

5.4树、森林

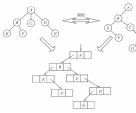
树的存储结构

- 双亲表示法
 - 采用一组连续空间来存储每个结点,同时在每个结点中增设一个伪指针,指示 其双亲结点在数组中的位置
 - 根结点下标为0,其伪指针域为-1
 - 优点 利用了每个结点(根结点除外)只有唯一双亲的性质,可以很快得到每个结点的双亲结点
 - 缺点 但求结点的孩子时需要遍历整个结构
- 孩子表示法
 - 将每个结点的孩子结点都用单链表链接起来形成一个线性结构,此时n个结点就有n个孩子链表(叶子结点的孩子链表为空表)
 - 优点 寻找子女的操作非常直接
 - 缺点 寻找双亲的操作需要遍历n个结点中孩子链表指针域所指向的n个孩子链表
- 孩子兄弟表示法
 - 以二叉链表作为树的存储结构,孩子兄弟表示法使每个结点包括三部分内容:结点值、指向结点第一个孩子结点的指针,及指向结点下一个兄弟结点的指针
 - 优点 这种存储表示法比较灵活,其最大的优点是可以方便地实现树转换为二叉树的操作,易于查找结点的孩子等
 - 缺点 从当前结点查找其双亲结点比较麻烦

若为每个结点增设一个parent 域指向其父结点,则查找结点的父结点也很方便

树、森林与二叉树的转换

- 树转换为二叉树的规则
 - 每个结点左指针指向它的第一个孩子,右指针指向它在树中的相邻右兄弟
- 树转换成二叉树的画法
 - 在兄弟结点之间加一连线
 - 对每个结点,只保留它与第一个 孩子的连线,而与其他孩子的连线全部抹掉
 - 以树根为轴心,顺时针旋转45度
- 森林转换成二叉树的画法
 - 将森林中的每棵树转换成相应的二叉树
 - 每棵树的根也可视为兄弟关系,在每棵树的根之间加一根连线
 - 以第一棵树的根为轴心顺时针旋转45°
- 二叉树转换为森林的规则
 - 若二叉树非空,则二叉树的根及其左子树为第一棵树的二叉树形式,故将根的右链断开
 - 二叉树根的右子树又可视为一个由除第一棵树外的森林转换后的二叉树
 - 应用同样的方法,直到最后只剩一棵没有右子树的二叉树为止,最后再将每棵二叉树依次转换成树,就得到了原森林



森林与二叉树的对应关系

树和森林的遍历

- 先根遍历
 - 若树非空,先访问根结点,再依次遍历根结点的每棵子树,遍历子树时仍遵循先根后子树的规则
 - 其遍历序列与这棵树相应二叉树的先序序列相同
- 后根遍历
 - 若树非空,先依次遍历根结点的每棵子树,再访问根结点,遍历子树时仍遵循先子树后根的规则
 - 其遍历序列与这棵树相应二叉树的中序序列相同
- 先序遍历森林
 - 访问森林中第一棵树的根结点
 - 先序遍历第一棵树中根结点的子树森林
 - 先序遍历除去第一棵树之后剩余的树构成的森林
- 中序遍历森林
 - 中序遍历森林中第一棵树的根结点的子树森林
 - 访问第一棵树的根结点
 - 中序遍历除去第一棵树之后剩余的树构成的森林

树和森林的遍历与二叉树遍历的对应关系

树	森林	二叉树
先根遍历	先序遍历	先序遍历
后根遍历	中序遍历	中序遍历

树的应用——并查集

- 支持操作
 - 1) Union (S,Root1,Root2):把集合S中的子集合Root2并入子集合Root1。要求Root1 和Root2互不相交,否则不执行合并
 - 2) Find(S,x):查找集合S中单元素x所在的子集合,并返回该子集合的名字
 - 3) Initial(S):将集合S中的每个元素都初始化为只有一个单元素的子集合
- 具体实现
 - 用树(森林)的双亲表示作为并查集的存储结构,每个子集合以一棵树表示
 - 所有表示子集合的树,构成表示全集的森林,存放在双亲表示数组内
 - 通常用数组元素的下标代表元素名,用根结点的下标代表子集合名,根结点的双亲结点为负数