

3.3~4矩阵的压缩存储（上）

栈的应用

括号匹配

- 中缀表达式转后缀表达式（手算）
 - ①确定中缀表达式中各个运算符的运算顺序
 - ② 选择下一个运算符，按照「左操作数 右操作数 运算符」的方式组合成一个新的操作数
 - ③ 如果还有运算符没被处理，就继续 ②

- 后缀表达式的计算
 - 手算 从左往右扫描，每遇到一个运算符，就让运算符前面最近的两个操作数执行对应运算，合体为一个操作数
 - 机算
 - ①从左往右扫描下一个元素，直到处理完所有元素
 - ②若扫描到操作数则压入栈，并回到①；否则执行③
 - ③若扫描到运算符，则弹出两个栈顶元素，执行相应运算，运算结果压回栈顶，回到①

- 中缀表达式转前缀表达式（手算）
 - ① 确定中缀表达式中各个运算符的运算顺序
 - ② 选择下一个运算符，按照「运算符 左操作数 右操作数」的方式组合成一个新的操作数
 - ③ 如果还有运算符没被处理，就继续 ②

- 前缀表达式的计算
 - ①从右往左扫描下一个元素，直到处理完所有元素
 - ②若扫描到操作数则压入栈，并回到①；否则执行③
 - ③若扫描到运算符，则弹出两个栈顶元素，执行相应运算，运算结果压回栈顶，回到①

- 中缀表达式转后缀表达式
 - 手算
 - ① 确定中缀表达式中各个运算符的运算顺序
 - ② 选择下一个运算符，按照「左操作数 右操作数 运算符」的方式组合成一个新的操作数
 - ③ 如果还有运算符没被处理，就继续 ②
 - “左优先”原则：只要左边的运算符能先计算，就优先算左边的
 - 机算
 - 从左到右处理各个元素，直到末尾。可能遇到三种情况：
 - ① 遇到操作数。直接加入后缀表达式。
 - ② 遇到界限符。遇到“(“ 直接入栈；遇到”)“ 则依次弹出栈内运算符并加入后缀表达式，直到弹出“(“ 为止。注意：“(“ 不加入后缀表达式。
 - ③ 遇到运算符。依次弹出栈中优先级高于或等于当前运算符的所有运算符，并加入后缀表达式，若碰到“(“ 或栈空则停止。之后再把当前运算符入栈。

- 中缀表达式的计算（用栈实现）
 - 初始化两个栈，操作数栈和运算符栈
 - 若扫描到操作数，压入操作数栈
 - 若扫描到运算符或界限符，则按照“中缀转后缀”相同的逻辑压入运算符栈（期间也会弹出运算符，每当弹出一个运算符时，就需要再弹出两个操作数栈的栈顶元素并执行相应运算，运算结果再压回操作数栈）

递归

- 计算正整数的阶乘 n!
 - 递归调用时，函数调用栈可称为“递归工作栈”
 - 每进入一层递归，就将递归调用所需信息压入栈顶
 - 每退出一层递归，就从栈顶弹出相应信息
- 求斐波那契数列

缺点：太多层递归可能会导致栈溢出

队列在计算机系统中的应用

- 解决主机与外部设备之间速度不匹配的问题 利用队列先进先出的性质，实现对打印机与主机速度不匹配的协调功能
- 解决由多用户引起的资源竞争问题 将多个用户排成一个队列，然后利用先进先出的性质分别将CPU分配给不同的用户使用
- 图的广度优先遍历