

南京邮电学院

2000 年攻读硕士学位研究生入学

数据结构 试题

一、完成下列各题（每小题 6 分，共 18 分）：

1. 设 n 是偶数，试计算运行下列程序段后 m 的值并给出该程序段的时间复杂度。

$m:=0;$

FOR $i:=1$ TO n DO

FOR $j:=2*i$ TO n DO

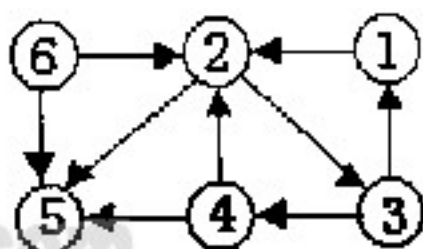
$m:=m+1;$

2. 已知字符串 'cddcdececdca', 计算每个字符的 next 和 nextval 函数的值。

3. 给出冒泡排序和快速排序的最好情况、平均情况和最坏情况下的时间复杂度。

二、完成下列各题：（每小题 8 分，共 24 分）

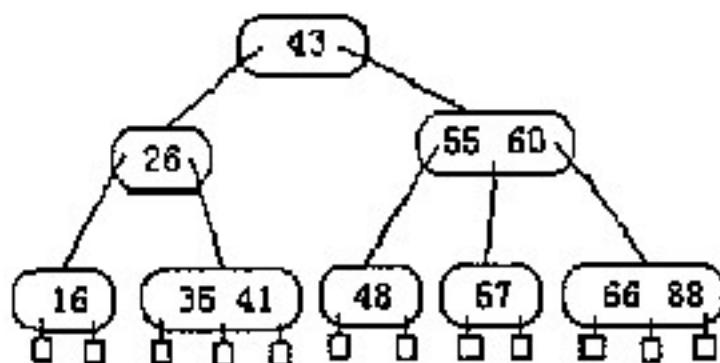
1. 设有下图所示的有向图，给出其邻接矩阵和强连通分量。



2. 设有 3 阶 B-树如下图所示，

(1) 在该 B-树上依次插入关键字 33, 97, 画出两次插入后的 B-树；

(2) 从 (1) 得到的 B-树上依次删除 66, 43, 画出两次删除后的 B-树。



3. 现有 8 个初始游程，每个游程的第一、二个记录的关键字分别为：

游程 \ 记录	1	2	3	4	5	6	7	8
一	11	8	21	9	7	12	45	19
二	15	33	25	14	16	18	48	23

(1) 画出据此构造的败方选择树；

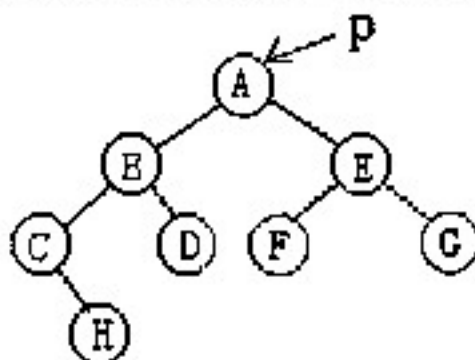
(2) 画出输出一个记录后的败方树。

三、阅读下列二叉树算法，每个结点三个域：lchild, element, rchild。(10分)

(1) $X(p)$ 对以 p 为根的二叉树执行什么功能？

(2) 以下图所示的二叉树调用此算法，则 $X(p)$ 的执行结果是什么？

(3) 执行中，堆栈 s 中元素个数最多时为多少？给出该时栈中元素的情况。



```
PROCEDURE X( p:pointer);
```

```
VAR s:stack; { 定义一个堆栈 s }
```

```
q:pointer;
```

```
BEGIN
```

```
push(s, nil); { push: 向堆栈 s 中压入一个元素 }
```

```
WHILE p <> nil DO
```

```
BEGIN
```

```
q:=p^.lchild;
```

```
p^.lchild:=p^.rchild;
```

```
p^.rchild:=q;
```

```
IF p^.lchild <> nil
```

```
THEN push (s, p^.lchild);
```

```
IF p^.rchild <> nil
```

```
THEN p:=p^.rchild
```

```
ELSE p:=pop(s); { pop: 从堆栈中弹出栈顶元素 }
```

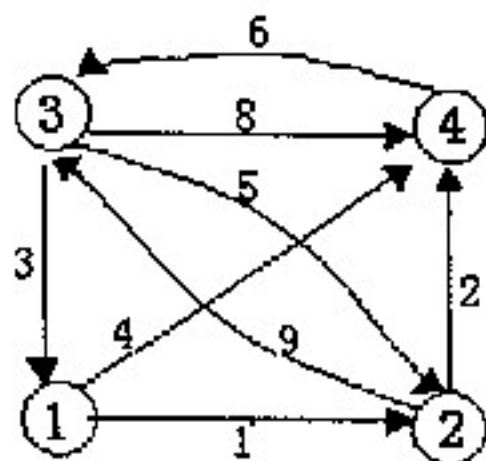
```
END
```

```
END;
```

四、阅读下列求每对顶点之间的最短路径的 Floyd 算法。(16分)

(1) 若对下图所示的有向图执行此算法，写出对 k 为 1 到 n 的各步中，二维数组 a 和 $path$ 的值。

(2) 试设计一个算法，打印每对顶点 $\langle i, j \rangle$ ($1 \leq i, j \leq n$) 之间的最短路径的长度 ($a[i, j]$ 的值) 及其对应的那条路径 (路径上的顶点序列)。



```

CONST n={ user supplied integer }
TYPE  graph=ARRAY[1..n, 1..n] OF real;
      pathtype=ARRAY[1..n, 1..n] OF integer;
PROCEDURE Floyd ( cost: graph; VAR a: graph ;VAR path: pathtype);
VAR ij,k:integer;
BEGIN
  FOR i:= 1 TO n DO
    FOR j:= 1 TO n DO
      BEGIN
        a[i, j]:=cost[i, j];
        IF (i<>j) and (a[i, j]<maxnum)
          THEN path[i, j]:=i
          ELSE path[i, j]:=0;
      END;
    FOR k:= 1 TO n DO
      FOR i:= 1 TO n DO
        FOR j:= 1 TO n DO
          IF a[i, k]+a[k, j] < a[i, j]
            THEN BEGIN
              a[i, j]:=a[i, k]+a[k, j];
              path[i, j]:=path[k, j];
            END;
        END;
      END;
    END;
  END;
END;

```

五、设计一个算法判断一个算术表达式中的括号是否配对。算术表达式保存在带表头结点的单循环链表中，每个结点有两个域：ch 和 link，其中 ch 域为字符类型。（16 分）

六、试设计一个递归算法，在一棵有 n 个结点的随机建立的二叉排序树上查找第 k ($1 \leq k \leq n$) 小元素，并返回指向该结点的指针。要求算法的平均时间复杂度为 $O(\log_2 n)$ ，并说明你所设计的算法具有该时间复杂度的理由。二叉排序树的每个结点有四个域：lchild, element, rchild, num。其中，num 域中已存有以该结点为根的树（子树）上的结点数。（16 分）