数，则 StrLength(S) = \_\_\_\_\_。
- A. 1      B. 2      C. 3      D. 4
- 12、若一棵二叉树具有 10 个度为 2 的结点，5 个度为 1 的结点，则度为 0 的结点个数是\_\_\_\_\_。
- A. 9      B. 11      C. 15      D. 不确定

- 13、 二叉树的先序遍历和中序遍历如下，先序遍历：EFHIGJK；中序遍历：HFIEJKG。该二叉树根的右子树的根是\_\_\_\_\_。
- A. E          B. F          C. G          D. H
- 14、 n 个结点的线索二叉树上含有的线索数为\_\_\_\_\_。
- A. n          B. 2n          C. n-1          D. n+1
- 15、 下述编码中，\_\_\_\_\_不是前缀码。
- A. (00, 01, 10, 11)          B. (0, 1, 00, 11)
- C. (0, 10, 110, 111)          D. (1, 01, 000, 001)
- 16、 设有向图的顶点个数为 n，则该图最多可以有\_\_\_\_\_条弧。
- A. n-1          B.  $n(n-1)/2$           C.  $n(n-1)$           D.  $n^2$
- 17、 一个 n 个顶点的连通无向图，其边的个数至少为\_\_\_\_\_。
- A. n-1          B. n          C. n+1          D.  $n*(n-1)$
- 18、 已知有向图  $G=(V,E)$ ，其中  $V=\{V_1,V_2,V_3,V_4,V_5,V_6,V_7\}$ ，  
 $E=\{<V_1,V_2>, <V_1,V_3>, <V_1,V_4>, <V_2,V_5>, <V_3,V_5>, <V_3,V_6>, <V_4,V_6>, <V_5,V_7>, <V_6,V_7>\}$ ，G 的拓扑序列是\_\_\_\_\_。
- A.  $V_1,V_3,V_4,V_6,V_2,V_5,V_7$           B.  $V_1,V_3,V_2,V_6,V_4,V_5,V_7$
- C.  $V_1,V_3,V_4,V_5,V_2,V_6,V_7$           D.  $V_1,V_2,V_5,V_3,V_4,V_6,V_7$
- 19、 若查找每个记录的概率均等，则在具有 n 个数据元素的顺序表中采用顺序查找法查找一个记录，其平均查找长度 ASL 为\_\_\_\_\_。
- A.  $(n-1)/2$           B.  $n/2$           C.  $(n+1)/2$           D. n
- 20、 设有一组记录的关键字为{19, 14, 23, 1, 68, 20, 84, 27, 55, 11, 10, 79}，用链地址法构造散列表，散列函数为  $H(key) = key \text{ MOD } 13$ ，散列地址为 1 的链中有个\_\_\_\_\_记录。
- A. 1          B. 2          C. 3          D. 4
- 21、 m 阶 B-树是一棵\_\_\_\_\_。
- A. m 叉排序树          B. m 叉平衡排序树
- C. m-1 叉平衡排序树          D. m+1 叉平衡排序树

- 22、就平均性能而言，目前最好的内部排序方法是\_\_\_\_\_排序法。  
A. 直接插入排序    B. 希尔排序    C. 冒泡排序    D. 快速排序
- 23、对  $n$  个记录的文件进行堆排序，最坏情况下的执行时间是\_\_\_\_\_。  
A.  $O(\log_2 n)$     B.  $O(n)$     C.  $O(n\log_2 n)$     D.  $O(n^2)$
- 24、若需在  $O(n\log_2 n)$  的时间内完成对数组的排序，且要求排序是稳定的，则可选的排序方法是\_\_\_\_\_。  
A. 快速排序    B. 堆排序    C. 归并排序    D. 直接插入排序
- 25、表达式  $a*(b+c)-d$  的后缀表达式是\_\_\_\_\_。  
A.  $abcd*+-$     B.  $abc+*d-$     C.  $abc*+d-$     D.  $abcd+*-$

## 二、填空题（每空 1 分，共 20 分）

- 数据的物理结构分为\_\_\_\_\_（1）\_\_\_\_\_的表示和\_\_\_\_\_（2）\_\_\_\_\_的表示。
- 一个算法具有 5 个特性：\_\_\_\_\_（3）\_\_\_\_\_、\_\_\_\_\_（4）\_\_\_\_\_、\_\_\_\_\_（5）\_\_\_\_\_、有零个或多个输入、有一个或多个输出。
- 在一个长度为  $n$  的顺序表中第  $i$  个元素 ( $1 \leq i \leq n$ ) 之前插入一个元素时，需向后移动\_\_\_\_\_（6）\_\_\_\_\_个元素。
- 对于一个具有  $n$  个结点的单链表，在已知的结点  $*p$  后插入一个新结点的时间复杂度为\_\_\_\_\_（7）\_\_\_\_\_，在给定值为  $x$  的结点后插入一个新结点的时间复杂度为\_\_\_\_\_（8）\_\_\_\_\_。
- 最大容量为  $n$  的循环队列，队尾指针是  $rear$ ，队头指针是  $front$ ，则队空的条件是\_\_\_\_\_（9）\_\_\_\_\_。
- 顺序栈用  $data[M]$  存储数据，栈顶指针是  $top$ ，已知栈未满，则值为  $x$  的元素入栈的操作是\_\_\_\_\_（10）\_\_\_\_\_。
- 零个字符的串为空串，其长度等于\_\_\_\_\_（11）\_\_\_\_\_。
- 两个字符串相等的充分必要条件是\_\_\_\_\_（12）\_\_\_\_\_。
- 具有 32 个结点的完全二叉树的深度为\_\_\_\_\_（13）\_\_\_\_\_。
- 如果结点  $A$  有 3 个兄弟，而且  $B$  是  $A$  的双亲，则  $B$  的度是\_\_\_\_\_（14）\_\_\_\_\_。
- 直接插入排序在最好的情况下时间复杂度为\_\_\_\_\_（15）\_\_\_\_\_。

- 12、Prim（普里姆）算法的时间复杂度是  $O(n^2)$ ，它适用于求\_\_\_\_（16）\_\_\_\_网的最小生成树；Kruskal（克鲁斯卡尔）算法的时间复杂度是  $O(n \log n)$ ，它适用于求\_\_\_\_（17）\_\_\_\_网的最小生成树。
- 13、高度为 4 的 3 阶 B-树中，最多有\_\_\_\_（18）\_\_\_\_个关键字。
- 14、对于具有 144 个记录的文件，若采用分块查找法，且每块长度为 8，则平均查找长度为\_\_\_\_（19）\_\_\_\_。
- 15、基数排序是将多关键字分配排序的\_\_\_\_（20）\_\_\_\_位优先法用于单关键字的排序方法。

### 三、算法填空题（每空 2 分，共 20 分）

- 1、构造一个空的顺序表。

```
#define OK 1
#define OVERFLOW -2
#define LIST_INIT_SIZE 100    // 表存储空间的初始分配量
typedef struct {
    ElemType *elem;           // 存储空间基址
    int length;               // 当前长度
    int listsize;             // 存储容量
} SqList;
typedef int Status;
Status InitList_Sq(SqList &L)    // 构造一个空的线性表 L
{
    L.elem = _____(1)_____;    // 分配 LIST_INIT_SIZE 个数据元素的空间
    if ( _____(2)_____ ) exit(OVERFLOW); // 存储分配失败
    L.length = 0;              // 长度为 0
    L.listsize = LIST_INIT_SIZE; // 初始存储容量
    return OK;
} // InitList_Sq
```

2、在顺序线性表 L 中删除第 i 个元素，并用 e 返回其值。

```
#define OK 1
#define ERROR 0

typedef struct {
    ElemType *elem;          // 存储空间基址
    int length;              // 当前长度
    int listsize;            // 存储容量
} SqList;

typedef int Status;

Status ListDelete_Sq (SqList &L, int i, ElemType &e)
{
    if ((i < 1) || (i > L.length)) return ERROR; // 删除位置不合法
    p = &(L.elem[i-1]);          // p 为被删除元素的位置
    e = *p;                      // 被删除元素的值赋给 e
    q = L.elem+L.length-1;      // 表尾元素的位置
    for (++p; p <= q; ++p)
        _____ (3) _____ // 被删除元素之后的元素左移
        _____ (4) _____ // 表长减 1
    return OK;
} // ListDelete_Sq
```

3、在带头结点的单链表中第 i 个位置前插入元素。

```
#define OK 1
#define ERROR 0
typedef struct LNode{
    ElemType data;
    struct LNode *next;
} LNode, *LinkList;
typedef int Status;
```

```

Status ListInsert_L(LinkList &L, int i, ElemType e)
{
    p = L; j = 0;
    while (p && j < i-1)          // 寻找第 i-1 个结点
    {
        p = p->next; ++j;
    }
    if (!p || j > i-1) return ERROR;          // i 小于 1 或者大于表长
    s = (LinkList) malloc ( sizeof (LNode)); // 生成新结点
    s->data = e;
    _____ (5) _____          // 插入 L 中
    _____ (6) _____
    return OK;
} // LinstInsert_L

```

#### 4、中序打印二叉树。

```

typedef int TElemType;
typedef struct BiTNode {
    TElemType data;
    struct BiTNode *lchild, *rchild; // 左右孩子指针
} BiTNode, *BiTree;
void InOrderTraverse(BiTree T)
{
    if(T)
    {
        InOrderTraverse(T->lchild); //中序打印左子树
        _____ (7) _____ //打印根结点的值
        _____ (8) _____ //中序打印右子树
    }
}

```

5、将增序有序的 SR[i..m] 和 SR[m+1..n] 归并为非递减的 TR[i..n]。

```
typedef int RedType;
void Merge(RedType SR[ ], RedType &TR[ ], int i, int m, int n )
{
    for(j = m+1, k=i; i<=m&&j<=n; ++k)
    {
        if( SR[i]<SR[j] ) TR[k]=SR[i++];
        else _____ (9) _____
    }
    if(i<=m)
        while (i<=m) TR[k++]=SR[i++];
    if(j<=n)
        _____ (10) _____ TR[k++]=SR[j++];
}
```

#### 四、计算题（每题 4 分，共 20 分）

1、分别计算下面两个程序段的时间复杂度。

(1) i=1;

While(i<=n)

i=i\*3;

(2) for (i=1; i<=n; ++i)

for (j=1; j<=n; ++j)

{

c[i,j] = 0;

for (k=1; k<=n; ++k)

c[i,j] += a[i,k]\*b[k,j];

}

2、假设 1 个字符占 8 位，1 个指针占 32 位。已知串的块链存储表示如下：

```
#define CHUNKSIZE 16 // 一块中字符的个数

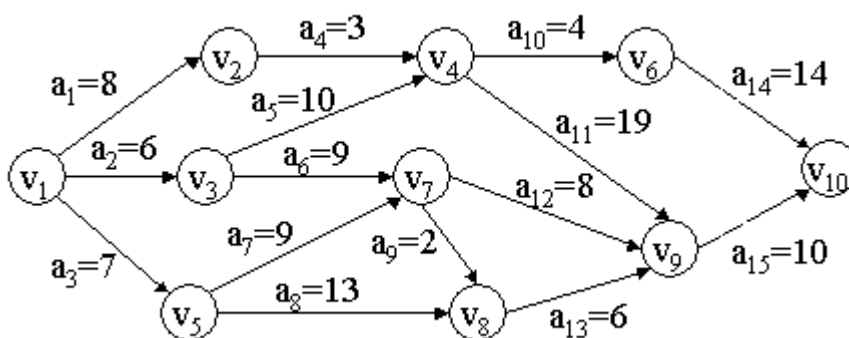
typedef struct Chunk { // 结点结构
    char ch[CHUNKSIZE];
    struct Chunk *next;
} Chunk;
```

计算此块链存储结构的存储密度。

3、以 {4, 5, 6, 7, 8} 作为叶子结点的权值构造哈夫曼树，计算此树的带权路径长度。

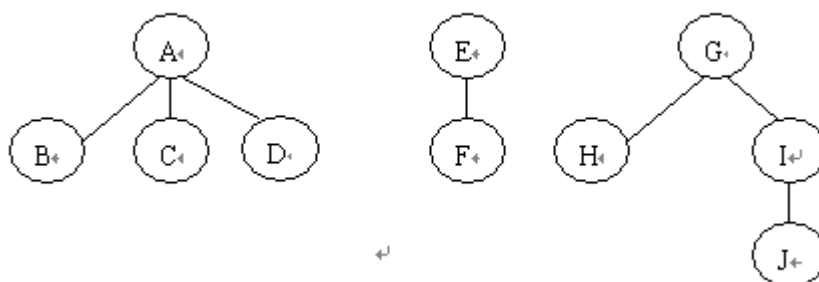
4、已知序列 {46, 88, 45, 39, 70, 58, 101, 10, 66, 34}，建立一个二叉排序树。计算在等概率情况下查找成功的平均查找长度。

5、下图是以边为活动的网 (AOE\_网)，以  $V_1$  为开始点，以  $V_{10}$  为终止点，计算该 AOE\_网的关键路径长度 (即，从开始点到终止点的最长路径长度)。

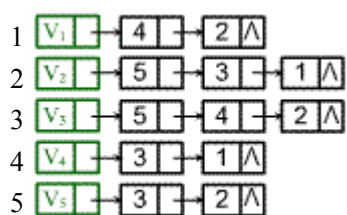


## 五、应用题 (每题 5 分，共 30 分)

1、将下图所示的森林转化为二叉树。



- 2、已知一棵二叉树的前序遍历序列是 ABDEGCFHIJ，中序遍历序列是 DBGEAHFIJC，①构造出该二叉树；②给出该二叉树的后序遍历序列。
- 3、已知无向图的邻接表如下，①画出这个图；②写出该图的邻接矩阵 A；  
③根据邻接表，写出从顶点  $V_1$  出发，深度优先遍历该图所得到的顶点序列。



提示：

$$A[i][j] = \begin{cases} 1 & \text{若}(v_i, v_j) \text{或} \langle v_i, v_j \rangle \text{是} E(G) \text{中的边} \\ 0 & \text{若}(v_i, v_j) \text{或} \langle v_i, v_j \rangle \text{不是} E(G) \text{中的边} \end{cases}$$

- 4、已知一组关键字为 (19, 14, 23, 01, 68, 20, 84, 27, 55, 11, 79, 40)，表长 15，请按哈希函数  $H(\text{key}) = \text{key} \bmod 13$  和线性探测处理冲突。①构造该哈希表；②说明查找关键字 27 必须依次和表中哪些关键字比较。
- 5、给出一组关键字序列 {12, 2, 16, 30, 8, 28, 4, 10, 20, 6, 18}，写出用下列算法按从小到大排序，第一趟结束时的序列。
- ① 希尔排序 (第一趟排序的增量为 5)；
  - ② 快速排序 (选第一个记录为枢轴(分隔))；
  - ③ 链式基数排序 (基数为 10)。
- 6、给出下列递归过程的执行结果

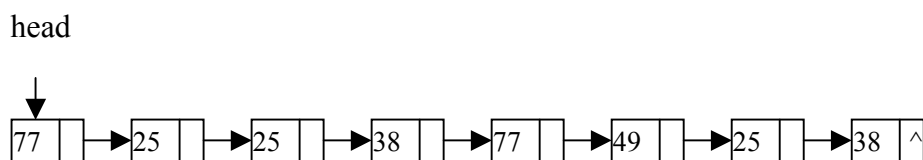
```
void unknown ( int w )
{
    int i;
    if ( w ) {
        unknown ( w-1 );
        for ( i = 1; i <= w; i++ ) printf("%d",w);
        printf("\n");
    }
}

void main()
{
    unknown(4);
}
```

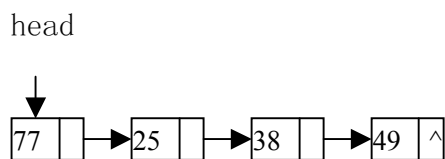
## 五、算法设计题（共 10 分）

已知一个不带头结点的单链表 `head`，结点结构为 `Node`。写出一个通用的算法，删除链表中值相同的结点（如果有值相同的结点只保留首次出现的结点）。

例如，删除前：



删除后：



预编译命令：

```
#define NULL 0 //NULL 表示空指针
```

结点结构：

```
typedef struct node
{
    int data;
    struct node *next;
}Node;
```

函数首部：

```
Node * Expurgate (Node * head)
```

共 11 页 第 11 页

