

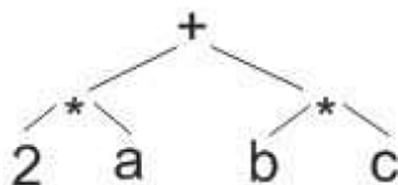


PROLOG

Rafael D. Ribeiro, M.Sc.
rafaeldiasribeiro@gmail.com
<http://www.rafaeldiasribeiro.com.br>

○ Aritmética

- Geralmente, quando se escreve uma expressão matemática a notação infixa é utilizada, por exemplo $2*a+b*c$, onde 2 ; a ; b e c são argumentos e $+$ e $*$ são operadores.
- Em Prolog uma expressão é representada internamente como uma árvore, assim a expressão anterior seria representada pela árvore abaixo.




○ Aritmética

- em Prolog a expressão em questão utiliza notação prefixa, a expressão seria representada como $+(*(2, a), *(b, c))$.
- Porém, por ser mais usual a representação infixa também é compreendida pela linguagem.
- Prolog possui definidos operadores para as quatro operações: +;-; *;/, para realizar soma, subtração, multiplicação e divisão, respectivamente.
- Para se obter o resultado de uma operação é necessário utilizar o operador is
- A precedência de operações aritméticas em Prolog é a mesma precedência adotada na matemática, assim, quando necessário devem ser utilizados parênteses para descrever uma expressão corretamente.

Strawberry PROLOG

```
?- X is 2+5, write(X).
```

 Output

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\ES  
0 errors, 0 warnings.
```


```
7Yes.
```

Strawberry PROLOG

```
lista
?- X is 5/2, write(X).

    Compiling the fil
    C:\Users\RDRIBEIR
    0 errors, 0 warni

2.5Yes.
```




Strawberry PROLOG

```
lista
?- X is 5//2, write(X).

    Compiling the fi
    C:\Users\RDRIBEI
    0 errors, 0 warn

2Yes.
```



○ Aritmética

- Alguns dos operadores reconhecidos são:

Many of the usual arithmetic operators are available in Prolog:

Operator	Meaning	Example
+	addition	2 is 1 + 1.
-	subtraction	1 is 2 - 1.
-	unary minus	Try the query X is 1, Y is - X.
*	multiplication	4 is 2 * 2.
/	division	2 is 6 / 3.
//	integer division	1 is 7 // 4.
mod	integer remainder	3 is 7 mod 4.
**	exponentiation	1.21 is 1.1 ** 2.



Strawberry PROLOG

```
?- X is 5 mod 2, write(X).
```

Output

```
Compiling the file:  
C:\Users\RDRIBEIRO\Document  
0 errors, 0 warnings.
```

```
1Yes.
```



Strawberry PROLOG

```
?- X is sqrt(144), write(X).
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\  
0 errors, 0 warnings.
```

```
12Yes.
```

○ Aritmética

- Existem também operações de conversão, algumas são automáticas outras precisam ser explicitamente solicitadas.
- Um exemplo de conversão automática ocorre quando um número inteiro é relacionado em uma expressão com números de ponto flutuante, automaticamente os inteiros são convertidos para números de ponto flutuante.
- Algumas conversões explícitas nativas são:
 - `integer(X)` - converte X para inteiro;
 - `float(X)` - converte X para ponto flutuante.

○ Aritmética

- Prolog também possui operações para comparação, os operadores são:
 - > - maior que;
 - < - menor que;
 - >= - maior ou igual;
 - =< - menor ou igual;
 - == - igual;
 - = / = - diferente.

Strawberry PROLOG

```
?- 13 > 5.
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documen  
0 errors, 0 warnings.
```

```
Yes.
```

○ Aritmética

- É importante explicitar a diferença entre os operadores = e ==, o primeiro operador verifica se dois objetos são iguais, enquanto o segundo verifica se o resultado da operação é igual.

```
?- 1+2=2+1.
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\E  
0 errors, 0 warnings.
```

```
No.
```

```
?- 1+2==2+1.
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\E  
0 errors, 0 warnings.
```

```
Yes.
```



Strawberry PROLOG

```
?- 1 + X = A + 2 , write(X),write("\n"), write(A).
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\E  
0 errors, 0 warnings.
```

```
2  
1Yes.
```



Strawberry PROLOG

```
impar(Num) :- 1 is Num mod 2.  
?-impar(5).
```

Output

```
Compiling the file:  
C:\Users\RDRIBEIRO\Document  
0 errors, 0 warnings.
```

```
Yes.  
|
```



Strawberry PROLOG

```
impar(Num) :- Num mod 2 == 1.  
?-impar(119).
```

Output

```
Compiling the file:  
C:\Users\RDRIBEIRO\Document  
0 errors, 0 warnings.
```

```
Yes.  
|
```



Strawberry PROLOG

```
impar(Num) :- Num mod 2 == 1.  
?-impar(1.0).
```

```
Compiling the file:  
C:\Users\RDRIBEIRO\Documents\ESTACIO\  
0 errors, 0 warnings.
```

```
Runtime Error at step 3:  
The argument 1 of "mod"  
is "float" but has to be "int".
```

```
No.
```



Strawberry PROLOG

```
impar(Num) :- integer(Num), 1 is Num mod 2.  
?-impar(1).
```

```
Output  
Compiling the file:  
C:\Users\RDRIBEIRO\Document  
0 errors, 0 warnings.  
  
Yes.  
|
```

