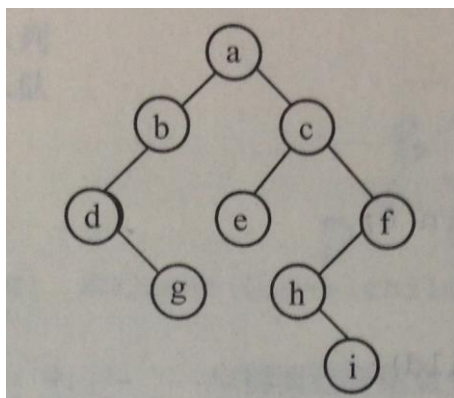


DS Homework 10

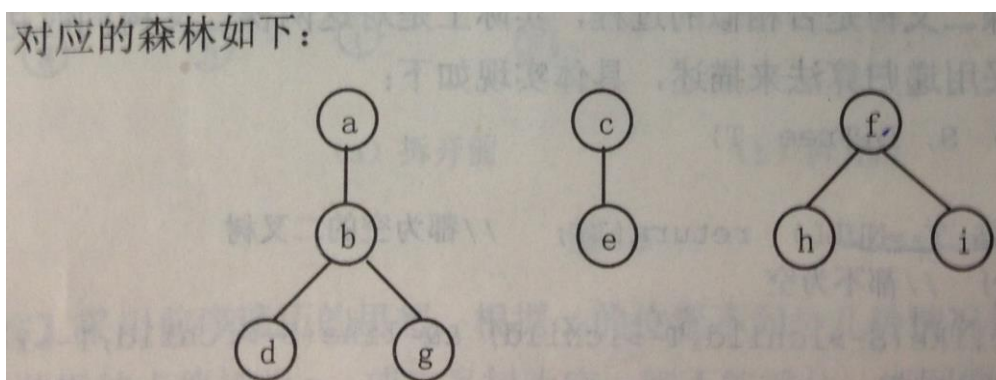
注：请使用 A4 纸作答，写上姓名学号，并于下一次上课时提交。

1、已知某二叉树的先序遍历为 **abdgcefhi** 和中序遍历为 **dgbaechif**，回答以下问题：

(1) 绘制出该二叉树；

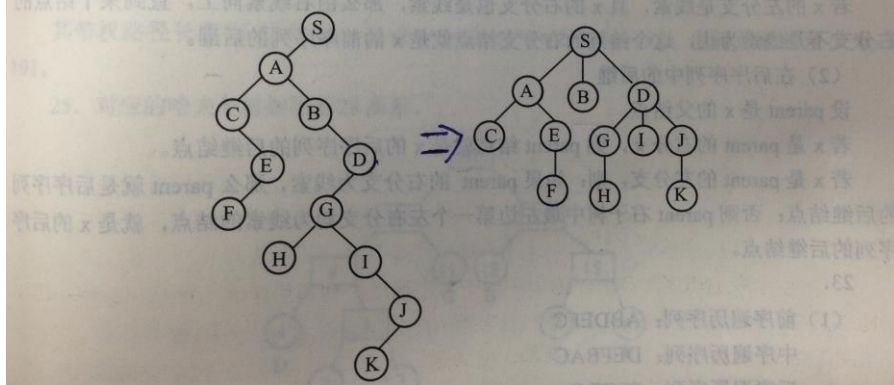


(2) 绘制出该二叉树对应的森林。



2、假设先根次序遍历某棵树的结点次序为 **SACEFBDGHIJK**，后根次序遍历该树的结点次序为 **CFEABHGKJDS**，请绘制出该树。

19. 树的先根次序遍历序列对应于二叉树的前序遍历序列，而后根遍历序列对应于二叉树的中序遍历序列，由给定的遍历序列可以确定一棵二叉树，再将二叉树转化为树，如图 6-24 所示。



3、设计一个复制一棵二叉树 T 的非递归算法。

A5. 算法结束。

33. 解：本题求解可借助于二叉树的层次遍历算法，当然也可以采用前、中、后三种遍历的非递归遍历来实现。基于层次遍历方法复制二叉树 T（新二叉树根为 T_1 ，二叉树用二叉链表作存储结构）的算法描述如下：

A1. 初始化队列 Q 和 Q_1 ；

A2. $T_1 \leftarrow$ 空指针；

A3. 如果 $T =$ 空指针，则 转(A8)；

A4. $T_1 \leftarrow$ 申请一个结点空间；

A5. [$T_1.data \leftarrow T.data$; $T_1.lchild \leftarrow$ 空指针; $T_1.rchild \leftarrow$ 空指针;]

A6. [EnQueue(Q, T); EnQueue(Q_1 , T_1);]

A7. 循环：当队列 Q 不为空时，重复执行

[A7.1. [DeQueue(Q, p); DeQueue(Q_1 , p_1);]

A7.2. 如果 $p.lchild \neq$ 空指针，则 //左孩子不空则复制

[A7.2.1. $r \leftarrow$ 申请一个结点空间；

A7.2.2. [$r.data \leftarrow p.lchild.data$;

$r.lchild \leftarrow$ 空指针;

$r.rchild \leftarrow$ 空指针;

A7.2.3. $p_1.lchild \leftarrow r$;

A7.2.4. [EnQueue(Q, $p.lchild$); EnQueue(Q_1 , r);]

]]

A7.3. 如果 $p.rchild \neq$ 空指针，则 //左孩子不空则复制

[A7.3.1. $r \leftarrow$ 申请一个结点空间;

A7.3.2. [$r.data \leftarrow p.rchild.data$;

$r.lchild \leftarrow$ 空指针; $r.rchild \leftarrow$ 空指针;

]]

A7.3.3. $p_1.rchild \leftarrow r$;

A7.3.4. [EnQueue(Q, $p.rchild$); EnQueue(Q_1 , r);]

]]

A8. 算法结束。