

得 分	评卷人

三、运算题(每小题 6 分,共 24 分)

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

先序:

中序:

按层:

2. 假定一个大根堆为 $(64,55,38,30,15,24)$, 则求出从中删除一个元素后得到的大根堆。

大根堆:

3. 已知一个图的顶点集 V 和边集 G 分别为:

$$V = \{0,1,2,3,4\};$$

$$E = \{(0,1)12, (0,2)2, (0,3)6, (1,3)9, (2,3)4, (1,4)5\};$$

试写出按照克鲁斯卡尔算法求解最小生成树的过程中依次得到的各条边, 及最小生成树的权。

最小生成树中的各条边:

最小生成树的权:

4. 假定一组数据为 $(35,58,62,25,43,18,65,74)$, 写出在二路归并排序的过程中第二次归并后的数据排列结果。

数据排列结果:

得 分	评卷人

四、阅读算法,回答问题(每小题 8 分,共 16 分)

1. `void AB(List& L)`

```
{
    InsertRear(L,30);
    InsertFront(L,58);
    Delete(L,12);
}
```

假定调用该算法时线性表 L 为 $(35,19,12,15)$, 则调用返回后线性表 L 变为:

2. 写出下面递归算法的功能,假定 HL 为一个单链表的表头指针。

```
void C2(LNode * HL)
{
    if(HL != NULL) {
        C2(HL->next);
        cout<<HL->data<<' ' ;
    }
}
```

算法功能:

得分	评卷人

五、算法填空,在画有横线的地方填写合适的内容(每小题 6 分,共 12 分)

1. 向以 BST 为树根指针的二叉搜索树上插入值为 item 的结点的递归算法。

```
void Insert(BTreeNode * & BST, const ElemType& item)
{
    if(BST == NULL) {
        BST = new BTreeNode;
        BST->data = item;
        BST->left = NULL;
        BST->right = NULL;
    }
    else if(item < BST->data) Insert(BST->left, item);
    else _____ ;
}
```

2. 下面算法的功能为:根据 BT 所指的二叉树生成一棵新的二叉树并返回树根指针,新二叉树是已知二叉树 BT 中所有结点的左、右子树交换的结果。

```
BTreeNode * BTreeSwopX(BTreeNode * BT)
{
```

```

if(BT==NULL) return NULL;
else {
    BTreeNode * pt=new BTreeNode;
    pt->data= BT->data;
    pt->left=BTtreeSwopX(BT->right);
    pt->right=_____ ;
    return pt;
}
}

```

得 分	评卷人

六、编写算法(8分)

根据下面函数原型编写出从二叉搜索树 BST 中查找并返回所有结点的最大值的算法,若为空树则退出运行。

```
ElemType FM(BTreeNode * BST);
```

试卷代号:2075

中央广播电视大学 2009—2010 学年度第一学期“开放专科”期末考试

数据结构 试题答案及评分标准

(供参考)

2010 年 1 月

一、单选题(每小题 2 分,共 20 分)

评分标准:选对者得 2 分,否则不得分。

- | | | | | |
|------|------|------|------|-------|
| 1. C | 2. A | 3. C | 4. B | 5. C |
| 6. B | 7. A | 8. B | 9. D | 10. D |

二、填空题(每小题 2 分,共 20 分)

1. 索引
2. $O(1)$
3. 21
4. 左
5. 堆顶
6. $n-1$
7. 理想平衡
8. 20
9. m
10. $O(n)$

三、运算题(每小题 6 分,共 24 分)

1. 先序: a,b,c,d,e,f,g //2 分
中序: c,b,d,a,f,g,e //2 分
按层: a,b,e,c,d,f,g //2 分
2. (55,30,38,24,15)
3. (0,2)2, (2,3)4, (1,4)5, (1,3)9 //4 分

最小生成树的权:20 //2分

4. (25,35,58,62,18,43,65,74)

四、阅读算法,回答问题(每小题8分,共16分)

评分标准:根据答案正确与叙述的完整程度酌情给分。

1. (58,35,19,15,30)

2. 按逆序(即原链接的相反次序)输出以HL为表头指针的单链表中的每个结点的值。

五、算法填空,在画有横线的地方填写合适的内容(每小题6分,共12分)

1. Insert(BST->right, item)

2. BTreeSwopX(BT->left)

六、编写算法(8分)

评分标准:根据编程完整程度酌情给分。

注:可以不用指针t,而直接使用值参BST。

```
ElemType FM(BTreeNode * BST) {
```

```
    if(BST==NULL) {cerr<<"此树为空"<<endl; exit(1);} //2分
```

```
    BTreeNode * t=BST;
```

```
    while(t->right!=NULL) t=t->right; //6分
```

```
    return t->data; //8分
```

```
}
```