

## 2009-2010 学年第二学期《数据结构》试卷 A 卷

授课班号: 242101 专业: 计算机 2008 级 学号 \_\_\_\_\_ 姓名 \_\_\_\_\_

题号	一	二	三	四		总分	审核
题分	10	30	24	36			
得分							

得分	评阅人

### 一、填空(10 分)

1. 数据元素是数据结构中的所处理的基本数据单位, 数据元素可以是不可分割的整体, 也可以是由 \_\_\_\_\_ 组成;

2. 与顺序表相比, 链表的主要特点是 \_\_\_\_\_ 操作的效率比较高。

3. 逻辑上的线性结构我们称为线性表, 线性表即可用顺序表, 也可以用链表存储, 如果有  $n$  个线性表同时并存, 并且在处理过程中各表的长度会动态发生变化, 在此情况下, 应选用 \_\_\_\_\_ 存储结构

4. 设有 10 行 20 列的二维数组  $A[10][20]$ , 其元素长度为 4 字节, 按行优先顺序存储, 基地址为 200, 则元素  $A[8][12]$  的存储地址为 \_\_\_\_\_。

5. 一个二叉树按顺序方式存储在一个一维数组中, 如图

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	B	C	D		E	F		G			H		I	J

结点 H 的父结点是 \_\_\_\_\_。

6. 对图的遍历主要有两种方法, 这两种方法是 \_\_\_\_\_ 和 \_\_\_\_\_。

7. 对于无向图  $G$ , 若用邻接矩阵  $A$  表示, 则  $G$  的第  $k$  个顶点的度等于 \_\_\_\_\_, 若用邻接表表示, 则  $G$  的第  $k$  个顶点的度等于 \_\_\_\_\_。

8. 为得到一棵排序二叉树的有序序列, 应该对该二叉树进行 \_\_\_\_\_ 遍历。

9. 对于一个关键字序列  $\{k_1, k_2, \dots, k_n\}$ , 若有  $k_i = k_j$  ( $i < j$ ), 排序之前  $k_i$  在  $k_j$  之前, 若某种排序方法使得排序后  $k_j$  在  $k_i$  之前, 则称这种排序方法是 \_\_\_\_\_。

10. 5 阶 B\_树中, 每个结点最多有\_\_\_\_\_个关键码, 最少必须有\_\_\_\_\_关键码。

得分	评阅人

二、选择(30 分,将你的选择填在下表中)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

1. 下列函数在最坏情况下的时间复杂度是[\_\_\_\_\_]

```
void unknown(int n)
{
    int i,n;
    for(i=n;i>0;i=i/2)
        cout<<i;
}
```

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

2. 设单链表中结点的结构为 (data, next)。已知指针 p 所指结点不是尾结点, 若在 \*p 之后插入结点 \*s, 则应执行下列[\_\_\_\_\_]操作?

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;

3. 设有单循环链表, 指针 rear 指向链表尾部, 现在要在链表尾部插入节点 s, 则应该执行下面哪一种操作?

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

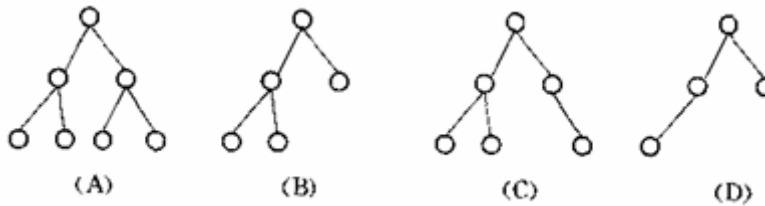
4. 设有一个顺序栈 S, 元素  $a_1, a_2, a_3, a_4, a_5$  依次进栈, 如果 5 个元素的出栈顺序为  $a_3, a_4, a_2, a_5, a_1$ , 则顺序栈的容量至少应为[\_\_\_\_\_]

A. 6    B. 5    C. 3    D. 2

5. 以顺序表实现的循环对列, front 表示队头, rear 表示列尾, 队列长度为 n, 顺序表下标从 0 开始, 则队列中当前元素个数是[\_\_\_\_\_]计算

A. rear-front    B. fron-rear  
C. (rear-front+n)%n    D. front+1-rear

6. 广义表  $(a, ((i, j), k), (a, b), d, e)$  的长度和深度分别是[\_\_\_\_\_]  
 A. 5, 2    B. 5, 3    C. 4, 3    D. 5, 1
7. 在下面的 4 棵二叉树中, [\_\_\_\_\_]不是完全二叉树



8. 已知某二叉树的先序遍历序列是  $abdecf$ , 中序遍历序列是  $dbaecf$ , 则这棵二叉树的后序遍历序列是[\_\_\_\_\_]
   
A.  $dbeacf$     B.  $debcfa$     C.  $deabcf$     D.  $debfca$
9. 在一棵二叉树的二叉链表表示中, 假设  $B_p$  表示非空指针域的个数,  $K_p$  表示空指针域个数, 则  $K_p$  与  $B_p$  的关系是[\_\_\_\_\_].
   
A.  $K_p=B_p+1$ ,    B.  $K_p=B_p+2$ ,    C.  $K_p=B_p$ ,    D.  $K_p=B_p-1$
10. 任何一棵二叉树的叶节点在先序、中序和后序遍历中, 其相对次序[\_\_\_\_\_]
   
A 不发生改变,    B. 会逆序改变    C. 会随机改变    D. 不确定
11. 无向图中一个顶点的度是指图中[\_\_\_\_\_].
   
A. 通过该顶点的简单路径数    B. 与该顶点相邻接的顶点数
   
C. 通过该顶点的回路数    D. 与该顶点连通的顶点数
12. 在 AOE 网络中, 关键路径指的是[\_\_\_\_\_].
   
A. 从源点到汇点的最长路径    B. 从源点到汇点的最短路径
   
C. 最长的回路    D. 最短的回路
13. 在待排序元素基本有序时, 最好使用[\_\_\_\_\_]排序方法。
   
A. 选择排序    B. 插入排序    C. 快速排序    D. 基数排序
14. AVL 树是一种平衡的二叉排序树, 树中任意节点的[\_\_\_\_\_]
   
A. 左、右子树的高度均相同,    B. 左子树的高均大于右子树的高度
   
C. 左、右子树的高度的差的绝对值不超过 1,    D. 右子树的高均大于左子树的高度
15. 一棵有  $n$  个结点的树, 其中所有分枝结点的度均为  $k$ , 则该树中叶子结点的个数为 [\_\_\_\_\_]
   
A.  $((k-1)n+1)/k$     B.  $((k-1)n-1)/k$     C.  $(k-1)n/k$     D.  $(k+1)n/k$

得分	评阅人

### 三、算法与程序填空 (24 分)

(第 5、第 6 题任选一道)

1.(3')写出下列递归过程的执行结果\_\_\_\_\_。

```
void unknown ( int w ) {
    if ( w ) {
        unknown ( w-1 );
        for ( int i = 1; i <= w; i++ ) cout <<i;
        cout << endl;
    }
}

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

2. (3')说明下述递归程序的功能\_\_\_\_\_

```
int unknown ( BinTreeNode * t )
{
    if ( t == NULL ) return 0;
    else if ( t->leftChild == NULL && t->rightChild == NULL ) return
1;
    else return  unknown ( t->leftChild ) + unknown ( t->
rightChild );
}
```

3. (4')下面是用 c++语言编写的对不带头结点的单链表进行就地逆置的算法, 请完成程序。

```
void List_reverse(ListNode *L) //ListNode 表示链表节点
{
    p=L;
    _____;
    while(p!=NULL) {
```

```

        s=p;
        p=p->next;
        _____;
        L=s;
    }
}

```

4. (6') 顺序表的二分查找，返回待查找的关键字 key 的数组下标，弱查找不成功，返回 0，完成程序。

```

#define N 1000
int Search_Bin(int A[N], int key)
{
    int low, high, mid;

    low =1, high=N-1;
    while( _____ ) {
        mid=_____ ;
        if (key == S[mid])      return  mid;
        else if ( key < S[mid])  high=mid-1;
        else _____ ;
    }
    return 0;
}

```

5. (8' ) 已知记录序列 a[1..n] 中的关键字各不相同，可按如下方法实现计数排序：另设数组 c[1..n]，对每条记录 a[i]，统计序列关键字值比它小的记录个数，存于 c[i] 中，这样 c[i]=0 所对应的记录必为关键字最小的记录，然后依以 c[i] 值的大小对 a 中记录进行重新排列。完成下面的计数排序算法。

```

void CountSort(int a[], int b[], int n)
{//a[1..n] 原输入序列，b[1..n] 排好序的序列
    int *c=new int[n+1];
    int i, j;
    for(i=1; i<=n; i++)

```

```

{
c[i]=0;
for(j=1;j<=n;j++)
    if(a[j]<a[i])
        [_____] ;
}
for(i=1;i<=n;i++)
    [_____];
}
该算法的时间复杂度是[_____]

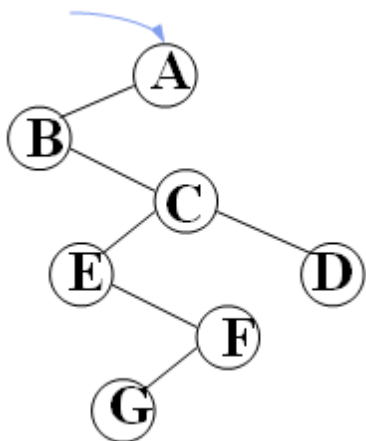
```

6.(8')根据快速排序算法的原理，写出快速排序算法程序。

得分	评阅人

#### 四、综合应用（36 分）

1.(6')下图是二叉树是由一查普通树转换而来，将其还原成普通树的形式（6 分）



2. (6') 假设用于通信的电文仅由 5 个符 (a, b, c, d, e, f) 组成, 这 8 个字符的频率为 (3, 12, 2, 4, 15, 45, 21, 8)。

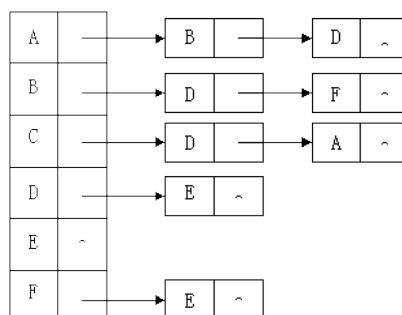
(1) 画出 huffman 树

(2) 写出每个字符的的编码

3. (6') 下图是用邻接表存储的有向图, 完成下面的问题

① 写出此图的深度优先遍历序列

② 出该图的一个拓排序列



4. (6') 给定数据序列 (1, 2, 3, 8, 4, 12, 5)

(1) 构造二叉排序树

(2) 构造平衡二叉排序树

5. (6') 设哈希表长度为 11, 哈希函数  $h(x)=x\%11$ , 给定的关键字序列为: 1, 13, 13, 34, 38, 33, 27, 22.

(1) 用哈希函数计算每个关键字的地址, 将关键字填入下表中 (如果关键字有冲突, 按顺序将其填入相同的空格中)

0	1	2	3	4	5	6	7	8	9	10

(2) 画出用线性探测法解决冲突, 构造的哈希表

0	1	2	3	4	5	6	7	8	9	10



6. (6') 给定数据序列 (42 , 76 , 157 , 137 , 93 , 114 , 159 , 12 , 121 , 11)

(1) 写出第二趟选择排序的结果

(2) 构建初始堆