

西安电子科技大学

2018 年硕士研究生招生考试初试答题纸

考试科目代码及名称 951 数据结构

考试时间 2017 年 12 月 24 日下午 (3 小时)

答题要求：所有答案（填空题按照标号写）必须写在答题纸上，写在试题上一律作废，准考证号写在指定位置！

一、单项选择题（在下列每小题的备选答案中选出一个正确答案。每空 2 分，共 30 分）

1. 数据的基本单位是（ ）。

- A. 数据结构
- B. 数据元素
- C. 数据项
- D. 文件

2. 在逻辑上可以把数据结构分成（ ）。

- A. 动态结构和静态结构
- B. 紧凑结构和非紧凑结构
- C. 内部结构和外部结构
- D. 线性结构和非线性结构

3. 不带头结点的单链表 head 为空的判定条件是（ ）。

- A. head=NULL
- B. head->next=NULL
- C. head->next=head
- D. head!=NULL

4. 在一个单链表中，若要在指针 p 所指结点之后插入指针 s 所指结点，则执行（ ）。

- A. s->next=p; p->next=s;
- B. s->next=p->next; p->next=s;
- C. s->next=p->next; p=s;
- D. p->next=s; s->next=p;

5. 串是一种特殊的线性表，其特殊性体现在（ ）。

- A. 可以顺序存储
- B. 数据元素是一个字符
- C. 可以连接存储
- D. 数据元素可以是多个字符

6. 二维数组 M 的成员是 6 个字符（每个字符占一个存储单元）组成的串，行下标 i 的范围从 0 到 8，列下标 j 的范围从 0 到 9，则存放 M 至少需要（ ）个存储单元。

951 数据结构 试题 共 8 页 第 1 页

A. 90

B. 180

C. 240

D. 540

7. 当利用大小为 n 的数组顺序存储一个栈时, 假定用 $\text{top} = n$ 表示栈空, 则向这个栈插入一个元素时, 首先应执行 () 语句修改 top 指针。(top 为栈顶指针)

A. top++

B. top--

C. top=0

D. top

8. 设长度为 n 的链队列用单循环链表表示, 若只设头指针, 则入队和出队操作的时间复杂度分别为 ()。

A. O (1) , O (1)

$$B, O(1), O(n)$$
$$C, O(n), O(1)$$
$$D, O(n), O(n)$$

9. 任何一棵二叉树的叶子结点在前序、中序和后序遍历序列中的相对次序 ()。

A. 不发生改变

B. 发生改变

C. 不能确定

D. 以上都不对

10. 线索二叉链表是利用 () 域存储后继结点的地址。

A. Ichild

B. data

C. rchild

D. root

11. 单链表中, 增加头结点的目的是为了 ()。

A. 使单链表至少有一个结点

B. 标示表结点中首结点的位置

C. 方便运算的实现

D. 说明单链表是线性表的链式存储实现

12. 无向图 $G=(V, E)$, 其中: $V=\{a, b, c, d, e, f\}$, $E=\{(a, b), (a, e), (a, c), (b, e), (c, f), (f, d), (e, d)\}$, 对该图进行深度优先遍历, 得到的顶点序列正确的是 ()。

A, a, b, e, c, d, f

B. a, c, f, e, b, d

C. a, e, b, c, f, d

D. a, e, d, f, c, b

13. 设散列表长度为 m , k 为关键字, 用 p 去除 k , 将所得的余数作为 k 的散列地址, 即 $H(k) = k \% p$ 。为了减少发生的冲突的频率, 一般取 p 为 ()。

A. 小于等于 m 的最大偶数

B. m

C. 大于等于 m 的最小素数

D. 小于等于 m 的最大素数

14. 矩阵连乘问题可以用下列哪种方法求解 ()。

A. 贪心算法

B. 分治递归算法

C. 动态规划

D. Kruskal 算法

15. 关于动态规划算法下列说法不正确的是 ()。

- A. 用于求解具有某种最优性质的问题。
- B. 以自顶向下的方式计算最优值。
- C. 适用于动态规划法求解的问题，经分解得到的子问题往往不是相互独立的。
- D. 先求解子问题，再从子问题的解得到原问题的解。

二、判断题（每题 1 分，共 15 分，在答题纸上标明题号，打√或×）

- 1. 算法的时间复杂度取决于问题的规模和待处理数据的初态。
- 2. 顺序表无需为表示结点间的逻辑关系而增加额外的存储空间。
- 3. 对双向链表来说，结点*P 的存储位置既存放在其前趋结点的后继指针域中，也存放在它的后继结点的前趋指针域中。
- 4. 二叉树的子树没有左右次序之分。
- 5. 树型结构中，任何结点可以有多个前驱结点和后继结点。
- 6. 模式匹配的改进算法-KMP 算法的最大特点是指主串的指针不需要回溯。
- 7. 空格串是零个字符组成的串。
- 8. 栈底元素是不能删除的元素。
- 9. 栈是一种对进栈、出栈操作总次数做了限制的线性表。
- 10. 无向图的邻接矩阵是一个对角矩阵。
- 11. 有向图的遍历不可采用广度优先搜索方法。
- 12. 把散列地址不同的结点，争夺同一个后继散列地址的现象称为“冲突”。
- 13. 当一个问题所有子问题都至少需要求解一次时，动态规划算法好于备忘录算法。
- 14. 能够用分治法求解的问题往往具有子问题重叠性质。
- 15. 贪心算法所做的贪心选择是仅在当前状态下做出最好的选择。

三、填空题（每空 2 分，共 30 分）

- 1. 当输入数据非法时，算法也能适当地做出反应或进行处理，而不会产生莫名其妙的输出结果，这称为算法的（1）。
- 2. 不要求逻辑上相邻的结点在物理位置上亦相邻，结点间的逻辑关系是由附加的指针字

段表示的。由此得到的存储表示称为 (2) 存储结构。

3. 一个顺序表的第一个元素的存储地址是 200, 每个元素的长度是 4, 则第 7 个元素的地址是 (3) 。

4. 单链表表示法的基本思想是用 (4) 表示结点间的逻辑关系。

5. 求子串在主串中首次出现的位置的运算称为 (5) 。

6. 二维数组 A 采用行序为主方式存储, 其中行下标 i 的范围从 0 到 10, 列下标 j 的范围从 0 到 5, 每个元素占 4 个存储单元, 并且 A[0][0] 的存储地址是 1000, 则 A[8][4] 的地址是 (6) 。

7. 已知一个栈的输入序列为 1, 2, 3, ..., n, 则其输出序列的第 2 个元素为 n 的输出序列的种数是 (7) 。

8. 一个循环队列 Q 的存储空间大小为 M, 其队头和队尾指针分别为 front 和 rear, 则循环队列中元素的个数为 (8) 。

9. 在二叉链表中空链域有 n 个, 则该链表共有 (9) 个结点。

10. 设某二叉树中度数为 0 的结点数为 N_0 , 度数为 2 的结点数为 N_2 , 则 N_0 、 N_2 之间关系: (10) 。

11. 某二叉树的前序遍历序列是 abdgcefh, 中序序列是 dgbacfh, 其后序序列为 (11) 。

12. 对于采用分块查找的数据表, 要求表是 (12) 。

13. 索引表的索引项的一般形式是: (关键字, (13))。

14. 贪心算法可以求解的问题应具有最优子结构和 (14) 的性质。

15. 一个有 n 个顶点的无向图最多有 (15) 条边。

四、问题求解题 (共 45 分)

1. (7 分) 在如下数组 A 中链接存储了一个线性表, 表头指针为 A[0].next, 试按照下面格式写出该线性表。

A	0	1	2	3	4	5	6	7
data		egg	if		for	go	and	do
next	5	7	4		1	2	0	6

线性表为：(, , , , ,)

2. (7分) 对给定的一组关键字：63, 83, 43, 22, 67, 65, 76, 54, 99, 68, 77, 14, 画出应用归并排序对上述序列进行排序中各趟归并的结果。

第一趟归并后[] [] [] [] [] []

第二趟归并后[] [] [] []

第三趟归并后[] []

第四趟归并后[]

3. (7分) 设有一组关键字序列 (23,34,56,33,45,28,67,11,50,07,47)。散列表长为 15。

(1) 请设计一个适当的散列函数 (采用除留余数法);

(2) 装填因子等于多少?

(3) 画出用拉链法构造的散列表 (链表中插入结点时采用头插法)。

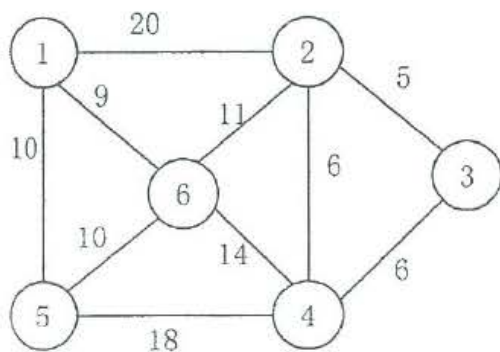
4. (8分) 设将整数 1、2、3、4 依次进栈, 但只要出栈时栈非空, 则可将出栈操作按任何次序夹入其中, 请回答下述问题:

(1) 若入、出栈次序为 Push(1), Pop(), Push(2), Push(3), Pop(), Pop(), Push(4), Pop(), 则出栈的数字序列是怎样的 (这里 Push(i)表示 i 进栈, Pop()表示出栈)?

(2) 能否得到出栈序列 1423 和 1432? 并说明为什么不能得到或者如何得到。

(3) 请分析 1、2、3、4 的 24 种排列中, 那些序列是可以通过相应的入出栈操作得到的。

5. (8分) 已知一个无向图如下图所示, 给出从顶点 1 出发用 Prim (普里姆) 算法生成所有可能的最小树的过程。



6 (8分) 输入一个正整数序列{40, 28, 6, 72, 100, 3, 54, 1, 80, 91, 38}, 建立一棵二叉排序树, 然后删除结点 72, 分别画出该二叉树及删除结点 72 后的二叉树。

五、算法设计题 (共 30 分)

1. (10 分) 假设二叉树采用二叉链表存储结构, 完成一个递归实现的算法, 计算一棵给定二叉树的叶子结点数。

```
typedef int Elemtype;
typedef struct node{
    Elemtype data;
    _____ ① _____ //定义左右子树
}BinNode, *BinTree;

int num=0;

void CountLeaves(BinNode *p, int num){
    //num is the number of leaves
    if(p!=Null){
        if(!(p->lchild)&&!( p->rchild))
            _____ ② _____
            _____ ③ _____ //计算左子树叶子节点个数
            _____ ④ _____ //计算右子树叶子节点个数
        }
    }
```

2. (10 分) 如果希望循环队列中的元素都能得到利用, 则需要设置一个标志域 tag, 并通过 tag 的值为 0 和 1 来区分尾指针和头指针值相同时的队列状态是“空”还是“满”。补全下面与此结构相应的入队列和出队算法。

```
typedef int QElemType; _____ //循环对列中元素的类型
```

```

typedef struct{
    QElemType base[maxsize];
    int    front;
    int    rear;
    int    tag;
}CyQueue;

Status EnCyQueue(CyQueue *Q , int x)           //带 tag 域的循环对列入队算法
{
    if(Q.front == Q.rear&&Q.tag == 1)           //tag 域的值为 0 表示“空”，1 表示“满”
        return OVERFLOW;

    Q.rear =(Q.rear+1)%MAXSIZE;
    Q.base[ ① ]=x;
    if( ② )
        Q.tag=1;
}

Status DeCyQueue(CyQueue *Q , int x)           //出队算法，出队的值赋给 x
{
    if(Q.front ==Q.rear&&Q.tag==0)
        return INFEASIBLE;

    ③ ;
    ④ ;
    if( ⑤ )
        Q.tag=0;
}

```


3 (10 分) 已知有 m 个顶点的无向图, 采用行序为主序的邻接矩阵结构存储, 完成下列算法:

(1) 计算图中有多少条边。

```
int edges(int *a, int m)    //a 为存储矩阵的结构
{
    int i, j;
    int e=0;
    for(i=0; i<m; i++)
        for(j=0; ① ; j++)
            if(② )
                ③ ;
    return e;
}
```

(2) 判断第 i 个顶点和第 j 顶点之间是否有边相连。

```
int isadj(int i, int j, int *a, int m)
{return ④ ;}
```


西安电子科技大学

2017 年硕士研究生招生考试初试试题

考试科目代码及名称 951 数据结构

考试时间 2016 年 12 月 25 日下午 (3 小时)

答题要求：所有答案（填空题按照标号写）必须写在答题纸上，写在试题上一律作废，准考证号写在指定位置！

一、单项选择题（在下列每小题的备选答案中选出一个正确答案。每空 2 分，共 30 分）

1. 算法能识别出错误的输入数据并进行适当的处理和反应，称为算法的（①）。

- A. 健壮性 B. 正确性 C. 并行性 D. 时间复杂度

2. 从一个具有 n 个结点的单链表中查找其值等于 x 的结点时，在查找成功的情况下，需要平均比较的节点个数是（②）。

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

3. 设有一个双端队列，元素进入该队列的次序为 $abcd$ ，则既不能由输入受限双端队列得到，也不能由输出受限双端队列得到的输出序列为（③）。

- A. $dbca$ B. $dcba$ C. $dbac$ D. $dcab$

4. 下列关于串的叙述中，正确的是（④）。

A. 一个串的字符个数即该串的长度

B. 空串是由一个空格字符组成的串

C. 一个串的长度至少是 1

D. 若两个串 S_1 和 S_2 的长度相同，则这两个串相等

5. 对一棵满二叉树， m 个树叶， n 个结点，深度为 k ，则（⑤）。

- A. $k+m=2n$ B. $n=k+m$ C. $n=2^k-1$ D. $m=k-1$

6. 一个具有 10 个顶点的有向图中，所有顶点的入度之和与所有顶点的出度之和的差等于（⑥）。

- A. 20 B. 5 C. 0 D. 2

7. 对于线性表 (7, 34, 55, 25, 64, 46, 20, 10) 进行散列存储时, 若选用 $H(K) = K \% 9$ 作为散列函数, 则散列地址为 1 的元素有 (⑦) 个。

A. 1 B. 2 C. 3 D. 4

8. 采用贪心算法不能求解的问题是 (⑧)。

A. 0-1 背包问题 B. 活动安排问题 C. 找零钱问题 D. 背包问题

9. 用数组 $Q[m]$ 存放循环队列的元素, $rear$ 和 len 分别表示该队列的队尾元素的位置和队列的长度, 则队列第一个元素的位置是 (⑨)。

A. $rear-len$ B. $(rear+m-len) \% m$ C. $m-len$ D. $rear-len+m$

10. 线性表若采用链表存储结构时, 要求内存中可用存储单元的地址 (⑩)。

A. 必须是连续的 B. 部分地址必须是连续的
C. 一定是不连续的 D. 连续不连续都可以

11. 在头指针为 $head$ 且表长大于 1 的单循环链表中, 指针 p 指向表中某个结点, 若 $p \rightarrow next \rightarrow next = head$, 则 (⑪)。

A. p 指向头结点 B. p 指向尾结点
C. $*p$ 的直接后继是头结点 D. $*p$ 的直接后继是尾结点

12. 快速排序在最坏的情况下的时间复杂度是 (⑫)。

A. $O(\log_2 n)$ B. $O(n \log_2 n)$ C. $O(n^2)$ D. $O(n^3)$

13. 设某棵二叉树中有 45 个结点, 则该二叉树的最小高度为 (⑬)。

A. 3 B. 4 C. 5 D. 6

14. 设 n 个元素进栈的序列是 1, 2, 3, ..., n , 其输出序列是 p_1, p_2, \dots, p_n , 若 $p_1=3$, 则 p_2 的值 (⑭)。

A. 一定是 1 B. 一定是 2 C. 可能是 1 D. 可能是 2

15. 在长度为 n 的顺序表的第 i ($1 \leq i \leq n+1$) 个位置上插入一个元素, 元素的移动次数为 (⑮)。

A. $n-i+1$ B. $n-i$ C. i D. $i-1$

二、判断题（每题 1 分，共 15 分）

1. 链式栈与顺序栈相比，一个明显的优点是通常不会出现栈满的情况。（ ① ）
2. 在一个顺序存储的循环队列中，队头指针指向队头元素的位置。（ ② ）
3. 数组一旦建立，结构中的元素个数和元素间的关系就不再发生变化。因此，一般都采用链式存储的方法来表示数组。（ ③ ）
4. 从串中取若干个字符组成的字符序列称为串的子串。（ ④ ）
5. 一个无向图的邻接矩阵中各元素之和与图中边的条数相等。（ ⑤ ）
6. 具有 6 个顶点的无向图至少有 6 条边才能确保是一个连通图。（ ⑥ ）
7. 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。（ ⑦ ）
8. 线性表的各种基本运算在顺序存储结构上的实现均比在链式存储结构上的实现效率要低。（ ⑧ ）
9. 在散列法中，一个可用的散列函数必须保证绝对不产生冲突。（ ⑨ ）
10. 在任何情况下，快速排序方法的时间性能总是最优的。（ ⑩ ）
11. 一个问题是否具有最优子结构性质是该问题是否可以用动态规划法求解的重要前提。（ ⑪ ）
12. 层次结构设计方法有两种：自顶向下和自底向上。（ ⑫ ）
13. 数据结构与算法的本质联系表现在失去一方，另一方将没有任何意义。（ ⑬ ）
14. 二叉树是一棵无序树。（ ⑭ ）
15. 对于一棵具有 n 个结点的任何二叉树，进行先序、中序或后序的任一种次序遍历的空间复杂度为 $O(\log 2n)$ 。（ ⑮ ）

三、填空题（每空 2 分，共 30 分）

1. ① 是数据的基本单位，即数据这个集合中的一个个体。
2. 数据的逻辑结构分为两大类：② 。
3. Huffman 树共有③ 个结点（ n 为叶子结点个数）。
4. 设一棵完全二叉树中有 500 个结点，若用二叉链表作为该完全二叉树的存储结构，则

共有 ④ 个空指针域。

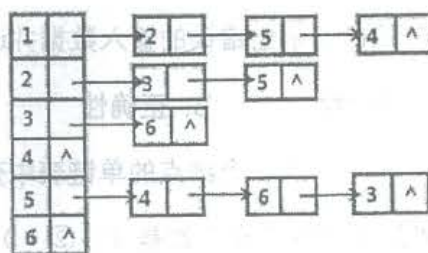
5. 设不含头节点的链队列中结点的格式为(data, next), front 为其头指针, rear 为其尾指针, 则该队列为空的条件是 ⑤ 。

6. 已知一个栈的输入序列为 1, 2, 3, ..., n, 则其输出序列的第 2 个元素为 n 的输出序列的种数是 ⑥ 。

7. 对于一维数组 A[15], 若一个数组元素占用字节数为 s, 则 A[i](i ≥ 0) 的存储地址为 ⑦ 。

8. 无向图中的极大连通子图称为该无向图的 ⑧ 。

9. 已知图 G 的邻接表如右图所示, 其从 1 顶点出发的深度优先搜索序列为 ⑨ 。



10. 单链表表示法的基本思想是用 ⑩ 表示结点间的逻辑关系。

11. 设 r 指向单链表的最后一个结点, 要在最后一个结点之后插入 s 所指的结点, 需执行的三条语句是 ⑪ ; r = s; r->next = null;。

12. 一个顺序表的第一个元素的存储地址是 200, 每个元素的长度是 2, 则第 6 个元素的地址是 ⑫ 。

13. 除留余数法选择一正整数 p, 以关键字除以 p 所得的余数作为散列地址, 通常选 p 为 ⑬ 。

14. 若某个散列函数 H 对于不相等的关键字 key₁ 和 key₂ 得到相同的散列地址 (即 H(key₁)=H(key₂)), 则将该现象称为 ⑭ 。

15. 二分搜索中查到每一个记录的比较次数可通过折半查找判定树来描述, 那么查找某个结点进行的比较次数即为被查找结点在树中的 ⑮ 。

四、问题求解 (共 45 分)

1. (7 分) 下图是一个单链表的每个结点的物理位置的示意图, 请用一般图示法画出该

单链表。

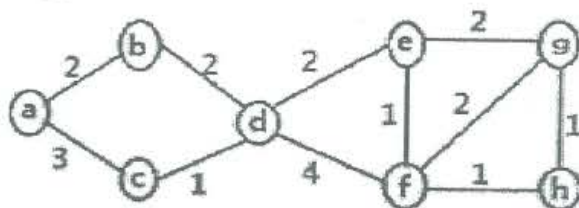
	数据域	指针域
	M	M
110	zhang	200
	M	M
125	wang	130
130	sun	NULL
	M	M
200	zhao	125
head 215	qiang	110
220	li	245
	M	M
245	wu	250
250	fu	125
	M	M

2. (7分) 假设二维数组 $A[6][8]$, 每个元素用相邻的 6 个字节存储, 存储器按字节编址, 已知 A 的基地址为 1000, 计算:

- (1) 数组 A 的占用的存储空间;
- (2) A 的最后一个元素第一个字节的地址;
- (3) 按行存储时, $A[1][4]$ 的第一个字节的地址;
- (4) 按列存储时, $A[4][7]$ 的第一个字节的地址。

3. (7分) 设有一组初始记录关键字为(45, 80, 48, 40, 22, 78), 要求构造一棵二叉排序树并给出构造过程。

4. (8分) 对下列连通图 (如下图所示), 请分别用 Prim(从定点 a 出发)和 Kruskal 算法构造其最小生成树。



5. (8分) 假设用于通信的电文由十种不同的符号来组成, 这些符号在电文中出现的频率为 8, 21, 37, 24, 6, 18, 23, 41, 56, 14, 试为这十个符号构造哈夫曼树, 并设计相应的哈夫曼编码 (要求树中左孩子结点的权值小于右孩子结点的权值)。

6. (8分) 对于下列一组关键字 46, 58, 15, 45, 90, 18, 10, 62, 试写出快速排序第一趟的一次划分过程, 并写出每一趟的排序结果。

五、算法设计题 (共 30 分)

1. (10分) 设计算法求关键字 x 在二叉排序树中的层次:

```
int lev=0;
```

```
typedef struct node{int key;
                    char data;
                    struct node *lchild,*rchild;
                    }bitree;
```

```
void level(bitree *bt, int x){
```

```
}
```

2. (10分) 请在空格处将算法补充完整。完成下面的算法, 识别一次读入的一个以@为结束符的字符序列是否为形如‘序列 1&序列 2’模式的字符序列。其中序列 1 和序列 2 中都不含字符‘&’, 且序列 2 是序列 1 的逆序列。例如, ‘a+b&b+a’是属该模式的字符序列, 而‘1+3&3-1’则不是。

```
typedef struct{
```

```
    SElemType data[M];
```

```
    int top;
```

```
} * Stack;
```

```
Push (Stack ST, SElemType x);    // 已定义的入栈操作
```

```
Pop (Stack ST, SElemType x); //已定义的退出栈顶元素赋给 x 的操作
```

```
BOOL Symmetry(char a[])
```

```
{
```

```
    int i=0;
```

```
    Stack s;    // 定义栈 s
```

```
    InitStack(s);
```

```
    ElemType x;
```

```
    while(① && a[i]!='@'){
```

```
        ②;
```

```
        i++;
```

```
    }
```

```
    if(③) return FALSE;
```

```
    i++;
```

```
    while(a[i]!='@'){
```

```
        ④;
```

```
        if(⑤){
```

```
            DestroyStack(s);
```

```
            return FALSE;
```

```
        }
```

```
        i++;
```

```
    }
```

```
return TRUE;
```

```
}
```

3. (10分) 已知一个顺序表, 编写算法, 实现在其值为x的元素之后插入m个元素的算法。

```
int InsertM(sequenlist *L, int x, int m)
```

```
{
```

```
} //InsertM
```