Lógica de Predicados

1) Determine o comprimento e as subfórmulas das seguintes fórmulas:

a) $p(X, Y, f(Z))$

b) $p(X, Y) \rightarrow (\forall X)q(X)$

2) Marque a fórmula que é equivalente à NEGAÇÃO de
$(\exists V)\ p(V) \rightarrow ((\exists X)\ q(X) \rightarrow (\forall Y)(\exists Z)\ r(Y, Z))$

a) $(\exists V)\ p(V) \land (\exists X)\ q(X) \land (\exists Y)(\forall Z)\ \neg r(Y,Z)$
b) $(\forall V)\ \neg p(V) \rightarrow ((\forall X)\ \neg q(X) \rightarrow (\exists Y)(\forall Z)\ \neg r(Y,Z))$
c) $(\exists V)\ p(V) \lor (\exists X)\ q(X) \lor (\exists Y)(\forall Z)\ \neg r(Y,Z)$

3) Marque a equivalência válida. W é uma fórmula na qual a variável X não ocorre livre.

a) $(\exists X)\ (p(X) \rightarrow W) \Leftrightarrow ((\exists X)\ p(X) \rightarrow W)$

b) $(\forall X)\ (p(X) \land W) \Leftrightarrow ((\exists X)\ p(X) \land W)$

c) $(\forall X)\ (p(X) \rightarrow W) \Leftrightarrow ((\exists X)\ p(X) \rightarrow W)$

4) Assinale as afirmações como verdadeiras ou falsas. Atenção, cada marcação errada anula uma marcação correta.

a) Toda variável é um termo.

b) Toda variável é um átomo.

c) Todo termo é uma fórmula.

d) Todo átomo é uma fórmula.

e) Todo literal é uma fórmula.

f) Todo átomo é um literal.

g) Se G é tautologia, todo *tableau* associado a ¬G é fechado.

h) Se um tableau de ¬G é fechado, G é tautologia.

i) Toda expansão por resolução é finita.

j) A resolução de duas cláusulas que têm no máximo um literal positivo gera uma cláusula com no máximo um literal positivo.

k) Uma cláusula com um único literal é chamada de cláusula unitária. A resolução de duas cláusulas unitárias é a cláusula vazia.

l) Todo tableau associado a uma fórmula insatisfatível é fechado.

m) Um programa lógico sem cláusulas unitárias é incapaz de gerar qualquer resultado positivo.

n) Se G é tautologia, há uma resolução fechada associada a ¬G.

5) Escreva a seguinte fórmula na notação de conjuntos:

$$(\exists X)\ (\forall Y)\ q(Y) \vee (\exists X)\ (\forall Y)\ (\neg(\forall Z)\ r(X, Y, Z))$$

6) Demonstre por resolução ou *tableau* se a seguinte fórmula é satisfatível:

$$(\forall^*)\ ((p(X, Y) \rightarrow \neg p(Y, X)) \wedge p(Z, Z) \wedge ((p(Q, R) \wedge p(S, R)) \rightarrow \neg p(S, R)))$$

7) Selecione a fórmula que representa a seguinte afirmação: “Nenhum dos meus amigos é perfeito”. Nas fórmulas, o predicado a(X) significa “X é meu amigo”, e p(X) significa “X é perfeito”.

a) $(\exists X)\ (a(X) \land \neg p(X))$

b) $(\exists X)\ (\neg a(X) \land \neg p(X))$

c) $(\exists X)\ (\neg a(X) \land p(X))$

d) $\neg(\exists X)\ (a(X) \land \neg p(X))$

8) Considere que o predicado eq(X,Y) significa que X é igual a Y e a função m(X,Y) é o produto de X por Y. Definimos o predicado p(X) como:
$p(X) \Leftrightarrow \neg eq(X, 1) \land (\forall Y)\ (\ (\exists Z)\ eq(X, m(Y, Z)) \rightarrow (eq(Y, X) \lor eq(Y, 1))\ )$.
Assinale a afirmação correta.

a) Se p(X) é verdade, então X é um número primo.
b) Se p(X) é verdade, então X não é um número primo.

9) Considere a afirmação “objetos de ouro são valiosos, objetos de prata também são”. Considere os predicados o(X): “X é de ouro”, p(X): “X é de prata” e v(X): “X é valioso”. Qual fórmula representa essa afirmação?

a) $(\forall X)\ (\ (o(X) \land p(X)) \rightarrow v(X)\ )$

b) $(\forall X)\ (\ (o(X) \lor p(X)) \rightarrow v(X)\ )$

c) $(\forall X)\ (\ v(X) \rightarrow (o(X) \lor p(X))\ )$

10) Considere a afirmação “nem todo dia chuvoso é frio” e os predicados c(X): “X é um dia chuvoso” e f(X): “X é um dia frio”. Qual a fórmula que representa a afirmação?

a) $(\forall X)\ (c(X) \land \neg f(X))$

b) $(\forall X)\ (\neg c(X) \rightarrow f(X)) \Leftrightarrow (\forall X)\ (c(X) \lor f(X))$

c) $(\exists X)\ (\neg c(X) \rightarrow f(X)) \Leftrightarrow (\exists X)\ (c(X) \lor f(X))$

d) $(\exists X)\ (c(X) \land \neg f(X))$

11) Considere a afirmação “Ou $-2 \leq X \leq -1$, ou $1 \leq X \leq 2$.” Qual das opções representa o complemento dessa afirmação?

a) $X < -2$ ou $X > 2$ ou $-1 < X < 1$
b) $X < -2$ ou $X > 2$
c) $-1 < X < 1$
d) $X \leq -2$ ou $2 \leq X$ ou $-1 \leq X \leq 1$

12) Um mapa de um certo jogo é formado por várias salas numeradas. Algumas salas estão conectadas através de portais unidirecionais. Ou seja, se há um portal de 1 para 2, é possível ir da sala 1 para a sala 2 mas o contrário não é verdade.

Considere a seguinte lista de portais, onde `p(X, Y)` indica que "Há um portal de `X` para `Y`":

```
p(1,2). p(3,4). p(5,6). p(7,8). p(9,10). p(12,13). p(13,14).
p(15,16). p(17,18). p(19,20). p(4,1). p(6,3). p(4,7).
p(6,11). p(14,9). p(11,15). p(16,12). p(14,17). p(16,19).
```

a) Escreva um predicado `vai(X,Y)` que significa "A sala `Y` pode ser alcançada a partir da sala `X` em um ou mais passos".

b) Há alguma sala inalcançável a partir das demais?

c) Quais salas, quando alcançadas, deixam o jogador preso?

13) Considere os seguintes predicados com informações sobre viagens:

```
onibus(recife, maceio).
onibus(maceio, aracaju).
onibus(liege, cologne).
onibus(liege, lille).

trem(lille, frankfurt).
trem(cologne, frankfurt).
trem(lille, paris).
trem(cologne, paris).

aviao(frankfurt, bangkok).
aviao(frankfurt, singapura).
aviao(paris, losAngeles).
aviao(bangkok, recife).
aviao(singapura, recife).
aviao(losAngeles, recife).
```

onde `carro(X,Y)` indica que é possível ir de carro de `X` para `Y`. `trem(X,Y)` indica que é possível ir de trem de `X` para `Y` e `aviao(X,Y)` indica que é possível ir de avião de `X` para `Y`.

- ***a)*** Escreva um predicado `viagem(X,Y)` que determina se é possível ir de `X` para `Y` através de alguma combinação de deslocamentos. Ex: `viagem(paris, aracaju)` deve retornar `True`.

b) Saber que é possível viajar é uma informação interessante, mas melhor ainda seria saber qual sequência de viagens devem ser feitas. Escreva um predicado `viagem(X,Y,Z)` que diz a rota que deve ser feita. Por exemplo, para a consulta `viagem(paris, aracaju, Z)` o Prolog deve retornar:

```
Z = viagem(paris, losAngeles, viagem(losAngeles, recife,
viagem(recife, maceio, viagem(maceio, aracaju)))).
```

c) Estenda o predicado definido anteriormente para dizer que tipo de transporte deve ser utilizado nas transições. Para a consulta `viagem(paris, aracaju, Z)` o programa deve retornar:

```
Z = viagem(paris, losAngeles, aviao, viagem(losAngeles,
recife, aviao, viagem(recife, maceio, onibus,
viagem(maceio, aracaju, onibus)))).
```

14) Vamos criar predicados para retornar listas de inteiros em um dado intervalo.

a) Crie o predicado `lista(I, S, L)`, que vai unificar em `L` uma lista contendo todos os inteiros de `I` até `S`.
Exemplo de consulta:
```
?- lista(-2, 3, L).
L = [-2, -1, 0, 1, 2, 3]
```

b) Crie o predicado `lista(S, L)`, que vai unificar em `L` uma lista contendo os naturais de `0` até `S`.
Exemplo de consulta:
```
?- lista(3, L).
L = [0, 1, 2, 3]
```

c) Crie o predicado `lista(L)`, que vai unificar em `L` uma lista contendo os naturais de `0` a `10`.
Exemplo de consulta:
```
?- lista(L).
L = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

15) Matrizes são geralmente representadas em linguagens de programação através de listas de listas. A matriz:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{bmatrix}$$

Pode ser representada no Prolog como:

```
A = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]]
```

a) Para ser uma matriz, todas as linhas precisam ter o mesmo comprimento. Crie um predicado `matriz(M)` que verifica se `M` é uma matriz ou não.

b) Crie um predicado `transpoe(M, T)` que unifica em `T` a transposta da matriz `M`.

Exemplo de consulta:

```
?- transpoe([[1,2,3,4], [5,6,7,8], [9,10,11,12]], T).
T = [[1, 5, 9], [2, 6, 10], [3, 7, 11], [4, 8, 12]].
```

$$A^T = \begin{bmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{bmatrix}$$