

Aula 01: Fatos, Regras e Consultas

Definição 01: Para escrever um programa em Prolog é necessário codificar uma base de conhecimento.

Definição 02: Uma base de conhecimento é formada por cláusulas.

Definição 03: Uma cláusula pode ser definida por um fato ou uma regra.

Definição 04: Fatos são utilizados para construir cláusulas que são incondicionalmente verdadeiras.

Definição 05: Os fatos são um meio de afirmar que existe uma relação entre os objetos.

Sintaxe de um Fato: predicado(atomo,...,atomo).

Exemplo 01:

In []:

```
is_friend(jennifer, sally).  
is_friend(jennifer, ana).  
is_friend(ana, daniel).  
is_friend(mark, jennifer).
```

Com esse Exemplo, podemos fazer algumas consultas:

Q1: Jennifer é amiga da Ana?

Q2: Mark é amigo da Jennifer?

Q3: Jennifer é amiga do Mark?

Q4: Jennifer é amigo de quem?

Q5: Quais as relações de amizades existentes?

Ao relizar uma consulta, podemos utilizar variáveis, por exemplo, a variável F. Nesse caso, será atribuído a F, todos os átomos que satisfazem a condição para que a cláusula seja verdadeira e retornada como resposta.

Em Prolog, as variáveis comuns devem iniciar com letra maiúscula. Caso você tente criar uma variável iniciando com letra minúscula, na verdade, o que você vai estar fazendo é criando um átomo.

Exemplo 02:

In []:

```
homem(pedro).  
mulher(maria).  
pai(pedro, josé).  
pai(pedro, maria).
```

Com esse Exemplo, vamos fazer algumas consultas:

Consulta 01: Pedro é homem?

Consulta 01: Pedro é homem?

Consulta 03: Pedro é pai de José?

Consulta 04: Pedro é pai de quem?

Consulta 05: Quem é o pai de Maria?

Consulta 06: Quais são as relações de pai existente na base de conhecimento?

Consulta 07: José é irmão de Maria?

Não é possível realizar a **Consulta 07**, uma vez que não existe na base de conhecimento nenhuma cláusula que tenha o predicado irmão.

Regra Diferentemente dos fatos que são incondicionalmente verdadeiros, a regra é uma cláusula condicionalmente verdadeira, ou seja, que pode ser verdadeira ou não.

Definição 06: Uma regra é formada por duas partes, denominadas cabeça e corpo. Sua sintaxe se dá da seguinte forma:

cabeça :- corpo.

Definição 07: Uma regra só será verdadeira se a sua cabeça for verdadeira.

Definição 08: A cabeça da regra só será verdadeira se o corpo for verdadeiro. Sendo assim, podemos considerar que a veracidade de uma regra é determinada pelo seu corpo.

O corpo de de uma regra pode ser formado por mais de uma cláusula da base de conhecimento. Quando este for o caso, entre as cláusulas do corpo de uma regra há a necessidade de utilização de um operador lógico de conjunção (E) ou disjunção (OU).

No Prolog, o operador de conjunção é representado pela vírgula e o operador de disjunção é representado pelo ponto e vírgula.

Sabendo disso, vamos criar a nossa regra para determinar a relação de irmão.

Exemplo 03:

In []:

```
homem(pedro).  
mulher(maria).  
pai(pedro, josé).  
pai(pedro, maria).  
  
irmão(X, Y) :- pai(P, X), pai(P, Y).
```

Agora vamos repetir a nossa **Consulta 07**: José é irmão de Maria?

Vamos fazer mais algumas consultas:

Consulta 08: Maria é irmão de José?

Consulta 09: José é irmão dele mesmo?

Na **Consulta 09**, temos um problema, pois, claramente, uma pessoa não é irmão dela mesma. Mas por que a resposta foi verdadeira para essa consulta?

A resposta está na forma como o algoritmo backtracking que faz a varredura da base de conhecimento trabalha. Explicarei mais sobre ele em sala de aula.

Para evitar que uma pessoa seja irmão dela mesma, devemos acrescentar mais uma cláusula para garantir que X seja diferente de Y. Podemos fazer isso da seguinte forma:

In []:

```
homem(pedro).  
mulher(maria).  
pai(pedro, josé).  
pai(pedro, maria).  
  
irmão(X, Y) :- pai(P, X), pai(P, Y), X \== Y.
```

Agora, o corpo da nossa regra possui três partes, todas separadas pelo operador de conjunção (E). Em linguagem natural, a nossa regra pode ser lida da seguinte forma:

X é irmão de Y, se P é pai de X e P é pai de Y e X é diferente de Y

Quando o corpo da regra possui muitas cláusulas, por questão de melhor leitura do código, podemos utilizar o seguinte estilo:

In []:

```
homem(pedro).  
mulher(maria).  
pai(pedro, josé).  
pai(pedro, maria).  
  
irmão(X, Y) :- pai(P, X),  
               pai(P, Y),  
               X \== Y.
```

Exercício: Vamos construir a base de conhecimento da Figura 1 da seguinte forma:

R1: Crie os fatos para estabelecer quem é homem e quem é mulher.

R2: Crie os fatos para estabelecer as relações de pai e mãe.

R3: Crie a regra gerou(X,Y) tal que X gerou Y se X é pai ou mãe de Y.

R4: Crie a regra para estabelecer a relação de irmão.

R5: Crie a regra para estabelecer a relação de irmã.

R6: Crie a regra para estabelecer a relação de tio.

R7: Crie a regra para estabelecer a relação de tia.

R8: Crie a regra para estabelecer a relação de avô.

R9: Crie a regra para estabelecer a relação de avó

R10: Codifique as regras equivalentes a seguinte sentença: Um casal é formado por duas pessoas que têm filhos em comum.

R11: Codifique as regras equivalentes a seguinte sentença: Todo mundo que tem filhos é feliz.

Utilize a sua base de conhecimento e faça consultas. Analise se as suas consultas retornam os resultados esperados.