

绝密★启用前

2021 年 4 月高等教育自学考试全国统一命题考试

## 数据结构

(课程代码 02331)

注意事项:

1. 本试卷分为两部分, 第一部分为选择题, 第二部分为非选择题。
2. 应考者必须按试题顺序在答题卡(纸)指定位置上作答, 答在试卷上无效。
3. 涂写部分、画图部分必须使用 2B 铅笔, 书写部分必须使用黑色字迹签字笔。

### 第一部分 选择题

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, 不属于线性结构的是  
A. 线性表      B. 双向链表      C. 循环队列      D. 二叉树
2. 某线性表 L 含有 n 个元素, 采用单循环链表保存, 仅有尾指针指向链表的终端结点。在最后一个结点之后插入一个结点及删除第一个结点的时间复杂度分别是  
A.  $O(1)$  和  $O(1)$       B.  $O(1)$  和  $O(n)$       C.  $O(n)$  和  $O(1)$       D.  $O(n)$  和  $O(n)$
3. 下列应用中会用到栈的是  
A. 计算后缀表达式的值      B. 图的广度优先遍历  
C. 对数组进行希尔排序      D. 对散列表进行查找
4. 设栈初始为空, 入栈序列为 1, 2, 3, 4, 5, 下列选项中, 不可能得到的出栈序列是  
A. 1, 2, 3, 4, 5      B. 3, 1, 4, 2, 5  
C. 4, 3, 2, 5, 1      D. 5, 4, 3, 2, 1
5. 已知广义表  $LS = (((c, (d)), (e, (f))), (g, h), ((m, n)))$ ,  $head(LS)$  是  
A. c      B. (c)      C. (c, (d))      D. ((c, (d)), (e, (f)))
6. 设线性表采用顺序存储方式保存, 每个元素占 8 个存储单元。第 1 个元素的存储地址为 200, 则第 5 个元素占用的最后一个存储单元的地址是  
A. 239      B. 240      C. 247      D. 248

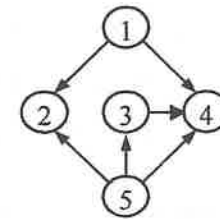
7. 一棵完全二叉树 T 的全部 k 个叶结点都在同一层中, 每个分支结点都有两个孩子结点。T 中包含的结点数是  
A. k      B.  $2k-1$       C.  $k^2$       D.  $2^k-1$

8. 设字符集中有 n 个字符, 对其进行哈夫曼编码, 得到的哈夫曼树的结点总数是  
A.  $2n-1$       B.  $2n$       C.  $2n+1$       D. 不确定

9. 设图 G 的邻接矩阵 A 如下所示。G 的各顶点的度依次是

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

- A. 1, 2, 1, 2      B. 2, 2, 1, 1      C. 3, 4, 2, 3      D. 4, 4, 2, 2
10. 对题 10-11 图进行深度优先遍历, 下列选项中, 正确的遍历序列是



题 10-11 图

- A. 1, 2, 3, 4, 5      B. 2, 3, 5, 4, 1      C. 3, 5, 1, 2, 4      D. 4, 3, 5, 1, 2
11. 对题 10-11 图进行拓扑排序, 下列选项中, 正确的拓扑序列是  
A. 1, 2, 3, 4, 5      B. 2, 3, 1, 4, 5      C. 3, 5, 1, 2, 4      D. 5, 3, 1, 2, 4
  12. 下列排序方法中, 不是稳定排序方法的是  
A. 直接插入排序      B. 冒泡排序      C. 归并排序      D. 快速排序
  13. 已知数据序列 (18, 19, 20, 4, 51, 6, 30, 1, 2) 是某种排序算法第二趟排序后得到的结果, 则该算法可能是  
A. 选择排序      B. 冒泡排序      C. 直接插入排序      D. 快速排序
  14. 对有序表 (1, 3, 9, 12, 32, 41, 45, 62, 75, 77) 进行二分查找, 查找关键字 9 时, 进行比较的关键字依次是  
A. 1, 3, 9      B. 32, 3, 9      C. 32, 12, 9      D. 41, 12, 9
  15. 分别使用下列数据序列建立二叉排序树, 能得到高度最高的二叉树的是  
A. 10, 8, 9, 6, 12, 11, 13      B. 10, 6, 8, 9, 12, 11, 13  
C. 10, 12, 11, 13, 8, 6, 9      D. 10, 8, 6, 9, 12, 13, 11

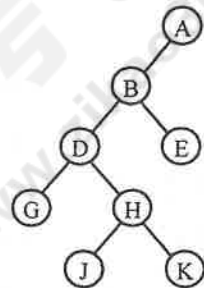
## 第二部分 非选择题

二、填空题：本大题共 10 小题，每小题 2 分，共 20 分。

16. 数据的运算，即对数据元素施加的操作，是定义在数据的\_\_\_\_\_结构上的。
17. 在顺序表中，因为访问任一结点的方式是\_\_\_\_\_，所以访问每个结点的时间复杂度均为  $O(1)$ 。
18. 带头结点的链队列可以由一个头指针和一个尾指针唯一确定。当头指针和尾指针相等时，表示队列\_\_\_\_\_。
19. 稀疏矩阵采用压缩存储，只保存非零元素，得到的顺序存储结构称为\_\_\_\_\_。
20. 广义表  $((a), (b, c), (d, e, (f, g, h)))$  的表尾是\_\_\_\_\_。
21. 中序线索化二叉树的过程，是在中序遍历过程中用线索取代\_\_\_\_\_。
22. 在有  $n$  个顶点、 $e$  条边的无向连通图中， $e$  的取值范围是\_\_\_\_\_。
23. 对数据序列进行升序排序。采用堆排序算法时，首先应对初始数据建立\_\_\_\_\_堆。
24. 在无序数组中进行查找操作，应使用的查找方法是\_\_\_\_\_。
25. 一棵高度为 2 的 4 阶 B 树中能够保存的关键字个数最多是\_\_\_\_\_。

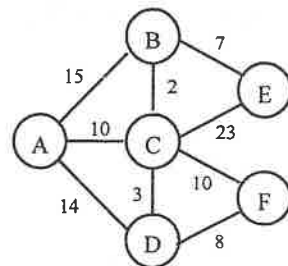
三、解答题：本大题共 4 小题，每小题 5 分，共 20 分。

26. 请画出题 26 图所示的二叉树对应的树或森林。



题 26 图

27. 求题 27 图从顶点 A 到其余各顶点的最短路径，给出各条路径包含的顶点序列及路径长度。



题 27 图

28. 有以下数据序列 (20, 84, 19, 14, 23, 01, 68, 27, 55, 11, 10, 79, 12)，使用二路归并排序算法将其排成升序序列。给出各趟排序结果。

29. 设有以下关键字：15, 72, 52, 65, 23, 68，散列函数  $H(\text{key}) = \text{key} \% 7$ ，散列表空间为  $0 \sim 6$ ，采用线性探查法解决冲突。请回答下列问题。

- (1) 构造散列表。
- (2) 计算等概率情况下查找成功时的平均查找长度。

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;
```

阅读下列函数，并回答问题。

```
SeqList * f30( int *pdata, int n)
{
    int k, m;
    SeqList *SL;
    SL = (SeqList *)malloc( sizeof(SeqList) );
    SL->length = 0;
    for ( k=0; k<n; k++)
    {
        if ( pdata[k]%2 != 0 )
            SL->data[ SL->length ] = pdata[ k ];
        else
        {
            for ( m=SL->length; m>0; m-- )
                SL->data[ m ] = SL->data[ m-1 ];
            SL->data[ 0 ] = pdata[k];
        }
        SL->length ++;
    }
}
```

```

for ( k=0; k < SL->length; k++ )
    printf("%d, ", SL->data[k]);
return SL;
}

```

(1) 若已知: `int array[] = {10, 2, 9, 5, 30, 3};`, 执行语句:

`f30( array, sizeof(array)/sizeof(int) );`

函数 `f30()` 的输出是什么?

(2) 函数 `f30()` 的功能是什么?

31. 阅读函数 `f31()`, 并回答问题。

`f31( int a[], int n )`

```

{   int  k, m = 0, temp;

```

```

    while ( a[m] < 0 && m < n ) // 说明本循环的作用
        m = m+1;

```

```

    k = m;

```

```

    while ( k < n )

```

```

    {   while ( a[k] >= 0 && k < n )

```

```

        k = k+1;

```

```

        if ( k < n )

```

```

        {   temp = a[k];

```

```

            a[k] = a[m];

```

```

            a[m] = temp;

```

```

            m = m+1;

```

```

        }

```

```

    }

```

```

}

```

(1) 若 `int a[] = {25, 4, 256, 9, -38, 47, 128, -256, 64};`, 则执行语句:

`f31(a, 9);`

数组 `a` 中的数据是什么?

(2) 说明方框内 `while` 语句的作用。

32. 二叉排序树的存储结构类型定义如下。

```

typedef int KeyType;

```

```

typedef struct node

```

```

{   KeyType data;           // data 是数据域

```

```

    struct node * lchild, * rchild; // 分别指向左右孩子

```

```

} BinNode;

```

```

typedef BinNode * BinTree;

```

阅读下列算法, 并回答问题。

```

void f32( BinTree t, KeyType K1, KeyType K2 )

```

```

{   if ( t != NULL )

```

```

    {   f32( t->lchild, K1, K2 );

```

```

        if ( t->data >= K1 && t->data <= K2 ) printf("%d ", t->data );

```

```

        f32( t->rchild, K1, K2 );

```

```

    }

```

```

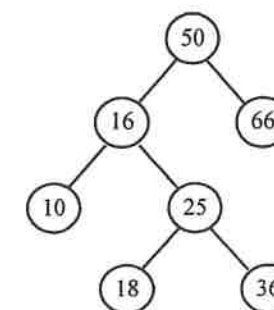
}

```

(1) 设二叉排序树 `T` 如题 32 图所示, 给出执行语句:

`f32(T, 14, 50);`

输出结果。



题 32 图

(2) 函数 `f32()` 的功能是什么?

33. 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 ];

```

函数 f33()对 R 采用直接选择排序进行升序排序，请在空白处填上适当内容使算法完整。

```
void f33(SeqList R, int n)
{
    int i, j, k;
    for ( i = 1; i <= n; i++ )
    {
        k = i;
        for ( j = i+1; (1); j++ )
            if ( (2) )
                k = j;
        if ( (3) )
        {
            R[0] = R[i];
            R[i] = R[k];
            R[k] = R[0];
        }
    }
}
```

五、算法设计题：本题 10 分。

34. 已知单链表的存储结构类型定义如下。

```
typedef int DataType;
typedef struct node{
    DataType data;
    struct node * next;
} ListNode;
typedef ListNode * LinkList;
ListNode *p;
LinkList head1;
```

双向链表的存储结构类型定义如下。

```
typedef struct dlnode{
    DataType data;
    struct dlnode *prior, * next;
} DLNode;
typedef DLNode * DLinkList;
DLinkList head2;
```

编写程序，对给定的单链表 head1，将其中的数据按原次序复制到双向链表 head2 中。函数原型为：void f34(head1, &head2);。