Cálculo de Programas

2.º Ano de LCC+LEI (Universidade do Minho)
Ano Lectivo de 2012/13

Miniteste — 22 de Maio de 2013
16h00
Salas CP2 201, 202, 203 e 204

---

*Este miniteste consta de **6** questões que valem, cada uma, 2 valores. O tempo médio estimado para resolução de cada questão é de 15 min.*

PROVA SEM CONSULTA (90m)

Questão 1 Considere as funções seguintes:

$$f = [i_1 \cdot i_1 \ , i_2 + id]$$
$$g = [id + i_1 \ , i_2 \cdot i_2]$$

Identifique (justificadamente) os seus tipos e mostre que $f \cdot g = id$.

RESOLUÇÃO: Cálculo do tipo de f:

De $(A+B)+C \xleftarrow{i_1} A+B \xleftarrow{i_1} A$ e $(G+D)+E \xleftarrow{i_2+id} D+E$ infere-se, por unificação de $(A+B)+C$ com $(G+D)+E$ ('either's têm que ter o mesmo tipo de saída) $(A+B)+C \xleftarrow{i_2+id} B+C$ e, daí

$$(A+B)+C \xleftarrow{f} A+(B+C)$$

O tipo $g : (A+B)+C \to A+(B+C)$ infere-se de imediato de $f \cdot g = id$ (g e f são inversas), tal como se calcula a seguir (NB: preencher as justificações):

$$
\begin{array}{ll}
& f \cdot [id + i_1 \ , i_2 \cdot i_2] = id \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& [f \cdot (id + i_1) \ , f \cdot i_2 \cdot i_2] = [i_1 \ , i_2] \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \left\{ \begin{array}{l} f \cdot (id + i_1) = i_1 \\ f \cdot i_2 \cdot i_2 = i_2 \end{array} \right. \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \left\{ \begin{array}{l} [i_1 \cdot i_1 \ , i_2 + id] \cdot (id + i_1) = i_1 \\ [i_1 \cdot i_1 \ , i_2 + id] \cdot i_2 \cdot i_2 = i_2 \end{array} \right. \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \left\{ \begin{array}{l} [i_1 \cdot i_1 \ , i_1 \cdot i_2] = i_1 \\ (i_2 + id) \cdot i_2 = i_2 \end{array} \right. \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \left\{ \begin{array}{l} i_1 \cdot id = i_1 \\ i_2 \cdot id = i_2 \end{array} \right.
\end{array}
$$

□

Questão 2 Seja dada uma função δ da qual só sabe duas propriedades: $\pi_1 \cdot \delta = id$ e $\pi_2 \cdot \delta = id$. Mostre que, necessariamente, δ satisfaz também a propriedade natural $(f \times f) \cdot \delta = \delta \cdot f$.

RESOLUÇÃO: Tem-se:

$$
\begin{array}{ll}
 & \left\{ \begin{array}{l} \pi_1 \cdot \delta = id \\ \pi_2 \cdot \delta = id \end{array} \right. \\
\equiv & \qquad \{ \text{ universal-}\times \ \} \\
 & \delta = \langle id, id \rangle
\end{array}
$$

Então (preencher justificações):

$$
\begin{array}{ll}
 & (f \times f) \cdot \delta = \delta \cdot f \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
 & (f \times f) \cdot \langle id, id \rangle = \langle id, id \rangle \cdot f \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
 & \langle f, f \rangle = \langle f, f \rangle \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
 & f = f
\end{array}
$$

$\square$

Questão 3 Demonstre a seguinte propriedade do combinador condicional de McCarthy:

$$(f + g) \cdot (p \rightarrow i_1 \cdot h, i_2 \cdot k) \quad = \quad p \rightarrow i_1 \cdot f \cdot h, i_2 \cdot g \cdot k$$

RESOLUÇÃO: Tem-se (preencher justificações):

$$
\begin{array}{ll}
 & (f + g) \cdot (p \rightarrow i_1 \cdot h, i_2 \cdot k) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
 & p \rightarrow (f + g) \cdot i_1 \cdot h, (f + g) \cdot i_2 \cdot k \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
 & p \rightarrow i_1 \cdot f \cdot h, i_2 \cdot g \cdot k
\end{array}
$$

$\square$

Questão 4 A função $\pi_2 : A \times B \rightarrow B$ é binária e, como tal, faz sentido a sua versão "curried" $\overline{\pi_2} : A \rightarrow (B \rightarrow B)$. Usando as leis da exponenciação mostre que, qualquer que seja f,

$$\overline{\pi_2} \cdot f = \overline{\pi_2}$$

Logo $\overline{\pi_2}$ é uma função constante. Qual?

RESOLUÇÃO: Tem-se (preencher justificações):

$$
\begin{array}{ll}
& \overline{\pi_2} \cdot f \\
\equiv & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \overline{\pi_2 \cdot (f \times id)} \\
\equiv & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \overline{id \cdot \pi_2} \\
\equiv & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \overline{\pi_2}
\end{array}
$$

Por análise de tipos, teremos que $(\overline{\pi_2}\ \ a) : B \rightarrow B$, qualquer que seja a. Ora a única função que conseguimos polimorficamente garantir como resultado é $id : B \rightarrow B$. Logo, $\overline{\pi_2} = \underline{id}$. E. de facto: $\pi_2\ (a, b) = b$ é equivalente a $(\overline{\pi_2}\ a)\ b = b$, isto é: $\overline{\pi_2}\ a = \lambda b.b = id$. $\square$

Questão 5 A função factorial pode definir-se com recurso a uma função auxiliar:

$$
\begin{array}{l}
\mathit{fac}\ 0 = 1 \\
\mathit{fac}\ (n+1) = \mathit{fsuc}\ n \times \mathit{fac}\ n \\
\mathit{fsuc}\ 0 = 1 \\
\mathit{fsuc}\ (n+1) = \mathit{fsuc}\ n + 1
\end{array}
$$

Recorrendo à lei de recursividade múltipla (entre outras) derive desse par de funções a seguinte implementação de fac como um ciclo-for:

$$
\begin{array}{l}
\mathit{fac} = \pi_2 \cdot \mathit{facfor} \\
\mathit{facfor} = \mathsf{for}\ (\langle \mathsf{succ} \cdot \pi_1, \mathit{mul} \rangle)\ (1, 1)
\end{array}
$$

tal que $\mathit{fsuc} = \pi_1 \cdot \mathit{facfor}$ e onde $\mathsf{succ} = (1+)$ e $\mathit{mul}\ (n, m) = n \times m$. **NB:** Recorde que todo o ciclo-for é um catamorfismo de naturais: $\mathsf{for}\ f\ k = (\![\underline{k}\ , f]\!)$.

RESOLUÇÃO: Começamos, como habitualmente, por tirar variáveis às funções dadas, obtendo-se, respectivamente,

$$
\begin{array}{l}
\mathit{fac} \cdot \underline{0} = \underline{1} \\
\mathit{fac} \cdot \mathsf{succ} = \mathit{mul} \cdot \langle \mathit{fsuc}, \mathit{fac} \rangle \\
\mathit{fsuc} \cdot \underline{0} = \underline{1} \\
\mathit{fsuc} \cdot \mathsf{succ} = \mathsf{succ} \cdot \mathit{fsucc}
\end{array}
$$

que comprimem, por introdução de $\mathbf{in} = [\underline{0}\ , \mathsf{succ}]$, em

$$
\begin{array}{l}
\mathit{fsuc} \cdot \mathbf{in} = [\underline{1}\ , \mathsf{succ} \cdot \mathit{fsuc}] \\
\mathit{fac} \cdot \mathbf{in} = [\underline{1}\ , \mathit{mul} \cdot \langle \mathit{fsuc}, \mathit{fac} \rangle]
\end{array}
$$

Para que haja recursividade múltipla é preciso que ambas as funções envolvam o termo $\langle \mathit{fsuc}, \mathit{fac} \rangle$, o que — no caso de fsuc, precisa que $\mathsf{succ} \cdot \mathit{fsuc}$ dê lugar a $\mathsf{succ} \cdot \pi_1 \cdot \langle \mathit{fsuc}, \mathit{fac} \rangle$. Daí (preencher justificações):

$$
\begin{array}{ll}
& \left\{ \begin{array}{l} \mathit{fsuc} \cdot \mathbf{in} = [\underline{1}\ , \mathsf{succ} \cdot \pi_1 \cdot \langle \mathit{fsuc}, \mathit{fac} \rangle] \\ \mathit{fac} \cdot \mathbf{in} = [\underline{1}\ , \mathit{mul} \cdot \langle \mathit{fsuc}, \mathit{fac} \rangle] \end{array} \right. \\
\equiv & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \}
\end{array}
$$

$$\begin{cases} fsuc \cdot \mathbf{in} = [\underline{1}\ , \mathsf{succ} \cdot \pi_1] \cdot (id + \langle fsuc, fac \rangle) \\ fac \cdot \mathbf{in} = [\underline{1}\ , mul] \cdot (id + \langle fsuc, fac \rangle) \end{cases}$$

$\equiv$ { .... }

$\langle fsuc, fac \rangle = (\!|\langle [\underline{1}\ , \mathsf{succ} \cdot \pi_1], [\underline{1}\ , mul] \rangle|\!)$

$\equiv$ { .... }

$\langle fsuc, fac \rangle = (\!|[\langle \underline{1}, \underline{1} \rangle\ , \langle \mathsf{succ} \cdot \pi_1, mul \rangle]|\!)$

$\equiv$ { .... }

$\langle fsuc, fac \rangle = \mathsf{for}\ (\langle \mathsf{succ} \cdot \pi_1, mul \rangle)\ (1, 1)$

Fazendo $facfor = \langle fsuc, fac \rangle$, ter-se-á $fac = \pi_2 \cdot facfor$ e $facfor = \mathsf{for}\ (\langle \mathsf{succ} \cdot \pi_1, mul \rangle)\ (1, 1)$.

□

Questão 6 Quem programou a função seguinte

$$\begin{array}{l} f :: \mathsf{LTree}\ a \to \mathbb{N}_0 \\ f\ (Leaf\ \ a) = 0 \\ f\ (Fork\ (t, t')) = max\ (f\ \ t, f\ \ t') \end{array}$$

deve ter-se enganado: f dará sempre 0 como resultado. Identifique o gene g do catamorfismo $f = (\!|g|\!)$ e mostre que, de facto, $(\!|g|\!) = \underline{0}$.

RESOLUÇÃO: Da função dada inferimos, removendo variáveis, $f \cdot Leaf = \underline{0}$ e $f \cdot Fork = max \cdot (f \times f)$, isto é:

$f \cdot \mathbf{in} = [\underline{0}\ , max] \cdot (id + f \times f)$

$\equiv$ { universal-cata (para $\mathsf{F}\, f = id + f \times f$) }

$f = (\!|[\underline{0}\ , max]|\!)$

isto é, o gene de f é $g = [\underline{0}\ , max]$. Então (preencher justificações):

$(\!|g|\!) = \underline{0}$

$\equiv$ { .... }

$\underline{0} \cdot \mathbf{in} = [\underline{0}\ , max] \cdot (id + \underline{0} \times \underline{0})$

$\equiv$ { .... }

$\underline{0} \cdot \mathbf{in} = [\underline{0}\ , max \cdot (\underline{0} \times \underline{0})]$

$\equiv$ { .... }

$[\underline{0}\ , \underline{0}] = [\underline{0}\ , max \cdot \underline{(0, 0)}]$

$\equiv$ { .... }

$\underline{0} = \underline{(max\ (0, 0))}$

$\equiv$ { .... }

$\underline{0} = \underline{0}$

□
