

# 昆明理工大学 2009 年硕士研究生招生入学考试试题(A 卷)

考试科目代码：801

考试科目名称：数据结构

试题适用招生专业：地图学与地理信息系统

## 考生答题须知

1. 所有题目（包括填空、选择、图表等类型题目）答题答案必须做在考点发给的答题纸上，做在本试题册上无效。请考生务必在答题纸上写清题号。
2. 评卷时不评阅本试题册，答题如有做在本试题册上而影响成绩的，后果由考生自己负责。
3. 答题时一律使用蓝、黑色墨水笔或圆珠笔作答（画图可用铅笔），用其它笔答题不给分。
4. 答题时不准使用涂改液等具有明显标记的涂改用品。

一、单项选择题（在每小题的四个备选答案中，选出一个正确答案，并将正确答案的序号填在答题纸上。每小题 3 分，共 45 分）

1. 下面几个符号串编码集合中，不是前缀编码的是（ ）。

- A. {0, 10, 110, 111}                      B. {11, 10, 001, 101, 0001}  
C. {00, 010, 0110, 1000}                D. {b,c,aa,ac,aba,abb,abc}

2. 在长度为  $n$  的顺序表中删除第  $i$  个元素( $1 \leq i \leq n$ )时，元素移动的次数为( )

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

3. 若不带头结点的单链表的头指针为  $head$ ，则该链表为空的判定条件是( )

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

4. 引起循环队列队头位置发生变化的操作是( )

- A. 出队                                         B. 入队  
C. 取队头元素                                D. 取队尾元素

5. 若进栈序列为 1, 2, 3, 4, 5, 6, 且进栈和出栈可以穿插进行，则不可能出现的出栈序列是( )

- A. 2, 4, 3, 1, 5, 6                            B. 3, 2, 4, 1, 6, 5  
C. 4, 3, 2, 1, 5, 6                            D. 2, 3, 5, 1, 6, 4

6. 字符串通常采用的两种存储方式是( )

- A. 散列存储和索引存储                    B. 索引存储和链式存储  
C. 顺序存储和链式存储                    D. 散列存储和顺序存储

7. 设主串长为  $n$ ，模式串长为  $m(m \leq n)$ ，则在匹配失败情况下，朴素匹配算法进行的无效位移次数为( )

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

8. 二维数组  $A[12][18]$  采用列优先的存储方法，若每个元素各占 3 个存储单元，且第 1 个元素的地址为 150，则元素  $A[9][7]$  的地址为( )

昆明理工大学 2009 年硕士研究生招生入学考试试题

A. 429 B. 432

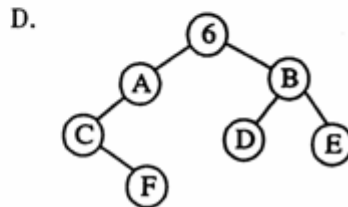
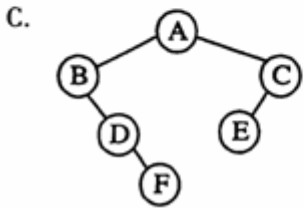
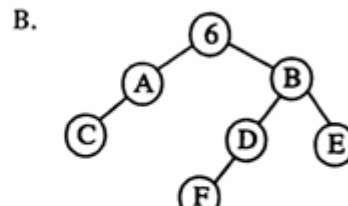
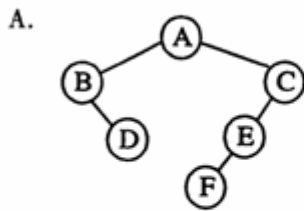
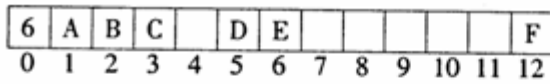
C. 435 D. 438

9. 对广义表  $L=((a,b),(c,d),(e,f))$  执行操作  $\text{tail}(\text{tail}(L))$  的结果是( )

A. (e,f) B. ((e,f))

C. (f) D. ()

10. 下列图示的顺序存储结构表示的二叉树是( )



11.  $n$  个顶点的强连通图中至少含有( )

A.  $n-1$  条有向边 B.  $n$  条有向边

C.  $n(n-1)/2$  条有向边 D.  $n(n-1)$  条有向边

12. 对关键字序列(56, 23, 78, 92, 88, 67, 19, 34)进行增量为 3 的一趟希尔排序的结果为 ( )

A. (19, 23, 56, 34, 78, 67, 88, 92) B. 23, 56, 78, 66, 88, 92, 19, 34)

C. (19, 23, 34, 56, 67, 78, 88, 92) D. (19, 23, 67, 56, 34, 78, 92, 88)

13. 设树  $T$  的度为 4, 其中度为 1, 2, 3 和 4 的结点数分别为 4, 2, 1, 1, 则  $T$  中的叶子数为( )

A. 5 B. 6 C. 7 D. 8

14. 由同一关键字集合构造的各棵二叉排序树( )

A. 其形态不一定相同, 但平均查找长度相同

B. 其形态不一定相同, 平均查找长度也不一定相同

C. 其形态均相同, 但平均查找长度不一定相同

D. 其形态均相同, 平均查找长度也都相同

15. 某二叉树中序序列为 ABCDEFG, 后序序列为 BDCAFGE, 则前序序列是( )

A. EGFACDB B. EACBDGF C. EAGCFBD D. 以上都不对

二、填空题 (每小题 3 分, 共 30 分)

16. 数据的逻辑结构在计算机存储器内的表示, 称为数据的\_\_\_\_\_。

## 昆明理工大学 2009 年硕士研究生招生入学考试试题

17. 删除双向循环链表中 p 的前驱结点(存在)应执行的语句是\_\_\_\_\_。
18. 栈下溢是指在\_\_\_\_\_时进行出栈操作。
19. 已知 substr(s, i, len) 函数的功能是返回串 s 中第 i 个字符开始长度为 len 的子串, strlen(s) 函数的功能是返回串 s 的长度。若 s=" ABCDEFGHIJK", t=" ABCD", 执行运算 substr(s, strlen(t), strlen(t)) 后的返回值为\_\_\_\_\_。
20. 去除广义表 LS=(a1, a2, a3, …, an) 中第 1 个元素, 由其余元素构成的广义表称为 LS 的\_\_\_\_\_。
21. 已知完全二叉树 T 的第 5 层只有 7 个结点, 则该树共有\_\_\_\_\_个叶子结点。
22. 在有向图中, 以顶点 v 为终点的边的数目称为 v 的\_\_\_\_\_。
23. G 是一个非连通无向图, 共有 28 条边, 则该图至少有\_\_\_\_\_个顶点。
24. 具有 n 个结点的二叉树, 采用二叉链表存储, 共有\_\_\_\_\_个空链域。
25. 对 n 个记录的表 r[1..n] 进行简单选择排序, 所需进行的关键字间的比较次数为\_\_\_\_\_。

### 三、解答题 ( 每小题 10 分, 共 20 分 )

26. 假设以数组 seqn [m] 存放循环队列的元素, 设变量 rear 和 quelen 分别指示循环队列中队尾元素的位置和元素的个数。

- (1) 写出队满的条件表达式;
- (2) 写出队空的条件表达式;
- (3) 设 m=40, rear=13, quelen=19, 求队头元素的位置;
- (4) 写出一般情况下队头元素位置的表达式。
  - (1)
  - (2)
  - (3)
  - (4)

27. 对二叉排序树的查找都是从根结点开始的, 查找失败时是否一定落在叶子上? 为什么?

### 四、算法阅读题 ( 每小题 10 分, 共 40 分 )

28. 阅读下列算法, 并回答问题:

- (1) 设顺序表 L=(3,7,11,14,20,51), 写出执行 f30(&L,15) 之后的 L;
- (2) 设顺序表 L=(4,7,10,14,20,51), 写出执行 f30(&L,10) 之后的 L;
- (3) 简述算法的功能。

```
void f30(SeqList*L, DataType x)
```

```
{   int i=0, j;  
    while (i<L->length && x>L->data [i] )i++;  
    if(i<L->length && x==L->data [i] ) {
```

昆明理工大学 2009 年硕士研究生招生入学考试试题

```
for(j=i+1;j<L->length;j++)
    L->data [j-1] =L->data [j] ;
L->length--;
}
else
{
    for(j=L->length;j>i;j--)
        L->data [j] =L->data [j-1] ;
    L->data [i] =x;
    L->length++;
}
}
(1)
(2)
(3)
```

29. 已知图的邻接表表示的形式说明如下:

```
#define MaxNum 50 //图的最大顶点数
typedef struct node {
    int adjvex; //邻接点域
    struct node *next; //链指针域
} EdgeNode; //边表结点结构描述
typedef struct {
    char vertex; //顶点域
    EdgeNode *firstedge; //边表头指针
} VertexNode; //顶点表结点结构描述
typedef struct {
    VertexNode adjlist [MaxNum] ; //邻接表
    int n, e; //图中当前的顶点数和边数
} ALGraph; //邻接表结构描述
```

下列算法输出图 G 的深度优先生成树(或森林)的边。阅读算法,并在空缺处填入合适的内容,使其成为一个完整的算法。

```
typedef enum {FALSE, TRUE} Boolean;
Boolean visited [MaxNum] ;
void DFSForest(ALGraph *G){
    int i;
```

昆明理工大学 2009 年硕士研究生招生入学考试试题

```

for(i=0;i<G->n;i++)
visited [i] = _____ (1) _____;
for(i=0;i<G->n;i++) if (!visited [i] ) DFSTree(G,i);
}
void DFSTree(ALGraph *G, int i) {
    EdgeNode *p;
    visited [i] =TRUE;
    p=G->adjlist [i] . firstedge;
    while(p!=NULL){
        if(!visited [p->adjvex] ){
            printf("<%c,%c>",G->adjlist [i] . vertex,G->adjlist [p->adjvex] .vertex);
            _____ (2) _____;
        }
        _____ (3) _____;
    }
}

```

(1)

(2)

(3)

30. 阅读下列算法，并回答问题：

(1)假设数组 L [8] ={3,0,5,1,6,4,2,7}，写出执行函数调用 f32(L, 8)后的 L;

(2)写出上述函数调用过程中进行元素交换操作的总次数。

```

void f32(int R [ ] ,int n){
    int i,t;
    for (i=0;i<n-1;i++)
        while (R [i] !=i){
            t=R [R [i] ] ;
            R [R [i] ] =R [i] ;
            R [i] =t;
        }
}

```

(1)

(2)

31. 已知单链表的结点结构为

data	next
------	------

## 昆明理工大学 2009 年硕士研究生招生入学考试试题

下列算法对带头结点的单链表 L 进行简单选择排序，使得 L 中的元素按值从小到大排列。  
请在空缺处填入合适的内容，使其成为完整的算法。

```
void SelectSort(LinkedList L)
{
    LinkedList p,q,min;
    DataType rcd;
    p=_____ (1) _____ ;
    while(p!=NULL) {
        min=p;
        q=p->next;
        while(q!=NULL){
            if(_____ (2) _____)min=q;
            q=q->next;
        }
        if(_____ (3) _____ ){
            rcd=p->data;
            p->data=min->data;
            min->data=rcd;
        }
        _____ (4) _____ ;
    }
}
```

五、算法设计题 ( 本题 15 分 )

32. 设线性表  $A=(a_1, a_2, a_3, \dots, a_n)$  以带头结点的单链表作为存储结构。编写一个函数，对 A 进行调整，使得当 n 为奇数时  $A=(a_2, a_4, \dots, a_{n-1}, a_1, a_3, \dots, a_n)$ ，当 n 为偶数时  $A=(a_2, a_4, \dots, a_n, a_1, a_3, \dots, a_{n-1})$ 。