

## Métodos de Programação I

2.º Ano de LESI (5303O7) + LMCC (7003N5)  
Ano Lectivo de 2005/06

Exame (1.ª chamada da época normal) — 10 de Janeiro 2006  
14h00  
Salas 2201 a 2210

---

**NB:** Esta prova consta de 8 alíneas que valem, cada uma, 2.5 valores. Utilize folhas de resposta diferentes para cada grupo.

PROVA SEM CONSULTA (2 horas)

GRUPO I

**Questão 1** Considere o seguinte isomorfismo.

$$(A + B) + C \cong A + (B + C)$$

Escreva a função `coassocr` que testemunha o isomorfismo da esquerda para a direita no estilo *pointwise*. Derive a versão *point-free* dessa função, acompanhando a definição obtida com o diagrama respectivo.

---

**Questão 2** Demonstre que `swap` é uma transformação natural, ou seja, que verifica a seguinte propriedade para quaisquer  $f$  e  $g$ .

$$(f \gg g) \cdot \text{swap} = \text{swap} \cdot (g \gg f)$$

Justifique todos os passos da demonstração usando às leis do cálculo estudado nesta disciplina.

---

**Questão 3** Considere o seguinte tipo de dados.

```
data LTree a = Leaf a | Fork (LTree a, LTree a)
```

Defina as funções `inLTree`, `outLTree`, `cataLTree` e `anaLTree` para este tipo de dados. Acompanhe as duas últimas definições com os diagramas respectivos.

---

GRUPO II

**Questão 4**

Considere a seguinte função definida sobre elementos do tipo `LTree`.

```
mirror (Leaf x) = Leaf x  
mirror (Fork (a,b)) = Fork (mirror b, mirror a)
```

Derive a seguinte versão no estilo *point-free*, justificando o melhor possível todos os passos da derivação (recorrendo, sempre que possível, às leis do cálculo estudado nesta disciplina).

```
mirror = cataLTree (inLTree . (id -|- swap))
```

---

**Questão 5** Utilizando a lei de fusão instanciada para `cataLTree`, a propriedade da comutatividade da soma (que em *point-free* pode ser expressa como `plus . swap = plus`), e outras do cálculo estudado nesta disciplina, demonstre o facto

```
nfolhas . mirror = nfolhas
```

onde `mirror` é o catamorfismo definido na questão anterior e `nfolhas` tem a seguinte definição.

```
nfolhas = cataLTree (either (const 1) plus)
```

---

**Questão 6** Uma das formas de calcular  $n^2$ , o quadrado de um número natural  $n$ , é somar os  $n$  primeiros ímpares. De facto,  $0^2 = 0$ ,  $1^2 = 1$ ,  $2^2 = 1 + 3$ ,  $3^2 = 1 + 3 + 5$ , etc. De acordo com esta sugestão, exprima a função `square` que determina o quadrado de um número sob a forma de um hilomorfismo de listas. Faça um diagrama ilustrativo identificando qual o tipo intermédio do hilomorfismo.

---

### GRUPO III

**Questão 7** Suponha que se pretende imprimir o conteúdo de uma árvore binária de procura prefixando cada elemento com o número de ordem em que este ocorre na travessia *in-order*. Assim,

```
1:3
2:5
3:8      (será o resultado para a árvore)
4:10
5:12
```

```
      8
     / \
    5  12
   /  /
  3  10
```

Para tal, optou-se por utilizar o monad de estado para registar o número de ordem à medida que se percorre a árvore. Complete o código apresentado para realizar a operação pedida (note a utilização de o *monad transformer* `StateT` para combinar o monad de estado com o `IO`).

```
data BTree a = Vazia | Nodo a (BTree a) (BTree a)
mostra :: (Show a) => BTree a -> IO ()
mostra t = evalStateT (mostraST t) 0
mostraST :: (Show a) => BinTree a -> StateT Int IO ()
...
```

---

**Questão 8** Complete a demonstração que se segue do facto

$$(F f) x = do \{ a \leftarrow x ; return(f a) \}$$

válido para toda a mónade  $FF$ :

$$\begin{aligned}
& do \{ a \leftarrow x ; return(f a) \} \\
= & \quad \{ \text{definição de } do \} \\
& x \gg = \lambda a. return(f a) \\
= & \quad \{ \dots(\text{justifique}) \dots \} \\
& x \gg = (return \cdot f) \\
= & \quad \{ \dots(\text{justifique}) \dots \} \\
& (\mu \cdot F(return \cdot f)) x \\
= & \quad \{ \dots(\text{justifique}) \dots \} \\
& (\mu \cdot (F return) \cdot (F f)) x \\
= & \quad \{ \dots(\text{justifique}) \dots \} \\
& (id \cdot (F f))x \\
= & \quad \{ \dots(\text{justifique}) \dots \} \\
& (F f)x
\end{aligned}$$


---