

## NOTAÇÕES: INFIXADA E PÓS-FIXADA

A notação tradicional para expressões aritméticas, que representa uma operação binária na forma  $x+y$ , ou seja, com o operador entre seus dois operandos, é conhecida como notação infixada. Uma notação alternativa para esse tipo de expressão é a notação pós-fixada.

Notação Polonesa Inversa (ou RPN na sigla em inglês, de Reverse Polish Notation), também conhecida como notação pós-fixada, foi inventada pelo filósofo e cientista da computação australiano Charles Hamblin em meados dos anos 1950, para habilitar armazenamento de memória de endereço zero. Ela deriva da notação polonesa, introduzida em 1920 pelo matemático polonês Jan Lukasiewicz. (Daí o nome sugerido de notação Zciweisakul.) Hamblin apresentou seu trabalho numa conferência em Junho de 1957, e o publicou em 1957 e 1962.

Quando a ideia de linguagens de programação de alto nível foi concebida, seus implementadores defrontaram-se com vários obstáculos técnicos. Um deles foi a questão de, durante a compilação, gerar instruções de máquina para avaliar corretamente uma expressão aritmética genérica. Uma expressão como:  $1/2*3+4*5-1*3$ , pode ter significados diferentes, se não forem definidas prioridades dos operandos, regras de avaliação ou uso de parênteses. Aparentemente, a tarefa de gerar as instruções de máquina corretas na ordem correta não é trivial, mas existem soluções simples e elegantes. Na verdade, a solução é considerada tão simples que este aspecto de construção de compiladores é visto como um dos mais triviais.

Uma expressão é composta por operandos e operadores. A expressão acima contém cinco operandos: 1, 2, 3, 4 e 5; e os operadores +, -, \* e /. O primeiro problema que deve ser resolvido para se entender o significado de uma expressão é decidir em qual ordem as operações são executadas. Isto significa que todas as linguagens devem definir alguma ordem. Para fixar a ordem de avaliação, as linguagens permitem o uso de vários níveis de parênteses e definem uma prioridade para cada operador. Assim, operadores com prioridades mais alta são avaliados primeiro. Um possível assinalamento de prioridades é:

Operador	Prioridade
----------	------------

*, /	2
------	---

+, -	1
------	---

Ainda resta o problema de uma expressão com dois operadores adjacentes de mesma prioridade (p.ex.,  $A/B*C$ ). Neste caso, uma regra adicional, dizendo, por exemplo, que a avaliação deve ser feita da esquerda para a direita resolve a ambiguidade.

Definidas as prioridades e regras de avaliação, sabemos que a expressão  $1/2*3+4*5-1*3$  será avaliada como  $((1/2)*3)+(4*5)-(1*3)$ .

Como um compilador aceita expressões como essas e produz código de máquina correto?

A resposta é reorganizar a expressão original numa forma que é conhecida como notação pós-fixada, ou notação polonesa. Expressões escritas na forma convencional estão em notação infixada, porque os operadores estão entre os operandos. Na notação pós-fixada, cada operador aparece após seus operandos. Por exemplo:

Infixada	Pós-fixada
$1*2/3$	$12*3/$
$1/2*3+4*5-1*3$	$12/3*45*+13*-$
$(1/2)*(3+4)*(5-1)*3$	$12/34+*51-*3*$

A notação pós-fixada tem várias vantagens sobre a convencional:

- Não há necessidade de parêntesis;
- Não há necessidade de se estabelecer a prioridade dos operadores;

Para se avaliar uma expressão na notação pós-fixada, basta dar uma passada na expressão da esquerda para a direita, procedendo-se assim:

- Se encontrar um operando, empilhe-o;
- Se encontrar um operador, execute a operação sobre os operandos que estiverem no topo da pilha e empilhe o resultado.
- Ao final, o resultado da expressão estará no topo da pilha.