

一、对于一个使用邻接表存储的带权有向图  $G$ ，试利用深度优先搜索法，对该图中所有顶点进行拓扑排序。若邻接表的数据类型定义为  $Graph$ ，则算法的首部为：

**FUNCTION** dfs-toposort( $G:Graph$ ):boolean;

若函数返回 **true**，则表示拓扑成功，图中不存在环；若函数返 **false**，则图中存在环，拓扑排序不成功。在这个算法中嵌套用一个递归的深度优先搜索算法：

**PROCEDURE** dfs( $G:Graph$ ;  $V:vtxnum$ );

在遍历图的同时进行拓扑排序。其中， $vtxnum$  是顶点号

(1) 给出该图的邻接表定义； (4 分)

(2) 定义在算法中使用的全局辅助数组； (4 分)

(3) 写出拓扑排序的算法。 (10 分)

二、设有一头指针为  $L$  的带有表结点的非循环双向链表，其每个结点中除有  $pred$ （前驱指针）， $data$ （数据）和  $next$ （后继指针）域外，还有一个访问频度域  $freq$ 。在链表被使用前，其值均初始化为零。每当在链表中进行一次  $Locate(L,x)$  运算时，令元素值为  $x$  的结点中  $freq$  域的值增 1，并使此链表中结点保持按访问频度非增（递减）的顺序排序，同时最近访问的结点排在频度相同的结点的最后，以便使频繁访问的结点总是靠近表头。试编写符合上述要求的  $Locate(L,x)$  运算的算法，该运算为函数过程，返回找到结点的地址，类型为指针型。

(10 分)

三、已知二叉树的链表存储结构定义如下：

**TYPE** bitreptR = ^bitrenode;

bitrenode = RECORD

data:char;

lchild,rchild:bitreptr

**END**;

编写一个递归算法，利用叶结点中空的右链指针域  $rchild$ ，将所有叶结点自左至右链接成一个单链表，算法返回最左叶结点的地址（链头）。 (10 分)

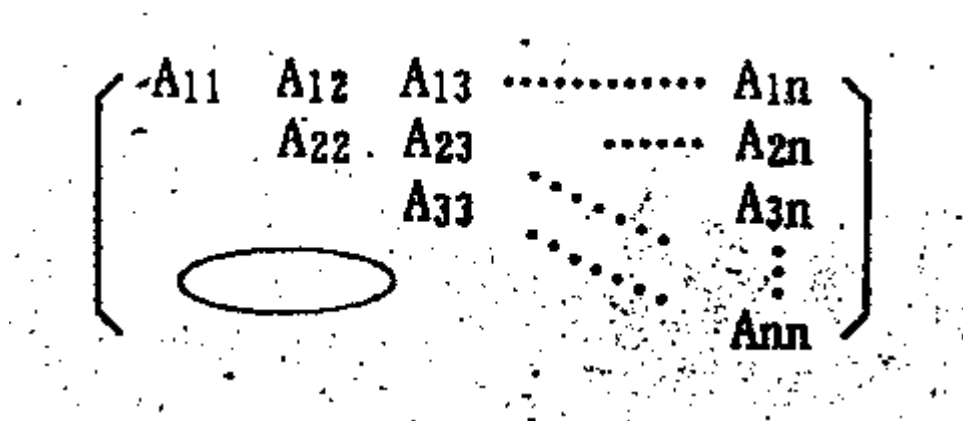
四、设目标为  $S = "abcaabbcaababababca"$ ，模是  $P = "babab"$ ，

(1) 手工计算模式  $P$  的  $nextval$  数组的值； (5 分)

(2) 写出利用求得的  $nextval$  数组，按 KMP 算法对目标  $S$  进行模式匹配的过程。 (5 分)

五、对于一个对称矩阵采用压缩存储，只存放它的上三角部分，并按列存放。例如对于一个

$n \times n$  的对称矩阵  $A$ ，

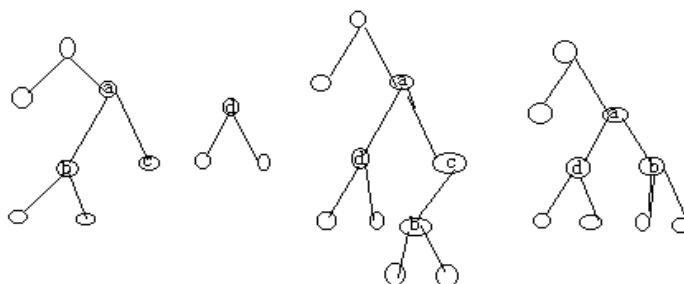


用一个一维数组  $B$  来存放它的上三角部分：

$B=[A_{11}, A_{12}, A_{22}, A_{13}, A_{23}, A_{33}, A_{14}, \dots, A_{1n}, A_{2n}, \dots, A_{nn}]$

同时有两个函数： $MAX(i, j)$  和  $MIN(i, j)$ ，分别计算下标  $i$  和  $j$  中的大者与 smaller。试利用它们给出求任意一个  $A_{ij}$  在  $B$  中存放位置的公式。（若式中没有  $MAX(i, j)$  和  $MIN(i, j)$  则不给分）。  
(10 分)

六、有一棵中序遍历二叉树，如下图 (a) 所示



(a)

(b)

(c)

(1) 现要把一棵根指针为  $d$  的中序线索二叉树插在另一棵中序先索二叉树中，使  $d$  成为的  $A$  左孩子，原来  $A$  的左孩子  $B$  变成  $A$  的右孩子  $C$  的左孩子，如图(B)所示（树中的线索自行画出 0。试针对图中的实例写出实现插入的几条语句。

(2) 现在想在插入后的中序线索二叉树中删去  $A$  右孩子  $C$  并用  $C$  的左孩子填补原来的  $c$  的位置，如图 (c) 所示。试写出实现删除的几条语句。  
(15 分)

七、设有一组数据 black,blue,green,purple,red,white,yellow，它们的查找概率分别为 0.10,0.08,0.12,0.05,0.20,0.25,0.20。试以它们的查找概率为权值，构造一棵次查找树，并计算其查找成功的平均查找长度。  
(12 分)

八、设有 11 个长度（即包含记录个数）不同的归段，它们所包含的记录个数分别为 25,40,16,38,77,64,53,88,9,48,98。

试根据它们做 4 路平均归并，要求：

- (1) 指出总的归并趟数； (3 分)
- (2) 构造最佳归并树； (8 分)
- (3) 根据最佳归并树计算每一趟及总的读记录数。 (5 分)