

全国 2015 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)

在每小题列出的四个备选项中只有一个是符合题目要求的,请将其选出并将“答题纸”的相应代码涂黑。错涂、多涂或未涂均无分。

1. 下列选项中,不属于线性结构的是

- A. 网                      B. 栈                      C. 队列                      D. 线性表

2. 长度为  $n$  的顺序表,删除位置  $i$  上的元素 ( $0 \leq i \leq n-1$ ),需要移动的元素个数为

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

3. 栈采用不同的存储方式时,下列关于出栈过程的叙述中,正确的是

- A. 顺序栈需要判定栈空,链栈也需要判定  
B. 顺序栈需要判定栈空,而链栈不需要判定  
C. 顺序栈不需要判定栈空,而链栈需要判定  
D. 顺序栈不需要判定栈空,链栈也不需要判定

4. 若一个栈以数组  $V[0..n-1]$  存储,初始栈顶指针  $top$  为  $n$ ,则  $x$  入栈的正确操作是

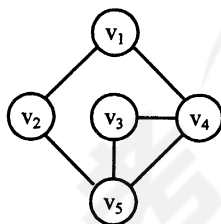
- A.  $top = top + 1; V[top] = x$                       B.  $V[top] = x; top = top + 1$   
C.  $top = top - 1; V[top] = x$                       D.  $V[top] = x; top = top - 1$

5. 在二维数组  $a[9][10]$  中,每个数组元素占用 3 个存储空间,从首地址  $SA$  开始按行优先连续存放,则元素  $a[8][5]$  的起始地址是

- A.  $SA + 141$                       B.  $SA + 144$                       C.  $SA + 222$                       D.  $SA + 255$

6. 广义表  $A = (x, ((y), ((a)), A))$  的深度是
- A. 2                      B. 3                      C. 4                      D.  $\infty$
7. 一棵左子树为空的二叉树在前序线索化后, 其空指针域个数为
- A. 0                      B. 1                      C. 2                      D. 不确定
8. 下列关于哈夫曼树的叙述中, 错误的是
- A. 用  $n$  个结点构造的哈夫曼树是唯一的
- B. 哈夫曼树中只有度为 0 或度为 2 的结点
- C. 树中两个权值最小的结点可能是兄弟结点
- D. 同一结点集构造的二叉树中, 哈夫曼树的 WPL 最小
9. 6 个顶点的强连通图中, 含有的边数至少是
- A. 4                      B. 5                      C. 6                      D. 7

10. 对题 10 图进行深度优先搜索遍历, 下列选项中, 正确的遍历序列是



题 10 图

- A.  $v_3 v_4 v_5 v_1 v_2$       B.  $v_3 v_5 v_2 v_1 v_4$       C.  $v_4 v_5 v_2 v_3 v_1$       D.  $v_5 v_1 v_2 v_4 v_3$
11. 下列选项中, 能构成题 10 图中一条路径的是
- A.  $v_1 v_2 v_4 v_5 v_3$       B.  $v_1 v_2 v_5 v_3 v_4$       C.  $v_2 v_5 v_1 v_3 v_4$       D.  $v_2 v_1 v_5 v_4 v_3$
12. 有向图采用邻接矩阵存储, 某一行中非零元素的个数等于
- A. 对应顶点  $v$  的度                      B. 对应顶点  $v$  的出度
- C. 对应顶点  $v$  的入度                      D. 依附于对应顶点  $v$  的边数
13. 以下选项中, 符合堆定义的是
- A. { 102, 24, 55, 60, 89, 93 }                      B. { 24, 89, 55, 60, 93, 102 }
- C. { 102, 93, 55, 60, 89, 24 }                      D. { 102, 60, 89, 93, 55, 24 }

14. 已知关键字序列为{ 66, 82, 25, 51, 98, 108 }, 利用快速排序方法, 以第一个元素为基准得到的一趟排序结果为
- A. { 25, 51, 66, 82, 98, 108 }                      B. { 25, 51, 66, 98, 82, 108 }
- C. { 51, 25, 66, 108, 98, 82 }                      D. { 51, 25, 66, 82, 98, 108 }
15. 下列选项中, 其平均查找性能与基于二叉排序树的查找相当的是
- A. 二分查找              B. 顺序查找              C. 分块查找              D. 索引顺序查找

## 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。

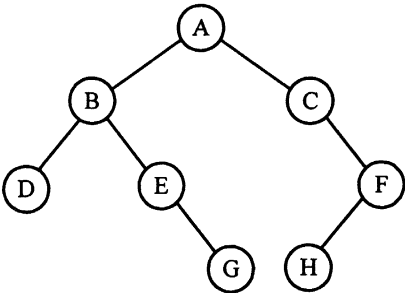
### 二、填空题 (本大题共 10 小题, 每小题 2 分, 共 20 分)

16. 线性表  $(a_1, a_2, \dots, a_n)$  中, 除\_\_\_\_\_外, 每个元素都有唯一的直接前趋。
17. 指针  $p$  指向单链表中某个结点, 在  $p$  所指结点后插入指针  $s$  所指的结点, 正确的操作序列是\_\_\_\_\_。
18. 设 Push、Pop 分别表示入栈和出栈操作,  $x = 10, y = 20, z = 30$ 。依次进行下列操作: Push( $y$ )、Push( $z$ )、Push( $z$ )、 $x = \text{Pop}()$ 、 $y = \text{Pop}()$ ,  $x$ 、 $y$  的值分别是\_\_\_\_\_。
19. 广义表  $L = (a, (b, c, (e, f, g, h)))$ ,  $\text{head}(L) =$ \_\_\_\_\_。
20. 设树  $T$  的度为 3, 其中度为 1、2 和 3 的结点个数分别为 3、2 和 1, 则  $T$  中叶子结点的个数为\_\_\_\_\_。
21. 由一棵二叉树的后序遍历序列和\_\_\_\_\_遍历序列可以唯一确定该二叉树。
22. 在有  $n$  个顶点的无向图中, 任一顶点的度不大于\_\_\_\_\_。
23. 借助于一个栈来实现的图的遍历算法是\_\_\_\_\_。
24. 若有向图中存在拓扑排序序列, 则该图一定不存在\_\_\_\_\_。
25. 已知关键字序列为{ 66, 82, 25, 51, 98, 108 }, 一趟二路归并排序的结果为\_\_\_\_\_。

### 三、解答题 (本大题共 4 小题, 每小题 5 分, 共 20 分)

26. 已知  $n$  阶对称矩阵  $A$  的元素为  $a_{ij}$  ( $0 \leq i, j \leq n-1$ ), 采用“按行优先”将下三角部分的元素 (含主对角线) 保存在一维数组  $sa$  中, 且约定元素  $a_{0,0}$  保存在  $sa[0]$  中, 元素  $a_{ij}$  ( $0 \leq i, j \leq n-1$ ) 保存在  $sa[k]$  中, 请给出由下标  $i, j$  计算下标  $k$  的计算公式。

27. 已知二叉树 T 如题 27 图所示。

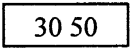


题 27 图

请问答下列问题：

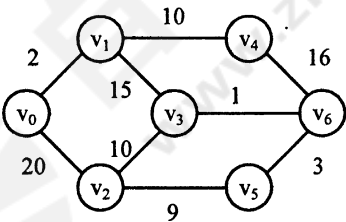
- (1) 画出该二叉树对应的森林。
- (2) 写出对森林进行前序遍历的遍历序列。

28. 题 28 图所示为一棵含 2 个关键字的 3 阶 B 树 T。现将关键字序列 {40, 60, 70, 20, 10} 依次插入到 T 中，画出每插入一个关键字后得到的树型。



题 28 图

29. 给定无向带权连通图 G 如题 29 图所示，从顶点  $v_0$  开始，使用普里姆 (Prim) 算法，求 G 的最小生成树 T。请回答下列问题。



题 29 图

- (1) 画出最小生成树 T。
- (2) 计算 T 中各边权值之和。

四、算法阅读题（本大题共 4 小题，每小题 5 分，共 20 分）

30. 请写出下列程序段的输出结果。

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

void f30( SeqList * list )
{   int i, j, k;
    for ( i=0; i<=list->length - 2; i++ )
    {   j = i+1;
        while ( j <= list->length - 1 )
        {   if ( list->data[i] == list->data[j] )
            {   for ( k=j; k < list->length-1; k++ )
                    list->data[k] = list->data[k+1];
                list->length--;
            }
            else
                j++;
        }
    }
}

void main()
{   SeqList list = { {0, 3, 7, 3, 3, 3, 4, 0, 3, 7 }, 10 };
    int i;
    f30( &list );
    printf( "len=%d\n", list.length );
    for ( i=0; i<list.length; i++ )
        printf("%d, ", list.data[i] );
    printf("\n");
}
```

运行结果：

31. 已知存储稀疏矩阵三元组表的类型定义如下:

```
#define MAX 100
```

```
typedef struct {
```

```
    int i, j;           // 非零元素的行号、列号 (下标)
```

```
    int v;              // 非零元素值
```

```
} TriTupleNode;
```

```
typedef struct {
```

```
    TriTupleNode data[ MAX ];    // 存储三元组的数组
```

```
    int m, n, t;                  // 矩阵的行数、列数和非零元素个数
```

```
} TSMatrix;                      // 稀疏矩阵类型
```

函数 f31 的功能是将 a 所表示的矩阵转置后保存在 \*b 中。请在答题纸相应位置填写适当内容, 使其完成指定功能。

```
int f31(TSMatrix a, TSMatrix * b) // 返回值: 1 表示出错, 0 表示正确
```

```
{ // a 和 *b 分别是矩阵 M、T 的三元组表, T 为稀疏矩阵 M 的转置
```

```
    int p, q, col;
```

```
    b->m = a.n;   b->n = a.m;   b->t = a.t;
```

```
    if ( b->t < 0 )
```

```
        return 1;
```

```
    else {
```

```
        q = 0;
```

```
        for ( col = 0; col < a.n; ++col )
```

```
            for ( p = 0; p < ____ (1) ____ ; ++p )
```

```
                if ( ____ (2) ____ == col ) {
```

```
                    b->data[q].i = ____ (3) ____ ;
```

```
                    b->data[q].j = ____ (4) ____ ;
```

```
                    b->data[q].v = a.data[p].v;
```

```
                    ++q;
```

```
                }
```

```
    }
```

```
    return 0;
```

```
}
```

32. 已知二叉树的二叉链表类型定义如下：

```
typedef struct node {
    char data;
    struct node * lchild, * rchild;
} BinTNode;
typedef BinTNode * BinTree;
```

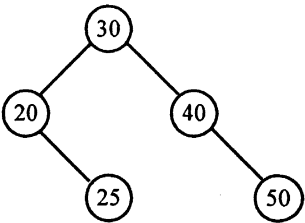
函数 CopyBin 的功能是完成二叉树 Bt 的复制，程序如下：

```
BinTree CopyBin( BinTree Bt )      // 函数返回值为指向复制后的二叉树根的指针
{
    BinTree p;
    if ( Bt == NULL )
        (1) ;
    else {
        p = (BinTNode *) malloc(sizeof(BinTNode));
        p->data = Bt->data;
        p->lchild = (2) ;
        p->rchild = (3) ;
    }
    return p;
}
```

为完成指定功能，请在答题纸相应位置填写适当内容，使其功能完整。

33. 函数 f33 的参数 t 指向题 33 图所示的二叉排序树的根，阅读程序，回答下列问题。

```
typedef int KeyType;
typedef struct node{
    KeyType key;
    node *lchild, *rchild;
} BSTNode, *BSTree;
```



题 33 图

```
BSTree f33( BSTree t, KeyType K )
{
    BSTree p;
    while ( t != NULL )
    {
        if ( t->key == K )
```

```

    {   printf ( "查找成功\n" );
        return  t;
    }
    p = t;
    if ( t->key > K )  t = t->lchild;
    else  t = t->rchild;
}
printf ( "查找不成功\n" );
t = (BSTree) malloc ( sizeof( BSTNode ) );
t->key = K;
t->lchild = NULL;
t->rchild = NULL;
if ( p->key > K ) p->lchild = t;
else  p->rchild = t;
return  NULL;
}

```

(1) 若连续 3 次调用函数 f33, 参数 K 的值依次取 10、25、10, 写出每次调用后函数的输出结果;

(2) 说明函数 f33 的功能。

## 五、算法设计题 (本题 10 分)

34. 已知顺序表 SeqList 定义如下:

```

typedef struct {
    KeyType  key;
    InfoType otherinfo;
} RecType;
typedef RecType  SeqList[ MAXSIZE + 1 ];

```

编写函数, 用冒泡排序法将 n 个元素的待排序列 R 按关键字降序排序。函数原型为:

int f34( SeqList R, int n )。