

Programação em Lógica

Mauro Hemerly Gazzani

mauro.hemerly@gmail.com

Câmpus de Ituiutaba

<https://github.com/mauro-hemerly/UEMG-2019-1>

Linguagem de Programação em Lógica

Bibliografia

- Bratko, I., “Prolog Programming for Artificial Intelligence” (3rd edition), Addison Wesley, 2000.
- Clocksin, W.F., Mellish, C.S., “Programming in Prolog” (5th edition), Springer, 2003.
- Sterling, L., Shapiro, E., “The Art of Prolog” (2th edition), MIT Press, 1994.
- Casanova, Marco Antônio. Programação em Lógica e a Linguagem Prolog. São Paulo: Edgard Blücher, 1987. 461p. (UEMG Ituiutaba)

Linguagem de Programação em Lógica

Prolog

- **Prolog: Programming in Logic**
- **Prolog** é uma linguagem de programação baseada em **lógica de predicados de primeira ordem**.
- A principal utilização da linguagem Prolog reside no **domínio da programação simbólica**, não numérica, sendo especialmente adequada à solução de problemas, envolvendo **objetos e relações entre objetos**.
- Algumas implementações: **SICStus Prolog, Borland Turbo Prolog, SWI-Prolog, GNU Prolog,...**
- Geralmente é interpretado, mas pode ser compilado.
- Não é padronizada.

Prolog x Outras Linguagens

- **Linguagens Imperativas** (C, Pascal, Basic,...): Especifica-se como realizar determinada tarefa. Permite estados ou dados mutáveis.
- **Linguagens Orientadas a Objetos** (C++, Java, C#...): Especifica-se classes, objetos, métodos e atributos.
- **Prolog**: Especifica-se o quê se sabe sobre um problema e o quê deve ser feito. É mais direcionada ao conhecimento e menos direcionada a algoritmos.
- **Linguagens Funcionais** (Haskell, Lisp,...): Trata da avaliação de funções matemáticas e que evita estados ou dados mutáveis.

Programação em Prolog

- Programar em Prolog envolve:

- Declarar alguns **fatos** a respeito de objetos e seus relacionamentos.

Exemplo: **pai(renato,marcelo).**

- Definir algumas **regras** sobre os objetos e seus relacionamentos.

Exemplo: **pai(P, F) :- filho(F,P).**

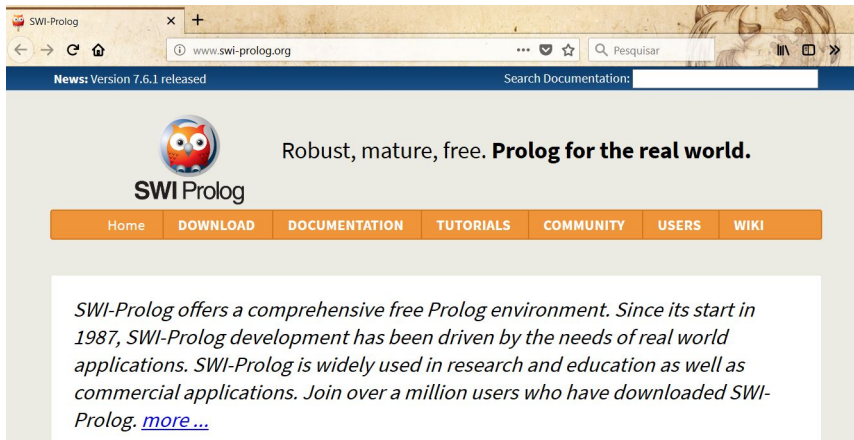
- Fazer **perguntas** sobre os objetos e seus relacionamentos.

Exemplo: **?- pai(renato,marcelo)**

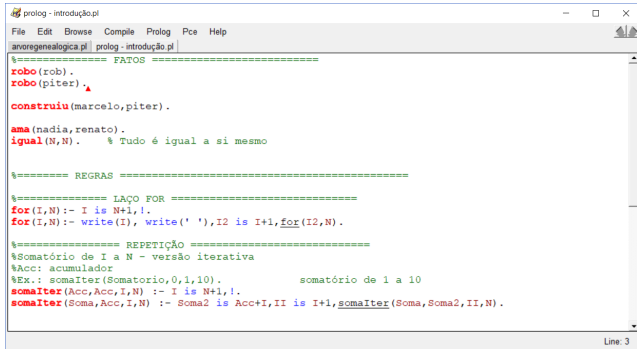
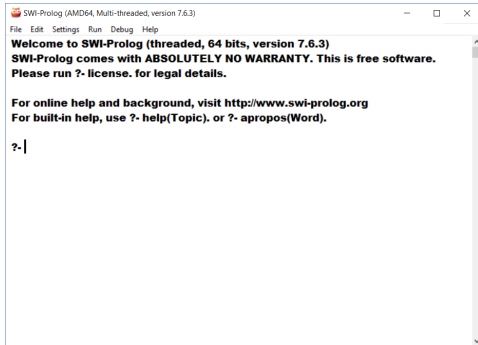
Resposta: **true**

SWI-Prolog

- Open Source.
- Multiplataforma.
- Possui interface com as linguagens C e C++.
- **www.swi-prolog.org**
- Download: **<http://www.swi-prolog.org/download/stable>**



SWI-Prolog Interface



Linguagem de Programação em Lógica

Prolog: Átomos

- São cadeias compostas pelos seguintes caracteres:
 - **letras maiúsculas:** A, B, \dots, Z
 - **letras minúsculas:** a, b, \dots, z
 - **dígitos:** $0, 1, \dots, 9$
 - **caracteres especiais:** $+ - * / < > = : . \& _$
- Podem ser construídos de três maneiras:
 - **cadeias de letras, dígitos e o caractere `'_'`,
começando com uma letra minúscula:
*anna, nil, x25, x_25, x_25AB, x_, x_y, tem_filhos, tem_um_filho***
 - **cadeias de caracteres especiais:** `--` `=====>`, `...`, `..`, `::`, `==`
 - **cadeias de caracteres entre apóstrofes:** `'Abraão'`, `'América_do_Sul'`,
`'América_Latina'`

Sentenças Prolog

- **Nomes de constantes e predicados** iniciam sempre com letra minúscula.
- O **predicado** (relação unária, n-ária ou função) é escrito primeiro e os objetos relacionados são escritos depois entre parênteses.
- **Variáveis** sempre começam por letra maiúscula.
- Toda sentença termina com ponto "."
- **Exemplo:**
 - **homem(renato).**
 - **mulher(nadia).**

Linguagem de Programação em Lógica

Prolog: Operadores Lógicos

Símbolo	Conectivo	Operação Lógica
<code>:-</code>	if	implicação
<code>,</code>	and	conjunção
<code>;</code>	or	disjunção
<code>not</code>	not	negação

Exemplos:

`humano(H) :- homem(H).`

`humano(H) :- mulher(H).`

ou:

`humano(H) :- homem(H);mulher(H).`

Linguagem de Programação em Lógica

- Números usados em **Prolog** incluem números inteiros e números reais

Prolog: Operadores Aritméticos

Operador	Significado	Exemplo	Notação Prolog
is	atribuição	$X = 6$	$X \text{ is } 6$
+	adição	$X = 6+2$	$X \text{ is } 6+2$
-	subtração	$X = 6-2$	$X \text{ is } 6-2$
*	multiplicação	$X = 6*2$	$X \text{ is } 6*2$
/	divisão	$X = 6/2$	$X \text{ is } 6/2$
//	divisão inteira (quociente)	$X = 6/2$	$X \text{ is } 6//2$
mod	divisão inteira (resto)	$X = 6/2$	$X \text{ is mod}(6,2)$
**	potência	$X = 6^2$	$X \text{ is } 6**2$

Linguagem de Programação em Lógica

Prolog: Números

- O operador **=** tenta unificar apenas
 - ?- $X = 1 + 2$.
 - $X = 1 + 2$
- O operador **is** força a avaliação aritmética
 - ?- $X \text{ is } 1 + 2$.
 - $X = 3$
- Se a variável à esquerda do operador **is** já estiver instanciada, **Prolog** apenas compara o valor da variável com o resultado da expressão à direita de **is**
 - ?- $X = 3, X \text{ is } 1 + 2$.
 - $X = 3$
 - ?- $X = 5, X \text{ is } 1 + 2$.
 - false

Linguagem de Programação em Lógica

Prolog: Operadores Relacionais

Operador	Significado
$X = Y$	X unifica com Y
$X =:= Y$	X é igual a Y
$X \neq Y$	X é diferente de Y
$X < Y$	X é menor do que Y
$X > Y$	X é maior do que Y
$X \leq Y$	X é menor ou igual a Y
$X \geq Y$	X é maior ou igual a Y

Prolog: Variáveis

- São cadeias de letras, dígitos e caracteres ' _ ', sempre começando com letra maiúscula ou com o caractere ' _ '
- X, Resultado, Objeto3, Lista_Alunos, ListaCompras, _x25, _32
- O **escopo** de uma variável é **dentro de uma mesma regra ou dentro de uma pergunta**
 - Isto significa que se a variável **X** ocorre em **duas regras/perguntas**, então são **duas variáveis distintas**
- Mas a ocorrência de **X** dentro de uma **mesma regra/pergunta** significa a mesma variável.

Prolog: Variáveis

- Uma variável pode estar:
 - **Instanciada**: quando a variável já referencia (**está unificada** a) algum objeto
 - **Livre ou não-instanciada**: quando a variável não referencia (**não está unificada** a) um objeto, ou seja, quando o objeto a que ela referencia ainda não é conhecido
- Uma vez instanciada, somente **Prolog** pode torná-la não-instanciada através de seu mecanismo de inferência (nunca o desenvolvedor).

Prolog: Variável Anônima

- Quando uma variável aparece em uma única cláusula, não é necessário utilizar um nome para ela
- Utiliza-se a variável anônima, que é escrita com um simples caracter '_'. Por exemplo

temfilho(X) :- progenitor(X,Y).

- Para definir **temfilho**, não é necessário o nome do filho(a)
- Assim, é o lugar ideal para a variável anônima:

temfilho(X) :- progenitor(X,_).

Linguagem de Programação em Lógica

Prolog: Variável Anônima

- Cada vez que um underscore '_' aparece em uma cláusula, ele representa uma nova variável anônima.
- Por exemplo: **alguem_tem_filho :- progenitor(,_).** equivale à:
alguem_tem_filho :- progenitor(X,Y).
que é bem diferente de:
alguém_tem_filho :- progenitor(X,X).
- Quando utilizada em uma pergunta, seu valor não é mostrado. Por exemplo, se queremos saber quem tem filhos mas sem mostrar os nomes dos filhos, podemos perguntar:
?- progenitor(X,_).

Prolog: Regras

- **Regras** são utilizadas para expressar dependência entre um **fato** e **outro fato**:
 - `criança(X) :- gosta(X,sorvete).`
 - `criança(X) :- not odeia(X,sorvete).`
- Ou **grupo de fatos**:
 - `avo(X,Z) :- (mae(X,Y),mae(Y,Z)); (mae(X,Y),pai(Y,Z)).`
- Podem conter **listas**:
 - `compra(ana, [roupa, comida, brinquedo]).`

Prolog: Regras

- Toda regra tem a seguinte forma: **a :- b1, b2,..., bi;bj...bn**
a cabeça, conclusão, conseqüente
b corpo, condição, antecedente
:- se
, e
; ou

Prolog: Definindo Relações por Fatos

- Exemplo de relações familiares:
 - O fato que **Abraão é um progenitor de Isaque** pode ser escrito como:
progenitor(abraao, isaque).
 - Neste caso definiu-se progenitor como o **nome de uma relação**; **abraão e isaque** são seus **argumentos**.

Prolog: Definindo Relações por Fatos

- **Árvore familiar** em Prolog:
 - `progenitor(sara,isaque).`
 - `progenitor(abraão,isaque).`
 - `progenitor(abraão,ismael).`
 - `progenitor(isaque,esaú).`
 - `progenitor(isaque,jacó).`
 - `progenitor(jacó,josé).`
- Cada cláusula declara um **fato** sobre a **relação progenitor**.

Linguagem de Programação em Lógica

Prolog: Exercícios

1. Expresse através de **fatos** e **regras** Prolog as informações contidas na seguinte frase: **“João é um pássaro. Pedro é um peixe. Maria é uma minhoca. Pássaros gostam de minhocas. Gatos gostam de peixes. Gatos gostam de pássaros. Amigos gostam uns dos outros. O meu gato é meu amigo. O meu gato come tudo o que gosta, exceto pessoas. O nome do meu gato é Chuck Norris.”**
2. Escreva um programa Prolog para representar o seguinte:
 - ① João nasceu em Ituiutaba e Jean nasceu em Paris.
 - ② Ituiutaba fica Minas Gerais.
 - ③ Bruxelas fica na Bélgica.
 - ④ Só é mineiro quem nasceu em Minas Gerais.

Prolog: Exercícios

3. Expresse através de **fatos** e **regras** em Prolog as informações contidas na seguinte frase: **“Cassia é bonita. Marcos é rico e bonito. Ana é rica e forte. Fabiano é forte e bonito. Silvio é amável e forte. Todos os homens gostam de mulheres bonitas. Todos os homens ricos são felizes. Qualquer homem que gosta de uma mulher que gosta dele é feliz. Qualquer mulher que gosta de um homem que gosta dela é feliz. Ana gosta de qualquer homem que gosta dela. Cassia gosta de qualquer homem que gosta dela, desde que ele seja rico, amável ou bonito, e forte”**.

Linguagem de Programação em Lógica

Prolog: Exercícios

4. Elabore um programa em PROLOG que considere que uma pessoa é irmã de outra se possuem o mesmo pai ou a mesma mãe. Se uma pessoa é tia de outra se a mãe da outra é irmã da primeira.

Defina fatos e regras para que se possa perguntar:

X é Irmã de Y ?

X é tia de Y ?

5. Escreva frases em Prolog que representem o seguinte conhecimento:
- **peru, frango, salmão, solha, cerveja, vinho branco, vinho tinto, Ana, Antônio, Barbara, Bruno, gosta, casado, combina** Escreva as seguintes questões em Prolog:
 - a) Ana e Bruno são casados e gostam de vinho verde?
 - b) Que bebida combina com salmão?
 - c) Que comidas combinam com vinho verde?