
PARADIGMAS DE PROGRAMAÇÃO III

2006/07 - Janeiro (2ª chamada)

LESI/LMCC - 2º Ano

31 de Janeiro de 2007

Duração Máxima: 2 horas

Obs: Responda a cada uma das partes em folhas separadas e bem identificadas (nome e nº).

PARTE I

1: LISTAS e RECURSIVIDADE. (6.5 pontos)

1.1.- Dada uma lista L contendo apenas dígitos 0 e 1, escreva:

- a) Um predicado *contaBits/2* que dê como resultado um par contendo, respectivamente, o número total de 0s e de 1s existentes na lista parâmetro.
- b) Um predicado *converteDec/2* que converta para um valor decimal os bits da lista, tendo em consideração que o bit mais significativo é o "head" da lista (sendo potência $\rightarrow a ** b$). Por exemplo, a lista [1,0,1,0] valeria 10.

1.2.- Dada uma lista L contendo listas escreva um predicado *tamanhos/2* que dê como resultado uma lista de pares Índice/Comprimento, onde cada par representa a posição e o comprimento de cada uma das listas contidas na lista parâmetro (cf. [1/4, 2/7, 3/10, ...]).

PARTE II

2: Queries e Múltiplas Soluções. (8 pontos)

Admitindo que existem numa base de conhecimento predicados sobre alunos com a estrutura *aluno(Numero, Curso, Media)* :

- a) Escreva um predicado *removeEntre/2* que permita remover da base de conhecimento todos os factos *aluno/3* para alunos com um número compreendido entre os valores N1 e N2 (sendo $N1 < N2$).
- b) Escreva um predicado *paresCursoMedia/1* que dê como resultado uma lista de pares Curso/Média contendo as médias das médias de todos os alunos para cada um dos cursos existentes.

c) Escreva um outro predicado *inserePorCurso/2* que leia de um ficheiro F factos *aluno* com a estrutura dada, e insira na base de conhecimento apenas os que pertencem a um dado Curso, tendo em atenção que não devem ser introduzidos duplicados.

d) Escreva um predicado *melhoresAlunos/1* que devolva uma lista com o(s) número(s) do(s) aluno(s) com melhor média.

PARTE III

4: Grafos. (5.5 pontos)

Numa BC existem factos **liga(cid1, cid2)**, indicando que existe uma ligação orientada directa entre a cidade cid1 e a cidade cid2.

a) A partir destes factos, escreva um predicado *criaLigações* que crie na BC os factos *viagemDirecta(cid1, listadeCidades)*, em que lista de cidades representa a lista dos códigos das cidades directamente ligadas a *cid1*. Estes factos devem ser criados para todas as cidades da BC. Uma lista de cidades vazia indicará que tal cidade não se liga a qualquer outra, ou seja, é terminal.

b) Dada uma cidade terminal Ct qualquer, escreva um predicado *daoAcesso/2*, que determine todas as cidades a partir das quais será possível chegar a Ct através de um qualquer percurso.