

全国 2017 年 10 月高等教育自学考试

数据结构试题

课程代码:02331

请考生按规定用笔将所有试题的答案涂、写在答题纸上。

选择题部分

注意事项:

1. 答题前,考生务必将自己的考试课程名称、姓名、准考证号用黑色字迹的签字笔或钢笔填写在答题纸规定的位置上。
2. 每小题选出答案后,用 2B 铅笔把答题纸上对应题目的答案标号涂黑。如需改动,用橡皮擦干净后,再选涂其他答案标号。不能答在试题卷上。

一、单项选择题:本大题共 15 小题,每小题 2 分,共 30 分。在每小题列出的备选项中只有一项是最符合题目要求的,请将其选出。

1. 下列选项中,与数据存储结构直接相关的是
A. 线性表 B. 双向链表 C. 二叉树 D. 有向图
2. 将 12 个数据元素保存在顺序表中,若第一个元素的存储地址是 100,第二个元素的存储地址是 105,则该顺序表最后一个元素的存储地址是
A. 111 B. 144 C. 155 D. 156
3. 设栈的初始状态为空,元素 1,2,3,4,5,6 依次入栈,栈的容量是 3,能够得到的出栈序列是
A. 1,2,6,4,3,5 B. 2,4,3,6,5,1
C. 3,1,2,5,4,6 D. 3,2,6,5,1,4
4. 设指针变量 head 指向非空单循环链表的头结点,指针变量 p 指向终端结点,next 是结点的指针域,则下列逻辑表达式中,值为真的是
A. $p \rightarrow next \rightarrow next == head$ B. $p \rightarrow next == head$
C. $p \rightarrow next \rightarrow next == NULL$ D. $p \rightarrow next == NULL$
5. 已知广义表 $LS = (((a,b)),((c,(d)),(e,(f))), (g,h))$,LS 的深度是
A. 2 B. 3 C. 4 D. 5
6. 已知一棵高度为 4 的完全二叉树 T 共有 5 个叶结点,则 T 中结点个数最少是
A. 9 B. 10 C. 11 D. 12

7. 在一棵非空二叉树的中序遍历序列中，所有列在根结点前面的是
- A. 左子树中的部分结点 B. 左子树中的全部结点
C. 右子树中的部分结点 D. 右子树中的全部结点
8. 用邻接矩阵表示有 n 个顶点和 e 条边的无向图，采用压缩方式存储，矩阵中零元素的个数是
- A. $n(n+1)/2-e$ B. $n(n+1)/2-2e$ C. $n \times n - e$ D. $n \times n - 2e$
9. 无向图 G 中所有顶点的度数之和是 20，则 G 中的边数是
- A. 10 B. 20 C. 30 D. 40
10. 设有向图 G 含有 n 个顶点、 e 条边，使用邻接表存储。对 G 进行广度优先遍历的算法的时间复杂度是
- A. $O(n)$ B. $O(e)$ C. $O(n+e)$ D. $O(n \times e)$
11. 对数据序列 (25, 15, 7, 18, 10, 0, 4) 采用直接插入排序进行升序排序，两趟排序后，得到的排序结果为
- A. 0, 4, 7, 18, 10, 25, 15 B. 0, 4, 25, 15, 7, 18, 10
C. 7, 15, 10, 0, 4, 18, 25 D. 7, 15, 25, 18, 10, 0, 4
12. 下列排序方法中，稳定的排序方法是
- A. 希尔排序 B. 归并排序 C. 堆排序 D. 快速排序
13. 一组记录的关键码为 (45, 68, 57, 13, 24, 89)，利用堆排序算法进行升序排序，建立的初始堆为
- A. 68, 45, 57, 13, 24, 89 B. 89, 68, 57, 13, 24, 45
C. 89, 68, 57, 45, 24, 13 D. 89, 57, 68, 24, 45, 13
14. 一棵二叉排序树中，关键字 n 所在结点是关键字 m 所在结点的祖先，则
- A. n 一定大于 m B. n 一定小于 m
C. n 一定等于 m D. n 与 m 的大小关系不确定
15. 设散列表长 $m=14$ ，散列函数 $H(\text{key})=\text{key}\%11$ 。表中已保存 4 个关键字： $\text{addr}(15)=4$ ， $\text{addr}(38)=5$ ， $\text{addr}(61)=6$ ， $\text{addr}(84)=7$ ，其余地址均为空。保存关键字 49 时存在冲突，采用线性探查法来处理。则查找关键字 49 时的探查次数是
- A. 1 B. 2 C. 4 D. 8

非选择题部分

注意事项:

用黑色字迹的签字笔或钢笔将答案写在答题纸上,不能答在试题卷上。

二、填空题:本大题共 10 小题,每小题 2 分,共 20 分。

16. 数据的逻辑结构是从逻辑关系上描述数据,它与数据元素的存储结构_____。
17. 指针 p 和指针 q 分别指向单链表 L 中的两个相邻结点,即 $q \rightarrow next = p$,且 p 所指结点不是终端结点。若要删除 p 所指结点,则执行的语句是_____。
18. 一个直接或间接调用自己的函数称为_____。
19. 广义表(a, (b, c, d), e, f, (g, h))的表尾是_____。
20. 二叉树的前序遍历序列和后序遍历序列中,叶结点之间的相对次序_____。
21. 如果图 G 存在拓扑排序序列,则 G 必为_____。
22. 将一棵树 T 转换为一棵二叉树 T1,在 T1 中结点 A 是结点 B 的父结点,则在 T 中,A 可能是 B 的父结点或_____。
23. 对含 n 个元素的数据序列采用快速排序算法进行排序,在最坏情况下的时间复杂度是_____。
24. 散列方法中,表示散列表装满程度的指标 α 称为_____。
25. 假设顺序存储的有序表 R 含有 12 个关键字,进行二分查找时,平均查找长度为_____。

三、解答题:本大题共 4 小题,每小题 5 分,共 20 分。

26. 设电文字符集是 $\{e_1, e_2, e_3, e_4, e_5\}$,它们出现的次数分别为: $\{50, 10, 16, 8, 12\}$ 。现要为该字符集设计哈夫曼编码。请回答下列问题。
 - (1) 画出得到的哈夫曼树。
 - (2) 给出各符号的哈夫曼编码。
27. 已知图 G 采用邻接矩阵存储,邻接矩阵如题 27 图所示。

	A	B	C	D	E	F	G
A	0	1	1	1	0	0	0
B	0	0	1	0	1	0	0
C	0	0	0	0	1	0	0
D	0	0	0	0	0	1	0
E	0	0	0	0	0	0	1
F	0	0	1	0	0	0	1
G	0	0	0	0	0	0	0

题 27 图

- (1) 写出从顶点 A 开始图 G 的 3 个不同的深度优先搜索遍历序列。
- (2) 写出从顶点 A 开始图 G 的 2 个不同的广度优先搜索遍历序列。

28. 有以下数据序列(19, 14, 23, 01, 68, 20, 84, 27, 55, 11, 10, 79, 12), 使用希尔排序方法将其排成升序序列。请回答下列问题。

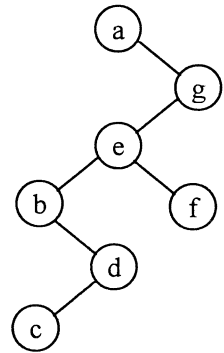
(1) 写出增量为 4 时对上述数据序列进行一趟希尔排序的结果。

(2) 给出一个可行的希尔排序增量序列。

29. 设有二叉排序树如题 29 图所示。请回答下列问题。

(1) 假定二叉排序树初始为空, 写出一个数据输入序列, 按序插入时能得到题 29 图所示的二叉排序树。

(2) 能得到题 29 图所示的二叉排序树的不同的输入数据序列有几个?



题 29 图

四、算法阅读题: 本大题共 4 小题, 每小题 5 分, 共 20 分。

30. 顺序表类型定义如下:

```
# define ListSize 100
typedef struct {
    int data[ListSize];
    int length;
```

```
} SeqList;
```

阅读下列算法, 并回答问题。

```
void change(SeqList *SL1, SeqList *SL2)
{
    int minlength;
    int k = 0, temp;
    if(SL1->length < SL2->length) return;
    minlength = SL2->length;
    while( k < minlength )
    {
        if ( SL1->data[k] < SL2->data[k] )
        {
            temp = SL1->data[k];
            SL1->data[k] = SL2->data[k];
            SL2->data[k] = temp;
        }
        k++;
    }
    return;
}
```

```

void f30( SeqList *SL1, SeqList *SL2 )
{
    if ( SL1->length > SL2->length ) change( SL1, SL2 );
    else change( SL2, SL1 );
    return;
}

```

(1) 若 SL1->data 中的数据为{25, 4, 256, 9, -38, 47, 128, 256, 64}, SL2->data 中的数据为{22, 4, -63, 15, 29, 34, 42, 3}, 则执行算法 f30(&SL1, &SL2)后 SL1->data 和 SL2->data 中的数据各是什么?

(2) 该算法的功能是什么?

31. 二叉树的存储结构类型定义如下:

```

typedef char DataType;
typedef struct node
{
    DataType data;           // data 是数据域
    struct node * lchild, * rchild; // 分别指向左右孩子
}BinTNode;
typedef BinTNode * BinTree;

```

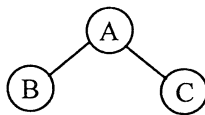
阅读下列算法, 并回答问题。

```

void A31 ( BinTree T )
{
    if (T != NULL)
    {
        printf( "%c ", T->data );
        A31( T->rchild );
        printf( "%c ", T->data );
        A31( T->lchild );
    }
    return;
}

```

(1) 设二叉树 T 如题 31 图所示, 给出执行 A31(T)的输出结果。



题 31 图

(2) 给出该算法的时间复杂度。

32. 待排序记录的数据类型定义如下:

```
#define MAXSIZE 100
typedef int KeyType;
typedef struct {
    KeyType key;
} RecType;
typedef RecType SeqList [MAXSIZE];
```

下列算法实现自底向上、自顶向下交替进行的双向扫描冒泡排序,请在空白处填上适当内容使算法完整。

```
void f32 ( SeqList R, int n)
{
    int i=0, j;
    RecType t;
    int NoSwap=1;
    while(NoSwap){
        NoSwap=0;
        for(j = n-i-1; _____ (1) _____)
            if(R[j].key < R[j-1].key) {
                t=R[j];
                R[j]=R[j-1];
                R[j-1]=t;
                NoSwap=1;
            }
        for( _____ (2) _____; j++)
            if(R[j].key>R[j+1].key){
                t=R[j];
                R[j]=R[j+1];
                R[j+1]=t;
                NoSwap=1;
            }
        _____ (3) _____;
    }
}
```

33. 二叉树的存储结构类型定义如下:

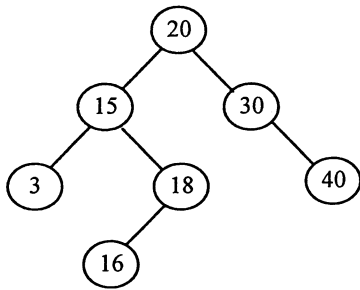
```
typedef int DataType;
typedef struct node
{
    DataType key;                // data 是数据域
    struct node * lchild, * rchild; // 分别指向左右孩子
}BinTNode;
typedef BinTNode * BinTree;
```

阅读下列算法, 并回答问题。

```
void A33(BinTree root, int k1, int k2, int end)
```

```
{
    if (root==NULL) return;
    A33(root->lchild, k1, k2, end);
    if (end) return;
    if (root->key > k2) {
        end = 1;
        return;
    }
    if (root->key >= k1) printf( "%d ",root->key);
    A33(root->rchild, k1, k2, end);
}
```

(1) 设二叉排序树 T 如题 33 图所示, bt 是指向根结点的指针。给出执行 A33(bt, 6, 100, 0)的输出结果。



题 33 图

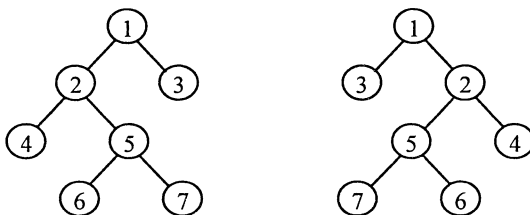
(2) 给出该算法的功能。

五、算法设计题：本题 10 分。

34. 已知二叉树的存储结构类型定义如下：

```
typedef struct node {  
    int data;  
    struct node *lchild, *rchild;  
} BinNode;  
typedef BinNode *BinTree;
```

编写递归算法，对于给定的一棵二叉树 T，将其修改为镜像二叉树。例如，题 34 图所示的两棵二叉树互为镜像二叉树。



题 34 图

函数的原型为：void f34 (BinTree T);