

换其他节点的儿子后这两棵树是相同的。

- a. 给出一个多项式时间算法以决定是否两棵树是同构的。
- *b. 你的程序的运行时间是多少(存在一个线性的解决方案)?

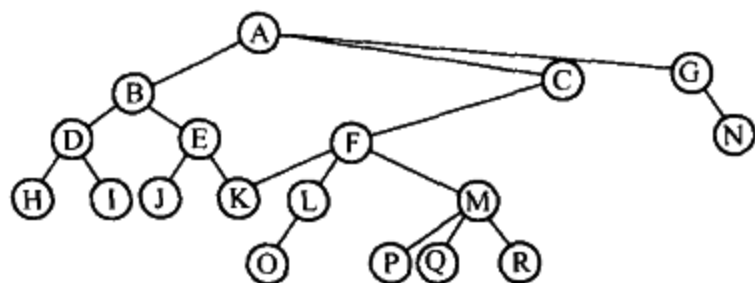


图 4-72 练习 4.44 中的树

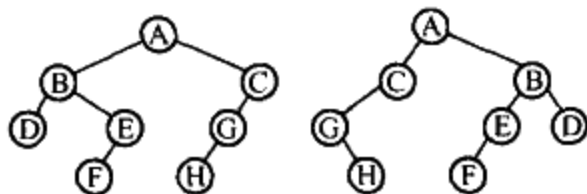


图 4-73 两棵同构的树

- 4.48 *a. 证明, 经过一些 AVL 单旋转, 任意二叉查找树 T_1 可以变换成另一棵(具有相同项的)查找树 T_2 。
- *b. 给出一个算法平均用 $O(N \log N)$ 次旋转完成这种变换。
- **c. 证明该变换在最坏的情形下可以用 $O(N)$ 次旋转完成。
- 4.49 设我们想要把运算 `findKth` 添加到指令集中。该运算 `findKth(k)` 返回树的第 k 个最小项。假设所有的项都是互异的。解释如何修改二叉树以平均 $O(\log N)$ 时间支持这种运算, 而又不影响任何其他操作的时间界。
- 4.50 由于具有 N 个节点的二叉查找树有 $N+1$ 个 `null` 引用, 因此在二叉查找树中指定给链接信息的空间的一半被浪费了。设若一个节点有一个 `null` 左儿子, 我们使它的左儿子链接到它的中序前驱元(inorder predecessor), 若一个节点有一个 `null` 右儿子, 我们让它的右儿子链接到它的中序后继元(inorder successor)。这就叫做线索树(threaded tree), 而附加的链就叫做线索(thread)。
- a. 我们如何能够从实际儿子的链中区分出线索?
- b. 编写执行向由上面描述的方式形成的线索树进行插入的例程和删除的例程。
- c. 使用线索树的优点是什么?
- 4.51 令 $f(N)$ 为一棵 N 节点二叉查找树中满节点的平均个数。
- a. 确定 $f(0)$ 和 $f(1)$ 的值。
- b. 证明, 对于 $N > 1$

$$f(N) = \frac{N-2}{N} + \frac{1}{N} \sum_{i=0}^{N-1} (f(i) + f(N-i-1))$$

- c. (用归纳法)证明 $f(N) = (N-2)/3$ 是问题(b)中的方程的解, 其初始条件在问题(a)中。
- d. 应用练习 4.6 的结果确定二叉查找树中树叶的平均个数。
- 4.52 编写一个程序, 该程序读 Java 源代码文件并以字母顺序输出所有的标识符(即变量名而非关键字, 并且这些变量名不是从注释和串常数中找出的)。每个标识符要和它所在的那些行的一列行号一起输出。
- 4.53 为一本书生成一个索引。输入文件由一组索引项组成。每行由串 `IX:` 组成, 后跟一个索引项的名字(封在大括号内), 后面是封在大括号内的页号。在索引项名字中的每个 `!` 代表一个子层(sub-level)。符号 `|` 代表一个范围的开始, 而 `|)` 则代表这个范围的结束。偶尔这个范围是同一页。在这种情形下只输出一个页号。在其他情况下不要套叠, 否则你自己就扩大了范围。例如, 图 4-74 显示一个样本输入, 而图 4-75 则显示对应的输出。