Cálculo de Programas

2.º Ano de LCC (Universidade do Minho)
Ano Lectivo de 2022/23

Teste — 2 de Junho de 2023, 10h00–12h00
Salas E1-0.04 + E1-0.20

PROVA PRESENCIAL INDIVIDUAL SEM CONSULTA (2h)

Importante — *Ler antes de iniciar a prova:*

- *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.*

Questão 1 Recorde o isomorfismo

$$\begin{tikzcd} Maybe\ B \arrow[rr, bend left, "\mathsf{out}=\mathsf{in}^{\circ}"] \arrow[rr, phantom, "\cong"] & & 1+B \arrow[ll, bend left, "\mathsf{in}=[\underline{\mathsf{Nothing}}\,,\mathsf{Just}]"] \end{tikzcd}$$

e considere a função:

$\mathsf{fromMaybe} :: a \rightarrow Maybe\ a \rightarrow a$
$\mathsf{fromMaybe}\ a = [\underline{a}\,, id] \cdot \mathsf{out}$

Derive a versão *pointwise* de fromMaybe por forma a não recorrer ao combinador de alternativa (vulg. 'either') de funções.

Questão 2 Suponha que apenas sabe a seguinte propriedade de uma dada função α,

$$\alpha \cdot \langle f, \langle g, h \rangle \rangle = \langle h, f \rangle \tag{E1}$$

válida para quaisquer f, g e h que a tipem correctamente.

Deduza a definição de α e, a partir do seu tipo mais geral, a respectiva propriedade *natural* (também chamada *grátis*) usando o habitual diagrama.

Questão 3 Considere a função:

$$x \ominus y = \mathbf{if}\ x \leqslant y\ \mathbf{then}\ 0\ \mathbf{else}\ 1 + x \ominus (y + 1)$$

Use o condicional de McCarthy para identificar o gene de $\widehat{\ominus} : \mathbb{N}_0 \times \mathbb{N}_0 \to \mathbb{N}_0$ escrita como um anamorfismo de naturais, fazendo o respectivo diagrama.

[1] Ver e.g. vídeo T9b, t=3.55 etc.

Questão 4 Considere o combinador $comb\ f$ definido por:

$$comb\ f = [id\,, f] \cdot (i_1 + i_2) \cdot f \tag{E2}$$

Mostre que o tipo mais geral de $comb$ é

$$comb : (C + B)^{A+B} \rightarrow (C + B)^{A+B}$$

e demonstre analiticamente que

$$comb\ id = id$$

Questão 5 Na estratégia algorítmica conhecida por *Google map-reduce* abordada nas aulas teóricas ocorre o catamorfismo de listas seguinte,

$$invert :: Eq\ a \Rightarrow [(a, String)] \rightarrow [(String, [a])]$$
$$invert = \llparenthesis\, [\mathsf{nil}\,, \widehat{col}] \cdot (id + (conv \cdot (id \times \mathsf{words})) \times id)\, \rrparenthesis$$

onde $\mathsf{words} : String \rightarrow String^*$ é a função que separa um *string* na lista das suas palavras.

Identifique os tipos X, Y e Z no diagrama abaixo e, assim, os das funções auxiliares $conv$ e col (cuja definição se omite). Justifique a sua resposta.

$$\begin{tikzcd}
(A \times String)^* \arrow[rrr, bend left, "\mathsf{out} = \mathsf{in}^\circ"] \arrow[d, "invert"'] & & & 1 + (A \times String) \times (A \times String)^* \arrow[lll, bend left, "\mathsf{in} = [\mathsf{singl}\,, \mathsf{cons}]"] \arrow[d, "id + id \times invert"] \arrow[lll, phantom, "\cong"] \\
(String \times A^*)^* & & & 1 + (A \times String) \times Z \arrow[dl, bend left, "id + (id \times \mathsf{words}) \times id"] \\
& 1 + X \times X \arrow[ul, bend left, "{[\mathsf{nil}\,, \widehat{col}]}"] & 1 + Y \times Z \arrow[l, "id + conv \times id"] &
\end{tikzcd}$$

Questão 6 Considere o catamorfismo $\mathsf{LTree}\ (A \times B) \xrightarrow{unzp} (\mathsf{LTree}\ A) \times (\mathsf{LTree}\ B)$ que divide uma árvore de pares num par de árvores

$$
\begin{array}{l}
unzp = (\!|\, \langle \mathsf{in}_1 \cdot (\mathsf{F}\ \pi_1), \mathsf{in}_2 \cdot (\mathsf{F}\ \pi_2) \rangle \,|\!)\ \mathbf{where} \\
\quad \mathsf{in}_1 = \mathsf{in} \cdot \mathsf{B}\ (\pi_1, id) \\
\quad \mathsf{in}_2 = \mathsf{in} \cdot \mathsf{B}\ (\pi_2, id)
\end{array}
$$

onde, como sabe, $\mathsf{B}\ (f, g) = f + g \times g$. Recorra a uma lei que conhece (e cujo nome é bastante sugestivo) para demonstrar a seguinte propriedade de cancelamento:

$$
\pi_1 \cdot unzp \quad = \quad \mathsf{LTree}\ \pi_1 \tag{E3}
$$

Questão 7 O conceito genérico de catamorfismo $(\!| g |\!)$ gerado pelo gene g é captado pela propriedade universal

$$k = (\!| g |\!) \quad \equiv \quad k \cdot in = g \cdot (\mathsf{F}\, k)$$

Mostre que:

$$(\!| f \cdot g |\!) \quad = \quad f \cdot (\!| g \cdot \mathsf{F}\, f |\!) \tag{E4}$$

Questão 8 Considere, definido em Haskell, o tipo

$\mathbf{data}$ $\mathsf{RTree}\ a = Ros\ a\ [\mathsf{RTree}\ a]$

das habitualmente designadas "rose trees", que tem bifunctor de base $\mathsf{B}\ (X, Y) = X \times Y^*$ e

$\mathsf{in} = \widehat{Ros}$
$\mathsf{out}\ (Ros\ a\ x) = (a, x)$

Considere $fmap\ f$ definida por

$$fmap\ f\ (Ros\ a\ xs) = Ros\ (f\ a)\ (\mathsf{map}\ (fmap\ f)\ xs) \tag{E5}$$

e mostre que $fmap\ f = (\!| g |\!)$ identificando g. Mostre ainda que esse catamorfismo se pode definir como um anamorfismo, calculando-o.

[2] Completar com as justificações.

$$
\begin{array}{ll}
& \mathit{fmap}\ f\ (\widehat{\mathit{Ros}}\ (a, xs)) = \widehat{\mathit{Ros}} \cdot (f \times \mathsf{map}\ (\mathit{fmap}\ f))\ (a, xs) \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{fmap}\ f \cdot \widehat{\mathit{Ros}} = \widehat{\mathit{Ros}} \cdot (f \times \mathsf{map}\ (\mathit{fmap}\ f)) \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{fmap}\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f \times id) \cdot (id \times \mathsf{map}\ (\mathit{fmap}\ f)) \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{fmap}\ f = (\!|\, \mathsf{in} \cdot (f \times id) \,|\!) \\
\Box &
\end{array}
$$

2ª parte:

$$
\begin{array}{ll}
& \mathit{fmap}\ f = (\!|\, \mathsf{in} \cdot (f \times id) \,|\!) \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{fmap}\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f \times id) \cdot (id \times \mathsf{map}\ (\mathit{fmap}\ f)) \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathsf{out} \cdot \mathit{fmap}\ f = (f \times id) \cdot (id \times \mathsf{map}\ (\mathit{fmap}\ f)) \cdot \mathsf{out} \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathsf{out} \cdot \mathit{fmap}\ f = (id \times \mathsf{map}\ (\mathit{fmap}\ f)) \cdot (f \times id) \cdot \mathsf{out} \\
\equiv & \qquad \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{fmap}\ f = [\!(\, (f \times id) \cdot \mathsf{out} \,)\!]
\end{array}
$$

$\Box$
