

试卷代号:1010

座位号

中央广播电视大学 2005—2006 学年度第一学期“开放本科”期末考试

计算机专业 数据结构 试题

2006 年 1 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(9 小题,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,删除第一个结点的语句是()。
A. $L=L->link;$
B. $L->link=L->link->link;$
C. $L=L;$
D. $L->link=L;$
4. 下列广义表中的线性表是()。
A. $E(a, (b, c))$
B. $E(a, E)$
C. $E(a, b)$
D. $E(a, ())$
5. 在一棵树的左子女—右兄弟表示法中,一个结点的右子女是该结点的()结点。
A. 兄弟
B. 父子
C. 祖先
D. 子孙

9. 在使用 Kruskal 算法构造连通网络的最小生成树时,只有当一条候选边的两个端点不在同一个_____上,才会被加入到生成树中。

10. 在堆排序中,对 n 个记录建立初始堆需要调用_____次调整算法。

11. 在对 n 个数据对象的二路归并排序中,每趟归并的时间复杂度为_____。

12. 在一棵 m 阶 B 树上,每个非根结点的关键码数最少为_____个。

得 分	评卷人

三、判断题,在每小题前面打对号表示正确或打叉号表示错误(10 小题,每小题 1 分,共 10 分)

- () 1. 多维数组是一种复杂的数据结构,数组元素之间的关系既不是线性的也不是树形的。
- () 2. 若每次从队列中取出的是具有最高优先权的元素,则称这种队列为优先级队列。
- () 3. 递归定义的数据结构通常不需要用递归的算法来实现对它的操作。
- () 4. 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。
- () 5. 对于一棵具有 n 个结点,其高度为 h 的二叉树,进行任一种次序遍历的时间复杂度为 $O(n)$ 。
- () 6. 对于同一组记录,生成二叉搜索树的形态与插入记录的次序无关。
- () 7. 在每个 AOE 网络中只有一条关键路径。
- () 8. 图的深度优先搜索是一种典型的回溯搜索的例子,可以通过递归算法求解。
- () 9. 装载因子是散列表的一个重要参数,它反映了散列表的装满程度。
- () 10. 在一棵 B 树中,所有叶结点都处在同一层上,所有叶结点中空指针数等于所有关键码的总数加 1。

得 分	评卷人

四、运算题(5 小题,每小题 6 分,共 30 分)

1. 假定一棵二叉树广义表表示为 $a(b(c), d(e, f))$, 分别写出对它进行中序、后序、按层遍历的结果。

中序:

后序:

按层:

2. 一个一维数组 $a[10]$ 中存储着有序表 $(15, 26, 34, 39, 45, 56, 58, 63, 74, 76)$, 根据折半搜索所对应的判定树, 写出该判定树中度为 1 的结点个数, 并求出在等概率情况下进行成功搜索时的平均搜索长度。

度为 1 的结点个数:

平均搜索长度:

3. 假定一个线性序列为 $(38, 42, 55, 15, 23, 44, 30, 74, 48, 26)$, 根据此线性序列中元素的排列次序生成一棵二叉搜索树, 求出该二叉搜索树中左子树为空的所有单支结点、右子树为空的所有单支结点和所有叶子结点, 请按照结点值从小到大的次序写出。

左子树为空的所有单支结点:

右子树为空的所有单支结点:

所有叶子结点:

4. 已知一个图的顶点集 V 和边集 G 分别为:

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

$E = \{ \langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 2, 4 \rangle, \langle 2, 5 \rangle, \langle 3, 4 \rangle, \langle 4, 5 \rangle, \langle 4, 6 \rangle, \langle 5, 1 \rangle, \langle 5, 3 \rangle, \langle 6, 5 \rangle \}$;

假定该图采用邻接表表示, 每个顶点邻接表中的边结点都是按照终点序号(即数值域的值)从小到大的次序链接的, 试写出:

(1) 从顶点 1 出发进行深度优先搜索所得到的顶点序列;

(2) 从顶点 1 出发进行广度优先搜索所得到的顶点序列。

(1):

(2):

5. 已知一个数据序列为 $\{6, 45, 27, 23, 41, 5, 56, 64\}$, 把它调整为最大堆并给出进行两趟交换和堆排序后的结果(即尾部得到 2 个最大数)。

最大堆:

两趟排序后结果:

得分	评卷人

五、算法分析题(3 小题, 每小题 6 分, 共 18 分)

1. 该算法功能为: 从表头指针为 la 的、按值从小到大排列的有序链表中删除所有值相同的多余元素, 并释放被删结点的动态存储空间。阅读算法, 按标号填写空缺的内容, 要求统一填写在算法后面的标记处。

```
void purge_linkst(ListNode * &la)
{
    ListNode * p, * q;
    if(la == NULL) return;
    q = la; p = la->link;
    while(p) {
        if( _____ (1) _____ ) {q = p; p = p->link; }
```

```

else {
    q->link= _____(2)_____ ;
    delete(p);
    p= _____(3)_____ ;
}
}
}

```

(1)

(2)

(3)

2. 请写出下面算法的功能,其中 Stack 表示栈类,Queue 表示队列类。

```

void unknown(Queue &Q)
{
    Stack S;int d;
    S.InitStack( );
    while(! Q.IsEmpty()){
        Q.DeQueue(d);    //出队列元素值由变量 d 带回
        S.Push(d);
    }
    while(! S.IsEmpty( )) {
        S.Pop(d);    //出栈元素值由变量 d 带回
        Q.Enqueue(d);
    }
}

```

算法功能:

3. 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。下面函数的功能是从二叉树 BT 中查找值为 X 的结点, 若查找成功则返回结点地址, 否则返回空。按标号填写空缺的内容, 要求统一填写在算法后面的标记处。

```
BinTreeNode * BTF(BinTreeNode * BT, ElemType x)
```

```
{  
    if(BT == NULL) _____ (1) _____;  
    else {  
        if(BT->data == x) _____ (2) _____;  
        else {  
            BinTreeNode * t;  
            if(t = BTF(BT->left, x)) return t;  
            _____ (3) _____;  
            return NULL;  
        }  
    }  
}
```

(1)

(2)

(3)

得分	评卷人

六、算法设计题(2 小题, 每小题 6 分, 共 12 分)

1. 在一个带头附加结点的单链表 L 中, 假定所有结点的值按递增顺序排列, 试编写一个 while 循环补充下面函数, 功能是删除表 L 中所有其值大于等于 min, 同时小于等于 max 的结点。

```
void rangeDelete(ListNode * &L, ElemType min, ElemType max)
```

```
{
```

```
    ListNode * q=L, * p=L->link;
```

```
    //添加的 while 循环位置
```

```
}
```

```
//请把 while 循环内容写在此行下面
```

2. 已知二叉搜索树中的结点类型 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。参数 BST 指向一棵二叉搜索树的根结点。试根据下面的函数声明编写一个非递归算法, 从 BST 树中搜索出具有 item 参数值的结点, 若搜索成功则返回该结点的地址, 否则返回 NULL。

```
BinTreeNode * Find (BinTreeNode * BST, const ElemType&item);
```

```
//请把函数定义写在此行下面
```


试卷代号:1010

中央广播电视大学 2005—2006 学年度第一学期“开放本科”期末考试

计算机专业 数据结构 试题答案及评分标准

(供参考)

2006 年 1 月

一、单项选择题,在括号内填写所选择的标号(9 小题,每小题 2 分,共 18 分)

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

二、填空题,在横线处填写合适内容(12 小题,每小题 1 分,共 12 分)

1. 数据封装

2. 值

3. 不一定

4. 栈顶指针

5. 递归

6. 85

7. $2i+2$

8. 48

9. 连通分量

10. $n/2$

11. $O(n)$

12. $\lceil m/2 \rceil - 1$

三、判断题,在每小题前面打对号表示正确或打叉号表示错误(10 小题,每小题 1 分,共 10 分)

1. 对 2. 对 3. 错 4. 对 5. 对
6. 错 7. 错 8. 对 9. 对 10. 对

四、运算题(5 小题,每小题 6 分,共 30 分)

1. 中序:c,b,a,e,d,f //2 分
后序:c,b,e,f,d,a //2 分
按层:a,b,d,c,e,f //2 分
2. 度为 1 的结点个数:3 //3 分
平均搜索长度:29/10 //3 分
3. 左子树为空的所有单支结点:15,23,42,44 //2 分
右子树为空的所有单支结点:30 //2 分
所有叶子结点:26,48,74 //2 分
4. (1)1,2,4,5,3,6 //3 分
(2)1,2,3,4,5,6 //3 分
5. 最大堆:{64,45,56,23,41,5,27,6} //3 分
两趟排序结果:{45,41,27,23,6,5,56,64} //3 分

五、算法分析题(3 小题,每小题 6 分,共 18 分)

1. (1)p->data>q->data(或 p->data! =q->data) //2 分
(2)p->link //2 分
(3)q->link //2 分
2. 利用“栈”作为辅助数据结构,将队列 Q 中的元素逆置(即按相反次序放置)。
3. (1)return NULL //2 分
(2)return BT //2 分
(3)if(t==BTF(BT->right,x))return t //2 分

六、算法设计题(2 小题,每小题 6 分,共 12 分)

评分标准:根据编写正确程度酌情给分。

1. while(p! =NULL) {
if(p->data>=min&& p->data<=max) {

```

q->link=p->link;delete p;p=q->link;
}
else{q=p;p=p->link; }
}

```

2. BinTreeNode * Find(BinTreeNode * BST, const ElemType&item)

```

{
while(BST! =NULL)
{
if(item==BST->data)return BST;
else if (item<BST->data)BST=BST->left;//关键字 else 可省略
else BST=BST->right;
}
return NULL;
}

```