

五邑大学 试 卷 参 考 答 案 及 评 分 标 准

学期: 2016 至 2017 学年度 第 2 学期

课程: 数据结构 课程代号: 0800610 使用班级: 150801~08、10

一、

得分	
----	--

 单项选择题 (10 小题, 每小题 1 分, 共 10 分)

1. 在解决计算机主机与打印机之间速度不匹配问题时通常设置一个打印缓冲区, 该缓冲区应该是一个 (**B**) 结构。

A. 栈 B. 队列 C. 数组 D. 线性表

2. 设某棵二叉树中有 2000 个结点, 则该二叉树的最小高度为 (**C**)。

A. 9 B. 10 C. 11 D. 12

3. 设输入序列为 1、2、3、4、5、6, 则通过栈的作用后可以得到的输出序列为(**B**)。

A. 5, 3, 4, 6, 1, 2 B. 3, 2, 5, 6, 4, 1
C. 3, 1, 2, 5, 4, 6 D. 1, 5, 4, 6, 2, 3

4. 设指针变量 p 指向单链表中结点 A, 若删除单链表中结点 A, 则需要修改指针的操作序列为(**A**)。

A. $q=p->next$; $p->data=q->data$; $p->next=q->next$; $free(q)$;
B. $q=p->next$; $q->data=p->data$; $p->next=q->next$; $free(q)$;
C. $q=p->next$; $p->next=q->next$; $free(q)$;
D. $q=p->next$; $p->data=q->data$; $free(q)$;

5. 对于 n 个关键字序列 $\{k_1, k_2, \dots, k_n\}$, 当且仅当满足关系 $k_i \leq k_{2i}$ 且 $k_i \leq k_{2i+1}$ ($2i \leq n, 2i+1 \leq n$) 称其为小根堆, 反之, 当且仅当满足关系 $k_i \geq k_{2i}$ 且 $k_i \geq k_{2i+1}$ ($2i \leq n, 2i+1 \leq n$) 则为大根堆, 以下序列中(**C**)不符合堆的定义。

A. (4,10,15,72,39,23,18) B. (58,27,36,12,8,23,9)
C. (4,10,18,72,39,23,15) D. (58,36,27,12,8,23,9)

6. 程序段

```
for(i=n-1;i>=1;i--)
```

```
    for(j=1;j<=i;j++)
```

```
        if(A[j]>A[j+1])
```

```
            A[j]与 A[j+1]对换;
```

其中 n 为正整数, 则最后一行的语句频度在最坏情况下是(**B**)。

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

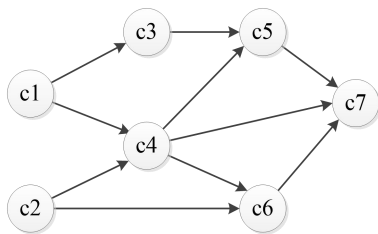
7. 表达式 $a*(b+c)-d$ 的后缀表达式是(**A**)。

A. $abc*+d-$ B. $abcd*+-$ C. $abc*+d-$ D. $-+*abcd$

8. 设某棵二叉树的中序遍历序列为 ABCD, 前序遍历序列为 CABD, 则后序遍历该二叉树得到序列为(**A**)。

A. BADC B. BCDA C. CDAB D. CBDA

9. 拓扑序列是无环有向图中所有顶点的一个线性序列, 图中任意路径中的各个顶点在该图的拓扑序列中保持先后关系, (**B**) 不是下图所示有向图的一个拓扑序列。



- A. c1 c2 c3 c4 c5 c6 c7
- B. c1 c2 c4 c5 c6 c3 c7
- C. c1 c2 c4 c6 c3 c5 c7
- D. c2 c1 c4 c6 c3 c5 c7

10. 设有一个 10 阶的下三角矩阵 A（包括对角线），按照从上到下、从左到右的顺序存储到连续的 55 个存储单元中，每个数组元素占 1 个字节的存储空间，则 A[5][4]地址与 A[0][0]的地址之差为(**B**)。

- A. 10
- B. 19
- C. 28
- D. 55

二、

得分	
----	--

 简单填空题（10 小题，20 个空，每空 1 分，共 20 分）

1. 数据元素是（ **数据的基本单位** ），数据对象是（ **具有相同性质的数据元素** ）的集合。
2. 在长度为 n 的数组 a[n]中（下标从 0 开始）存储两个栈时，假设左栈的栈顶指针（下标）是 topl，右栈的栈顶指针是 topr，那么数据元素 x 进左栈的操作是（ **topl=topl+1;a[topl]=x;** ），右栈的退栈操作是（ **topr=topr+1;** ）。
3. 假设将线性表 L 存储在以 head 为头指针的不带头结点的循环单链表中，则当（ **head==NULL** ）时，表示线性表 L 是空表。假设将线性表 L 存储在以 head 为头指针的带头结点的循环单链表中，则当（ **head->next==head** ）时，表示线性表 L 是空表。
4. 在长度为 n 的数组 a[n]中（下标从 0 开始）存储一个循环队列时，假设队头指针（下标）是 front，队尾指针是 rear，而且 a[(front+1)%n]是队头元素，a[rear]是队尾元素，那么当队列不满时，数据元素 x 进队的操作是（ **rear=(rear+1)%n;a[rear]=x;** ），当队列不空时，出队的操作是（ **front=(front+1)%n;** ）。
5. 对于一个 n*n 的三对角线数组 a，按照行优先的原则，把 a 的三对角线上的元素存储在起始地址为 start 的连续的内存空间中，假定数组的下标从 0 开始，一个数组元素占内存的 c 个字节，那么一共要存储（ **3*n-2** ）个数组元素，三对角线上的数组元素 a[i][j]的地址是（ **start+(2*i+j)*c** ）。
6. (d, (, (e, (x, y)), (a,b,c)) 是一个长度为（ **4** ），深度为（ **3** ）的广义表。
7. 含有 n 个结点的完全二叉树的深度是（ **$\lfloor \log_2 n \rfloor + 1$** ）。如果从 1 开始，按照层序顺序对完全二叉树的结点进行编号，则当 $2*i+1 > n$ 时，表明编号为 i 的结点（ **没有右孩子** ）。
8. 在树的二叉树表示中，根结点一定没有（右孩子）。一个结点与它的左孩子是（ **父子关系** ），与它的右孩子是兄弟关系。
9. AOE-网是一个（ **顶点表示事件，有向边表示活动，有向边上的权值表示活动的持续时间** ）的有向图。如果 AOE-网中的一个活动，它的最早开始时间等于它的最迟开始时间，则表明这个活动是（ **关键活动** ）。
10. 对于 n 个记录进行快速排序，在最坏情况下的时间复杂度是 $O(\mathbf{n^2})$ ，在最好情况下的时间复杂度是 $O(\mathbf{n \log_2 n})$ 。

三、

得分	
----	--

应用题（10 小题，每小题 6 分，共 60 分）

1. 在如下数组 A 中存储了一个静态链表，表头指针为 A[0].next，请将其改成带头结点的单链表，画出该单链表。

A		0	1	2	3	4	5	6	7
data			60	50	78	90	34		40
next		3	5	7	2	0	4		1

答案：

该单链表



评分标准：1 个一分

2. 设有输入序列(2,3,1,5,4)，借助辅助栈或辅助队列得到序列(1,2,3,4,5)，问最少需要（ ）个辅助栈，或最少需要（ ）个辅助队列。

答案： 2 2

评分标准：3 分 3 分

3. 设某棵二叉树的中序遍历序列为 DBEAC，前序遍历序列为 ABDEC，要求给出该二叉树的的后序遍历序列为()。

答案： 后序遍历序列 DEBCA

评分标准：错 1 个扣 1 分

4. 设有 5 个使用频率不等的字符（A，B，C，D，E），使用的频率依次为 W=（20，15，10，9，7），构造哈夫曼树，该哈夫曼树的带权路径长度 WPL 为（ ），对这些字符进行哈夫曼编码，最长的码长为（ ）位。

答案： WPL=(7+9)*3+(10+15+20)*2=138 3

评分标准：4 分 2 分

5. 设一组初始记录关键字序列为(19，21，16，5，18，23)，要求给出以 19 为基准的一趟快速排序结果是()以及第 2 趟直接选择排序后的结果是()。

答案： (18,5,16,19,21,23), (5, 16, 21, 19, 18, 23)

评分标准：4 分 2 分

6. 已知一个散列表如下所示：

0	1	2	3	4	5	6	7	8	9	10	11	12
		35		20			33		48			59

其散列函数为 $h(\text{key}) = \text{key} \% 13$ ，处理冲突的方法为双重散列法，探查序列为：

$$h_i = (h(\text{key}) + i * h_1(\text{key})) \% m \quad i = 0, 1, \dots, m-1$$

其中

$h1(key)=key\%11+1$

回答下列问题:

- (1) 对表中关键字 35, 20, 33 和 48 进行查找时, 所需进行的比较次数各为多少?
(2) 该散列表在等概率查找时查找成功的平均查找长度为多少?

答案:

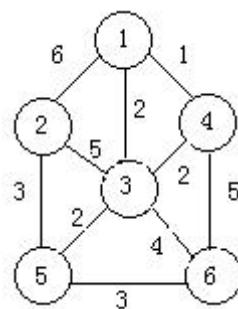
评分标准:

关键字 35 20 33 48

比较次数 3 2 1 1 每个 1 分

$ASL=(3+2+1+1+2)/5=9/5$ 2 分

7. 设无向图 G (所右图所示), 采用邻接表存储, 边表中顶点顺序从大到小, 那么该图从顶点 1 开始的深度优先遍历的序列为(), 广度优先遍历的序列为(), 该图的最小生成树各边和为()。



答案:

评分标准:

1 4 6 5 3 2 2 分

1 4 3 2 6 5 2 分

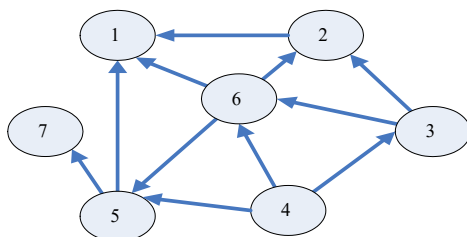
$1+2+2+3+3=11$ 2 分

8. 已知一个图的顶点集 V 和边集 E 分别为:

$V=\{1,2,3,4,5,6,7\};$

$E=\{<2,1>,<3,2>,<3,6>,<4,3>,<4,5>,<4,6>,<5,1>,<5,7>,<6,1>,<6,2>,<6,5>\};$

若存储它采用邻接表, 并且每个顶点邻接表中的边结点都是按照终点序号从小到大的次序链接的, 试给出得到的拓扑排序的序列。



答案:

拓扑排序为: 4 3 6 5 7 2 1/4362517/4365217/4365271

评分标准: 错 1 个扣 1 分

9. 设用序列 (23,20,11,31,13,27,44), 构造一个大顶堆, 该大顶堆的叶子元素包括 () 该大顶堆对应的顺序表为 (), 进行堆排序, 第一趟排序并调整后的顺序表为 ()。

答案:

评分标准:

叶子元素: 20, 13, 23, 11 2 分

大顶堆对应的顺序表: 44,31,27,20,13,23,11 2 分

第一趟排序并调整后的顺序表: 31,20,27,11,13,23 2 分

10. 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, 问该二叉排序树的中序遍历序列为(), 每个元素查找成功时的 ASL 为()。

答案:

评分标准:

中序遍历序列 : 22 , 40,45,48,78,80 2 分

$ASL=(1*1+2*2+2*3+1*4)/6=5/2$ 4 分

四、

得分	
----	--

 算法设计题 (2 小题, 每小题 5 分, 共 10 分)

1. 一个顺序表, 存放在数组 $a[]$ 中, 元素的类型为整型 (int), 顺序表的长度为 n 。设计算法调整 a , 使其左边的所有元素为奇数, 右边的所有元素为非奇数 (要求算法的时间复杂度为 $O(n)$, 空间复杂度为 $O(1)$)。

评分: 算法的时间复杂度 $> O(n)$ 扣 1 分, 空间复杂度 $> O(1)$ 扣 1 分, 具体评分见程序标注, 多解请参照评分。

```
void quickSwapList(int a[], int n)
{
    int i = 0, j = n - 1;    // n 为顺序表的长度, 即数的个数
    while( i < j )           //初值1分
    {
        while( a[i] %2==1 && i < j ) i++;    // 找到左边的非奇数    //1分
        while( a[j] %2== 0 && i < j ) j--;    // 找到右边的奇数    //1分
        if( i < j ) // 交换
        {
            int t = a[i]; a[i] = a[j]; a[j] = t;
        }
        i++;
        j--;
    }
}
```

解析: 本题考得的是线性表的基本操作, 数组的遍历技巧, 此题类似的字符串处理很多见, 快速排序中的划分也有类似的处理方法。用两个游标 i, j , i 的初值为 0(最低位置), $j=n-1$ (最高位置), i 增加, j 减少, 向中间靠拢, 符合条件继续增加(i)或减少(j), 不符合条件交换数据($a[i] \leftrightarrow a[j]$)。

2. 利用栈 (已提供的栈和基本的栈操作, 并假设不会出现栈满情况) 设计一个算法, 识别依次读入的以 “#” 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” 都不含字符 @, 且 “序列 2” 是 “序列 1” 的逆序列。例如 $abc@cba$ 属于该模式的字符序列, $abc@cab$ 则不是。

评分: 没有利用栈扣 2 分, 具体评分见程序标注, 多解请参照评分。

栈: $\text{Stack} *s$, 基本操作如下:

```
栈的初始化: Stack * StackInit();           //构造一个空栈
入栈: void Push(Stack *s, char x);          //将数据元素 x 插入栈顶
出栈: char Pop(Stack *s);                   //删除栈顶元素 (通过返回值得到栈顶数据)
判栈空: int StackEmpty(Stack *s);          //栈空返回 TRUE (或 1); 否则返回 FALSE (或 0)
void Palindrome(char expr[ ], int n)        // expr 存储的字符序列, 字符序列长度为 n
{
    int i, j;
    Stack s=StackInit();                    //以上 1 分
    for(i=0; i<n; i++)
    {
        if(expr[i]!='@')
            Push(s, expr[i]);
        else break;
    }
}
```

```

    }
    for(j=i+1; j<n&&!StackEmpty(s); j++)
    {
        char c= Pop(s );
        if(expr[j ]!=c) {    printf(“不是\n” );return;    }
    }
    if(expr[j ]=='#'&& StackEmpty(s)) {printf(“是\n” ); return;}
    或 if(j==n-1&& StackEmpty(s)) {printf(“是\n” ); return;}
    else {    printf(“不是\n” );return;    }
}

```

//以上 2 分

//以上 2 分

解析：此题类似回文，略有不同，一个是有分界字符@,字符串有结束字符#，题目要求采用堆栈进行辅助操作，概念上更简单。算法设计好后调用时 n 包括@#两个字符，比如 str="112@211#",n=8,输出结果"是"。 Palindrome(str,8).