

# Lista de Exercício I – Lab. Inteligência Artificial

Antônio Sousa

## Assuntos

- Lógica;
- Prolog;

### 1 Qual a resposta Prolog para as seguintes consultas?

1. ?-  $A = b$ .  $\Rightarrow$  **Resposta:**  $A = b$
2. ?-  $a = b$ .  $\Rightarrow$  **Resposta:** false
3. ?-  $A = 1+2$ .  $\Rightarrow$  **Resposta:**  $A = 1+2$
4. ?-  $A = p(a,b)$ .  $\Rightarrow$  **Resposta:**  $A = p(a,b)$ .
5. ?-  $a = p(a,B)$ .  $\Rightarrow$  **Resposta:** false
6. ?-  $p(A,b) = p(a,B)$ .  $\Rightarrow$  **Resposta:**  $A = a, B = b$
7. ?-  $p(x,A,y) = p(x,B)$ .  $\Rightarrow$  **Resposta:** false
8. ?-  $p(x,A,y) = p(x,f(b),y)$ .  $\Rightarrow$  **Resposta:**  $A = f(b)$

### 2 Para o seguinte programa, quais serão as respostas Prolog para as perguntas abaixo?

$p(a,c)$ .  
 $p(a,b)$ .  
 $p(d,a)$ .  
 $p(d,b)$ .

1. ?- $p(a,b)$ .  $\Rightarrow$  **Resposta:** true
2. ?- $p(a,d)$ .  $\Rightarrow$  **Resposta:** false

3.  $?-p(a,X).$   $\Rightarrow$  **Resposta:**  $X = c$  [ENTER]  $X = b$

4.  $?-p(X,Y).$   $\Rightarrow$  **Resposta:**

$X = a, Y = c$  [ENTER]

$X = a, Y = b$  [ENTER]

$X = d, Y = a$  [ENTER]

$X = d, Y = b$

5.  $?-p(X,Y),p(Z,X).$   $\Rightarrow$  **Resposta:**

$X = a, Y = c, Z = d$  [ENTER]

$X = a, Y = b, Z = d$

6.  $?-p(X,Y),p(Z,Y).$   $\Rightarrow$  **Resposta:**

$X = Z, Z = a, Y = c$  [ENTER]

$X = Z, Z = a, Y = b$  [ENTER]

$X = a, Y = b, Z = d$  [ENTER]

$X = Z, Z = d, Y = a$  [ENTER]

$X = d, Y = b, Z = a$  [ENTER]

$X = Z, Z = d, Y = b$

### 3 Considere o seguinte programa:

$p(a).$

$p(b).$

$p(d).$

$q(a).$

$q(c).$

$q(d).$

$r(X):-p(X),q(X).$

$s(X):-q(X).$

$s(X):-p(X).$

O que o Prolog responde para as seguintes consultas:

1.  $?-r(a).$   $\Rightarrow$  **Resposta:** true

2.  $?-r(b).$   $\Rightarrow$  **Resposta:** false

3.  $?-r(X).$   $\Rightarrow$  **Resposta:**  $X = a$  [ENTER]  $X = d$

4.  $?-s(a).$   $\Rightarrow$  **Resposta:** true [ENTER] true

5.  $?-s(b).$   $\Rightarrow$  **Resposta:** true

6.  $?-s(X).$   $\Rightarrow$  **Resposta:**

X = a [ENTER]  
X = c [ENTER]  
X = d [ENTER]  
X = a [ENTER]  
X = b [ENTER]  
X = d

O que você pode observar sobre a ordem das respostas Prolog para a consulta final ?

**Resposta:** Que o PROLOG segue a ordem da escrita da regra que está sendo consultado, e no caso da última consulta, como  $s(X)$  é declarado primeiro sendo igual a  $q(X)$ , o programa retorna todas as ocorrências de  $q(X)$ , para depois retornar as ocorrências de  $p(X)$ .

#### 4 Considere a seguinte base de dados:

a(1,1).  
a(2,1).  
a(3,2).  
a(4,4).  
b(1,2).  
b(1,3).  
b(2,3).  
b(3,2).  
b(4,4).

Sem usar o computador responda:  
(Considerarei que primeiro seria escrito as regras a e depois as b)

1. Quais são (e em que ordem) as respostas obtidas para a pergunta:  $?-a(X,Y),b(X,Y).$

**Resposta:** X = 3, Y = 2 [DEPOIS] X = Y, Y = 4

2. idem para a pergunta:  $?- a(X,Y), b(X,Y), a(Y,Y).$

**Resposta:** X = Y, Y = 4

**5 Considere a interrogação Prolog: ?- r(X,Y), s(Y,Z), not(r(Y,X)), not(s(Y,Y)). com a seguinte base de dados:**

r(a,b).  
r(a,c).  
r(b,a).  
r(a,d).  
s(b,a).  
s(b,d).  
s(c,a).  
s(c,c).  
s(d,a).

Sem usar o computador, responda:

1. Qual é a primeira resposta?

**Resposta:**  $X = Z, Y = d, Z = a$

2. Quantas vezes Prolog retrocede do terceiro para o segundo predicado antes de obter a primeira resposta?

**Resposta:** 2 vezes

**6 Considere a pergunta, ?-u(X,Y),u(Y,Z),not(u(X,Z))., com a seguinte base de dados:**

u(a,b).  
u(b,b).  
u(c,d).  
u(c,a).  
u(d,a).  
u(d,c).

Sem usar o computador, responda:

1. quantas vezes Prolog retrocede até o primeiro predicado - u(X,Y) para encontrar a primeira resposta?

**Resposta:** 2 vezes

2. quantas vezes Prolog retrocede até o segundo predicado - u(Y,Z) para encontrar a primeira resposta?

**Resposta:** 3 vezes

3. quantas vezes Prolog retrocede até o primeiro predicado -  $u(X,Y)$  para encontrar a segunda resposta?

**Resposta:** 0 vezes após encontrar o primeiro

## 7 Considere o programa descrito abaixo:

```
fatorial(0,1).  
fatorial(N,F):- N1 is N-1, fatorial(N1,F1), F is N*F1,!
```

1. Qual a função do operador `is`? **Resposta:** Atribuição
2. Qual a função do operador `!`? **Resposta:** Ele indica que quando uma das regras aplicar, ele não deverá testar outras, para impedir o backtracking

## 8 Considere o programa descrito abaixo:

```
p(X):- r(X).  
q(X):- s(X).  
q(X):- t(X).  
r(0).  
r(1).  
s(2).  
s(3).  
t(4).  
t(5).
```

Pede-se:

1. Execute e analise o programa para o objetivo  $p(X)$ . Quantas e quais são as instâncias encontradas?

**Resposta:** 2 instâncias:  $X = 0$  e  $X = 1$

2. Execute e analise o programa para o objetivo  $p(X):-!, q(X)$ . (altere no programa e chame  $p(X)$ ) Quantas e quais são as instâncias encontradas? Explique.

**Resposta:** 4 instâncias.  $X = 2$  e  $X = 3$  e  $X = 4$  e  $X = 5$ . Ele retorna todas as ocorrências de  $s$  e  $t$ , pois o  $p(x)$  chama eles. Como o `cut` está antes, ele não fará a sua devida função

3. Execute e analise o programa para o objetivo  $p(X):-q(X),!$ . (altere no programa e chame  $p(X)$ ). Quantas e quais são as instâncias encontradas? Explique.

**Resposta:** 1 ocorrência,  $X = 2$ . Como o cut foi inserido após, agora ele funciona, na primeira ocorrência ele já para, por isso ele imprime apenas o primeiro resultado.

## 9 Dado o programa em Prolog:

```
p(1).  
p(2):- !.  
p(3).
```

Encontre as respostas para as questões abaixo e as justifique:

1. ?- p(X).

**Resposta:**  $X = 1$  [ENTER]  $X = 2$ . Ao encontrar o cut ele para de procurar mais ocorrências.

2. ?- p(X), p(Y).

**Resposta:**

$X = Y, Y = 1$

$X = 1, Y = 2$

$X = 2, Y = 1$

$X = Y, Y = 2$

Ele testa todas as possibilidades até a segunda regra, pois o cut não permite ele pesquisar mais.

3. ?- p(X), !, p(Y).

**Resposta:**

$X = Y, Y = 1$

$X = 1, Y = 2$

O cut inserido na pergunta faz com que ele pare na primeira ocorrência do primeiro p(x), por isso ele vai variar apenas o segundo p(x).

## 10 Dado o programa em Prolog com os predicados:

```
gosta(gato,rato).  
deseja(gato,queijo).
```

Quais os resultados das seguintes consultas: (Justifique a resposta).

1. ?- not deseja(X,Y), gosta(gato,X).

2. ?- gosta(gato,X), not deseja(X,Y).

**Resposta:** Ambos ficaram rodando infinitamente, suponho que o not faz com que exista infinitas possibilidades, pois quaisquer valores que colocam para X e para Y que não estejam definidas, retornará false, que com o not, retornará true.