

Notação Polonesa

Notação Polonesa ou Notação pré-fixada foi inventada pelo matemático polonês Jan Łukasiewicz em torno de 1924 para simplificar a lógica nas sentenças matemáticas. A notação não utiliza parênteses ou outros delimitadores para indicar a precedência dos cálculos e mesmo assim não há ambiguidade quanto à ordem de resolução. Os operadores devem preceder os dois valores numéricos associados. Não é muito usado na matemática convencional, mas muito usado na ciência da computação.

Na notação pré-fixada os operadores são colocados antes dos operandos. A expressão $A + B$ fica $+AB$. Na notação convencional os operadores $*$ e $/$ tem maior precedência sobre os operadores $+$ e $-$. Dessa forma, a expressão $A+B*C$ pode ser interpretada de forma ambígua, pois primeiro é realizada a multiplicação e posteriormente a adição. Caso queiramos realizar primeiramente a operação de soma, precisamos usar o delimitador parênteses para alterar a ordem de cálculo da expressão $(A+B)*C$. Essa ambiguidade, na ciência da computação, torna mais complexo o processo de verificação e validação das expressões por linguagens de programação. Na notação pré-fixa a expressão $A+B*C$ ficaria $+*BCA$ e a expressão $(A+B)*C$ ficaria $*+ABC$.

Notação Polonesa Inversa

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. Ela deriva da notação polonesa, introduzida em 1920 pelo matemático polonês Jan Łukasiewicz.

Na notação pós-fixada os operadores são colocados depois dos operandos. A expressão $A+B$ fica $AB+$. Veja alguns exemplos.

Notação convencional	Notação Pós-fixada	Notação Pré-fixada
$A+B*C$	$ABC*+$	$+*BCA / +A*BC$
$(A+B)*C$	$AB+C*$	$*+ABC$
$A*(B+C)$	$ABC+*$	$*+BCA$
$(A+B)/(C-D)$	$AB+CD- /$	$/+AB-CD$
$(A+B)/(C-D)*E$	$AB+CD- /E*$	$* /+AB-CDE$

Passo a passo

Notação convencional	Notação Pós-fixada	Notação Pré-fixada
$(A+B)/(C-D)*E$	$AB+CD- /E*$	$* /+AB-CDE$
$(6+4)/(4-2)*3$	$6\ 4\ +\ 4\ 2\ -\ /\ 3\ *$	$*\ /\ +\ 6\ 4\ -\ 4\ 2\ 3$
$10 / 2 * 3$	$10\ 2\ /\ 3\ *$	$*\ /\ 10\ 2\ 3$
$5 * 3$	$5\ 3\ *$	$*\ 5\ 3$
15	15	15

Tarefa– Calculadora RPN

Escreva um programa que simule uma calculadora RPN que seja capaz de resolver expressões escritas na notação pós-fixa.

Entrada

A entrada do programa deve ser um arquivo texto no qual cada linha possui uma expressão escrita na notação pós-fixa.

Saída

A saída do programa deve ser outro arquivo texto em que cada linha possui o resultado do cálculo da expressão pós-fixa correspondente às linhas do arquivo de entrada.

Validação

- Assuma que as expressões somente conterão números e não haverá nenhuma variável.

Processamento

Segue a descrição dos passos de um possível algoritmo para realizar o cálculo da expressão escrita na notação pós-fixa.

1. Inicialize pilha vazia
2. Varrer a expressão por token. (Utilize o TokenRead)
 1. Se for operando, empilha o valor.
 2. Se for operador, desempilhar os últimos dois valores, efetuar a operação com eles e empilhar o resultado obtido.
3. Ao final, o resultado da expressão estará no topo da pilha.

Exemplo: 7 3 + 6 4 - *

Token	Ação	Pilha
7	Empilha 7	P:[7]
3	Empilha 3	P:[3,7]
+	Desempilha 3	P:[7]
	Desempilha 7	P:[]
	Efetuar operação 7+3 e empilha 10	P:[10]
6	Empilha 6	P:[6,10]
4	Empilha 4	P:[4,6,10]
-	Desempilha 4	P:[6,10]
	Desempilha 6	P:[10]
	Efetuar operação 6-4 e empilha 2	P:[2,10]
*	Desempilha 2	P:[10]
	Desempilha 10	P:[]
	Efetuar a operação 2*10, empilha 20	P:[20]

Observação

Para implementar a calculadora RPN, utilize os tipos abstratos TokenReader e Stack implementados anteriormente.

Exemplo

Entrada: Teste1_RNP.txt

```
1 3 +
20 3 5 * +
2 3 + 5 *
2 3 10 * +
6 4 + 4 2 - / 3 *
4 10 15 + * 20 /
15 7 1 1 +- / 3 * 2 1 1 + + -
```

Saída: Teste1_RPN_resultado.txt

```
4
35
25
32
15
5
5
```