

---

---

## PARADIGMAS DE PROGRAMAÇÃO III

2004/2005 – Fevereiro (Exame de Recurso)  
LESI e LMCC - 2º ANO – 16 de Fevereiro de 2005  
Duração Máxima: 2h00

---

---

**OBS: Responda a cada uma das Partes em folhas separadas e bem identificadas.**

### PARTE I

#### 1.- LISTAS E RECURSIVIDADE (7 pontos)

1.1. – Escreva um predicado *elem(N, L, E)* que determine o elemento E que se encontra na posição N da lista L. Caso N não tenha um valor entre 1 e o comprimento de L o predicado deverá falhar.

1.2.- Dada uma lista contendo pares da forma *clube/pontos*, escreva um predicado que dê como resultado uma lista ordenada por ordem crescente de pares *nº pontos/lista de clubes com tal número de pontos* e ainda outra lista contendo o(s) nome(s) do(s) clube(s) em 1º lugar.

1.3. – Escreva o código de um predicado *flattenNew/2* que, dada uma lista de listas, devolve como resultado uma lista contendo todos os elementos das listas contidas na lista parâmetro mas sem elementos em duplicado (sem usar os predicados *sort* nem *list\_to\_set*).

#### 2.- Semântica do Prolog (2 valores)

2.1- Considere o seguinte programa Prolog abstracto:

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

Indique quais as respostas obtidas para os seguintes "queries":

- a) ?- p(X).
- b) ?- p(X), !, p(Y).
- c) ?- p(X), p(3).

## PARTE II

### Queries e Múltiplas Soluções (6 valores)

Defina predicados Prolog que permitam determinar, tendo por base factos *enviamail(X, ListaNomes)* em que *ListaNomes* não pode ser vazia mas pode conter 1 ou mais nomes, as respostas às seguintes questões:

- a) *Lista de quem envia mails ?*
- b) *Lista de quem recebe mails ?*
- c) *Quem enviou mais emails ?*
- d) *Quantos emails recebeu a pessoa P ?*
- d) *Ler de um ficheiro F mais factos enviamail/2 e juntá-los à BC, mas apenas emails de pessoas que já tenham enviado emails anteriormente (cf. a);*
- e) *Eliminar da BC todos os emails enviados por uma lista de pessoas dada como parâmetro;*

## PARTE III

### Grafos (5 valores)

Numa BC foram introduzidos predicados **dista(cid1, cid2, km)**, cf. por exemplo *dista(braga, porto, 60)*.

- a) *Dadas duas cidades determine o caminho mais curto entre ambas;*
- b) *Determine o nome da cidade mais distante da cidade C dada;*

*Prof. F. Mário Martins*