

# Paradigmas da Programação I

## MiECom (2º ano)

Exame de 2ª Época (Recurso)

Data: 21 de Fevereiro de 2007  
Hora: 09:30

Dispõe de 2:30 horas para realizar este exame

Leia as questões com toda a atenção  
e responda com calma e clareza em folha convencional

### Questão 1: Bases de Conhecimento

Considere a BC formada apenas por factos correspondentes aos predicados `fresco/6` (produto perecível que é comercializado no dia, não se armazena), `classe/1`, `unidade/1` e `fornecedor/3` que podia ser um extracto do programa Prolog que alguém desenvolveu para apoio à gestão da MDume.

```
% fresco(Ref, Nome, PrecUnit, Unidade, Classe, Proveniencia).
fresco(a1, cenouras, 1, quilo, legume, f1).
fresco(a2, grelos, 2, molho, legume, f1).
fresco(b11, salsa, '0.5', molho, salada, f1).
fresco(c21, leite, '1.5', litro, lacticinio, f2).
fresco(dda, truta, 6, peca, peixe, f3).

% classe(Nome).
classe(fruta).
classe(salada).
classe(legume).
classe(lacticinio).
classe(peixe).

% unidade(Nome).
unidade(quilo).
unidade(litro).
unidade(molho).
unidade(peca).
unidade(caixa).

% fornecedor(Codigo, Regiao, Contacto).
fornecedor(f1, minhho, 919191911).
fornecedor(f2, beira, 933332221).
fornecedor(f3, minhho, 966665554).
fornecedor(f4, douro, 918881117).
```

Responda, então, às alíneas seguintes:

- Uma base de conhecimento, em programação lógica, é um conjunto de *cláusulas de Horn*. Diga porque razão as cláusulas acima não são designadas por *regras* e explique qual o seu papel na teoria que é representada pelo programa Prolog.
- Escreva uma nova cláusula que produz automaticamente, sem requer a intervenção do utilizador, as respostas às questões seguintes

1. a região de todos os fornecedores de peixe?
2. o nome de todos os legumes ou saladas que custem menos de 10euros?

Diga como poderia obter a mesma resposta à 1ª questão sem ter necessidade de acrescentar nenhuma cláusula à BC, apenas usando as facilidades do Interpretador.

c) Observe as 2 seguintes questões

```
?- fornecedor( f, minho, _ ).
?- fornecedor( F, Minho, _ ).
```

e diga qual a diferença entre ambas; indique a resposta dada pelo Interpretador de Prolog a cada uma delas. Explique o que se alteraria se a 2ª questão passasse a ser: `fornecedor( F, Minho, 919876543 )`.

- d) Explique porque é que o programa baseado na BC acima teria um comportamento diferente se o facto `classe(fruta)` fosse removido da posição actual e passasse para depois do facto `classe(peixe)`.
- e) Recorrendo aos predicados standard do Prolog para gestão dinâmica da BC, escreva um predicado `actualFornec/2` que recebe o Código de um Fornecedor e um número de telemóvel e actualiza o telefone desse fornecedor; caso o fornecedor não exista na BC não faz nada.
- f) Escreva um predicado `valida/0` que verifica se o *fornecedor* de todos os *frescos* está correcto, isto é, se existe definido na BC através do predicado `fornecedor/3`. Cada anomalia detectada deve ser reportada.
- g) Escreva um predicado `limpa/0` que remove da BC todos os *frescos* cuja classe não é válida, isto é, não está definida pelo predicado `classe/1`.

## Questão 2: Manuseamento de Listas

Sobre operações com Listas em Prolog, e considerando `vinhos` uma lista de ternos (`região,tipo,lista de marcas`), que descreve a garrafeira de um clube de enologia, responda às alíneas seguintes:

- a) Implemente um predicado `roses/2` que, dados os `vinhos` do clube, devolva como segundo argumento a lista de todas as regiões onde essa variedade é produzida. Exemplo:

```
?- roses([(douro,tinto,[v1,v2]),(douro,branco,[v3,v4]),(douro,rose,[v5])], Rs).
Rs = [douro]
```

- b) Implemente um predicado `conta/3` que receba `vinhos`, como 1º argumento, e uma variedade (tinto, branco, ou rose), como 2º argumento, e devolva, como terceiro argumento, o número marcas diferentes dessa variedade. Exemplo:

```
?- conta([(douro,tinto,[v1,v2]),(dao,tinto,[v3,v4]),(algarve,tinto,[v5]),
          (algarve,branco[v5])], tinto, C).
C = 5
```

- c) Implemente um predicado `sumario/2` que, recebendo `vinhos` como 1º argumento, devolva como 2º argumento uma lista de pares (`região,nMarcas`) que indica para cada região o número de marcas diferentes (de qualquer variedade) que existem na garrafeira. Exemplo:

```
?- sumario([(douro,tinto,[v1,v2]),(douro,branco,[v3,v4]),(algarve,tinto,[v5]),
            (algarve,branco[v5])], L).
L = [(douro,4),(algarve,1)]
```

### Questão 3: Representação de Conhecimento

Usando a abordagem seguida nas aulas para modelar sistemas de informação em lógica, recorrendo à linguagem de programação Prolog para poder interrogar de seguida o seu universo de discurso, construa uma Base de Conhecimento (BC) que descreva a garrafeira (vinhos existentes, identificando a marca, variedade, região e ano de colheita), sócios e eventos do clube de enologia referido acima. Sobre os sócios importa saber, além dos dados de identificação (incluindo sexo e idade), a profissão e a variedade e regiões preferidas. Quanto aos eventos (palestras, cursos, ou visitas de estudo), além da descrição, quer-se registar a data, duração e responsável, local de realização, bem como a lista de regiões em apreço no evento e os sócios que participaram.

Após identificar o tipo de cláusulas (factos ou regras) que deve usar para modelar o sistema, indique algum tipo de perguntas que poderiam ser respondidas pela sua BC.

### Questão 4: Autómatos Deterministas Reactivos

Pretende-se desenvolver um autómato reactivo que modele o comportamento, que se descreve abaixo, da balança-calculadora (BiCalc) usada na MDume.

Implemente, então, em Prolog esse autómato determinista reactivo, que vá descrevendo as operações que a BiCalc irá executar à medida que recebe comandos/estímulos; escreva apenas os predicados `automato/7`, `deltalfa/5` e `acciaoSem/1`.

Uma vez ligada a balança, esta aceita apenas impulsos (1 ou mais) correspondentes ao peso do produto colocado no prato; 1 impulso por cada 10gramas. A sequência de impulsos de pesagem termina com um sinal de fim-de-pesagem (FdP). Atingido o fim da pesagem, a BiCalc aguarda o código do produto, o qual é uma sequência de comprimento fixo formada por uma letra e dois dígitos. Após a introdução do código já é possível calcular o preço a pagar graças a uma tabela de preços que a BiCalc tem na sua memória. Nesse estado, a balança pode recomeçar novo processo de pesagem que se vai acrescentar ao anterior, se voltar a receber impulsos de pesagem; ou terminar o processo, caso receba uma ordem de `emitir-talão`. Em qualquer estado, a BiCalc pode voltar ao estado inicial, perdendo tudo que está em memória, caso receba o comando `desligar`.