

## 四川轻化工大学 2021 年研究生招生考试业务课试卷

(满分: 150 分, 所有答案一律写在答题纸上)

适用专业: 0811 控制科学与工程、0854 电子信息 ✓

考试科目: 816 数据结构与算法 A 卷 ✓

考试时间: 3 小时 ✓

### 一、 选择题 (每题 2 分, 共 60 分)

- 下列不属于数据的存储结构的是( )。  
A. 逻辑结构 B. 顺序存储结构 C. 链式存储结构 D. Hash 存储结构
- 算法分析的两个主要方面是( )。  
A. 正确性和简明性 B. 空间复杂度和时间复杂度  
C. 可读性和文档性 D. 数据复杂性和程序复杂性
- 下面关于算法说法错误的是( )。  
A. 算法原地工作的含义是指不需要其他计算机辅助实现  
B. 复杂度  $O(n)$  的算法在时间上不一定总是优于复杂度  $O(n^2)$  的算法  
C. 同一个算法, 不同的程序员实现, 用低级语言实现的效率不一定比高级语言实现效率高  
D. 工程总不一定总是选择算法时间复杂度低的算法
- 以下程序段的时间复杂度正确表示是( )。  
for (i=1; i<n; i++){  
    y=y+1;  
    for (j=0; j<=(2\*n); j++){  
        x++;  
    }  
}  
A.  $O((2n)^2)$  B.  $n*(n+1)/2$  C.  $O(n^2)$  D.  $2n^2$
- 某线性表中最常用的操作是在最后一个元素之后插入一个元素和删除第一个元素, 则采用( )存储方式最节省运算时间。  
A. 仅有头指针的单循环链表 B. 仅有尾指针的单循环链表  
C. 单链表 D. 双链表
- 在表长为  $n$  的顺序表中, 算法时间复杂度为  $O(1)$  的操作为( )。  
A. 在第  $i$  个元素前插入一个元素  
B. 删除第  $i$  个元素

- C. 查找其值与给定值相等的一个元素  
D. 在表尾插入一个元素
- 7、关于线性表下面的叙述正确的是( )。
- A. 线性表在用单链表存储时, 查找第  $i$  个元素的时间同  $i$  的值成正比  
B. 线性表在顺序存储时, 查找第  $i$  个元素的时间同  $i$  的值成正比  
C. 线性表在顺序存储时, 查找第  $i$  个元素的时间同第  $i$  个元素的概率成正比  
D. 线性表在用循环双链表存储时, 查找第  $i$  个元素的时间比数组快
- 8、对于顺序表, 以下说法错误的是( )。
- A. 顺序表的所有存储结点按相应数据元素间的逻辑关系决定的次序依次排列  
B. 顺序表的特点是: 可以随机存取元素  
C. 顺序表是用一维数组实现的线性表, 数组的下标可以看成是元素的绝对地址  
D. 顺序表的特点是: 逻辑上相邻的元素, 存储在物理位置也相邻的单元中
- 9、顺序表的插入算法中, 当  $n$  个空间已满时, 可再申请增加分配  $m$  个空间, 若申请失败, 则说明系统没有( )可分配的存储空间。
- A.  $n+m$  个      B.  $n+m$  个连续      C.  $m$  个      D.  $n+1$  个离散
- 10、设有一个栈, 元素依次进栈的顺序为 A、B、C、D、E。下列( )是不可能的出栈序列。
- A. E,A,B,C,D      B. A,B,C,D,E      C. B,C,D,E,A      D. E,D,C,B,A
- 11、在循环队列中, 若  $front$  与  $rear$  分别表示对头元素和队尾元素的位置, 则判断循环队列空的条件是( )。
- A.  $front==0$       B.  $front==rear+1$   
C.  $rear==front+1$       D.  $front==rear$
- 12、对 5 个不同的数据元素进行直接插入排序, 最多需要进行的比较次数是( )。
- A. 8      B. 10      C. 15      D. 25
- 13、在原始记录基本有序时, 以下说法正确的是( )。
- A. 对冒泡排序没有影响      B. 对直接插入排序没有影响  
C. 对直接选择排序没有影响      D. 对 Shell 排序没有影响
- 14、设有两个串  $p$  和  $q$ , 其中  $q$  是  $p$  的子串, 求  $q$  在  $p$  中首次出现的位置的算法称为( )。
- A. 求子串      B. 串联接      C. 串匹配      D. 求串长
- 15、设有一个二维数组  $A[m][n]$ , 设  $A[0][0]$  存放在 644,  $A[2][2]$  存放在 676, 每个元素占一个空间, 问  $A[3][3]$  存放在( )位置。
- A. 692      B. 693      C. 689      D. 690
- 16、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

- 数是( )。
- A.9          B.11          C.15          D.不确定
- 17、对具有 1000 个结点的完全二叉树按从上到下、从左到右的顺序依次编号 1, 2, 3, ..., N, 则第 501 个结点的双亲结点编号为: ( )。
- A.250          B.251          C.500          D.不确定
- 18、利用二叉链表存储树, 则根结点的右指针是( )。
- A.指向最左孩子          B.指向最右孩子          C.空          D.不确定
- 19、在下列存储形式中, ( )不是树的存储形式?
- A.双亲表示法          B.孩子链表表示法  
C.孩子兄弟表示法          D.顺序存储表示法
- 20、设 F 是一个森林, B 是由 F 变换得的二叉树。若 F 中有 n 个非终端结点, 则 B 中右指针域为空的结点有( )个。
- A.n-1          B.n          C.n+1          D.n+2
- 21、若一棵二叉树的前序遍历序列和后序遍历序列分别为 abcd 和 dcba, 则该二叉树的中序遍历序列不会是( )。
- A.abcd          B.bcda          C.cbda          D.dcba
- 22、一个有 N 个顶点和 N 条边的无向图, 一定是( )。
- A.连通的          B.不连通的          C.无环的          D.有环的
- 23、若图的邻接矩阵中主对角线上的元素全是 0, 其余元素全是 1, 则可以断定该图一定( )。
- A.是无向图          B.不是带权图  
C.是有向图          D.是完全图
- 24、邻接矩阵是对称矩阵的图为( )。
- A.有向网          B.无向网  
C.有向网和无向网都是          D.有向网和无向网都不是
- 25、用邻接表存储的图的深度优先遍历算法, 类似于树的( )。
- A.中序遍历          B.先序遍历          C.后序遍历          D.按层次遍历
- 26、任何一个无向连通图的最小生成树( )。
- A.有一棵或多棵          B.只有一棵          C.一定有多棵          D.可能不存在
- 27、折半查找和二叉排序树查找的时间性能( )。
- A.相同          B.有时不同          C.完全不同          D.无法比较
- 28、散列 (Hash) 查找一般适用于( )情况下的查找。
- A.查找表为链表  
B.查找表为有序表

- C.关键字集合比地址集合大得多  
D.关键字集合与地址集合之间存在对应关系
- 29、 假设有  $k$  个关键字互为同义词, 若用线性探查法把这  $k$  个关键字存入, 至少要进行探查次数是( )。
- A. $k-1$       B. $k$       C. $k+1$       D. $k(k+1)/2$
- 30、 在采用链地址法处理冲突所构成的散列表上查找某一个关键字, 则查找成功的情况下, 所探测的这些位置上的键值( )。
- A.一定都是同义词      B.不一定是同义词  
C.都相同      D.一定都不是同义词

## 二、填空题 (每空 2 分, 共 28 分)

- 1、 双链表的两个域分别是 `prior` 和 `next`, 完成在双循环链表结点 `p` 之前插入 `s` 的操作是(不填分号)\_\_\_\_\_ ; \_\_\_\_\_ ; `s->next=p`; `p->prior=s`;
- 2、 对于一个以顺序实现的循环队列 `Q[0...m-1]`, 队首、队尾指针分别为 `Q.f` 和 `Q.r`, 其队列长度是\_\_\_\_\_ (`Q` 为队列变量)
- 3、 一组记录的关键码为 (36,29,38,18,39,20,36,64), 则利用快速排序的方法, 以第一个记录为基准得到的一次划分结果为\_\_\_\_\_,  $dk=3$  的第一趟希尔排序结果为\_\_\_\_\_。
- 4、 已知广义表为  $D=(a,(b,(),c),((d),e))$ , 该表的长度为\_\_\_\_\_。
- 5、 由 3 个结点可以构造出\_\_\_\_\_ 种不同的二叉树。
- 6、 深度为  $h$  的满  $m$  叉树的第  $k$  层有\_\_\_\_\_ 个结点。 ( $1 \leq k \leq h$ )
- 7、 设有 5 个结点的权值分别为 3,4,5,6,7, 根据这些权值构造一棵 Huffman 树, 则该树的带权路径长度 WPL 为\_\_\_\_\_。
- 8、 已知一棵完全二叉树的第 7 层 (设根为第 1 层) 有 8 个叶子结点, 则该完全二叉树的结点个数最少是\_\_\_\_\_。
- 9、 在有  $N(N>1)$  个顶点的有向图中, 每个顶点的度最大可达\_\_\_\_\_。
- 10、 假设一个图有  $n$  个顶点、 $e$  条边, 则在最小生成树算法中, 普利姆(prim)算法的时间复杂度为\_\_\_\_\_, 因此适合于求\_\_\_\_\_ (选填稀疏、稠密) 图的最小生成树。
- 11、 对于长度为 10 的有序顺序表, 若查找每个元素的概率相等, 则用折半查找法查找顺序表中任意元素的查找成功的平均查找长度为\_\_\_\_\_ (保留一位小数)。

## 三、程序填空题 (每空 3 分, 共 42 分)

- 1、 在顺序线性表 `L` 中删除第  $i$  个元素 ( $1 \leq i$ ), 并用 `e` 返回其值。完成该算法。

```
typedef struct {
    ElemType *elem;      //存储空间基址
```

```

        int length;                //当前长度
        int listsize;              //当前分配的存储容量
    } SqList;
    Status ListDelete_sq(SqList &L, int i, ElemType &e) {
        if(i < 1 || ___ (1) ___) return ERROR;    // i 值合法性判断
        e=L.elem[i-1];                            // 被删元素的值赋给 e
        for(j=i; ___ (2) ___; j++)
            ___ (3) ___ =L.elem[j];
        --L.length;
        return OK;
    }

```

- 2、已知单链表的表首指针为 head，下面的函数 delete 是从单链表中删除指针为 p 的结点（假定单链表中一定存在被删结点），并返回新的表首指针。请完成如下程序：

```

typedef struct LinkNode {
    int data;
    struct LinkNode *next;
}Node;
Node *delete(Node *head, Node *p) {
    Node *pf;
    if(head==p) { // 若被删结点是第一个结点
        head= ___ (1) ___;
        free(p);
    }
    else {
        for(pf=head; pf->next!=p; pf=pf->next); // 定位
        ___ (2) ___ = p->next;
        free(p);
    }
    return ___ (3) ___;
}

```

- 3、用栈实现判断回文数，填空；

栈的操作函数如下：

将 x 入栈 s: push(s,x); 出栈,并赋值给 x:

pop(s,x) 如果栈空则 pop 函数返回为假

//ch 接收回文字符数组的基地址, length 接收回文字符串长度

```
int HuiWen(char *ch, int length){
    Stack s; char x;
    int i,j, m = length/2;
    InitStack(s); //初始化栈 s
    for(i=0;i<m;++i) ___ (1) ___;
    for(j= ___ (2) ___; j<length; j++){ //从一半后开始
        if(___ (3) ___ && x!=ch[j]) break;
    }
    if(j !=length) return ERROR; //不是回文, 返回 0
    return OK; //否则返回 1
}
```

4、直接插入排序算法实现填空

线性表 L, 其成员包括: int length, int r[MAXSIZE]

```
void InsertSort( SqList &L) {
    int i, j; int temp, len=L.length;
    for (i = 1; ___ (1) ___ ; i++) {
        j = i;
        temp =L.r[i]; //待排序元素赋值给临时变量
        //查找插入位置
        while (j >= 0 && ___ (2) ___ ) {
            L.r[j] = L.r[j - 1];
            j--;
        }
        ___ (3) ___; //插入位置已经找到
    }
}
```

5、以二叉链表作为二叉树的存储结构, 编写算法交换二叉树每个结点的左孩子和右孩子。

(说明: 对任何非叶子结点, 要求先访问左子树, 后访问右子树)

```
void ChangeLR(BiTree &T)
{
    BiTree temp;
```

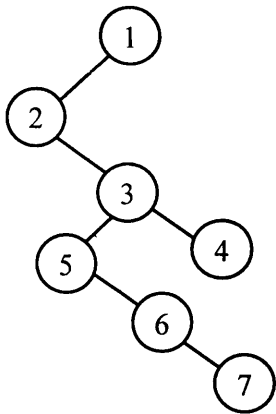
```

if(T==NULL)                // 若为空树
    return;
if(T->lchild==NULL &&__ (1) __) // 若为叶子结点
    return;
else
{
    temp = T->lchild;        // 交换
    T->lchild = T->rchild;
    T->rchild = temp;
}
ChangeLR(__ (2) __);
ChangeLR( 代码与 (2) 类似 );
}

```

四、应用题 (共 20 分)

1、(共 6 分) 已知二叉树如下图所示:



先序遍历该二叉树得到的序列为:

\_\_\_\_\_ (1) \_\_\_\_\_

中序遍历该二叉树得到的序列为:

\_\_\_\_\_ (2) \_\_\_\_\_

将该二叉树转化为树或森林, 请在答题纸上绘制转化结果。

\_\_\_\_\_ (3) \_\_\_\_\_

2、(共 6 分) 已知图的邻接矩阵如下。

	1	2	3	4	5	6
1	0	0	1	0	1	1
2	0	0	0	0	1	0
3	0	0	0	0	0	1
4	1	1	0	0	0	0
5	0	0	1	1	0	0
6	0	0	0	0	0	0

自顶点 1 出发进行深度优先遍历所得的遍历序列是：\_\_\_\_\_ (1) \_\_\_\_\_，

自顶点 1 出发进行广度优先遍历所得的遍历序列是：\_\_\_\_\_ (2) \_\_\_\_\_。

((1)(2) 问答题说明：小序号优先原则，无需任何分隔符)

请在答题纸上绘制该图的逆邻接表。\_\_\_\_\_ (3) \_\_\_\_\_

3、(共 8 分) 从空树开始依次插入如下元素{7, 3, 5, 15, 11, 1, 9, 13}，建立一棵排序二叉树。

(1) 请在答题纸上画该排序二叉树。

(2) 该二叉树是否为平衡二叉树？

(3) 若查找 12，将依次哪些元素比较？

(4) 计算查找成功的平均查找长度和查找不成功的平均查找长度。