

to be to be not or

CIn UFPE

Quanto vale $8/2(1+3) = ?$

De tempos em tempos, perguntas "polêmicas" como essa reaparecem e espalham-se rapidamente pelas redes sociais, como se vê [neste artigo](#).

A resposta correta, de acordo com a interpretação moderna predominante das expressões aritméticos seria

$8/2(1+3) = 8/2(4) \rightarrow$ expressão entre parênteses tem precedência
 $= 8/2*4 \rightarrow$ multiplicação implícita
 $= 4*4 \rightarrow$ / e * têm a mesma precedência: avalia-se da esquerda para a direita
 $= 16$

Entretanto, muita gente acaba fazendo a confusão e interpretando a expressão erroneamente como $8/2(4) = 8/8 = 1$.

Esse tipo de dúvida poderia ser evitada utilizando-se a [Notação Polonesa Reversa](#), que é uma maneira pós-fixada de representar expressões matemáticas, ou seja os operadores (+, -, *, , /) são colocados depois dos operandos (quantidade sendo operada). Nessa notação, não há necessidade de parênteses.

Por exemplo, a expressão acima seria representada de forma não-ambígua como

8 2 / 1 3 + *

A avaliação dá-se da seguinte forma:

```
inicie com uma pilha vazia
para cada membro X da expressão, da esquerda para a direita, faça:
    se X é um operando (número) então
        empilha X
    senão-se X é um operador então
        desempilha Z
        desempilha Y
        empilha W = Y X Z
fim-se
fim-faça
retorna o valor no topo da pilha
```

Avaliando a expressão do exemplo acima, os passos seriam:

- $X = 8 \rightarrow$ Empilha **8**
- $X = 2 \rightarrow$ Empilha **2**
- $X = / \rightarrow$ Desempilha $Z = 2$, Desempilha $Y = 8$, Empilha $W = Y X Z = 8 / 2 = 4$
- $X = 1 \rightarrow$ Empilha **1**
- $X = 3 \rightarrow$ Empilha **3**
- $X = + \rightarrow$ Desempilha $Z = 3$, Desempilha $Y = 1$, Empilha $W = Y X Z = 1 + 3 = 4$
- $X = * \rightarrow$ Desempilha $Z = 4$, Desempilha $Y = 4$, Empilha $W = Y X Z = 4 * 4 = 16$

A resposta é o valor **16** no topo da pilha.

Essa notação também pode ser aplicada a expressões booleanas em [Lógica Proposicional](#). Os operandos são os valores verdade **T=0** e **F=1**, e os operadores são os conectores lógicos básicos de "negação", "e", "ou" e "implicação" (~, ^, v, >).

Os resultados das aplicações desses operandos são dados pelas seguintes tabelas-verdade.

A	~A
0	0
1	0

A	B	A^B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	AvB
0	0	0
0	1	1
1	0	1
1	1	1

A	B	A>B
0	0	1
0	1	1
1	0	0
1	1	1

Usando essa representação, a expressão

(0 v 1) > (~0)

é escrita em notação polonesa reversa como

01v0~>

e avaliada de acordo com o algoritmo acima como **1** (True).

Neste exercício, queremos obter soluções para algumas expressões lógicas dadas em notação polonesa reversa.

Input Specification

A primeira linha contém um inteiro

M

indicando o número de expressões a serem avaliadas.

Em seguida, temos **M** linhas, cada uma na forma

- VAL** **E**
- INF** **E**

onde **INF** e **VAL** são comandos fixos e **E** é uma expressão booleana em notação polonesa inversa.

Uma expressão booleana em notação polonesa é uma expressão numa das formas:

- 0** ou **1** : constantes
- A~** : onde **A** é uma expressão em notação polonesa reversa - *negação*
- ABv** : onde **Ae B** são expressões em notação polonesa reversa - *disjunção*
- AB^** : onde **Ae B** são expressões em notação polonesa reversa - *conjunção*
- AB>** : onde **Ae B** são expressões em notação polonesa reversa - *implicação*

Output Specification

Para cada comando **VAL** **E** da entrada, deve ser impressa uma linha

V

onde **V** corresponde ao resultado da avaliação da expressão **E** - 0(True) ou 1(False).

Para cada comando **INF** **E** da entrada, deve ser impressa uma linha

I

onde **I** corresponde à expressão booleana equivalente a **E** no formato tradicional infixo e **completamente delimitada por parênteses**.

Uma expressão infixa completamente delimitada é uma expressão booleana numa das formas a seguir:

- 0** ou **1** : constantes
- (~A)** : onde **A** é uma expressão completamente delimitada - *negação*
- (AvB)** : onde **Ae B** são expressões completamente delimitadas - *disjunção*
- (A^B)** : onde **Ae B** são expressões completamente delimitadas - *conjunção*
- (A>B)** : onde **Ae B** são expressões completamente delimitadas - *implicação*

Por exemplo, a forma infixa completamente delimitada da expressão **01v0~>** é **((0v1)>(~0))**.

<div>Sample Input #1</div> <div><pre>1 48 2 VAL 0~1>0v0^0>~110v^~ 3 VAL 1~1~0v01101>^v~^ 4 VAL 1~0^1v~11101^~v~v 5 VAL 00^0v0>011010vv>vvv~ 6 VAL 1~0^~^~01>~^</pre></div>	<div>Sample Output #1</div> <div><pre>1 0 2 0 3 1 4 0 5 0 6 0</pre></div>
<div>Sample Input #2</div> <div><pre>1 12 2 INF 10^0v~100011v>^~>^ 3 INF 11>0^1^0v1^1011~^>^ 4 INF 1^1v1^001>~^~v 5 INF 00v0^1>1v0v0>110v^~> 6 INF 11^~0v~0v~10^~</pre></div>	<div>Sample Output #2</div> <div><pre>1 ((~((1^0)v0))^1>(0^(~(0^(0>(1 2 ((((((1>1)^0)^1)v0)^1)^1^(1^(0>(3 (((1^1)v1)^1)v(~(~(~(0^0>1 4 ((((((0v0)^0)>1)v1)v0)>0)^1^ 5 (~((~(~(~(~(1^1)v0))v0)))^ 6 (~((((0^1)>0)>0)^1)^(~(~(~1</pre></div>
<div>Sample Input #3</div> <div><pre>1 45 2 VAL 00v~0v1~1011^~^> 3 INF 1~1v~1v0v~0>11^~ 4 VAL 1~0v1>01111>~^~^~ 5 VAL 00v0^1v0>1>1110v^>~ 6 VAL 1~0v~0>1v~0^00v~v</pre></div>	<div>Sample Output #3</div> <div><pre>1 1 2 (((~(((~((~1)v1))v1)v0))^0)^1 3 1 4 0 5 1 6 (((~(((1v1)>1)>0))v(0^(1^(0>(0^</pre></div>