

2017《数据结构》期末考试试题及答案

《数据结构》期末考试试题及答案 1 .....2

试题 1 答案 .....7..

《数据结构》期末考试试题及答案 2 .....9

试题 2 答案 .....14

《数据结构》期末考试试题及答案 3 .....16

试题 3 答案 .....21

# 《数据结构》期末考试试题及答案 1

## 一、单选题（每题 2 分，共 20 分）

1. 栈和队列的共同特点是（ ）。  
A.只允许在端点处插入和删除元素  
B.都是先进后出  
C.都是先进先出  
D.没有共同点
2. 用链接方式存储的队列，在进行插入运算时（ ）。  
A. 仅修改头指针  
B. 头、尾指针都要修改  
C. 仅修改尾指针  
D.头、尾指针可能都要修改
3. 以下数据结构中哪一个是非线性结构？（ ）  
A. 队列  
B. 栈  
C. 线性表  
D. 二叉树
4. 设有一个二维数组  $A[m][n]$ ，假设  $A[0][0]$  存放在位置  $644_{(10)}$ ， $A[2][2]$  存放在位置  $676_{(10)}$ ，每个元素占一个空间，问  $A[3][3]_{(10)}$  存放在什么位置？脚注  $_{(10)}$ 表示用 10 进制表示。  
A . 688  
B . 678  
C . 692  
D . 696
5. 树最适合用来表示（ ）。  
A.有序数据元素  
B.无序数据元素  
C.元素之间具有分支层次关系的数据  
D.元素之间无联系的数据
6. 二叉树的第  $k$  层的结点数最多为（ ）。  
A .  $2^k-1$   
B.  $2K+1$   
C.  $2K-1$   
D.  $2^{k-1}$
7. 若有 18 个元素的有序表存放在一维数组  $A[19]$  中，第一个元素放  $A[1]$  中，现进行二分查找，则查找  $A[3]$  的比较序列的下标依次为（ ）。  
A. 1 , 2 , 3  
B. 9 , 5 , 2 , 3  
C. 9 , 5 , 3  
D. 9 , 4 , 2 , 3
8. 对  $n$  个记录的文件进行快速排序，所需要的辅助存储空间大致为  
A.  $O(1)$   
B.  $O(n)$   
C.  $O(\log_2 n)$   
D.  $O(n^2)$
9. 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 ( ) 个，

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

10. 设有 6 个结点的无向图，该图至少应有 ( ) 条边才能确保是一个连通图。

A.5                      B.6                      C.7                      D.8

## 二、填空题 (每空 1 分，共 26 分)

- 通常从四个方面评价算法的质量：\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。
- 一个算法的时间复杂度为  $(n^3 + n^2 \log_2 n + 14n)/n^2$ ，其数量级表示为\_\_\_\_\_。
- 假定一棵树的广义表表示为  $A(C, D(E, F, G), H(I, J))$ ，则树中所含的结点数为\_\_\_\_\_个，树的深度为\_\_\_\_\_，树的度为\_\_\_\_\_。
- 后缀算式  $9\ 2\ 3\ +\ -\ 10\ 2\ /\ -$  的值为\_\_\_\_\_。中缀算式  $(3+4X) - 2Y/3$  对应的后缀算式为\_\_\_\_\_。
- 若用链表存储一棵二叉树时，每个结点除数据域外，还有指向左孩子和右孩子的两个指针。在这种存储结构中， $n$  个结点的二叉树共有\_\_\_\_\_个指针域，其中有\_\_\_\_\_个指针域是存放了地址，有\_\_\_\_\_个指针是空指针。
- 对于一个具有  $n$  个顶点和  $e$  条边的有向图和无向图，在其对应的邻接表中，所含边结点分别有\_\_\_\_\_个和\_\_\_\_\_个。
- AOV 网是一种\_\_\_\_\_的图。
- 在一个具有  $n$  个顶点的无向完全图中，包含有\_\_\_\_\_条边，在一个具有  $n$  个顶点的有向完全图中，包含有\_\_\_\_\_条边。
- 假定一个线性表为  $(12, 23, 74, 55, 63, 40)$ ，若按  $\text{Key} \% 4$  条件进行划分，使得同一余数的元素成为一个子表，则得到的四个子表分别为\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。
- 向一棵 B\_树插入元素的过程中，若最终引起树根结点的分裂，则新树比原树的高度\_\_\_\_\_。
- 在堆排序的过程中，对任一分支结点进行筛运算的时间复杂度为\_\_\_\_\_，整个堆排序过程的时间复杂度为\_\_\_\_\_。

12. 在快速排序、堆排序、归并排序中，\_\_\_\_\_排序是稳定的。

三、运算题（每题 6 分，共 24 分）

1. 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，试写出该线性表。

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

2. 请画出图 10 的邻接矩阵和邻接表。

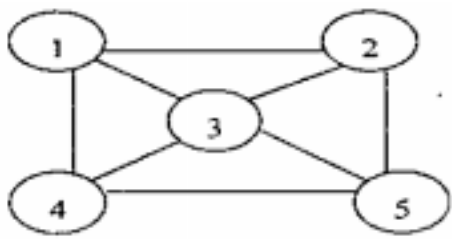


图 10

3. 已知一个图的顶点集 V 和边集 E 分别为：  
(1,2)3, (4,6)4, (1,3)5, (1,4)8, (2,5)10,  
(4,7)20  
V={1,2,3,4,5,6,7};  
E={(1,2)3,(1,3)5,(1,4)8,(2,5)10,(2,3)6,(3,4)15,  
(3,5)12,(3,6)9,(4,6)4,(4,7)20,(5,6)18,(6,7)25};  
用克鲁斯卡尔算法得到最小生成树，试写出

在最小生成树中依次得到的各条边。

4. 画出向小根堆中加入数据 4, 2, 5, 8, 3 时，每加入一个数据后堆的变化。

四、阅读算法（每题 7 分，共 14 分）

```
1. LinkList mynote(LinkList L)
    { //L 是不带头结点的单链表的头指针
        if(L&&L->next){
            q=L ; L=L ->next ; p=L ;
            S1 : while(p ->next) p=p ->next ;
            S2 : p ->next=q ; q ->next=NULL ;
        }
        return L ;
    }
```

}

请回答下列问题：

1. (1) 查询链表的尾结点  
(2) 将第一个结点链接到链表的尾部，作为新的尾结点  
(3) 返回的线性表为 ( a<sub>2</sub>,a<sub>3</sub>, ,a<sub>n</sub>,a<sub>1</sub>)

(1) 说明语句 S1 的功能；

(2) 说明语句组 S2 的功能；

(3) 设链表表示的线性表为 ( a<sub>1</sub>,a<sub>2</sub>, , ,a<sub>n</sub> ) ,写出算法执行后的返回值所表示的线性表。

2. void ABC(BTNode \* BT)

```
{  
    if BT {  
        ABC (BT->left);  
        ABC (BT->right);  
        cout<<BT->data<<' '  
    }  
}
```

2. 该算法的功能是：

五、 算法填空 ( 共 8 分 )

二叉搜索树的查找 —— 递归算法：

bool Find(BTreeNode\* BST,ElemType& item)

```
{  
    if (BST==NULL)  
        return false; //查找失败  
    else {  
        if (item==BST->data){  
            item=BST->data;//查找成功  
            return _____;}  
        else if(item<BST->data)  
            return Find(_____,item);  
        else return Find(_____,item);  
    }
```

```
        }//if  
    }
```

六、 编写算法（共 8 分）

统计出单链表 HL 中结点的值等于给定值 X 的结点数。

```
int CountX(LNode* HL,ElemType x)
```

# 试题 1 答案

一、 单选题（每题 2 分，共 20 分）  
1.A 2.D 3.D 4.C 5.C 6.D 7.D 8.C 9.D 10.A

二、 填空题（每空 1 分，共 26 分）  
1. 正确性 易读性 强壮性 高效率  
2.  $O(n)$   
3. 9 3 3  
4.  $-1 \quad 3 \quad 4 \quad X * + 2 \quad Y * 3 \quad / -$   
5.  $2n \quad n-1 \quad n+1$   
6.  $e \quad 2e$   
7. 有向无回路  
8.  $n(n-1)/2 \quad n(n-1)$   
9.  $(12, 40) \quad ( ) \quad (74) \quad (23, 55, 63)$   
10. 增加 1  
11.  $O(\log_2 n) \quad O(n \log_2 n)$   
12. 归并

三、 运算题（每题 6 分，共 24 分）  
1. 线性表为： $(78, 50, 40, 60, 34, 90)$

0	1	1	1	0
1	0	1	0	1
1	1	0	1	1
1	0	1	0	1
0	1	1	1	0

2. 邻接矩阵：  
邻接表如图 11 所示：

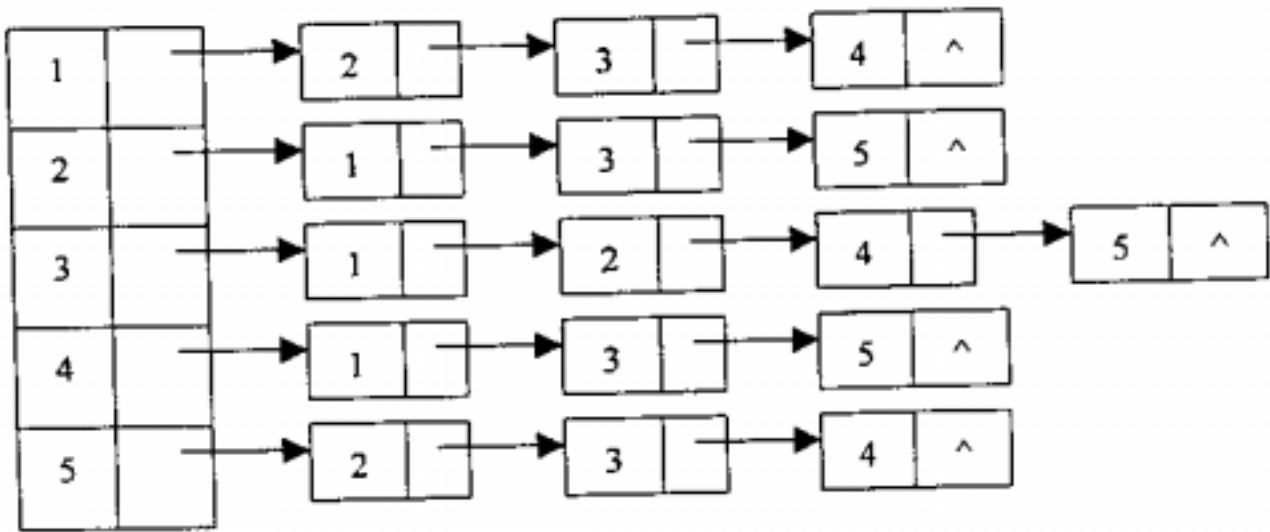


图 11

3. 用克鲁斯卡尔算法得到的最小生成树为：  
 $(1,2)3, (4,6)4, (1,3)5, (1,4)8, (2,5)10, (4,7)20$   
4. 见图 12

5. 4. 画出向小根堆中加入数据 4, 2, 5, 8, 3 时，每加入一个数据后堆的变化。

6.

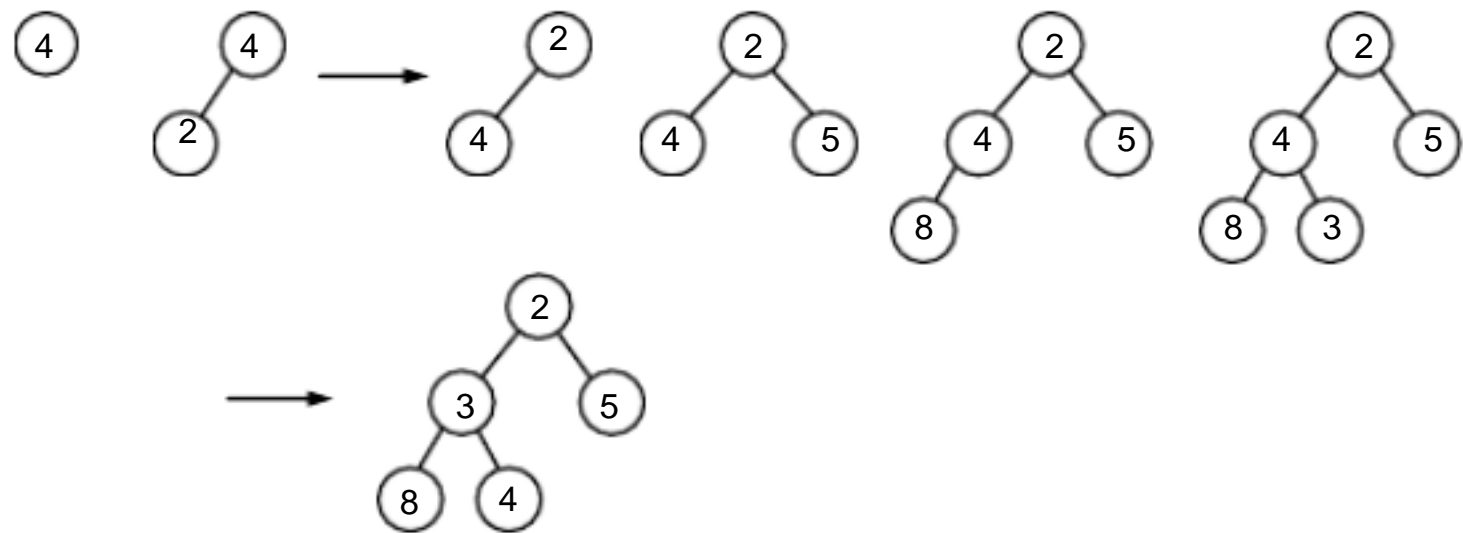


图 12

四、 阅读算法（每题 7 分，共 14 分）

3. （1）查询链表的尾结点

（2）将第一个结点链接到链表的尾部，作为新的尾结点

（3）返回的线性表为（ $a_2, a_3, \dots, a_n, a_1$ ）

4. 递归地后序遍历链式存储的二叉树。

五、 算法填空（每空 2 分，共 8 分）

true      BST->left      BST->right

六、 编写算法（8 分）

```

int CountX(LNode* HL, ElemType x)
{
    int i=0; LNode* p=HL; //i 为计数器
    while(p!=NULL)
    {
        if (P->data==x) i++;
        p=p->next;
    } //while, 出循环时 i 中的值即为 x 结点个数
    return i;
} //CountX

```



《数据结构》期末考试试题及答案 2

一、 单选题（每小题 2 分，共 8 分）

- 1、 在一个长度为  $n$  的顺序线性表中顺序查找值为  $x$  的元素时，查找成功时的平均查找长度（即  $x$  与元素的平均比较次数，假定查找每个元素的概率都相等）为（ ）。  
A  $n$                       B  $n/2$                       C  $(n+1)/2$                       D  $(n-1)/2$
- 2、 在一个单链表中，若  $q$  所指结点是  $p$  所指结点的前驱结点，若在  $q$  与  $p$  之间插入一个  $s$  所指的结点，则执行（ ）。  
A  $s \rightarrow \text{link} = p \rightarrow \text{link}; \quad p \rightarrow \text{link} = s;$       B  $p \rightarrow \text{link} = s; \quad s \rightarrow \text{link} = q;$   
C  $p \rightarrow \text{link} = s \rightarrow \text{link}; \quad s \rightarrow \text{link} = p;$       D  $q \rightarrow \text{link} = s; \quad s \rightarrow \text{link} = p;$
- 3、 栈的插入和删除操作在（ ）进行。  
A 栈顶                      B 栈底                      C 任意位置                      D 指定位置
- 4、 由权值分别为  $11, 8, 6, 2, 5$  的叶子结点生成一棵哈夫曼树，它的带权路径长度为（ ）。  
A 24                      B 71                      C 48                      D 53

二、 填空题（每空 1 分，共 32 分）

- 1、 数据的逻辑结构被分为 \_\_\_\_\_、 \_\_\_\_\_、 \_\_\_\_\_ 和 \_\_\_\_\_ 四种。
- 2、 一种抽象数据类型包括 \_\_\_\_\_ 和 \_\_\_\_\_ 两个部分。
- 3、 在下面的数组  $a$  中链接存储着一个线性表，表头指针为  $a[0].\text{next}$ ，则该线性表为 \_\_\_\_\_。

$a$	0	1	2	3	4	5	6	7	8
	60	56	42	38		74	25		
4	3	7	6	2		0	1		

data  
next

- 4、 在以  $HL$  为表头指针的带表头附加结点的单链表和循环单链表中，判断链表为空的条件分别为 \_\_\_\_\_ 和 \_\_\_\_\_。

- \_\_\_\_\_。
- 5、用具有  $n$  个元素的一维数组存储一个循环队列，则其队首指针总是指向队首元素的 \_\_\_\_\_，该循环队列的最大长度为 \_\_\_\_\_。
- 6、当堆栈采用顺序存储结构时，栈顶元素的值可用 \_\_\_\_\_ 表示；当堆栈采用链接存储结构时，栈顶元素的值可用 \_\_\_\_\_ 表示。
- 7、一棵高度为 5 的二叉树中最少含有 \_\_\_\_\_ 个结点，最多含有 \_\_\_\_\_ 个结点；  
一棵高度为 5 的理想平衡树中，最少含有 \_\_\_\_\_ 个结点，最多含有 \_\_\_\_\_ 个结点。
- 8、在图的邻接表中，每个结点被称为 \_\_\_\_\_，通常它包含三个域：一是 \_\_\_\_\_；二是 \_\_\_\_\_；三是 \_\_\_\_\_。
- 9、在一个索引文件的索引表中，每个索引项包含对应记录的 \_\_\_\_\_ 和 \_\_\_\_\_ 两项数据。
- 10、假定一棵树的广义表表示为  $A(B(C, D(E, F, G), H(I, J)))$ ，则树中所含的结点数为 \_\_\_\_\_ 个，树的深度为 \_\_\_\_\_，树的度为 \_\_\_\_\_，结点  $H$  的双亲结点为 \_\_\_\_\_，孩子结点为 \_\_\_\_\_。
- 11、在堆排序的过程中，对任一分支结点进行筛运算的时间复杂度为 \_\_\_\_\_，整个堆排序过程的时间复杂度为 \_\_\_\_\_。
- 12、在对  $m$  阶的  $B$ -树插入元素的过程中，每向一个结点插入一个索引项（叶子结点中的索引项为关键字和空指针）后，若该结点的索引项数等于 \_\_\_\_\_ 个，则必须把它分裂为 \_\_\_\_\_ 个结点。

### 三、 运算题（每小题 6 分，共 24 分）

- 1、已知一组记录的排序码为  $(46, 79, 56, 38, 40, 80, 95, 24)$ ，写出对其进行快速排序的每一次划分结果。
- 2、一个线性表为  $B = (12, 23, 45, 57, 20, 03, 78, 31, 15, 36)$ ，

设散列表为 HT[0..12]，散列函数为  $H(key) = key \% 13$  并用线性探查法解决冲突，请画出散列表，并计算等概率情况下查找成功的平均查找长度。

3、已知一棵二叉树的前序遍历的结果序列是 ABECKFGHIJ，中序遍历的结果是 EBCDAFHIGJ，试写出这棵二叉树的后序遍历结果。

4、已知一个图的顶点集 V 各边集 G 如下：

$$V = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\};$$
$$E = \{(0, 1), (0, 4), (1, 2), (1, 7), (2, 8), (3, 4), (3, 8), (5, 6), (5, 8), (5, 9), (6, 7), (7, 8), (8, 9)\}$$

当它用邻接矩阵表示和邻接表表示时，分别写出从顶点  $V_0$  出发按深度优先搜索遍历得到的顶点序列和按广度优先搜索遍历等到的顶点序列。

假定每个顶点邻接表中的结点是按顶点序号从大到小的次序链接的。

图	深度优先序列	广度优先序列
邻接矩阵表示时		
邻接表表示时		

四、 阅读算法，回答问题（每小题 8 分，共 16 分）

1、假定从键盘上输入一批整数，依次为： 78 63 45 30 91 34 - 1，请写出输出结果。

```
#include <iostream.h>
#include <stdlib.h>
const int stackmaxsize = 30;
typedef int elemtype;
struct stack {
    elemtype stack [stackmaxsize];
    int top;
```

```

};
#include "stack.h"
Void main ( )
{
    stack a;
    initstack(a);
    int x;
    cin >>x;
    while (x!= -1) {
        push (a, x );
        cin >>x;
    }
    while (!stackempty (a))
        cout <<pop (a) << " " ;
    cout <<end1;
}

```

该算法的输出结果为：

---

\_\_\_\_\_.

2、阅读以下二叉树操作算法，指出该算法的功能。

```

Template <calss type > void BinTree <Type> ::
unknown (BinTreeNode<Type>*t) {
    BinTreeNode< Type> *p =t, *temp;
    if (p!=NULL) {
        temp = p leftchild;
        p leftchild = p rightchild;
        p rightchild = temp;
        unknown(p leftchild);
        undnown(p rightchild);
    }
}

```

该算法的功能是： \_\_\_\_\_

五、 算法填空，在画有横线的地方填写合适的内容（ 10 分）

对顺序存储的有序表进行二分查找的递归算法 。

```
int Binsch( ElemType A[ ],int low ,int high,KeyType K )
{
    if (low <= high)
    {
        int mid = 1
        if ( K== A[ mid ].key )
            return mid;
        else if ( K < A[mid].key)
            return 2
        else
            return 3
    }
    else
        return 4
```

六、 编写算法（ 10 分）

编写算法，将一个结点类型为 Lnode 的单链表按逆序链接，即若原单链表中存储元素的次序为  $a_1, \dots, a_{n-1}, a_n$ ，则逆序链接后变为  $a_n, a_{n-1}, \dots, a_1$ 。

Void contrary (Lnode \* & HL)

试题 2 答案

一、单选题（每小题 2 分，共 8 分）

题 号	1	2	3	4
答 案	C	D	A	B

二、填空题（每空 1 分，共 32 分）

- 1： 集合、线性、树、图；  
2： 数据描述、操作声名；  
3： ( 38 , 56 , 25 , 60 , 42 , 74 )；  
4： HL next=NULL ； HL=HL next ；  
5： 前一个位置； n-1 ；  
6： S.stack [S.top]; HS data;  
7： 5 31  
8： 边结点、邻接点域、权域、链域；  
9： 索引值域、开始位置域；  
10： 10、 3、 3、 B、 I 和 J；  
11：  $O(\log_2 n)$   $O(n\log_2 n)$ ;  
12： m 、 m - 1

三、运算题（每小题 6 分，共 24 分）

1、

划分次序	划分结果							
第一次	[38 24 40] 46 [56 80 95 79]							
第二次	24 [38 40] 46 [56 80 95 79]							
第三次	24 38 40 46 [56 80 95 79]							
第四次	24 38 40 46 56 [80 95 79]							
第五次	24 38 40 46 56 79 [80 95]							
第六次	24 38 40 46 56 79 80 95							

2、

0      1      2      3      4      5      6      7      8      9      10      11      12

78		15	03		57	45	20	31		23	36	12
----	--	----	----	--	----	----	----	----	--	----	----	----

查找成功的平均查找长度：  $ASL_{succ}=14/10= 1.4$

3、此二叉树的后序遍历结果是： EDCBIHJGFA

4、

图	深度优先序列	广度优先序列
邻接矩阵表示时	0 , 1 , 2 , 8 , 3 , 4 , 5 , 6 , 7 , 9	0 , 1 , 4 , 2 , 7 , 3 , 8 , 6 , 5 , 9
邻接表表示时	0 , 4 , 3 , 8 , 9 , 5 , 6 , 7 , 1 , 2	0 , 4 , 1 , 3 , 7 , 2 , 8 , 6 , 9 , 5

四、阅读算法，回答问题（每小题 8 分，共 16 分）

- 1、1、 该算法的输入结果是： 34 91 30 45 63 78
- 2、2、 该算法的功能是：交换二叉树的左右子树的递归算法。

五、算法填空，在画有横线的地方填写合适的内容（ 10 分）

- 1、1 是：( low + high ) /2;
- 2 是： Binsch(A,low,mid – 1,K);
- 3 是： Binsch(A,mid+1,high,K);
- 4 是： -1 ;

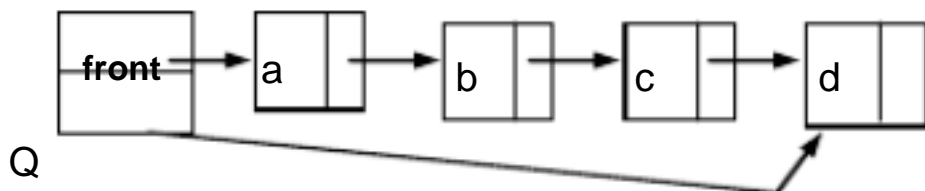
六、编写算法（ 10 分）

根据编程情况，酌情给分。

```
{
    Lnode *P=HL;
    HL=NULL;
    While (p!=null)
    {
        Lnode*q=p;
        P=p next;
        q next=HL;
        HL=q;
    }
}
```

## 《数据结构》期末考试试题及答案 3

### 一、单项选择题

1. 对于一个算法，当输入非法数据时，也要能作出相应的处理，这种要求称为 ( )。
- (A)、正确性 (B)、可行性 (C)、健壮性 (D)、输入性
2. 设  $S$  为 C 语言的语句，计算机执行下面算法时，算法的时间复杂度为 ( )。
- ```
for(i=n-1 ; i>=0 ; i--)  
    for(j=0 ; j<i ; j++) S ;
```
- (A)、 $n^2$  (B)、 $O(n \lg n)$  (C)、 $O(n)$  (D)、 $O(n^2)$
3. 折半查找法适用于 ( )。
- (A)、有序顺序表 (B)、有序单链表
- (C)、有序顺序表和有序单链表都可以 (D)、无限制
4. 顺序存储结构的优点是 ( )。
- (A)、利于插入操作 (B)、利于删除操作
- (C)、利于顺序访问 (D)、利于随机访问
5. 深度为  $k$  的完全二叉树，其叶子结点必在第 ( ) 层上。
- (A)、 $k-1$  (B)、 $k$  (C)、 $k-1$  和  $k$  (D)、1 至  $k$
6. 具有 60 个结点的二叉树，其叶子结点有 12 个，则度为 1 的结点数为 ( )。
- (A)、11 (B)、13 (C)、48 (D)、37
7. 图的 Depth-First Search (DFS) 遍历思想实际上是二叉树 ( ) 遍历方法的推广。
- (A)、先序 (B)、中序 (C)、后序 (D)、层序
8. 在下列链队列  $Q$  中，元素  $a$  出队的操作序列为 ( )。
- 
- (A)  $p=Q.front \rightarrow next; p \rightarrow next=Q.front \rightarrow next;$
- (B)  $p=Q.front \rightarrow next; Q.front \rightarrow next=p \rightarrow next;$
- (C)  $p=Q.rear \rightarrow next; p \rightarrow next=Q.rear \rightarrow next;$
- (D)  $p=Q \rightarrow next; Q \rightarrow next=p \rightarrow next;$
9. Huffman 树的带权路径长度 WPL 等于 ( )。
- (A)、除根结点之外的所有结点权值之和 (B)、所有结点权值之和
- (C)、各叶子结点的带权路径长度之和 (D)、根结点的值
10. 线索二叉链表是利用 ( ) 域存储后继结点的地址。
- (A)、lchild (B)、data (C)、rchild (D)、root



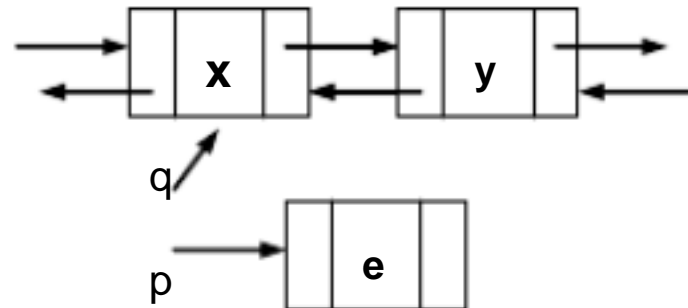
## 二、填空题

1. 逻辑结构决定了算法的 \_\_\_\_\_, 而存储结构决定了算法的 \_\_\_\_\_。

2. 栈和队列都是一种 \_\_\_\_\_ 的线性表, 栈的插入和删除只能在 \_\_\_\_\_ 进行。

3. 线性表 ( $a_1, a_2, \dots, a_n$ ) 的顺序存储结构中, 设每个单元的长度为  $L$ , 元素  $a_i$  的存储地址  $LOC(a_i)$  为 \_\_\_\_\_。

4. 已知一双向链表如下 (指针域名为  $next$  和  $prior$ ) :



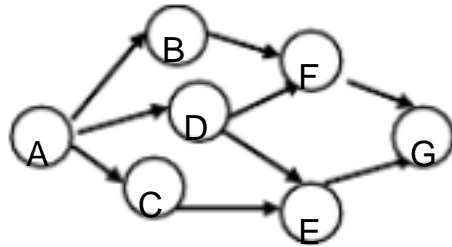
现将  $p$  所指的结点插入到  $x$  和  $y$  结点之间, 其操作步骤为: \_\_\_\_\_;

\_\_\_\_\_ ; \_\_\_\_\_ ; \_\_\_\_\_ ;

5.  $n$  个结点无向完全图的的边数为 \_\_\_\_\_ ,

$n$  个结点的生成树的边数为 \_\_\_\_\_。

6. 已知一有向无环图如下:



任意写出二种拓扑排序序列: \_\_\_\_\_、\_\_\_\_\_。

7. 已知二叉树的中序遍历序列为  $BCA$ , 后序遍历序列为  $CBA$ , 则该二叉树的先序遍历序列为 \_\_\_\_\_, 层序遍历序列为 \_\_\_\_\_。

## 三、应用题

1. 设散列函数  $H(k)=k \% 13$ , 设关键字系列为  $\{22, 12, 24, 6, 45, 7, 8, 13, 21\}$ , 要求用线性探测法处理冲突。 (6 分)

(1) 构造 HASH 表。

(2) 分别求查找成功和不成功时的平均查找长度。

2. 给定表 ( $19, 14, 22, 15, 20, 21, 56, 10$ )。 (8 分)

(1) 按元素在表中的次序, 建立一棵二叉排序树

(2) 对 (1) 中所建立的二叉排序树进行中序遍历, 写出遍历序列。

(3) 画出对 (2) 中的遍历序列进行折半查找过程的判定树。

3. 已知二个稀疏矩阵 A 和 B 的压缩存储三元组表如下：

| A |   |    | B |   |    |
|---|---|----|---|---|----|
| i | j | V  | i | j | V  |
| 1 | 3 | -5 | 2 | 5 | 2  |
| 2 | 4 | 6  | 3 | 3 | 7  |
| 2 | 5 | 2  | 4 | 1 | 3  |
| 4 | 2 | -1 | 5 | 2 | -9 |
| 5 | 2 | 9  | 5 | 5 | 8  |

写出 A-B 压缩存储的三元组表。（5 分）

4. 已知一维数组中的数据为（ 18,12,25,53,18\_\_ ），试写出插入排序（升序）过程。并指出具有 n 个元素的插入排序的时间复杂度是多少？（5 分）

5. 已知一网络的邻接矩阵如下，求从顶点 A 开始的最小生成树。（8 分，要有过程）

|   | A        | B        | C        | D        | E        | F        |
|---|----------|----------|----------|----------|----------|----------|
| A | $\infty$ | 6        | 5        | 1        | $\infty$ | $\infty$ |
| B | 6        | $\infty$ | $\infty$ | 5        | 3        | $\infty$ |
| C | 5        | $\infty$ | $\infty$ | 7        | $\infty$ | 2        |
| D | 1        | 5        | 7        | $\infty$ | 6        | 4        |
| E | $\infty$ | 3        | $\infty$ | 6        | $\infty$ | 6        |
| F | $\infty$ | $\infty$ | 2        | 4        | 6        | $\infty$ |

- (1) 求从顶点 A 开始的最小生成树。  
(2) 分别画出以 A 为起点的 DFS 生成树和 BFS 生成树。

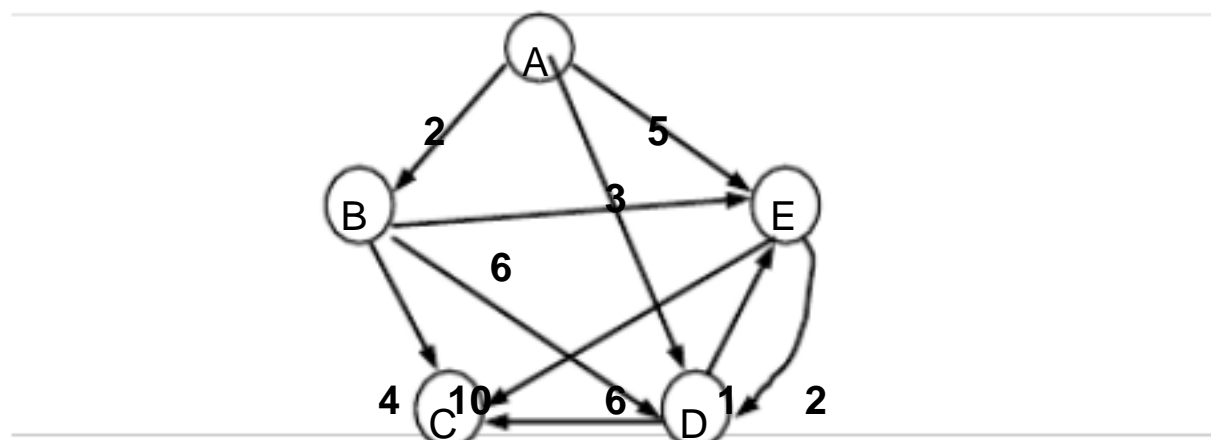
6. 已知数据六个字母及在通信中出现频率如下表：

| A    | B    | C   | D   | E   | F   |
|------|------|-----|-----|-----|-----|
| 0.15 | 0.15 | 0.1 | 0.1 | 0.2 | 0.3 |

把这些字母和频率作为叶子结点及权值，完成如下工作（7 分，要有过程）。

- (1) 画出对应的 Huffman 树。  
(2) 计算带权路径长度 WPL。  
(3) 求 A、B、C、D、E、F 的 Huffman 编码。

7. 已知有如下的有向网：



求顶点 A 到其它各顶点的最短路径（采用 Dijkstra 算法，要有过程）。（6 分）

### 三、 设计题（30 分，每题 10 分，用 C 语言写出算法，做在答题纸上）

1. 已知线性表（ $a_1, a_2, \dots, a_n$ ）以顺序存储结构为存储结构，其类型定义如下：

```
#define LIST_INIT_SIZE 100 //顺序表初始分配容量
typedef struct {
    Elemtype *elem; //顺序存储空间基址
    int length; //当前长度（存储元素个数）
}SqList;
```

设计一个算法，删除其元素值为  $x$  的结点（假若  $x$  是唯一的）。并求出其算法的平均时间复杂度。其算法函数头部如下：

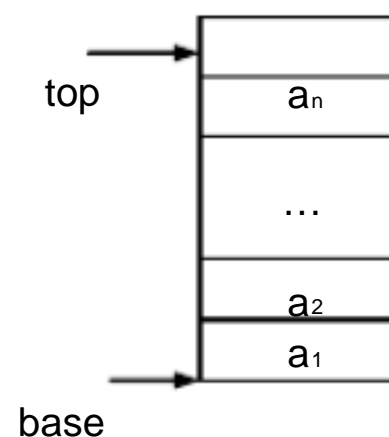
```
S tatus ListDelete(SqList &L,Elemtype x)
{
    .....
}
```

2. 设顺序栈如左图所示。

其中结点定义如下：

```
typedef struct {
    Elemtype *base; //栈底指针
    Elemtype *top; //栈顶指针
}Stack;
```

设计算法，将栈顶元素出栈并存入  $e$  中。



3. 设二叉链树的类型定义如下：

```
typedef int Elemtype;
typedef struct node{
    Elemtype data;
    struct node *lchild, *rchild;
}BinNode, *BinTree;
```

试写出求该二叉树叶子结点数的算法：

```
S tatus CountLeaves(BinTree &root,int &n)
```

{//n is the number of leaves  
.....  
}