Métodos de Programação I

2.º Ano da LMCC (701055) + LESI (531316)
Ano Lectivo de 1998/99

Exame (época de especial) — 17 de Novembro de 1999
14h30

IDENTIFICAÇÃO DO ALUNO:
Nome: ....
Número: | | | | | | **Curso**: ....

RESERVADO À EQUIPA DOCENTE
O Vigilante: .... **O Docente que corrigiu a prova:** ....

NB: *pode utilizar o verso de cada folha deste enunciado para continuar as suas respostas às respectivas questões.*

PROVA SEM CONSULTA ($3\ horas$)

Questão 1 Uma *árvore de pedigree* ($\mathsf{P}a$) descreve um animal (a) — por exemplo, um canídeo — e indica quais os seus ascendentes conhecidos (o pai e/ou a mãe), o seu pedigree, e assim sucessivamente:

$$\mathsf{P}a \quad \cong \quad a \times (1 + \mathsf{P}a) \times (1 + \mathsf{P}a)$$

Codifique o tipo $\mathsf{P}a$ em HASKELL.

RESPOSTA:

Questão 2 Uma possível alternativa para a codificação da informação requerida na questão anterior seria:

```
data PedAlt x = N x
              | SPai (x,PedAlt x)
              | SMae (x,PedAlt x)
              | Ambos (x,PedAlt x,PedAlt x)
```

Codifique em `Halkell` as funções que estabelecem o isomorfismo entre ambos os tipos de dados.

RESPOSTA:

Questão 3 Por inferência de tipos, escolha a função que, de entre as seguintes,

$$[id, id] \tag{1}$$
$$[\langle \underline{V}, id\rangle, \langle \underline{F}, id\rangle] \tag{2}$$
$$[\langle \underline{V}, \underline{F}\rangle, id] \tag{3}$$
$$id + id \tag{4}$$

estabelece o isomorfismo

$$2 \times A \quad \cong \quad A + A$$

da direita para a esquerda.

Aplique-lhe a *lei da troca* e codifique o resultado em HASKELL.

RESPOSTA:

Questão 4 Para representar listas não vazias pode usar-se a seguinte definição

```
data NeList a = U a | Cons (a,NeList a)
```

1. Apresente os diagramas de {ana,cata,hilo}-morfismos para este tipo de dados.
2. Defina em HASKELL as usuais funções `inNeList`, `outNeList`, `ana`, `cata`, `hylo` e `rec`.
3. Usando as funções da alínea anterior, complete a seguinte definição

```
instance Functor NeList where
  ...
```

RESPOSTA:

Questão 5 Na sequência da Questão 4, responda às seguintes alíneas:

1. Defina como catamorfismos as funções `maxim` e `minim` que calculam, respectivamente, o maior e o menor elemento de uma lista não vazia.
2. Defina como um catamorfismo a função `maxmin :: Nelist a -> (a,a)` que calcula o maior e o menor elementos de uma lista não vazia.

RESPOSTA:

Questão 6 Na sequência da Questão 4, defina como um anamorfismo em `NeList` a função `inits :: [a] -> NeList [a]` que calcula os segmentos iniciais de uma lista.

RESPOSTA:

Questão 7 Podemos representar os números interios (não negativos) como listas do tipo singular (i.e. o tipo `[()]` do *Haskell*).

1. Defina em *Haskell* as funções de representação e abstracção

```
repres :: Int -> [()]
abstr :: [()] -> Int
```

2. Codifique funções que calculem a *soma* e *multiplicação* de inteiros utilizando **na representaão sugerida**.

RESPOSTA:

Questão 8 Demonstre a seguinte igualdade

$$[id \times i_1, id \times i_2] = \langle [\pi_1, \pi_1], \pi_2 + \pi_2 \rangle$$

Qual o isomorfismo que esta função estabelece?

RESPOSTA:

Questão 9 Seja $distr$ (ler: *'distribute right'*) a bijecção que estabelece o isomorfismo $A \times (B + C) \cong A \times B + A \times C$. Preencha as reticências no diagrama que se segue por forma a ver nele especificada a bijecção $distl$ (ler: *'distribute left'*) que establece o isomorfismo $(B + C) \times A \cong$

$B \times A + C \times A$:

$$
\begin{tikzcd}
(B+C) \times A \arrow[r, "swap"] \arrow[rrr, bend right, "distl"'] & \cdots \arrow[r, "distr"] & \cdots \arrow[r, "\cdots"] & B \times A + C \times A
\end{tikzcd}
$$

RESPOSTA:

Questão 10 Na sequência da Questão 9, mostre que

$$[g,h] \times f \quad = \quad [g \times f, h \times f] \cdot distl \tag{5}$$

é uma propriedade válida sobre $distl$, aplicando, entre outras leis que conhece, as seguintes:

$$f \times [g,h] \quad = \quad [f \times g, f \times h] \cdot distr \tag{6}$$

$$swap \cdot (f \times g) \quad = \quad (g \times f) \cdot swap \tag{7}$$

RESPOSTA: