

全国 2019 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,不宜采用链式存储的是

- A. 无向图                      B. 单链表                      C. 最优二叉树                      D. 数组

2. 将 10 个数据元素保存在顺序栈 S 中,若栈顶元素的存储地址是 100,栈中每个元素占 4 个存储单元,进栈按  $S.top=S.top+1$  修改栈顶,则栈底元素的存储地址是

- A. 60                              B. 64                              C. 136                              D. 140

3. 设指针变量 head 指向循环链表的头结点, next 是结点的指针域,则判断此链表为空的条件是

- A.  $head \rightarrow next == NULL$                               B.  $head \rightarrow next == head$   
C.  $head \rightarrow next != NULL$                               D.  $head \rightarrow next != head \rightarrow next$

4. 已知广义表  $LS = (((a, b, c)), ((d, (e)), (f, (g))), (h, g), i)$ , LS 的深度是

- A. 4                              B. 3                              C. 2                              D. 1

5. 已知一棵完全二叉树 T 共有 7 个分支结点,则 T 中叶子结点个数最少是

- A. 7                              B. 8                              C. 9                              D. 10

6. 在一棵非空二叉树的后序遍历序列中,所有列在根结点前面的是

- A. 左子树中的部分结点                              B. 右子树中的全部结点  
C. 左右子树中的全部结点                              D. 左右子树中的部分结点

7. 用邻接表保存有  $n$  个顶点和  $e$  条边的无向图，邻接表中指针个数是
- A.  $e$                       B.  $n-e$                       C.  $n+e$                       D.  $n+2e$
8. 有向图  $G$  中某个顶点的出度和入度均为 2，则  $G$  中的顶点个数最少是
- A. 2                      B. 3                      C. 4                      D. 5
9. 在带权图的最短路径问题中，路径长度是指
- A. 路径上边的数目                      B. 路径上结点的数目  
C. 路径上边的权值之和                      D. 到达终点的最短路径数目
10. 对数据序列 (15, 10, 8, 12, 15, 8, 10) 按升序进行希尔排序，增量序列为 5, 3，两趟排序后，得到的排序结果为
- A. 8, 8, 10, 10, 15, 15, 12                      B. 8, 8, 10, 10, 12, 15, 15  
C. 8, 10, 8, 10, 15, 15, 12                      D. 8, 10, 8, 10, 12, 15, 15
11. 下列排序方法中，不稳定的排序方法是
- A. 直接选择排序    B. 归并排序                      C. 直接插入排序    D. 基数排序
12. 一组记录的关键字为 (35, 58, 24, 13, 44, 19, 10)，利用堆排序算法进行降序排序，要求空间复杂度为  $O(1)$ ，建立的初始堆为
- A. 10, 13, 19, 58, 44, 35, 24                      B. 10, 13, 35, 58, 44, 19, 24  
C. 58, 44, 24, 13, 35, 19, 10                      D. 58, 35, 24, 13, 44, 19, 10
13. 一棵二叉排序树中，关键字  $n$  所在结点的层数大于关键字  $m$  所在结点的层数，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
14. 设散列表长  $m=10$ ，散列函数  $H(\text{key})=\text{key}\%9$ 。表中已保存 3 个关键字： $H(13)=4$ ， $H(32)=5$ ， $H(15)=6$ ，其余地址均为空。保存关键字 23 时存在冲突，采用线性探查法来处理。则查找关键字 23 时的探查次数是
- A. 1                      B. 2                      C. 3                      D. 4
15. 下面关于  $m$  阶 ( $m\geq 3$ ) B 树的叙述中，正确的是
- A. 终端结点可位于不同层  
B. 非终端结点至多有  $m+1$  棵子树  
C. 若树非空，则根结点至少有 2 个关键字  
D. 每个非根结点包含  $n$  个关键字， $\lceil m/2 \rceil - 1 \leq n \leq m-1$

# 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. 数据的四种基本存储方法是顺序存储、链接存储、\_\_\_\_\_和散列存储。
17. 指针  $p$  和指针  $q$  分别指向单链表  $L$  中的两个结点， $next$  为指针域，则判断这两个结点是否相邻的条件是\_\_\_\_\_。
18. 递归求解过程中的最小子问题称为\_\_\_\_\_。
19. 广义表  $((a, b), (c, d, e)), (f, g), h)$  的表头是\_\_\_\_\_。
20. 3 个结点的不同形状的二叉树有\_\_\_\_\_棵。
21. 若有向无环图  $G$  存在 2 个入度为 0 的结点，则  $G$  至少存在\_\_\_\_\_个不同的拓扑序列。
22. 将一棵树  $T$  转换为一棵二叉树，则这棵二叉树的右子树\_\_\_\_\_。
23. 对含  $n$  个元素的数据序列采用直接选择排序算法进行排序，最好情况下的时间复杂度是\_\_\_\_\_。
24. 散列存储中，拉链法（链地址法）是处理\_\_\_\_\_的方法。
25. 假设顺序存储的有序表  $R$  含有 14 个关键字，进行二分查找时，查找失败时关键字的最大比较次数为\_\_\_\_\_。

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  $\{e_1, e_2, e_3, e_4, e_5, e_6\}$ ，它们出现的次数分别为：38, 12, 17, 26, 14, 20。现要为该字符集设计一种哈夫曼编码。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	0	0	0	0
$B$	0	0	1	1	0	0	0
$C$	0	0	0	1	0	0	0
$D$	0	0	0	0	1	0	0
$E$	0	0	0	0	0	1	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  $A$  开始到顶点  $C$  结束、包含所有顶点的 2 个深度优先遍历序列。

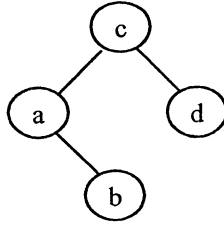
(2) 写出从顶点 A 开始的 3 个广度优先遍历序列。

28. 有以下关键字序列 (15, 20, 24, 32, 15, 7, 14, 23)，使用快速排序方法将其按升序排列。请回答下列问题。

(1) 若取第一个关键字为基准，写出第一趟快速排序的结果。

(2) 若取最后一个关键字为基准，写出第一趟快速排序的结果。

29. 设有二叉排序树 T 如题 29 图所示。请回答下列问题。



题 29 图

(1) 画出插入新结点 f 后的二叉排序树 T1。

(2) 在 T1 中再删除结点 c 得到二叉排序树 T2，画出 T2，并简要说明删除过程。

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。

```
void f30( SeqList *SL, int *pdata, int n )
{
    int k, m;
    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 ++;
    }
}
```

```

}
void out( SeqList *SList )
{   int k = 0;
    while ( k < SList->length )      // 顺序输出 SList 中的各个元素
        printf("%d, ", SList->data[ k++ ]);
}

int main()
{   int array[] = {10, 2, 9, 5, 30, 3};
    SeqList slist;
    slist.length = 0;
    f30( &slist, array, sizeof(array)/sizeof(int) );
    out( &slist );      // 输出 slist
    return 0;
}

```

(1) 执行程序后程序的输出是什么？

(2) 函数 f30()的功能是什么？

31. 二叉树的存储结构类型定义如下。

```

typedef int  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;

```

阅读下列函数并回答问题。

```

void f31( BinTree Bt )
{   if ( Bt != NULL )
    {   if ( Bt->lchild == NULL && Bt->rchild == NULL )
        Bt->data = Bt->data*2;
    }
}

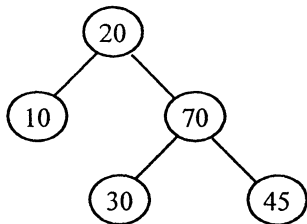
```

```

else
{   f31( Bt->lchild );
    f31( Bt->rchild );
}
}
}

```

(1) 设二叉树 Bt 如题 31 图所示，给出执行 f31(Bt)的输出结果。



题 31 图

(2) 该算法的功能是什么？

32. 待查找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 ];

```

下列算法实现对按升序排列的数据进行二分查找。请在空白处填上适当内容使算法完整。

```

int BinSearch( SeqList R, KeyType k, int n )
{   int low = 0, high = n-1, mid;
    while ( low <= high )
    {   mid = ( low+high )/2;
        if ( _____ (1) _____ ) return mid;
        else if ( R[mid].key > k ) high = _____ (2) _____;
        else low = _____ (3) _____;
    }
    return -1;
}

```

```
}
```

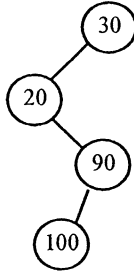
33. 二叉树的存储结构类型定义如下。

```
typedef int DataType;  
typedef struct node  
{   DataType data;           // data 是数据域，其值大于 0  
    struct node *lchild, *rchild; // 分别指向左右孩子  
} BinTNode;  
typedef BinTNode * BinTree;
```

下列程序的功能是：将一棵二叉树的顺序存储结构转换为对应的链式存储结构。

例如，对如题 33 图所示的二叉树，二叉树的顺序存储序列如下。

```
int data[ ] = { 30, 20, 0, 0, 90, 0, 0, 0, 0, 100 };
```



题 33 图

程序如下。

```
BinTree create( int *data, int n )  
{   BinTNode *Q[100], *Bt = NULL, *p;  
    int front = 0, rear = 0, k;  
    for ( k = 0; k < n; k++ )  
    {   p = NULL;  
        if ( data[k] != 0 )  
        {   p = (BinTree)malloc( sizeof(BinTNode) );  
            p->data = data[k];  
            p->lchild = p->rchild = NULL;  
        }  
        Q[ rear++ ] = p;  
        if ( rear == 1 ) Bt = p;
```

```

else
{
    if ( p != NULL && ____ (1) ____ )
        if ( ____ (2) ____ == 0 )
            Q[front]->lchild = p;
        else
            Q[front]->rchild = p;
    if ( rear%2 != 0 )
        ____ (3) ____;
}
}
return Bt;
}

```

请在程序的空白处填入适当的语句，使程序完整正确。

### 五、算法设计题：本题 10 分。

34. 单链表类型定义如下。

```

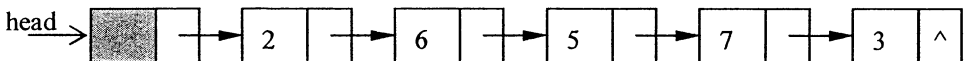
typedef struct node
{
    int data;
    struct node *next;
} ListNode;
typedef ListNode *LinkList;

```

请编写一个函数，在带头结点的单链表 L 中删除数值在指定范围内 ( $x \leq \text{data} \leq y$ ) 的结点。函数的原型如下。

```
void f34( LinkList L, int x, int y );
```

例如，对于如下的链表 head，



要删除链表中 data 在 4 到 7 范围内的结点，可调用函数 f34( head, 4, 7 )，结果如下。

