4.栈与队列

中缀表达式求值

构思

邓俊辉

deng@tsinghua.edu.cn

## 优先级

❖ 难点:如何高效地 找到 可优先计算的 S。(亦即,其对应的运算符)?

- ❖与括号匹配 迭代版类似,但亦不尽相同
  - 不能简单地按"左先右后"次序处理各运算符
  - 此时,需要考虑更多因素...
- ❖ (约定俗成的) 优先级 : 1 + 2 \* 3 ^ 4 !

可强行改变次序的 括号:(((1+2)\*3)^4)!

## 延迟缓冲

- ◇ 仅根据表达式的 前缀 , 不足以 确定各运算符的计算次序只有在获得足够的 后续信息 之后 , 才能 确定其中哪些运算符可以执行
- ❖ 体现在求值算法的流程上
  为处理某一前缀,必须提前预读并分析,更长的前缀
- ❖ 为此,需借助某种支持延迟缓冲的机制...

scanned to scan
buffered done

## 求值算法 = 栈 + 线性扫描

❖ 自左向右扫描表达式,用栈记录已扫描的部分(含已执行运算的结果) 在每一字符处

while ( 栈的 顶部 存在 可优先计算 的子表达式 ) 该子表达式退栈;计算其数值;计算结果进栈 当前字符进栈,转入下一字符

❖ 只要语法正确,则栈内最终应只剩一个元素

//即表达式对应的数值

//如何判断?

\0 0 3 5 × × 10 10 2 2 1 [10] 2 10 10 10 10 10 10 8

Data Structures & Algorithms, Tsinghua University