

栈与队列

逆波兰表达式：定义与求值

04-G1

日月逝矣，岁不我与

碰见我爱人的面，要替我说到
说我在这儿，依旧在为她颠倒

邓俊辉

deng@tsinghua.edu.cn

Reverse Polish Notation

- ❖ 逆波兰表达式: J. Lukasiewicz (1878 ~ 1956)
- ❖ 在由运算符 (operator) 和操作数 (operand) 组成的表达式中
不使用括号 (parenthesis-free), 即可表示带优先级的运算关系
- ❖ 例如: $0! + 123 + 4 * (5 * 6! + 7! / 8) / 9$
 $0! \ 123 + 4 \ 5 \ 6! \ * \ 7! \ 8 / + * 9 / +$
- ❖ 又如: $(0! + 1)^{(2 * 3! + 4 - 5) - 6! / (7 + 8 + 9)}$
 $0! \ 1 + 2 \ 3! \ * \ 4 + 5 - ^ 6! \ 7 \ 8 + 9 + / -$
- ❖ 相对于日常使用的中缀式 (infix), RPN亦称作后缀式 (postfix)
- ❖ 作为补偿, 须额外引入一个起分隔作用的**元字符** (比如空格) //较之原表达式, 未必更短

栈式求值

0 ! 123 + 4 5 6 ! * 7 ! 8 / + * 9 / +

❖ 引入栈s //存放操作数

逐个处理下一元素x

if (x是操作数) 将x压入s

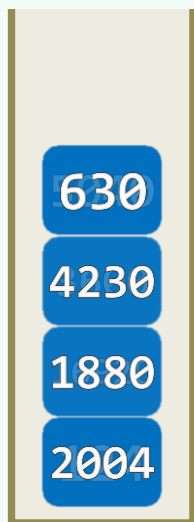
else //运算符无需缓冲

从s中弹出x所需数目的操作数

执行相应的计算，结果压入s //无需顾及优先级！

返回栈顶

❖ 只要输入的RPN语法正确，此时的栈顶亦是栈底，对应于最终的计算结果



0 ! 1 + 2 3 ^ 4 ! - 5 ! 6 / - 7 * 8 * - 9 -

