

试卷代号:1010

座位号

中央广播电视大学 2002—2003 学年度第二学期“开放本科”期末考试

计算机专业数据结构试题

2003 年 7 月

题号	一	二	三	四	五	六	总分
分数							

得分	评卷人

一、选择题(每小题 1 分,共 10 分)

(得分)1. 在一个长度为 n 的顺序表的任一位置插入一个新元素的渐进时间复杂度为()。

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

(得分)2. 带头结点的单链表 $first$ 为空的判定条件是:()

- A. $first == NULL$
- B. $first \rightarrow link == NULL$
- C. $first \rightarrow link == first$
- D. $first! = NULL$

(得分)3. 当利用大小为 n 的数组顺序存储一个队列时,该队列的最大长度为()。

- A. $n-2$
- B. $n-1$
- C. n
- D. $n+1$

得分	评卷人

二、填空题(每小题 1 分,共 10 分)

- (得分)11. 二维数组是一种非线性结构,其中的每一个数组元素最多有_____个直接前驱(或直接后继)。
- (得分)12. 将一个 n 阶三对角矩阵 A 的三条对角线上的元素按行压缩存放于一个一维数组 B 中, $A[0][0]$ 存放于 $B[0]$ 中。对于任意给定数组元素 $B[K]$,它应是 A 中第_____行的元素。
- (得分)13. 链表对于数据元素的插入和删除不需移动结点,只需改变相关结点的_____域的值。
- (得分)14. 在一个链式栈中,若栈顶指针等于 $NULL$ 则为_____。
- (得分)15. 主程序第一次调用递归函数被称为外部调用,递归函数自己调用自己被称为内部调用,它们都需要利用栈保存调用后的_____地址。
- (得分)16. 在一棵树中,_____结点没有后继结点。
- (得分)17. 一棵树的广义表表示为 $a((c,d(e,f),g(h)),i(j,k(x,y)))$,结点 f 的层数为_____。假定根结点的层数为 0。
- (得分)18. 在一棵 AVL 树(高度平衡的二叉搜索树)中,每个结点的左子树高度与右子树高度之差的绝对值不超过_____。

(得分)19. $n(n>0)$ 个顶点的无向图最多有_____条边,最少有_____条边。

(得分)20. 在索引存储中,若一个索引项对应数据对象表中的一个表项(记录),则称此索引为_____索引,若对应数据对象表中的若干个表项,则称此索引为_____索引。

得分	评卷人

三、判断题(每小题 1 分,共 10 分)

(得分)21. 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。 ()

(得分)22. 链式存储在插入和删除时需要保持物理存储空间的顺序分配,不需要保持数据元素之间的逻辑顺序。 ()

(得分)23. 在用循环单链表表示的链式队列中,可以不设队头指针,仅在链尾设置队尾指针。 ()

(得分)24. 通常递归的算法简单、易懂、容易编写,而且执行的效率也高。 ()

(得分)25. 一个广义表的表尾总是一个广义表。 ()

(得分)26. 当从一个小根堆(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。 ()

(得分)27. 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 $O(h)$ 。 ()

(得分)28. 存储图的邻接矩阵中,邻接矩阵的大小不但与图的顶点个数有关,而且与图边数也有关。 ()

(得分)29. 直接选择排序是一种稳定的排序方法。 ()

(得分)30. 闭散列法通常比开散列法时间效率更高。 ()

得分	评卷人

四、运算题(每小题 8 分,共 40 分)

(得分)31. 设有一个 10×10 的对称矩阵 A,将其下三角部分按行存放于一个一维数组 B 中,A[0][0]存放于 B[0]中,那么 A[8][5]存放于 B 中什么位置。

(得分)32. 这是一个统计单链表中结点的值等于给定值 x 的结点数的算法,其中 while 循环有错,请重新编写出正确的 while 循环。

```
int count (ListNode * Ha,ElemType x)
{ //Ha 为不带头结点的单链表的头指针
  int n=0;
  while (Ha->link! =NULL){
    Ha=Ha->link;
    if (Ha->data==x)n++;
  }
  return n;
}
```

(得分)33. 已知一棵二叉树的前序和中序序列,求该二叉树的后序序列。

前序序列:A,B,C,D,E,F,G,H,I,J

中序序列:C,B,A,E,F,D,I,H,J,G

后序序列:

(得分)34. 已知一个有序表(15,26,34,39,45,56,58,63,74,76,83,94)顺序存储于一维数组 a[12]中,根据折半搜索过程填写成功搜索下表中所给元素 34,56,58,63,94 时的比较次数。

元素值	34	56	58	63	94
比较次数					

(得分)35. 设散列表为 HT[17],待插入关键码序列为{Jan, Feb,Mar, Apr,May,June, July, Aug,Sep,Oct,Nov,Dec},散列函数为 $H(\text{key}) = \lfloor i/2 \rfloor$,其中,i 是关键码第一个字母在字母表中的序号。现采用线性探查法解决冲突。

字母	A	B	C	D	E	F	G	H	I	J	K	L	M
序号	1	2	3	4	5	6	7	8	9	10	11	12	13
字母	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
序号	14	15	16	17	18	19	20	21	22	23	24	25	26

(1) 试画出相应的散列表；

(2) 计算等概率下搜索成功的平均搜索长度；

得分	评卷人

五、算法分析题(每小题 8 分,共 24 分)

(得分) 36. 阅读下列算法,并补充所缺语句

```

void purge_linkst(LisNode * & la){
//从头指针为 la 的带头结点的有序链表中删除所有值相同的多余元素,
//并释放被删结点空间。
    ListNode p,q t; ElemType temp;
    p=la->link;
    while (p!=NULL){
        q=p;
        temp=p->data;
        p=p->link;
        if (p!=NULL && _____)p=p->link;
        else {
            while (p!=NULL && _____){
                t=p; p=p->link;
                delete t;
            }
            q->link=p;
        }
    }
}

```

(得分)37. 下面给出一个排序算法,它属于数据表类的成员函数,其中 `currentSize` 是数据表实例的当前长度,`Vector[]`是存放数据表元素的一维数组。

```
template<class T>
void dataList<T>::unknown(){
    T temp; int i,j,n=currentSize;
    for (i=1;i<n;i++)
        if(Vector[i].key<Vector[i-1].key){
            temp=Vector[i]; Vector[i]=Vector[i-1];
            for (j=i-2;j>=0;j--)
                if(temp.key<Vector[j].key)Vector[j+1]=Vector[j];
            else break;
            Vector [j+1]=temp;
        }
}
```

(1)写出该算法的功能。

(2)针对有 n 个数据对象的待排序的数据表,在最好情况下,算法的排序码比较次数和对象移动次数分别是多少?

比较次数:

移动次数:

(得分)38. 已知二叉树中的结点类型用 `BinTreeNode` 表示,被定义为:

```
struct BinTreeNode{ElemType data; BinTreeNode * leftChild, * rightChild;};
```

其中 `data` 为结点值域,`leftChild` 和 `rightChild` 分别为指向左、右子女结点的指针域。根据下面函数的定义指出函数的功能。算法中参数 `BT` 指向一棵二叉树的树根结点。

```
BinTreeNode * BinTreeSwopX(BinTreeNode * BT){
    if (BT==NULL)return NULL;
    else {
        BinTreeNode * pt=new BinTreeNode;
        pt->data=BT->data;
        pt->rightChild=BinTreeSwopX(BT->leftChild);
        pt->leftChild=BinTreeSwopX(BT->rightChild);
        return pt;
    }
}
```

得 分	评卷人

六、算法设计题(6分)

(得分)39. 已知二叉树中的结点类型用 BinTreeNode 表示,被定义为:

struct BinTreeNode { **char** data; BinTreeNode * leftChild, * rightChild; };其中 data 为点值域, leftChild 和 rightChild 分别为指向左、右子女结点的指针域,根据下面函数声明编出求一棵二叉树中结点总数的算法,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 BinTreeCount (BinTreeNode * BT);
```


试卷代号:1010

中央广播电视大学 2002—2003 学年度第二学期“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)

2003 年 7 月

一、选择题(每小题 1 分,共 10 分)

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

二、填空题(每小题 1 分,共 10 分)

11. 2
12. $\lfloor (K+1)/3 \rfloor$
13. 指针
14. 空栈
15. 返回
16. 叶子
17. 3
18. 1
19. $n(n-1)/2$ 0
20. 稠密 稀疏

三、判断题(每小题 1 分,共 10 分)

21. 对
22. 错
23. 对
24. 错
25. 对

- 26. 对
- 27. 错
- 28. 错
- 29. 错
- 30. 错

四、运算题(每小题 8 分,共 40 分)

31. 根据题意,矩阵 A 中当元素下标 I 与 J 满足 $I \geq J$ 时,任意元素 $A[I][J]$ 在一维数组 B 中的存放位置为 $I * (I+1)/2 + J$,因此, $A[8][5]$ 在数组 B 中位置为

$$8 * (8+1)/2 + 5 = 41。$$

```
32. while(Ha! = NULL){
    if(Ha->data == x)n++;
    Ha = Ha->link;
}
```

33. 后序序列:C,B,F,E,I,J,H,G,D,A

34. 判断结果:

元素值	34	56	58	63	94
比较次数	2	1	3	4	4

//对 1 个给 1 分,全对给 8 分

35. $H(\text{Jan}) = \lfloor 10/2 \rfloor = 5$, 成功. $H(\text{Feb}) = \lfloor 6/2 \rfloor = 3$, 成功.
 $H(\text{Mar}) = \lfloor 13/2 \rfloor = 6$, 成功. $H(\text{Apr}) = \lfloor 1/2 \rfloor = 0$, 成功.
 $H(\text{May}) = \lfloor 13/2 \rfloor = 6, = 7$, 成功, $H(\text{June}) = \lfloor 10/2 \rfloor = 5, = 6, = 7, = 8$, 成功.
 $H(\text{July}) = \lfloor 10/2 \rfloor = 5, = 6, = 7, = 8, = 9$, 成功.
 $H(\text{Aug}) = \lfloor 1/2 \rfloor = 0, = 1$, 成功. $H(\text{Sep}) = \lfloor 19/2 \rfloor = 9, = 10$, 成功.
 $H(\text{Oct}) = \lfloor 15/2 \rfloor = 7, = 8, = 9, = 10, = 11$, 成功.
 $H(\text{Nov}) = \lfloor 14/2 \rfloor = 7, = 8, = 9, = 10, = 11, = 12$, 成功.

$H(\text{Dec}) = \lfloor 4/2 \rfloor = 2$, 成功.

(1) 相应的散列表 (6 分) 错一个存储位置扣 1 分, 最多扣 6 分。

0	1	2	3	4	5	6	7	8	9	10	11	12	13
Apr	Aug	Dec	Feb		Jan	Mar	May	June	July	Sep	Oct	Nov	
(1)	(2)	(1)	(1)		(1)	(1)	(2)	(4)	(5)	(2)	(5)	(6)	

(2) 搜索成功的平均搜索长度为

$$1/12 * (1+2+1+1+1+1+1+2+4+5+2+5+6) = 31/12 \quad (2 \text{ 分})$$

五、算法分析题 (每小题 8 分, 共 24 分)

36. `p->data>temp` //4 分

`p->data=temp` //4 分

37. 算法功能及执行效率

(1) 该算法的功能是直接插入排序。(4 分)

(2) $n-1$ 0 (2 分 2 分)

38. 算法功能: 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、右子树交换的结果。

六、6 分, 请根据编程情况酌情给分。

39.

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

```
    if (BT==NULL)return 0;           //2 分
```

```
    else return BTreeCount (BT->leftChild)+BTreeCount (BT->rightChild)+1;
```

```
                                     //4 分
```

```
}
```