Cálculo de Programas

2.º Ano de LCC+MiEI (Universidade do Minho)
Ano Lectivo de 2020/21

Teste — 2 de Junho de 2021
15h30–17h30
Salas CP2-0.28, CP2-0.32 e Cantina

- *Esta prova consta de **8** questões que valem, cada uma, 2.5 valores. O tempo médio estimado para resolução de cada questão é de 15 min.*
- *Recomenda-se que os alunos leiam a prova antes de decidirem por que ordem querem responder às questões que são colocadas.*

PROVA PRESENCIAL INDIVIDUAL SEM CONSULTA (2h)

Questão 1 Mostre que a equação em θ

$$\theta \cdot \mathsf{distl} = [f\,,g] \times h \tag{E1}$$

só tem uma solução: $\theta = [f \times h\,, g \times h]$. **NB:** recorde que o isomorfismo distl tem $[i_1 \times id, i_2 \times id]$ como converso.

Questão 2 Identifique, apoiando a sua resolução num diagrama, qual é a definição da função polimórfica α cuja propriedade natural ("grátis") é:

$$(f + h) \cdot \alpha \quad = \quad \alpha \cdot (f + g \times h)$$

[1]Completar com as justificações.

Questão 3 Demonstre a 2ª-lei do combinador condicional de McCarthy:

$$(p \to f\ ,\ g) \cdot h \quad = \quad (p \cdot h) \to (f \cdot h), (g \cdot h)$$

Questão 4 Considere o isomorfismo de ordem superior *flip* definido pela composição de isomorfismos seguinte:

$$\begin{array}{ccccccc} (C^B)^A & \cong & C^{A \times B} & \cong & C^{B \times A} & \cong & (C^A)^B \\ f & \mapsto & \widehat{f} & \mapsto & \widehat{f}.\mathsf{swap} & \mapsto & \overline{\widehat{f} \cdot \mathsf{swap}} = \mathit{flip}\ f \end{array}$$

[2] Completar com as justificações.

Mostre que *flip*, acima definida por $\mathit{flip}\ f = \overline{\widehat{f} \cdot \mathsf{swap}}$, é um isomorfismo por ser a sua própria inversa, isto é, por

$$\mathit{flip}\ (\mathit{flip}\ f) = f \tag{E2}$$

se verificar.

Questão 5 Seja

$$\mathit{com}\ n\ p = \left(\begin{array}{c} n \\ p \end{array}\right)$$

a função que calcula o número de combinações de n objectos tomados p a p, para $n \geqslant p \geqslant 0$. Pode mostrar-se que:

$$\left\{\begin{array}{l} \mathit{com}\ n\ 0 = 1 \\ \mathit{com}\ n\ (p+1) = \frac{n-p}{p+1} \times (\mathit{com}\ n\ p) \end{array}\right. \tag{E3}$$

Mostre, por aplicação da lei de recursividade mútua, que a seguinte implementação da função *com*

$$\begin{array}{l} \mathit{com}\ n = \pi_2 \cdot (\mathit{aux}\ n)\ \mathbf{where} \\ \quad \mathit{aux}\ n = \mathsf{for}\ (\mathit{loop}\ n)\ (0,1) \\ \quad \mathit{loop}\ n\ (p,c) = (p+1, (n-p) * c \div (p+1)) \end{array}$$

é equivalente a (E3).

[3]Completar com as justificações.

Questão 6 Derive a versão *pointwise* do seguinte catamorfismo de BTrees,

$$
\begin{array}{l}
tar = (\!|\, [\mathsf{singl} \cdot \mathsf{nil}\, , g] \,|\!) \ \mathbf{where} \\
\quad g = \mathsf{map\ cons} \cdot lstr \cdot (id \times \mathsf{conc}) \\
\quad lstr\ (b, x) = [(b, a) \mid a \leftarrow x]
\end{array}
$$

entregando no final uma versão da função em que não ocorrem os nomes das funções map, cons, singl, nil, conc e $lstr$.
NB: recorda-se o tipo $\mathbf{data}\ \mathsf{BTree}\ a = Empty \mid Node\ (a, (\mathsf{BTree}\ a, \mathsf{BTree}\ a))$ tal como definido em Haskell, nas aulas, com base $\mathsf{B}\ (X, Y) = 1 + X \times Y^2$ e álgebra de construção $\mathsf{in} = [\underline{Empty}\, , Node]$. Recorda-se ainda $\mathsf{map}\ f\ x = [f\ a \mid a \leftarrow x]$ como definição *pointwise* de map em listas.

[4] Completar com as justificações.

□

Questão 7 Considere o catamorfismo

$$count\ p = (\![\,[(p \rightarrow \underline{1}\ ,\ \underline{0})\,,\mathsf{add}]\,]\!)\ \textbf{where}\ \mathsf{add} = \widehat{(+)} \tag{E4}$$

que conta as folhas de uma árvore de tipo LTree que satisfazem o predicado p e o isomorfismo:

$$\mathsf{mirror} = (\!|\,\mathsf{in}\cdot(id + \mathsf{swap})\,|\!) \tag{E5}$$

Mostre que $count\ p$ é invariante em relação a rotações da árvore, isto é, que

$$count\ p\cdot\mathsf{mirror} = count\ p \tag{E6}$$

NB: recorda-se que LTree X tem por base $\mathsf{B}\ (X, Y) = X + Y^2$ e $\mathsf{in} = [Leaf\ , Fork]$.

Questão 8 O seguinte programa em Haskell

$$pbin :: Ord\ p \Rightarrow p \rightarrow [p] \rightarrow \mathbb{B}$$
$$pbin\ a\ db = loop\ (0, \mathsf{length}\ db - 1)\ \textbf{where}$$

[5] Completar com as justificações.

$$
\begin{array}{l}
\mathit{loop}\ (l, r) \\
\quad \mid l > r \quad = \mathsf{False} \\
\quad \mid a < a' = \mathit{loop}\ (l, m - 1) \\
\quad \mid a > a' = \mathit{loop}\ (m + 1, r) \\
\quad \mid \mathsf{otherwise} = \mathsf{True} \quad \text{-- a=a'} \\
\quad \mathbf{where}\ m = (l + r) \div 2 \\
\qquad a' = db\ !!\ m
\end{array}
$$

implementa o algoritmo de pesquisa binária sobre uma lista ordenada. Mostre que $\mathit{loop} = [\![f, g]\!]$, identificando os dois genes f e g do hilomorfismo e o bifunctor do tipo indutivo intermédio. **Sugestão**: recorde o combinador **tailr**.

RESOLUÇÃO: Seguindo a sugestão dada de se tentar usar o combinador $\mathbf{tailr} = [\![[id, id], g]\!]$, fazendo $\mathit{loop} = \mathbf{tailr}\ d$, ter-se-á de imediato $f = [id, id]$. Quanto a g, a função surge quando re-escrevemos o código dado usando **tailr**:

$$
\begin{array}{l}
\mathit{pbin}\ a\ db = \mathit{loop}\ (0, \mathsf{length}\ db - 1)\ \mathbf{where} \\
\quad \mathit{loop} = \mathbf{tailr}\ g \\
\quad g\ (l, r) \\
\qquad \mid l > r \quad = i_1\ \mathsf{False} \\
\qquad \mid a < a' = i_2\ (l, m - 1) \\
\qquad \mid a > a' = i_2\ (m + 1, r) \\
\qquad \mid \mathsf{otherwise} = i_1\ \mathsf{True} \quad \text{-- a=a'} \\
\qquad \mathbf{where}\ m = (l + r) \div 2 \\
\qquad\quad a' = db\ !!\ m
\end{array}
$$

□