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

$$\mathit{Maybe}\ B \underset{\mathsf{in}=[\underline{\mathsf{Nothing}}\,,\mathsf{Just}]}{\overset{\mathsf{out}=\mathsf{in}^{\circ}}{\cong}} 1 + B$$

e considere a função:

$$\begin{array}{l} \mathsf{fromMaybe} :: a \rightarrow \mathit{Maybe}\ a \rightarrow a \\ \mathsf{fromMaybe}\ a = [\underline{a}\,, id] \cdot \mathsf{out} \end{array}$$

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

RESOLUÇÃO: Tem-se (justificar os passos com reticências):

$$\begin{array}{ll} & \mathsf{fromMaybe}\ a = [\underline{a}\,, id] \cdot \mathsf{out} \\ \equiv & \quad \{\ \mathsf{in}\ /\ \mathsf{out};\ \mathit{inMaybe}\ \} \\ & \mathsf{fromMaybe}\ a \cdot [\underline{\mathsf{Nothing}}\,, \mathsf{Just}] = [\underline{a}\,, id] \\ \equiv & \quad \{\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ \} \\ & \left\{ \begin{array}{l} \mathsf{fromMaybe}\ a \cdot \underline{\mathsf{Nothing}} = \underline{a} \\ \mathsf{fromMaybe}\ a \cdot \mathsf{Just} = id \end{array} \right. \\ \equiv & \quad \{\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ \} \\ & \left\{ \begin{array}{l} \mathsf{fromMaybe}\ a\ \mathsf{Nothing} = a \\ \mathsf{fromMaybe}\ a\ (\mathsf{Just}\ a') = a' \end{array} \right. \end{array}$$

□

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.

RESOLUÇÃO: A propriedade dar-nos-á uma definição de α desde que consigamos anular o termo $\langle f, \langle g, h\rangle\rangle$ em (E1), ou seja, reduzi-lo à identidade:

$$\begin{array}{ll} & \langle f, \langle g, h\rangle\rangle = id \\ \equiv & \quad \{ \text{ universal-}\times \text{ ou reflexão-}\times \text{ seguida de eq-}\times \ \} \\ & \left\{ \begin{array}{l} f = \pi_1 \\ \langle g, h\rangle = \pi_2 \end{array} \right. \\ \equiv & \quad \{ \text{ universal-}\times \ \} \\ & \left\{ \begin{array}{l} f = \pi_1 \\ \left\{ \begin{array}{l} g = \pi_1 \cdot \pi_2 \\ h = \pi_2 \cdot \pi_2 \end{array} \right. \end{array} \right. \end{array}$$

□

Logo, voltando a (E1) :

$$\begin{array}{ll} & \alpha \cdot id = \langle \pi_2 \cdot \pi_2, \pi_1\rangle \\ \equiv & \quad \{ \text{ natural-id } \} \\ & \alpha = \langle \pi_2 \cdot \pi_2, \pi_1\rangle \end{array}$$

Logo α tem tipo $B \times C \leftarrow C \times (A \times B)$ de que decorre, fazendo o diagrama habitual, a propriedade grátis

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

□

Questão 3 Considere a função:

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

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.

RESOLUÇÃO: Parte-se de $\widehat{\ominus}\ (x, y) = \textbf{if } x \leqslant y \textbf{ then } 0 \textbf{ else } 1 + \widehat{\ominus}\ (x, y + 1)$. Diagrama:

$$\begin{array}{ccc} \mathbb{N}_0 \times \mathbb{N}_0 & \xrightarrow{g} & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \\ {\scriptstyle \widehat{\ominus}}\downarrow & & \downarrow{\scriptstyle id+\widehat{\ominus}} \\ \mathbb{N}_0 & \xleftarrow[\text{in}]{} & 1 + \mathbb{N}_0 \end{array}$$

Pelo procedimento habitual,[1]

$$g\ (x, y) = \textbf{if } x \leqslant y \textbf{ then } i_1\ () \textbf{ else } i_2\ (x, y + 1)$$

[1] Ver e.g. vídeo T9b, t=3.55 etc.

Passando a *pointfree*, introduz-se o condicional,:

$$g = \widehat{(\leqslant)} \to i_1 \cdot !, i_2 \cdot (id \times \mathsf{succ})$$

ou seja:

$$g = \widehat{(\leqslant)} \to (! + id \times \mathsf{succ})$$

□

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

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

e demonstre analiticamente que

$$comb\ id = id$$

RESOLUÇÃO: Por absorção, $comb\ f = [i_1, f \cdot i_2] \cdot f$. Tipos iniciais:

$$f : X \to Z$$
$$i_2 : B \to A + B$$
$$i_1 : C \to C + D$$

Por $[i_1, f \cdot i_2]$ a saída de f tem de ser $C + D$ e a sua entrada tem de ser $A + B$. Logo tem-se

$f : A + B \to C + D$.

Mas o segundo f terá que ter saída $C + B$. Logo $D = B$ e $f : A + B \to C + B$.

O tipo de $comb\ f$ é o mesmo que o de f, pois a sua entrada e saída coincidem com as de f.

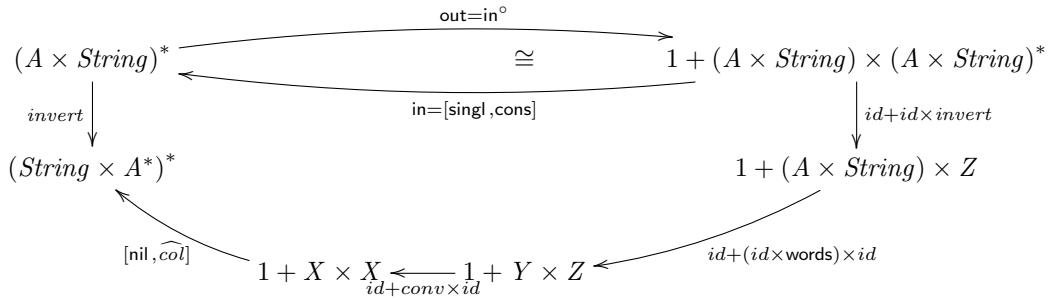
Finalmente, $comb\ id = [i_1, id \cdot i_2] \cdot id = [i_1, i_2] = id$ é imediato (apresentar justificações em casa). □

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

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

onde $\mathsf{words} : String \to String^*$ 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.

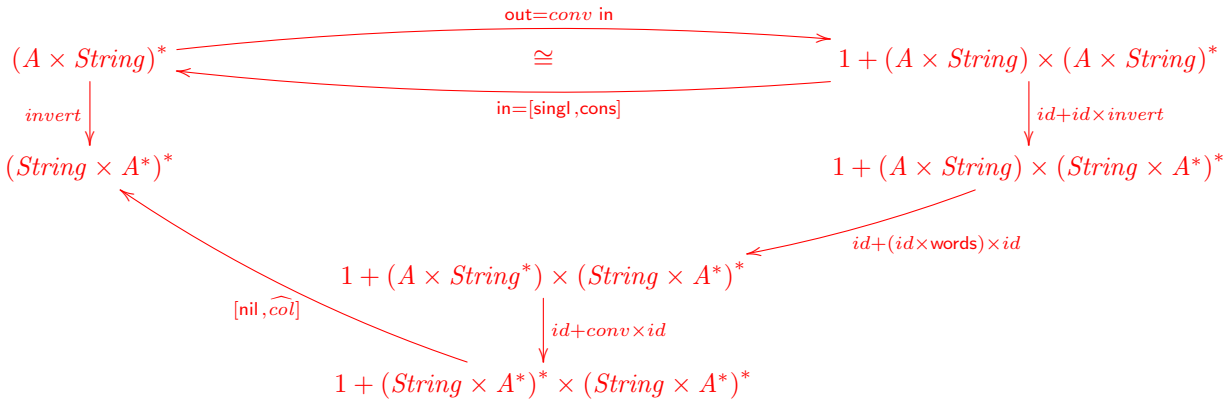


RESOLUÇÃO: Temos

$$X = (String \times A^*)^*$$
$$Y = A \times String^*$$
$$Z = X$$

$$conv : Y \to X$$
$$col : X \to X \to X$$

cf:



□

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

$$unzp = (\!|\ \langle \mathsf{in}_1 \cdot (\mathsf{F}\ \pi_1), \mathsf{in}_2 \cdot (\mathsf{F}\ \pi_2) \rangle\ |\!) \ \textbf{where}$$
$$\mathsf{in}_1 = \mathsf{in} \cdot \mathsf{B}\ (\pi_1, id)$$
$$\mathsf{in}_2 = \mathsf{in} \cdot \mathsf{B}\ (\pi_2, id)$$

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:

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

RESOLUÇÃO: A lei que é sugerida é conhecida por "banana-split". Investigar as justificações que faltam em

$$\begin{array}{ll} & \pi_1 \cdot unzp \\ = & \quad \{\ unzp = (\!|\ \langle \mathsf{in}_1 \cdot (\mathsf{F}\ \pi_1), \mathsf{in}_2 \cdot (\mathsf{F}\ \pi_2) \rangle\ |\!)\ \} \\ & \pi_1 \cdot (\!|\ \langle \mathsf{in}_1 \cdot (\mathsf{F}\ \pi_1), \mathsf{in}_2 \cdot (\mathsf{F}\ \pi_2) \rangle\ |\!) \\ = & \quad \{\ \ldots\ \} \\ & \pi_1 \cdot \langle (\!|\ \mathsf{in}_1\ |\!), (\!|\ \mathsf{in}_2\ |\!) \rangle \\ = & \quad \{\ \ldots\ \} \\ & (\!|\ \mathsf{in} \cdot \mathsf{B}\ (\pi_1, id)\ |\!) \\ = & \quad \{\ \ldots\ \} \\ & \mathsf{LTree}\ \pi_1 \end{array}$$

□

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:

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

RESOLUÇÃO: O termo $f \cdot (\!| g \cdot \mathsf{F}\, f |\!)$ sugere o recurso à fusão-cata:

$$\begin{array}{ll} & (\!| f \cdot g |\!) = f \cdot (\!| g \cdot \mathsf{F}\, f |\!) \\ \Leftarrow & \quad \{ \text{ fusão-cata (atenção ao } \Leftarrow\text{: vamos à procura de uma condição } \textbf{suficiente} \text{ para a fusão se dar) } \} \\ & f \cdot (g \cdot \mathsf{F}\, f) = (f \cdot g) \cdot \mathsf{F}\, f \\ \equiv & \quad \{ \text{ composição é associativa } \} \\ & \mathrm{TRUE} \end{array}$$

□

Questão 8 Considere, definido em Haskell, o tipo

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

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

$$\begin{array}{l} \mathsf{in} = \widehat{Ros} \\ \mathsf{out}\ (Ros\ a\ x) = (a, x) \end{array}$$

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.

RESOLUÇÃO: Vai ser preciso

$$\mathsf{B}\ (f, g) = f \times \mathsf{map}\ g \tag{E6}$$

$$\mathsf{F}\ f = \mathsf{B}\ (id, g) = id \times \mathsf{map}\ g \tag{E7}$$

Tem-se:[2]

$$\begin{array}{ll} & fmap\ f\ (Ros\ a\ xs) = Ros\ (f\ a)\ (\mathsf{map}\ (fmap\ f)\ xs) \\ \equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \end{array}$$

[2]Completar com as justificações.

$$fmap\ f\ (\widehat{Ros}\ (a, xs)) = \widehat{Ros} \cdot (f \times \mathsf{map}\ (fmap\ f))\ (a, xs)$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$fmap\ f \cdot \widehat{Ros} = \widehat{Ros} \cdot (f \times \mathsf{map}\ (fmap\ f))$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$fmap\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f \times id) \cdot (id \times \mathsf{map}\ (fmap\ f))$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$fmap\ f = (\!|\, \mathsf{in} \cdot (f \times id) \,|\!)$$

$\square$

2ª parte:

$$fmap\ f = (\!|\, \mathsf{in} \cdot (f \times id) \,|\!)$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$fmap\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f \times id) \cdot (id \times \mathsf{map}\ (fmap\ f))$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$\mathsf{out} \cdot fmap\ f = (f \times id) \cdot (id \times \mathsf{map}\ (fmap\ f)) \cdot \mathsf{out}$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$\mathsf{out} \cdot fmap\ f = (id \times \mathsf{map}\ (fmap\ f)) \cdot (f \times id) \cdot \mathsf{out}$$

$\equiv \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}$

$$fmap\ f = [\!(\, (f \times id) \cdot \mathsf{out} \,)\!]$$

$\square$
