

# 2012 年 10 月高等教育自学考试全国统一命题考试

## 数据结构导论试题

课程代码：02142

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。错选、多选或未选均无分。

1. 下面几种算法时间复杂度阶数中，值最大的是

- A.  $O(n \log_2 n)$
- B.  $O(n^2)$
- C.  $O(n)$
- D.  $O(2^n)$

2. 即使输入非法数据，算法也能适当地做出反应或进行处理，不会产生预料不到的运行结果，这种算法好坏的评价因素称为

- A. 正确性
- B. 易读性
- C. 健壮性
- D. 时空性

3. 设顺序表的长度为 100，则在第 40 个元素之后插入一个元素所需移动元素的个数为

- A. 40
- B. 60
- C. 61
- D. 100

4. 设带头结点的单循环链表的头指针为 head，则判断该链表是否为空的条件是

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

5. 在链栈的运算中，不需要判断栈是否为空的是

- A. 出栈
- B. 进栈
- C. 取栈顶元素
- D. 求链栈的元素个数

6. 一个队列的输入序列是 A, B, C, D，则该队列的输出序列是

- A. A, B, C, D
- B. B, C, D, A
- C. D, C, B, A
- D. C, D, B, A

7. 以行序为主序的二维数组  $a[3][5]$  中，第一个元素  $a[0][0]$  的存储地址是 100，每个元素占 2 个存储单元，则  $a[1][2]$  的存储地址是

- A. 100
- B. 108
- C. 114
- D. 116

8. 对任何一棵二叉树 T，若叶结点数为 5 个，则度为 2 的结点个数为

- A. 4
- B. 5
- C. 6
- D. 无法确定

9. m 个叶结点的哈夫曼树中，其结点总数为

- A.m  
C.2m
- B.2m+1  
D.2m-1
- 10.二叉树的中序遍历序列中, 结点 P 排在结点 Q 之前的条件是  
A.在二叉树中 P 在 Q 的左边  
B.在二叉树中 P 在 Q 的右边  
C.在二叉树中 P 是 Q 的祖先  
D.在二叉树中 P 是 Q 的子孙
- 11.有 10 个顶点的无向完全图的边数是  
A.11  
B.45  
C.55  
D.90
- 12.在带权有向图中求两个结点之间的最短路径可以采用的算法是  
A.迪杰斯特拉 (Dijkstra) 算法  
B.克鲁斯卡尔 (Kruskal) 算法  
C.普里姆 (Prim) 算法  
D.深度优先搜索 (DFS) 算法
- 13.二分查找 (Binary Search) 算法的时间复杂度是  
A.O (n<sup>2</sup>)  
B.O (nlog<sub>2</sub>n)  
C.O (n)  
D.O (log<sub>2</sub>n)
- 14.在一棵初始时空的二叉树中, 依次插入键值序列 50, 72, 43, 85, 75, 20, 38, 45, 65, 60, 构造对应的二叉排序树以后, 查找元素 60 要进行的比较次数是  
A.2  
B.3  
C.4  
D.5
- 15.快速排序属于  
A.插入排序  
B.交换排序  
C.选择排序  
D.归并排序

## 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 13 小题, 每小题 2 分, 共 26 分)

- 16.下面算法程序段的时间复杂度为\_\_\_\_\_。
- ```
for (i=1; i<=n; i++)
    for (j=1; j<=n; j++)
        for (k=1; k<=n; k++)
            x++;
```
- 17.所有存储结点存放在一个连续的存储区里, 利用结点在存储器中的相对位置来表示数据元素之间的逻辑关系。这种存储方式是\_\_\_\_\_。
- 18.单链表中指针 p 指向结点 A, 若要删除 A 之后的结点 (存在且不释放存储空间), 则需要修改指针的操作为 p->next=\_\_\_\_\_。
- 19.在带有头结点的单链表 head 中, 首结点的指针为\_\_\_\_\_。
- 20.在栈结构中, 允许插入和删除的一端称为\_\_\_\_\_。
- 21.C 程序中, 将对称矩阵 A [n] [n] 的下三角元素压缩存储到 n(n+1)/2 个元素的一维数组 M 中, 设 a [i] [j] (i≥j) 存放在数组 M [k] 中, 则 k 的值 (用 i,j 表示) 为\_\_\_\_\_。
- 22.具有 64 个结点的完全二叉树的深度为\_\_\_\_\_。
- 23.某二叉树的先序遍历序列为 AJKLMNO, 中序遍历序列为 JLKANMO, 则根结点 A 的右子树中的结点个数为\_\_\_\_\_。

24.三个顶点  $v_1, v_2, v_3$  的图的邻接矩阵为  $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ , 则该图中顶点  $v_2$  的出度为\_\_\_\_\_。

25.除第一个顶点和最后一个顶点相同外, 其余顶点不重复的回路, 称为\_\_\_\_\_。

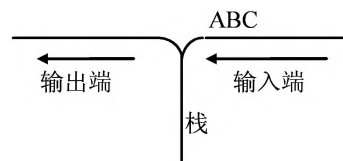
26.在顺序查找、二分查找、散列查找和索引顺序查找四种查找方法中, 平均查找长度与元素个数没有关系的查找方法是\_\_\_\_\_。

27.堆排序算法的时间复杂度为\_\_\_\_\_。

28.如果要将序列 {60, 18, 28, 69, 99, 75, 78} 建成堆, 则只需把 60 与\_\_\_\_\_相互交换。

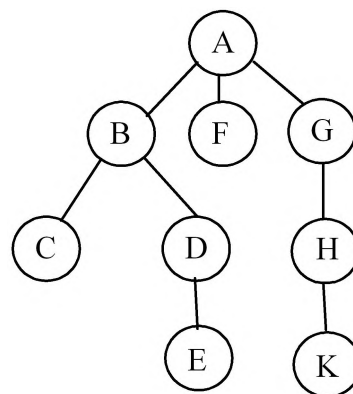
### 三、应用题(本大题共 5 小题, 每小题 6 分, 共 30 分)

29.如题 29 图所示, 在栈的输入端依次输入元素 A, B, C, 试写出在栈的输出端可以得到的所有输出序列, 并给出每个序列的操作过程 (用  $\text{push}(A)$  表示 A 进栈,  $\text{pop}(A)$  表示 A 出栈)。



题 29 图

30.将题 30 图所示的一棵树转换为对应的二叉树。



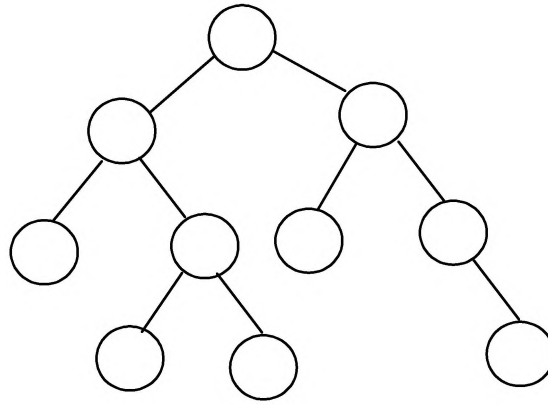
题 30 图

31.已知含五个顶点 A, B, C, D, E 的连通带权图的邻接矩阵如题 31 图所示, 试画出它所表示的连通带权图及该连通带权图的最小生成树。

$$\begin{bmatrix} \infty & 5 & \infty & 4 & 2 \\ 5 & \infty & 1 & \infty & \infty \\ \infty & 1 & \infty & 2 & 3 \\ 4 & \infty & 2 & \infty & \infty \\ 2 & \infty & 3 & \infty & \infty \end{bmatrix}$$

题 31 图

32.题 32 图所示二叉排序树的各结点的值为 1~10 中的数, 试标出各结点的数值。



题 32 图

33. 设散列函数  $H(\text{key}) = \text{key} \bmod 11$  ( $\bmod$  表示求余运算), 给出键值序列为 66,13,41,15,44,6,68,17,26,31,39,46,用链地址法解决冲突, 试画出相应的散列表, 并计算在等概率情况下查找成功时的平均查找长度。

四、算法设计题(本大题共 2 小题, 每小题 7 分, 共 14 分)

34. 带头结点的单链表的结点结构如下:

```

typedef struct node
{
    int data;
    struct node *next;
} Node, *LinkList;
  
```

试编写单链表的删除运算算法 `void DeleteLinklist( LinkList head, int i)`

35. 写出直接选择排序算法。

2012年10月高等教育自学考试全国统一命题考试

数据结构导论试题答案及评分参考

(课程代码 02142)

一、单项选择题(本大题共15小题,每小题2分,共30分)

- |       |       |       |       |       |
|-------|-------|-------|-------|-------|
| 1. D  | 2. C  | 3. B  | 4. A  | 5. B  |
| 6. A  | 7. C  | 8. A  | 9. D  | 10. A |
| 11. B | 12. A | 13. D | 14. C | 15. B |

二、填空题(本大题共13小题,每小题2分,共26分)

- |                             |            |                                           |
|-----------------------------|------------|-------------------------------------------|
| 16. $O(n')$                 | 17. 顺序存储方式 | 18. $p \rightarrow next \rightarrow next$ |
| 19. $head \rightarrow next$ | 20. 栈顶     | 21. $(i+1)i/2+j$                          |
| 22. 7                       | 23. 3      | 24. 2                                     |
| 25. 简单回路或简单环                | 26. 散列查找   | 27. $O(n \log_2 n)$                       |
| 28. 18                      |            |                                           |

三、应用题(本大题共5小题,每小题6分,共30分)

29. 解:共有5种:

ABC push(A); pop(A); push(B); pop(B); push(C); pop(C) (1分)

ACB push(A); pop(A); push(B); push(C); pop(C); pop(B) (1分)

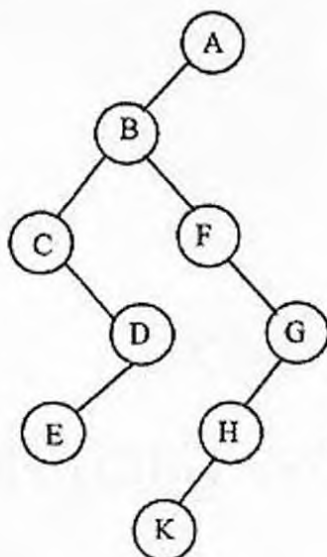
BAC push(A); push(B); pop(B); pop(A); push(C); pop(C) (1分)

BCA push(A); push(B); pop(B); push(C); pop(C); pop(A) (1分)

CBA push(A); push(B); push(C); pop(C); pop(B); pop(A) (2分)

没有 CAB

30. 解:对应的二叉树为

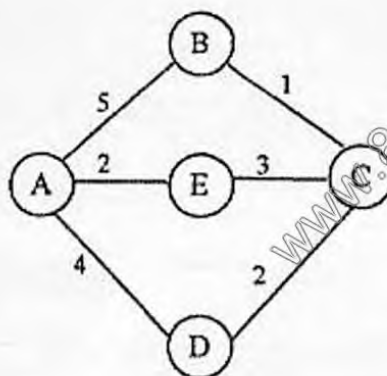


答30图

(6分)

31. 解:

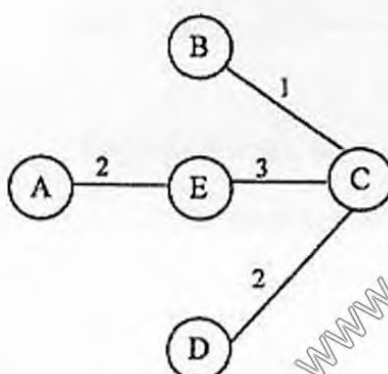
连通带权图为:



答 31 图 (a)

(3 分)

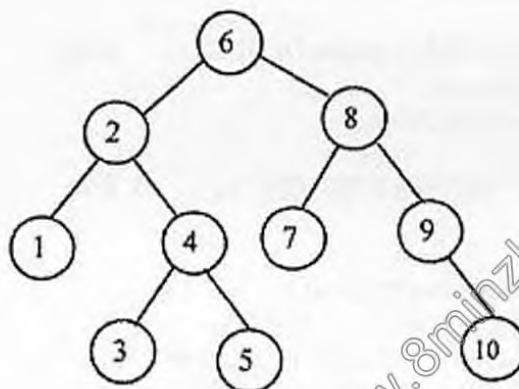
最小生成树为:



答 31 图 (b)

(3 分)

32. 解:

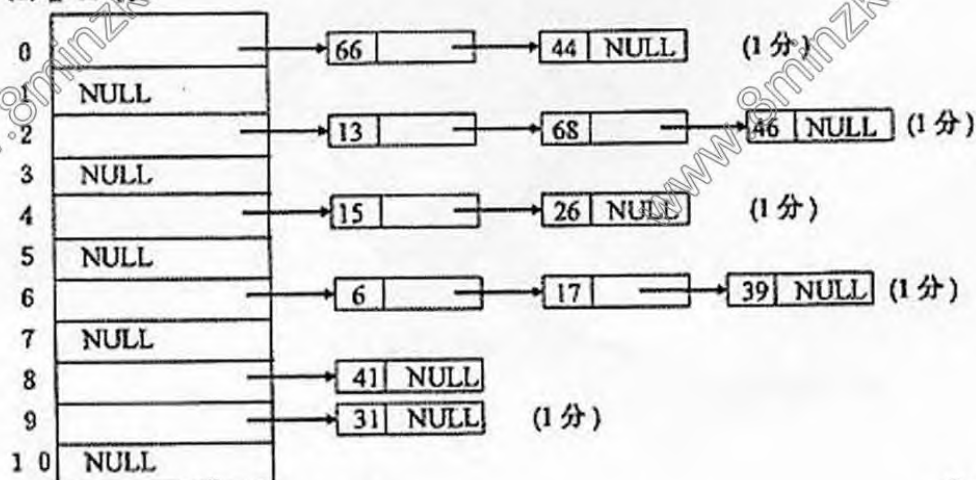


答 32 图

(注: 左边 3 分, 右边 3 分。)

33. 解:

(1) 答 33 表



(2) 平均查找长度为 5/3。 (1 分)

四、算法设计题 (本大题共 2 小题, 每小题 7 分, 共 14 分)

34. 解:

```
void DeleteLinklist( LinkList head, int i)
{ Node * q;
  if (i == 1) q = head;      (1 分)
  else {
    q = head->next;          (1 分)
    int c = 1;
    while ((c < i-1) && (q != NULL))
      { q = q->next; c++; }    (1 分)
  }
  if (q != NULL && q->next != NULL)    (1 分)
    { p = q->next;      (1 分)
      q->next = p->next;
      free(p);          (1 分)
    }
  else exit("找不到要删除的结点");    (1 分)
}
```

35. 解:

```
void SelectSort(List R, int n)
{ int min, i, j; List t;      (1 分)
  for (i = 1; i <= n-1; i++)    (1 分)
    { min = i;                  (1 分)
      for (j = i+1; j <= n; j++)    (1 分)
        if (R[j].key < R[min].key) min = j;    (1 分)
        if (min != i)              (1 分)
          { t = R[min]; R[min] = R[i]; R[i] = t; }    (1 分)
    }
}
```