

试卷代号:1010

座位号

中央广播电视大学 2010—2011 学年度第二学期“开放本科”期末考试

## 数据结构 试题

2011 年 7 月

题号	一	二	三	四	五	六	总分
分数							

得分	评卷人

一、单项选择题(在括号内填写所选择的标号。每小题 2 分,共 18 分)

1. 一种抽象数据类型包括数据和( )两个部分。  
A. 数据类型  
B. 操作  
C. 数据抽象  
D. 类型说明
2. 在一个长度为  $n$  的顺序表的表尾插入一个新元素的时间复杂度为( )。  
A.  $O(1)$   
B.  $O(n)$   
C.  $O(n^2)$   
D.  $O(\log_2 n)$
3. 已知  $L$  是带表头附加结点的单链表,删除第一个元素结点的语句是( )。  
A.  $L = L \rightarrow \text{link};$   
B.  $L \rightarrow \text{link} = L \rightarrow \text{link} \rightarrow \text{link};$   
C.  $L \rightarrow \text{link} \rightarrow \text{link} = L;$   
D.  $L \rightarrow \text{link} = L;$
4. 在下列广义表中,( )又是一个线性表。  
A.  $E(a, (b, c))$   
B.  $E(a, E)$   
C.  $E(a, b)$   
D.  $E(a, ( ))$
5. 在一棵深度为 3 的三叉树中,最多含有( )个结点。假定树根结点的深度为 1。  
A. 10  
B. 11  
C. 12  
D. 13

6. 向一棵 AVL 树插入元素时,可能引起对最小不平衡子树的双向旋转的调整过程,此时需要修改相关( )个指针域的值。

- A. 2
- B. 3
- C. 4
- D. 5

7. 在一个具有  $n$  个顶点的有向图的邻接矩阵表示中,删除一条边  $\langle i, j \rangle$  需要的时间复杂度为( )。

- A.  $O(1)$
- B.  $O(i)$
- C.  $O(n)$
- D.  $O(n^2)$

8. 在一棵 B 树中,当插入一个元素时,若最终引起树根结点的分裂,则新树的高度比原树的高度( )。

- A. 减 1
- B. 减 2
- C. 增 1
- D. 增 2

9. 对存储有  $n$  个元素的长度为  $m$  的散列表进行搜索,平均搜索长度与( )有关。

- A.  $n$
- B.  $m$
- C.  $n/m$
- D.  $n * m$

得 分	评卷人

二、填空题(在横线处填写合适的内容。每小题 2 分,共 14 分)

1. 链表只适用于\_\_\_\_\_查找。

2. 归并排序算法的时间复杂度为\_\_\_\_\_。

3. 在一个链式队列中,若队头指针与队尾指针的值相同,则表示该队列至多有\_\_\_\_\_个结点。

4. 假定一棵树的广义表表示为  $a(b, c, d(e, f), g(h))$ ,则结点  $f$  的层数为\_\_\_\_\_。假定树根结点的层数为 0。

5. 从一棵二叉搜索树中搜索一个元素时,若给定值大于根结点的值,则需要向根的\_\_\_\_\_继续搜索。

6. 每次从第  $i$  至第  $n$  个元素中顺序挑选出一个最小元素,把它交换到第  $i$  个位置,此种排序方法叫做\_\_\_\_\_排序。

7. 快速排序在最坏情况下的时间复杂度为\_\_\_\_\_。

得 分	评卷人

三、判断题(在每小题后面的括号内打对号“√”表示叙述正确或打叉号“×”表示叙述错误。每小题 2 分,共 14 分)

1. 若每次从队列中取出的是具有最高优先权的元素,则称此队列为优先级队列。

( )

2. 递归定义的数据结构通常不需要采用递归的算法对其运算。( )

3. 当从一个最小堆中删除一个元素时,需要把堆尾元素填补到堆顶位置,然后再按条件把它逐层向下调整,直到调整到合适位置为止。( )

4. 对于一棵具有  $n$  个结点、高度为  $h$  的二叉树,进行任一种次序遍历的时间复杂度均为  $O(n)$ 。( )

5. 对于同一组记录集合,生成二叉搜索树的形态与插入记录的次序无关。( )

6. 装载因子是散列存储中的一个重要指标,它反映了散列表的装满程度。( )

7. 在一棵 B 树中,所有叶结点都处在同一层上。( )

得 分	评卷人

四、运算题(每小题 6 分,共 30 分)

1. 假定一棵普通树的广义表表示为  $a(b(e),c(f(h,i,j),g))$ ,分别写出对其进行先根和按层遍历的结果。

先根:

按层:

2. 假定一个线性表为  $(38,52,25,74,68,16,30)$ ,根据此表中的元素排列次序生成一棵二叉搜索树,求出该二叉搜索树中分支结点数和叶子结点数。

分支结点数:

叶子结点数:

3. 已知一个数据集合为{28,12,16,49,34,30},试把它调整为一个最大堆。

最大堆:

4. 已知一个图的顶点集 V 和边集 G 分别为:

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

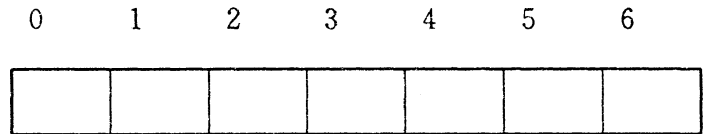
$E = \{ \langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 4 \rangle, \langle 2, 5 \rangle, \langle 3, 4 \rangle, \langle 4, 5 \rangle, \langle 5, 1 \rangle, \langle 5, 2 \rangle, \langle 5, 3 \rangle \};$

假定此图采用邻接矩阵表表示,根据图的遍历算法分别写出从顶点 1 出发进行深度优先搜索和广度优先搜索所得到的顶点序列。

深度优先搜索序列:

广度优先搜索序列:

5. 设散列表的长度  $m=7$ ,散列函数为  $H(K) = K \text{ mod } m$ ,若采用线性探查法解决冲突,待依次插入的关键码序列为{19,14,23,68,20},试求出最后得到的散列表。



得分	评卷人

**五、算法分析题(每小题 8 分,共 16 分)**

1. 下面算法的功能为:将两个有序单链表合并成一个有序单链表并返回其表头指针。阅读算法,在划有横线的上面填写合适的内容。

```

ListNode * Merge1(ListNode * & p1, ListNode * & p2)
{
    ListNode * p=new ListNode, * f=p;
    while(p1! =NULL && p2! =NULL)
    {
        if(p1->data<=p2->data) {
            p->link=p1; _____;
        }
    }
}
    
```

```

else {p->link=p2; p2=p2->link;}
_____};
}
if(p1!=NULL) p->link=p1; else p->link=p2;
p1=p2=NULL;
return f->link;
}

```

2. 已知二叉树中的结点类型 BinTreeNode 定义为:

```
struct BinTreeNode {ElemType data; BinTreeNode * left, * right};
```

其中 data 为结点值域, left 和 right 分别为指向左、右子女结点的指针域。根据下面算法的定义指出其功能。算法中参数 BT 指向一棵二叉树。

```
BinTreeNode * BTreeSwopX(BinTreeNode * BT)
{
if(BT==NULL) return NULL;
else {
BinTreeNode * pt=new BinTreeNode;
pt->data=BT->data;
pt->right=BTreeSwopX(BT->left);
pt->left=BTreeSwopX(BT->right);
return pt;
}
}

```

算法功能:

得 分	评卷人

### 六、算法设计题(8分)

已知二叉树中的结点类型 BinTreeNode 定义为：

```
struct BinTreeNode {char data; BinTreeNode * left, * right;};
```

其中 data 为结点值域, left 和 right 分别为指向左、右子女结点的指针域, 根据下面函数声明编写出求一棵二叉树中分支结点总数的算法, 该总数值由函数返回。假定参数 BT 初始指向这棵二叉树的根结点。

```
int BTreeCount(BinTreeNode * BT);
```

试卷代号:1010

中央广播电视大学 2010—2011 学年度第二学期“开放本科”期末考试

## 数据结构 试题答案及评分标准

(供参考)

2011 年 7 月

一、单项选择题(在括号内填写所选择的标号。每小题 2 分,共 18 分)

1. B            2. A            3. B            4. C            5. D  
6. D            7. A            8. C            9. C

二、填空题(在横线处填写合适内容。每小题 2 分,共 14 分)

1. 顺序  
2.  $O(n\log_2 n)$   
3. 1  
4. 2  
5. 右子树  
6. 直接选择  
7.  $O(n^2)$

三、判断题(在每小题后面的括号内打对号“√”表示叙述正确或打叉号“×”表示叙述错误。每小题 2 分,共 14 分)

1. 对            2. 错            3. 对            4. 对            5. 错  
6. 对            7. 对

四、运算题(每小题 6 分,共 30 分)

1. 先根:a,b,e,c,f,h,i,j,g    //3 分  
按层:a,b,c,e,f,g,h,i,j    //3 分  
2. 分支结点数:4    //3 分  
叶子结点数:3    //3 分  
3. {49,34,30,12,28,16}

4. 深度搜索序列:1,2,4,5,3 //3 分

广度搜索序列:1,2,3,4,5 //3 分

5. 散列表中的每个元素占 1 分,全对得 6 分。

0	1	2	3	4	5	6
14	20	23			19	68

### 五、算法分析题(每小题 8 分,共 16 分)

1.  $p1=p1->link, p=p->link$  //每空 4 分

2. 生成一棵新二叉树并返回树根指针,该二叉树是已知二叉树 BT 中所有结点的左、右子树(或左、右孩子的值)交换的结果。

### 六、算法设计题(8 分)

评分标准:根据编程酌情给分。

```
int BTreeCount(BinTreeNode * BT)
```

```
{  
    if(BT==NULL) return 0; //2 分  
    else if(BT->left==NULL && BT->right==NULL) return 0; //4 分  
    else return BTreeCount(BT->left)+BTreeCount(BT->right)+1; //8 分  
}
```

说明:函数体中的每个 else 保留字可以省略。