**1º Exercício LAB – Sarah Rodrigues**

**1.** Qual a resposta Prolog para as seguintes consultas?

**a)** ?- A = b. *A = b.*

**b)** ?- a = b. *False.*

**c)** ?- A = 1+2. *A = 1+2.*

**d)** ?- A = p(a,b). *A = p(a,b).*

**e)** ?- a = p(a,B). *False.*

**f)** ?- p(A,b) = p(a,B). *A = a, B = b.*

**g)** ?- p(x,A,y) = p(x,B). *False.*

**h)** ?- p(x,A,y) = p(x,f(b),y). *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).

**a)** ?-p(a,b). *True*

**b)** ?-p(a,d). *False.*

**c)** ?-p(a,X). *X = c, X = b.*

**d)** ?-p(X,Y). *X = a, Y = c; X = a, Y = b; X = d, Y = a; X = d, Y = b.*

**e)** ?-p(X,Y), p(Z,X).  *X = a, Y = c, Z = d; X = a, Y = b, Z = d.*

**f)** ?-p(X,Y), p(Z,Y). *X = Z, Z = a, Y = c; X = Z, Z = a, Y = b; X = a, Y = b, Z = d; X = Z, Z = d, Y = a; X = d, Y = b, Z = a; 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:

**a)** ?-r(a). *True.*

**b)** ?-r(b). *False.*

**c)** ?-r(X). *X = a; X = d.*

**d)** ?-s(a). *True.*

**e)** ?-s(b). *True.*

**f)** ?-s(X). *X = a; X = c; X = d; X = a; X = b; X = d.*

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

*O Prolog usa pesquisa em profundidade em uma árvore, de forma a pesquisar o programa de cima para baixo e realizar unificações e retrocessos, para testar todas as possibilidades.*

**4.** Considere a seguinte base de dados:

a(1,1).

b(1,2).

a(2,1).

b(1,3).

a(3,2).

b(2,3).

a(4,4).

b(3,2).

b(4,4).

Sem usar o computador responda:

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

*X = 3,Y = 2; X = Y, Y = 4.*

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

*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).

s(b,a).

r(a,c).

s(b,d).

r(b,a).

s(c,a).

r(a,d).

s(c,c).

s(d,a).

Sem usar o computador, responda:

**a)** qual é a primeira resposta? *Resposta = X=Z, Z=a, Y=d.*

*X=a,Y=b, Z=a, not(true)*

*X=a,Y=b, Z=d, not(true)*

*X=a,Y=c, Z=a, not(false), not(true)*

*X=a,Y=c, Z=c, not(false), not(true)*

*X=b,Y=a, false*

*X=a,Y=d, Z=a, not(false), not(false)*

**b)** quantas vezes Prolog retrocede do terceiro para o segundo predicado antes de obter a primeira resposta? *6 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:

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

*X=a,Y=b, Z=b, not(true)*

*X=b,Y=b, Z=b, not(true)*

*X=c,Y=d, Z=a, not(true)*

*X=c,Y=d, Z=c, not(false)*

**b)** quantas vezes Prolog retrocede até o segundo predicado - u(Y,Z) para encontrar a primeira resposta? *4 vezes.*

**c)** quantas vezes Prolog retrocede até o primeiro predicado - u(X,Y) para encontrar a segunda

resposta? 5 vezes.

**8.** Considere o programa descrito abaixo:

*fatorial(0,1).*

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

**8.1.** Qual a função do operador is?

*É usado para operações aritméticas.*

**8.2.** Qual a função do operador ! ?

*É usado para impedir o backtracking.*

**9.** 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:

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

*X = 0; X = 1.*

**9.2.** 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.

*X = 2; X = 3; X = 4; X = 5. Não houve nenhuma alteração no processo de busca, uma vez que o cut está no início, antes de qualquer predicado, de forma que não há “casamento de sucesso” (algum predicado resulta em true para o X escolhido) para que o backtracking seja interrompido a partir desse ponto.*

**9.3.** 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. *X = 2. Como o cut está depois de q(X), ele impede o backtracking a partir do primeiro ponto em que o q(X) realizou um “casamento de sucesso”. Dessa forma, ao testar q(X) para s(X), como há sucesso em s(2), ele impede a busca por mais resultados.*

**10.** Dado o programa em Prolog,

p(1).

p(2):- !.

p(3).

Encontre as respostas para as questões abaixo e as justifique:

**a)** ?- p(X).

*X = 1; X = 2. O cut impede o backtracking a partir do p(2), ou seja, impede a busca de mais resultados.*

**b)** ?- p(X), p(Y).

*X = Y, Y = 1; X = 1, Y = 2; X = 2, Y = 1; X = Y, Y = 2. O cut realiza a mesma interferência da questão anterior. Os resultados mostrados são as combinações para X e Y dentro dos valores 1 e 2 encontrados na busca.*

**c)** ?- p(X), !, p(Y).

*X = Y, Y = 1; X = 1, Y = 2. O cut impede o backtracking para a variável X, de forma que apenas os valores de Y para X=1 são buscados.*

**11.** Dado o programa em Prolog com os predicados:

gosta(gato,rato).

deseja(gato,queijo).

Quais os resultados das seguintes consultas:

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

*False, pois* deseja(X,Y) retorna X=gato e Y=queijo e a negação do fato retorna false.

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

*X = rato, pois temos que gosta(gato,X) retorna X=rato e deseja(rato,Y) retorna false, virando true com a negação.*

**12.** Dado um banco de dados com informações geográficas definindo os seguintes fatos:

oceano(X)

pais(X)

continente(X)

fronteira(X,Y), onde X e Y podem ser países ou oceanos.

loc(X,Y), onde X está localizado em Y (X é um país e Y um continente )

Responda as perguntas:

**a)** Quais países europeus fazem fronteira com oceanos?

*pais(X), localizado(X,europa), fronteira(X,Y), oceano(Y).*

**b)** Ache todos os pares de países A e B onde: A e B tem a mesma fronteira; A faz fronteira com

um oceano e B com outro; A e B não fazem fronteira com o mesmo oceano.

*pais(A), pais(B), A\=B, ( (fronteira(A,X);fronteira(X,A)), (fronteira(B,X);fronteira(X,B)) );*

*pais(A), pais(B), A\=B, fronteira(A,X), fronteira(B,Y), oceano(X), oceano(Y), X\=Y;*

*pais(A), pais(B), A\=B, not((fronteira(A,X),fronteira(B,X),oceano(X))).*

**13.** Escreva um programa Prolog para representar o seguinte:

**a)** João nasceu em Pelotas e Jean nasceu em Paris.

**b)** Pelotas fica no Rio Grande do Sul.

**c)** Paris fica na França.

**d)** Só é gaúcho quem nasceu no Rio Grande do Sul.

*nasceu(joao,pelotas).*

*nasceu(jean,paris).*

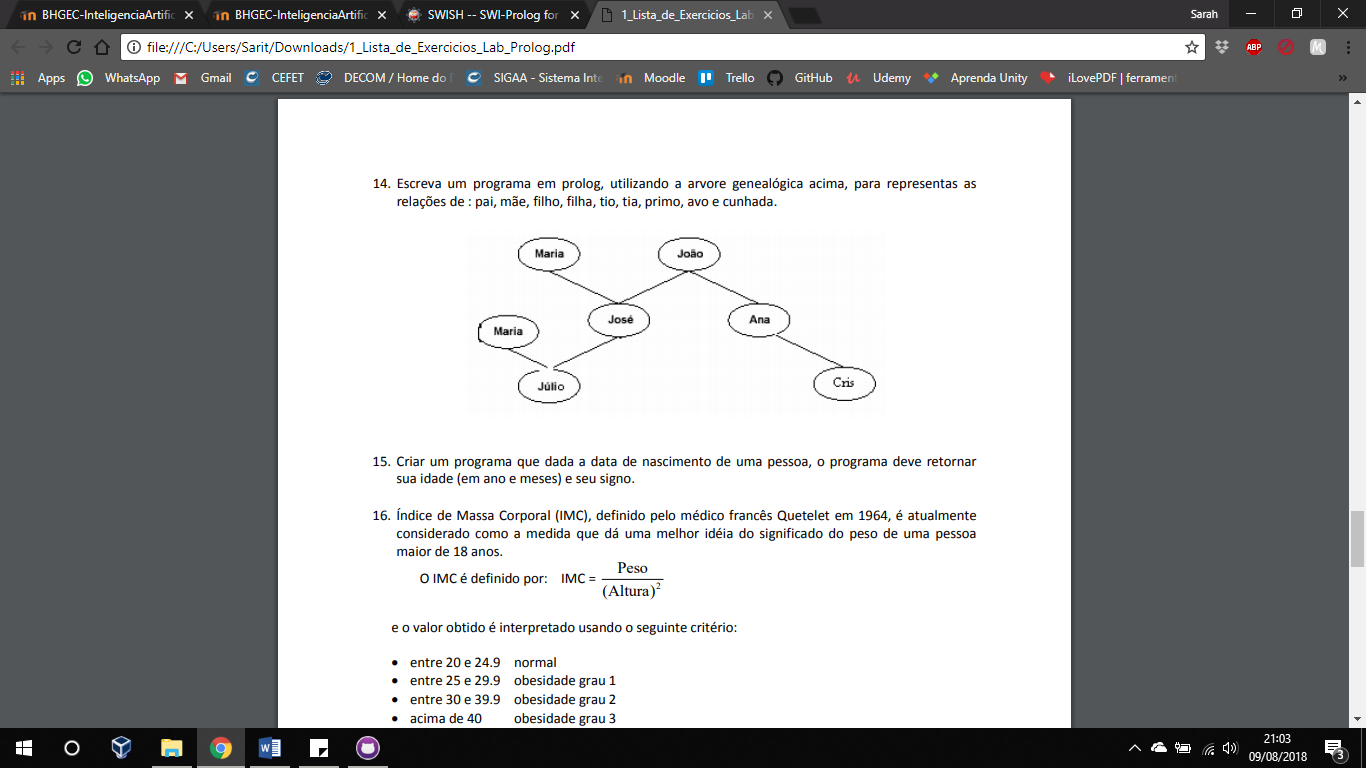
*localizado(pelotas,rio\_grande\_sul).*

*localizado(paris,franca).*

*gaucho(X):-nasceu(X,Y),localizado(Y,rio\_grande\_sul).*

**14.** Escreva um programa em prolog, utilizando a arvore genealógica acima, para representas as

relações de : pai, mãe, filho, filha, tio, tia, primo, avo e cunhada.



*homem(joao).*

*homem(jose).*

*homem(julio).*

*mulher(maria1).*

*mulher(maria2).*

*mulher(ana).*

*mulher(cris).*

*pai(joao,jose).*

*pai(joao,ana).*

*pai(jose,julio).*

*mae(ana,cris).*

*mae(maria1,jose).*

*mae(maria2,julio).*

*filho(X,Y):-homem(X),(pai(Y,X);mae(Y,X)).*

*filha(X,Y):-mulher(X),(pai(Y,X);mae(Y,X)).*

*irmaos(X,Y):-X\=Y,( (pai(Z,X),pai(Z,Y));(mae(W,X),mae(W,Y)) ).*

*tio(X,Y):-homem(X),( (pai(Y,Z),irmaos(X,Z));(mae(Y,Z),irmaos(X,Z)) ).*

*tia(X,Y):-mulher(X),( (pai(Y,Z),irmaos(X,Z));(mae(Y,Z),irmaos(X,Z)) ).*

*primo(X,Y):-homem(X), (pai(Z,X),tio(Z,Y));(mae(Z,X),tia(Z,Y)).*

*prima(X,Y):-mulher(X), (pai(Z,X),tio(Z,Y));(mae(Z,X),tia(Z,Y)).*

**15.** Criar um programa que dada a data de nascimento de uma pessoa, o programa deve retornar

sua idade (em ano e meses) e seu signo.

*signo(capricornio,22,12,20,1).*

*signo(aquario,21,1,19,2).*

*signo(peixes,20,2,20,3).*

*signo(aries,21,3,19,4).*

*signo(touro,20,4,20,5).*

*signo(gemeos,21,5,21,6).*

*signo(cancer,22,6,23,7).*

*signo(leao,24,7,23,8).*

*signo(virgem,24,8,22,9).*

*signo(libra,23,9,22,10).*

*signo(escorpiao,23,10,22,11).*

*signo(sagitario,23,11,21,12).*

*data(D,M,A):- D>0, M>0, D<32, M<13, A>0.*

*idade(A,I):- I is 2018 - A.*

*consulta(D,M,A,S,I):- data(D,M,A), idade(A,I), signo(S,DI,MI,DF,MF), ((M=MI,D>=DI);(M=MF,D=<DF)).*