

# 利用栈实现表达式求值

常州市第一中学 林厚从

# 例、算术表达式的求值

$$12+(2+5*4)/2-5$$

## 四则混合运算规则：

- (1) 先算括号内，再算括号外；
- (2) 在无括号或同层括号的情况下，先做乘除运算，后做加减运算，即乘除运算的优先级高于加减运算的优先级；
- (3) 同一优先级运算，从左向右依次进行。

第1遍  $12+(2+5*4)/2-5$

第2遍  $12+(2+20)/2-5$

第3遍  $12+22/2-5$

第4遍  $12+11-5$

第5遍  $23-5$

18

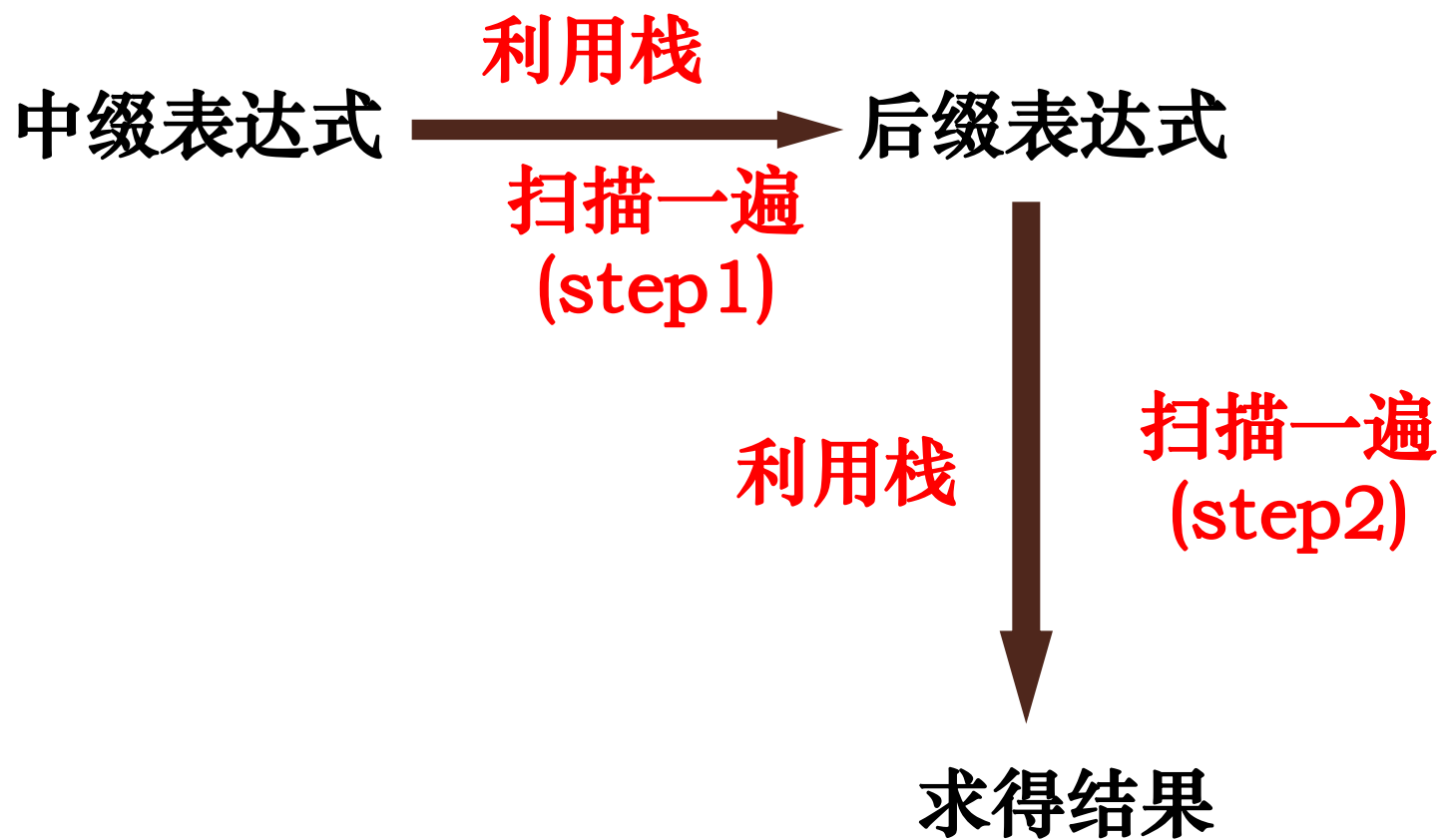
各运算符的实际运算次序与它们出现的先后次序是不一致的，而且是不可预测的。

计算机程序实现时只能一个字符一个字符的扫描，要知道哪一个先算，就必须对整个表达式扫描一次。

因此，理论上表达式中有多少个运算符就要扫描多少遍，算法效率太低！

## 我们换个角度分析这个问题：

- 1、表达式树是什么？
- 2、表达式树的先序、中序、后序三种遍历方法和结果；
- 3、三种遍历对应的三种表达式：前缀表达式、中缀表达式、后缀表达式；
- 4、计算机中表达式的处理：后缀表达式（逆波兰式）；
- 5、后缀表达式的优点：无括号、无优先级、从左往后顺序处理一边便可得出结果。



## Step2、后缀表达式的计算

**1 2 2 5 4 \* + 2 / + 5 - @**

扫描一遍字符串，遇到运算数压栈，遇到运算符（假设为f），则依次从栈中弹出运算数x和y，做y f x运算，再把运算的结果压栈，继续扫描处理字符串，直至结束。最后的结果在栈底。

## Step1、中缀表达式转换为后缀表达式

### 转换规律：

把每个运算符都移到它的两个运算数的后面，  
然后删除所有的括号即可。

中缀表达式：  $12+(2+5*4)/2-5$

                  ↑      ↑      ↑      ↑      ↑  
                  ④      ②      ①      ③      ⑤

后缀表达式：  $12\ 2\ 5\ 4\ *\ +\ 2\ /\ +\ 5\ -$

**观察发现：**

- 1、操作数的先后顺序并没有发生变化；**
- 2、运算符的先后顺序发生了改变。**

**课堂练习：将下列中缀表达式改成后缀表达式**

**1、 $3/5+6$**

**2、 $2*(x+y)/(1-x)$**

**3、 $b^2-4*a*c$**

## 中缀表达式转后缀表达式算法实现

先设置一个运算符的优先级比较表，然后在栈底压入一个设定的优先级最低的运算符（如@），再给原中缀表达式末尾加上一个@。

从左往右扫描中缀表达式，遇到运算数直接输出。遇到运算符(P2)，则比较P2与栈顶运算符(P1),即 $P_1$ 和 $P_2$ 分别表示中缀表达式前后相邻的两个运算符，如果 $P_2$ 优先级高于 $P_1$ ，则将 $P_2$ 压栈，继续往后扫描；低于则出栈输出 $P_1$ ，并继续比较 $P_2$ 与新的栈顶元素 $P_1$ ；如果遇到 $P_2=P_1$ ，一是“（”遇到“）”，则“（”出栈不输出，二是“@”遇到“@”，则表示结束。

此算法也可以用来判断表达式是否合法。

设 $P_1$ 和 $P_2$ 分别表示中缀表达式前后相邻的两个算符

>表示 $P_1$  运算优先于 $P_2$  运算

<表示 $P_2$  运算优先于 $P_1$  运算

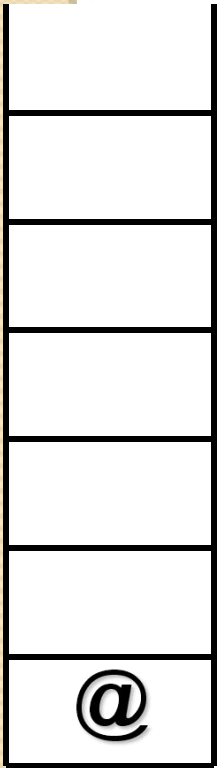
同一优先级运算  
从左向右依次进

在无括号或同层括号  
的情况下  
先算括号内，  
再算括号外

规定

| $P_1 \backslash P_2$ | + | - | * | / | ( | ) | @ |
|----------------------|---|---|---|---|---|---|---|
| +                    | > | > | < | < | < | > | > |
| -                    | > | > | < | < | < | > | > |
| *                    | > | > | > | > | < | > | > |
| /                    | > | > | > | > | < | > | > |
| (                    | < | < | < | < | < | = |   |
| )                    | > | > | > | > |   | > | > |
| @                    | < | < | < | < | < |   | = |

相邻运算符优先关系表



1 遍  $12 + (2 + 5 * 4) / 2 - 5 @$

) = {  
 + >  
 \* > )

@ = @  
 - > @



+ < \*

@ < 1 2 2 5 4 \* + 2 / + 5 - @

( < +

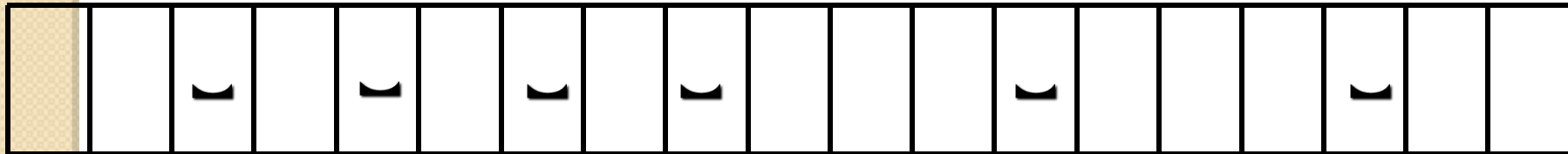
+ > -

+ < (

/ > -

@ < +

+ < /



1 2 2 5 4 \* + 2 / + 5 - @



上机1：NOIP2013复赛普及组T2

上机2：NOIP2011复赛普及组T4