



INSTITUTO
SUPERIOR
TÉCNICO

Ciências de Engenharia Informática e de Computadores

Lógica para Programação

Solução do Segundo Teste

20 de Dezembro de 2007

15:00–16:30

Nome: _____ Número: _____

1. (6.0) Para cada uma das seguintes questões, indique se é verdadeira ou falsa. NOTA: Uma resposta errada desconta 0.4 valores.

- (a) Na conversão para a forma clausal normal de uma *fbf* em lógica de primeira ordem, os quantificadores existenciais são eliminados antes dos quantificadores universais.

Resposta: V

- (b) Na conversão para a forma clausal normal de uma *fbf* em lógica de primeira ordem, a eliminação do quantificador existencial consiste em substituir todas as variáveis quantificadas existencialmente por uma constante de Skolem.

Resposta: F

- (c) O processo de unificação permite determinar se duas *fbfs* atómicas podem ficar iguais através de substituições apropriadas para as suas variáveis livres.

Resposta: V

- (d) O unificador mais geral para um conjunto de *fbfs* atómicas é único excepto para variantes alfabéticas de variáveis.

Resposta: V

- (e) Uma conceptualização é um triplo $\langle D, F, R \rangle$ em que D é o universo do discurso, F é o conjunto de funções e R é o conjunto de relações ou predicados.

Resposta: V

- (f) Uma interpretação é uma função que tem como domínio as entidades da conceptualização e como contradomínio as entidades da linguagem.

Resposta: F

- (g) Um conjunto de cláusulas Δ é não satisfazível se e só se um conjunto finito de instâncias fechadas de cláusulas de Δ é não satisfazível.

Resposta: V

- (h) Uma cláusula de Horn é uma cláusula que contém, no máximo, um literal negativo.

Resposta: F

- (i) A programação em lógica combina a representação de um subconjunto de fórmulas de primeira ordem com uma estratégia de resolução.

Resposta: V

- (j) O conjunto das cláusulas de Horn constituído por regras e factos diz-se o conjunto das cláusulas determinadas.

Resposta: V

- (k) Para usar resolução com as cláusulas de Horn, um dos resolventes tem de ser necessariamente um objectivo.

Resposta: F

- (l) Uma afirmação em PROLOG corresponde a uma cláusula em que o corpo não contém literais.

Resposta: V

- (m) Uma função de selecção permite escolher o literal de uma cláusula objectivo como candidato na aplicação do princípio da resolução.

Resposta: V

- (n) O PROLOG não permite que o mesmo símbolo de predicado seja utilizado com diferentes números de argumentos.

Resposta: F

- (o) O funcionamento do PROLOG corresponde a uma refutação SLD com uma função de selecção que escolhe o primeiro literal numa cláusula objectivo e com uma regra de procura que escolhe a última cláusula na sequência de cláusulas que constitui o programa.

Resposta: F

2. (1.5) Determine o unificador mais geral para o seguinte conjunto de *fbfs*. Apresente todos os passos intermédios.

$$\Delta = \{P(a, f(x), g(z)), P(x, f(a), y)\}$$

Resposta:

Conjunto	Conjunto de desacordo	Substituição
$\{P(a, f(x), g(z)), P(x, f(a), y)\}$	$\{a, x\}$	$\{a/x\}$
$\{P(a, f(a), g(z)), P(a, f(a), y)\}$	$\{g(z), y\}$	$\{g(z)/y\}$
$\{P(a, f(a), g(z))\}$		

O unificador mais geral é $\{a/x, g(z)/y\}$.

3. (1.5) Considere a conceptualização $C = (D, F, R)$ em que:

- $D = \{\square, \triangle, \diamond\}$
- $F = \{(\square, \square), (\triangle, \triangle), (\diamond, \diamond)\}$

- $R = \{(\square, \diamond), (\square, \triangle), (\diamond, \triangle)\}$

e a seguinte interpretação:

- $I(Q) = \square$
- $I(T) = \triangle$
- $I(L) = \diamond$
- $I(\text{Identidade}) = \{(\square, \square), (\triangle, \triangle), (\diamond, \diamond)\}$
- $I(\text{LadosIguais}) = \{(\square, \diamond)\}$
- $I(\text{MaisLados}) = \{(\square, \triangle), (\diamond, \triangle)\}$

Diga, justificando, se a seguinte *fbf* é satisfeita por esta interpretação para esta conceptualização:

$$(\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)) \rightarrow \text{LadosIguais}(\text{Identidade}(L))$$

Resposta:

A interpretação I satisfaz a *fbf*

$$(\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)) \rightarrow \text{LadosIguais}(\text{Identidade}(L))$$

se e só se

- I não satisfaz $\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)$ ou
- I satisfaz $\text{LadosIguais}(\text{Identidade}(L))$.

Consideremos cada um destes casos separadamente:

- (a) A interpretação I satisfaz a *fbf* $\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)$ se e só se I satisfizer pelo menos uma das *fbfs*, $\neg \text{LadosIguais}(Q)$ ou $\text{MaisLados}(Q, T)$.

Consideremos o segundo elemento da disjunção:

- i. A interpretação I satisfaz a *fbf* $\text{MaisLados}(Q, T)$ se e só se o par $(I(Q), I(T))$ for um elemento da relação $I(\text{MaisLados})$.

$$(I(Q), I(T)) = (\square, \triangle)$$

$$I(\text{MaisLados}) = \{(\square, \triangle), (\diamond, \triangle)\}$$

Como $(\square, \triangle) \in \{(\square, \triangle), (\diamond, \triangle)\}$, a interpretação I satisfaz a *fbf* $\text{MaisLados}(Q, T)$.

Por consequência, a interpretação I satisfaz a *fbf* $\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)$.

- (b) A interpretação I satisfaz a *fbf* $\text{LadosIguais}(\text{Identidade}(L))$ se e só se $(I(\text{Identidade}(L)))$ for um elemento da relação $I(\text{LadosIguais})$.

$$(I(\text{Identidade}(L))) = (I(\text{identidade})(I(L))) = (I(\text{identidade})(\diamond)) = (\diamond)$$

$$I(\text{LadosIguais}) = \{(\square, \diamond)\}$$

Como $(\diamond) \in \{(\square, \diamond)\}$, a interpretação I satisfaz a *fbf* $\text{LadosIguais}(\text{Identidade}(L))$

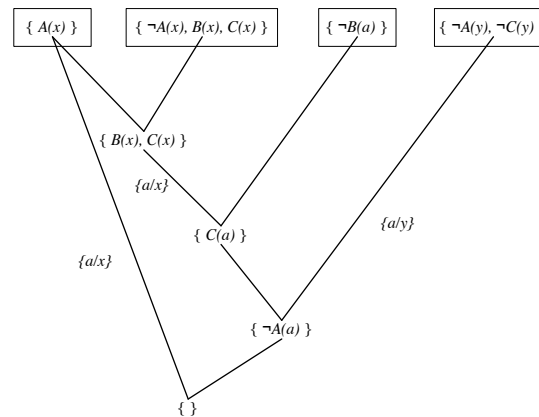
Portanto, a interpretação I satisfaz a *fbf*

$$(\neg \text{LadosIguais}(Q) \vee \text{MaisLados}(Q, T)) \rightarrow \text{LadosIguais}(\text{Identidade}(L)).$$

4. (1.0) Desenvolva uma demonstração por refutação, usando as seguintes cláusulas:

$$\{\neg A(x), B(x), C(x)\}, \{A(x)\}, \{\neg B(a)\}, \{\neg A(y), \neg C(y)\}.$$

Resposta:



5. (2.0) Considere o seguinte conjunto de cláusulas de Horn:

$$A(x) \leftarrow B(x), C(x)$$

$$B(x) \leftarrow D(x)$$

$$C(x) \leftarrow E(x)$$

$$B(a_1) \leftarrow$$

$$E(a_1) \leftarrow$$

$$C(a_2) \leftarrow$$

$$C(a_3) \leftarrow$$

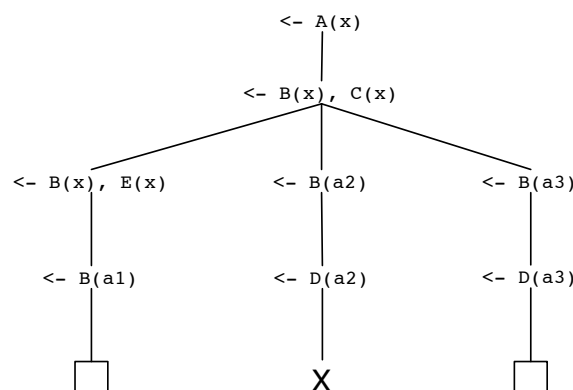
$$D(a_3) \leftarrow$$

Usando uma árvore de resolução SLD e uma função de selecção que escolha para unificar o *último* literal do objectivo, mostre todas as soluções para o seguinte objectivo:

$$\leftarrow A(x).$$

NOTA: pode usar a estratégia de procura que preferir. No final, indique explicitamente as soluções.

Resposta:



As respostas são $x = a_1$ e $x = a_3$.

6. Considere a seguinte estrutura de dados:

```
paciente(Numero, Nome, Idade, Morada, ListaMaleitas)
```

Supondo disponível uma base de dados com os dados dos pacientes de um hospital (BDPacientes.pl), apresente o código em PROLOG para calcular:

- (a) (0.5) A lista das maleitas do paciente “Joaquim Silva”.

Resposta:

```
paciente(_, 'Joaquim Silva', _, _, ListaMaleitas).
```

- (b) (0.5) Uma lista de pares (Numero, Nome), com o número (Numero) e o nome (Nome) de cada paciente do hospital.

Resposta:

```
setof((Numero, Nome),
      paciente(Numero, Nome, _, _, _),
      ListaDePares).
```

- (c) (2.0) Supondo que o hospital queria enviar um cartão de boas festas a todos os pacientes *com mais de 60 anos*, escreva um pequeno programa em PROLOG que usa BDPacientes.pl para gerar os destinatários, na forma:

```
Exmo Senhor(a) Nome
Morada
```

Resposta:

```
boasfestas :- ['BDPacientes.pl'],
              /* Guarda em ListaPacientes o número de todos os pacientes
               com mais de 60 anos */
              findall(NumPaciente,
                      (paciente(NumPaciente, _, Idade, _, _),
                       Idade > 60),
                      ListaPacientes),
              escreve-destinatarios(ListaPacientes).

/* Escreve o texto pedido, relativo a cada paciente */
escreve-destinatarios([]).
escreve-destinatarios([NumPaciente | RestoListaPacientes]) :-
    paciente(NumPaciente, Nome, _, Morada, _),
    write('Exmo Senhor(a) '),
    write(Nome),
    nl,
    write(Morada),
    nl,
    escreve-destinatarios(RestoListaPacientes).
```

7. (1.0) Atente no seguinte programa:

```
maximo(X,Y,X) :- X >= Y, !.
maximo(X,Y,Y).
```

Explique porque razão o PROLOG responde yes ao que se segue:

```
?- maximo(5,2,2).
```

Resposta:

O Prolog responde “yes” ao programa dado, pois é possível unificar `maximo(5, 2, 2)` com o facto `maximo(X, Y, Y)` (repare-se que não é possível unificar `maximo(5, 2, 2)` com a primeira regra do programa e que o corte não chega a ser atingido).

Para que o Prolog devolvesse o resultado esperado (“No”), seria necessário acrescentar a condição `X < Y`, à segunda regra do programa.

8. (1.0) Dado o seguinte programa em PROLOG:

```
soldado(X) :- \+ casado(X), homem(X).
casado(manuel).
homem(manuel).
homem(pedro).
```

Indique e justifique a resposta do PROLOG ao seguinte objectivo:

```
?- soldado(X).
```

Resposta:

```
?- soldado(X).
No
```

O meta-predicado `\=` tem sucesso quando uma instância da sua variável não é derivável. No caso da primeira regra ter sido escrita

```
soldado(X) :- homem(X), \+ casado(X).
```

o programa respondia `pedro`.

9. (a) (2.0) Considere a definição dos *números de Fibonacci*:

$$fib(n) = \begin{cases} 0 & \text{se } n = 0 \\ 1 & \text{se } n = 1 \\ fib(n-1) + fib(n-2) & \text{se } n > 1 \end{cases}$$

Seja `fib` o predicado com o seguinte significado: `fib(N, V)` afirma que o N-ésimo número de Fibonacci é V. Escreva um programa em PROLOG que implementa o predicado `fib`.

Resposta:

```
fib(0, 0).
fib(1, 1).
fib(X, Fib_X) :-
    X_menos_1 is X - 1,
    fib(X_menos_1, Fib_X_menos_1),
    X_menos_2 is X - 2,
    fib(X_menos_2, Fib_X_menos_2),
    Fib_X is Fib_X_menos_1 + Fib_X_menos_2.
```

(b) (1.0) Qual a resposta do seu programa ao objectivo `fib(X, 21)`? Justifique a sua resposta.**Resposta:**

```
?- fib(X, 21).
ERROR: is/2: Arguments are not sufficiently instantiated
^ Exception: (8) _L134 is _G180-1
```

O operador `is` necessita que todas as variáveis da expressão a avaliar estejam instanciadas.