

1999 年西南交通大学数据结构试题  
考研加油站收集整理 <http://www.kaoyan.com>

1999 年西南交通大学数据结构试题

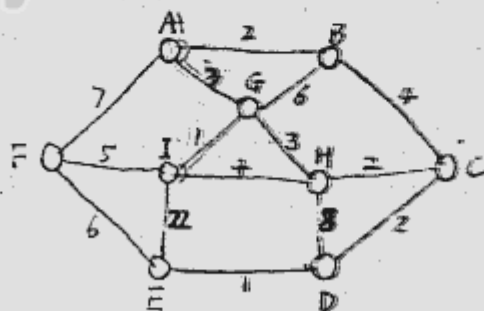
一、填空 (24分)

1.  $a_1, a_2, a_3$  依次进出栈, 写出出栈所有可能的组合序列: \_\_\_\_\_。
2. 设  $S = '000000000001'$ ,  $t = '000001'$ ;  
 a) 模式  $t$  的  $Next[i]$  为: \_\_\_\_\_;  
 b) 用 KMP 算法在  $S$  中查找到  $t$  的比较次数为: \_\_\_\_\_。
3. 从一棵空的平衡二叉树 (AVL) 开始, 依次将关键字插入 AVL 中, 使得四种平衡调整动作至少各执行一次, 这样的插入序列至少应有多少个关键字组成 \_\_\_\_\_。
4. 设有  $n$  个记录组成的二叉排序树, 每个记录查找的概率相等。在最坏情况下, 平均查找长度  $ASL =$  \_\_\_\_\_。

本试题共 6 页, 本页为第 1 页

## 二. 作图 (24分)

1. 画出广义链表  $A = ((a), b, A, (c, d))$ 。
2. 用堆排序法对关键字序列: 23, 17, 12, 60, 25, 8, 68, 71, 52 进行排序,  
a) 画出得到的初始堆;  
b) 画出输出两个最小关键字后的剩余堆。
3. 设元件  $\{e_1, e_2, \dots, e_9\}$  的权值分别为:  
 $\{50, 10, 30, 5, 20, 15, 60, 40, 2\}$ , 求元件的  
哈夫曼树及哈夫曼编码。
4. 求出下图的最少生成树



## 程序阅读及填空 (20分)

阅读下面程序, 假设  $D[1..2] = [3, 2]$ ;

$A[1..9] = [15, 3, 20, 19, 1, 7, 5, 8, 6]$ ;

当执行  $\text{Sort}(A, 9, D)$  后,  $A$  中的值是什么?

```
PROC Sort (VAR A, n, D);
```

```
BEGIN
```

```
FOR k := 1 to 2 DO
```

```
  BEGIN
```

```
    Dh := D[k];
```

```
    FOR j := Dh+1 TO n DO
```

```
      BEGIN
```

```
        temp := A[j]; j := j;
```

```
        while (j > Dh) and (A[j-Dh] > temp) DO
```

```
          BEGIN
```

```
            A[j] := A[j-Dh]; j := j-Dh;
```

```
          END;
```

```
          A[j] := temp;
```

```
        END
```

```
      END
```

```
    ENDP;
```

2. 下面为 Dijkstra 算法, 请填入适当信息, 完善该算法。

```

PROC shortpath(cost, n, v0, dist, path);
FOR i:=1 to n DO
    { dist[i]:=∞; path[i]:={v0} };
dist[v0]:=0; S:={v0}; j:=v0;
FOR k:=1 to n-1 DO
    { FOR i:=1 to ____ DO
        if not(i in S) and (____ + cost[j, i] < dist[i])
        then { dist[i]:=dist[j] + cost[j, i];
              path[i]:=____; };
        min:=∞; j:=v0;
    }
    FOR i:=1 to n DO
        if not(____) and (dist[i] < min)
        then { j:=i; min:=dist[i] };
    S:=____;
    }
ENDP;

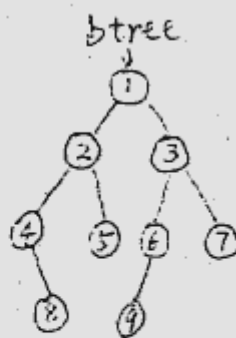
```

#### 四 程序设计 (32分)

- 右图为一二叉树, 结点结构为: 

Lchild	Data	Rchild
--------	------	--------

 编写一二叉树遍历算法, 使得结点输出顺序为 7, 9, 6, 3, 5, 8, 4, 2, 1。



- 设有一指针 DLink 指向一双向链表, 结点结构为: 

LUNK	Data	RUNK
------	------	------

, 分别设计在双向链表中插入和删除一结点的算法:

PROC insert (DLink, x);

PROC Delete (DLink, P);

(其中 x 为待插入结点的值, P 指向要删除的结点, 且  $P \neq DLink$ )

- 所有汉字拼音及字内码构成一个顺序静态查找表, 假设拼音总数目为 400 个。

① 如何最有效地组织该字库? (用结构图表示);

- ② 给出各层次的数据描述;
- ③ 指出各层次的最有效的查找方法;
- ④ 用称法语言写出用输入码

$x = x_1 x_2 \dots x_m$  检索对应的字内码的  
检索算法:

PROC Find( $x$ , outBuf);

{outBuf为字内码缓冲}