Cálculo de Programas

2.º Ano de LCC+LEI (Universidade do Minho)
Ano Lectivo de 2013/14

Exame de Recurso — 4 de Julho de 2014
09h00
Salas CP2 201, 202, 203 e 204

---

*Este teste consta de **10** questões que valem, cada uma, 2 valores. O tempo médio estimado para resolução de cada questão é de 15 min.*

PROVA SEM CONSULTA (2h30m)

Questão 1 Qualquer programador de C sabe que, ao definir uma `struct` de apontadores $(A+1)\times(B+1)$, acaba por ter um tipo de dados que pode representar alternativamente o produto $A\times B$ (`struct`), o co-produto $A+B$ (`union`) ou nada (`void`) — tal como se mostra na sequência de composições de isomorfismos:

$$
\begin{tikzcd}
(A+1)\times(B+1) & ((A+1)\times B)+((A+1)\times 1) \arrow[l, "\mathsf{undistr}"'] & (A\times B+1\times B)+(A\times 1+1\times 1) \arrow[l, "\mathsf{iso}_2"'] \\
A\times B+((B+A)+1) \arrow[r, "\mathsf{iso}_5"'] & A\times B+(B+(A+1)) \arrow[r, "\mathsf{coassocl}"'] & (A\times B+B)+(A+1) \arrow[u, "(id+bl)+(br+br)"']
\end{tikzcd}
$$

Apresente, em notação *pointfree*, as definições dos ismorfismos iso_2 e iso_5 que completam o diagrama, bem como as das funções br e bl que participam no terceiro passo da transformação.

RESOLUÇÃO:

$bl = \langle !, id\rangle$

$br = \langle id, !\rangle$

$iso2 = \mathsf{undistl} + \mathsf{undistl}$
  where $\mathsf{undistl} = [i_1 \times id\ , i_2 \times id]$

$iso5 = id + \mathsf{coassocr}$
  where $\mathsf{coassocr} = [id + i_1\ , i_2 \cdot i_2]$

□

Questão 2 Sabendo que uma dada função rot satisfaz a propriedade

$$[[f\ ,g]\ ,h]\cdot \mathsf{rot} = [[h\ ,f]\ ,g] \qquad \text{(E1)}$$

quaisquer que sejam f, g e h,

- complete o diagrama genérico

$$
\begin{tikzcd}
\ldots \arrow[d, "{[[f\ ,g]\ ,h]}"'] & \ldots \arrow[l, "\mathsf{rot}"'] \arrow[dl, "{[[h\ ,f]\ ,g]}"] \\
\ldots &
\end{tikzcd}
$$

  que representa (E1) e

- formule a propriedade *natural* (i.é, *grátis*) de rot.

RESOLUÇÃO: Sejam $f : A \to B$, $g : C \to D$, $h : E \to G$ os tipos das funções do diagrama. De imediato se vê que $B = D = G$ pois 'eithers' têm o mesmo tipo de saída e, assim:

$$[[f\ ,g]\ ,h] : ((A + C) + E \to B$$
$$[[h\ ,f]\ ,g] : ((E + A) + C \to B$$

No diagrama, ter-se-á

$$\begin{tikzcd}
(A + C) + E \arrow[d, "{[[f\ ,g]\ ,h]}"'] & (E + A) + C \arrow[l, "\mathsf{rot}"'] \arrow[dl, "{[[h\ ,f]\ ,g]}"] \\
B &
\end{tikzcd}$$

de que sai a propriedade natural

$$((f + g) + h) \cdot \mathsf{rot} = \mathsf{rot} \cdot ((h + f) + g)$$

NB: é fácil de ver que

$$*\ Main > \mathbf{let}\ \mathsf{rot} = \mathsf{coswap} \cdot (\mathsf{coassocr})$$
$$*\ Main > :t\ \mathsf{rot}$$
$$\mathsf{rot} :: (a + b) + c \to (b + c) + a$$

□

Questão 3 Demonstre a seguinte propriedade do combinador condicional de McCarthy

$$f \times (p \to g, h) \quad = \quad p \cdot \pi_2 \to f \times g, f \times h \tag{E2}$$

sabendo que a igualdade

$$\langle f, (p \to q, h)\rangle \quad = \quad p \to \langle f, q\rangle, \langle f, h\rangle \tag{E3}$$

se verifica.

RESOLUÇÃO: Propõe-se (completar as justificações):

$$\begin{array}{ll}
& f \times (p \to g, h) \\
= & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \langle f \cdot \pi_1, (p \to g, h) \cdot \pi_2\rangle \\
= & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& \langle f \cdot \pi_1, p \cdot \pi_2 \to g \cdot \pi_2, h \cdot \pi_2\rangle \\
= & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& p \cdot \pi_2 \to \langle f \cdot \pi_1, g \cdot \pi_2\rangle, \langle f \cdot \pi_1, h \cdot \pi_2\rangle \\
= & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& p \cdot \pi_2 \to f \times g, f \times h \\
\Box &
\end{array}$$

□

Questão 4 A nova linguagem da Apple — SWIFT — é inspirada no Haskell. Em particular, presta atenção a um isomorfismo célebre que conhece

$$
\begin{tikzcd}
A \times B \to C \arrow[r, bend left, "\mathsf{curry}"] \arrow[r, phantom, "\cong"] & A \to C^B \arrow[l, bend left, "\mathsf{uncurry}"]
\end{tikzcd}
$$

e que permite converter funções binárias (eg. $f : A \times B \to C$) em funções de ordem superior (eg. $\mathsf{curry}\ f : A \to C^B$). Mostre que a igualdade

$$\overline{f}\ a = f \cdot \langle \underline{a}, id \rangle \tag{E4}$$

se verifica, onde $\overline{f}$ abrevia $\mathsf{curry}\ f$.

RESOLUÇÃO: Propõe-se (justificar cada passo):

$$
\begin{array}{cl}
& f \cdot \langle \underline{a}, id \rangle \\
= & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{ap} \cdot (\overline{f} \times id) \cdot \langle \underline{a}, id \rangle \\
= & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{ap} \cdot \langle \overline{f} \cdot \underline{a}, id \rangle \\
= & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{ap} \cdot \langle \underline{\overline{f}\ a}, id \rangle \\
= & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \overline{f}\ a
\end{array}
$$

$\square$

Versão mais *pointwise*:

$$
\begin{array}{cl}
& \overline{f}\ a = f \cdot \langle \underline{a}, id \rangle \\
\equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \overline{f}\ a\ b = (f \cdot \langle \underline{a}, id \rangle)\ b \\
\equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \overline{f}\ a\ b = f\ (\langle \underline{a}, id \rangle\ b) \\
\equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \overline{f}\ a\ b = f\ (a, b) \\
\equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& (\mathsf{ap} \cdot (\overline{f} \times id))\ (a, b) = f\ (a, b) \\
\equiv & \quad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{ap} \cdot (\overline{f} \times id) = f
\end{array}
$$

$\square$

$\square$

Questão 5 Seja dado um tipo indutivo T com base B, isto é,

$$\mathsf{T}\, f = (\!| \, \mathbf{in} \cdot \mathsf{B}\,(f, id) \, |\!)$$

Defina-se agora o chamado *combinador triangular* de T, $tri\ f$, tal como se segue:

$$tri\ f = (\!| \, \mathbf{in} \cdot \mathsf{B}\,(id, \mathsf{T}\, f) \, |\!)$$

Mostre que, para o caso das listas, se tem

$$\begin{array}{l} tri\ f\ [\,] = [\,] \\ tri\ f\ (h : t) = h : (\mathsf{map}\ f\ (tri\ f\ t)) \end{array}$$

com tipo $tri :: (a \to a) \to [\,a\,] \to [\,a\,]$, e calcule o resultado da expressão

$$tri\ \mathsf{succ}\ [1\,..\,5] \tag{E5}$$

NB: recorda-se que $\mathsf{B}\,(f, g) = id + f \times g$ em catamorfismos de listas.

RESOLUÇÃO: Para listas tem-se

$$\begin{array}{l} \mathsf{F}\, f = \mathsf{B}\,(id, f) = id + id \times f \\ \mathbf{in} = [\mathsf{nil}\,, \mathsf{cons}] \\ \mathsf{T}\, f = \mathsf{map}\, f \end{array}$$

Logo:

$$\begin{array}{cl}
 & tri\ f = (\!| \, \mathbf{in} \cdot \mathsf{B}\,(id, \mathsf{T}\, f) \, |\!) \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & tri\ f \cdot \mathbf{in} = \mathbf{in} \cdot \mathsf{B}\,(id, \mathsf{T}\, f) \cdot \mathsf{F}\,(tri\ f) \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & tri\ f \cdot \mathbf{in} = \mathbf{in} \cdot (id + id \times \mathsf{T}\, f) \cdot (id + id \times tri\ f) \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & tri\ f \cdot \mathbf{in} = \mathbf{in} \cdot (id + id \times ((\mathsf{map}\ f) \cdot tri\ f) \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & tri\ f \cdot [\mathsf{nil}\,, \mathsf{cons}] = [\mathsf{nil}\,, \mathsf{cons} \cdot (id \times ((\mathsf{map}\ f) \cdot tri\ f))] \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & \left\{ \begin{array}{l} tri\ f\ [\,] = [\,] \\ tri\ f \cdot \mathsf{cons} = \mathsf{cons} \cdot (id \times ((\mathsf{map}\ f) \cdot tri\ f)) \end{array} \right. \\
\equiv & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & \left\{ \begin{array}{l} tri\ f\ [\,] = [\,] \\ tri\ f\ (h : t) = h : (\mathsf{map}\ f\ (tri\ f\ t)) \end{array} \right. \\
\Box &
\end{array}$$

Cálculo pedido:

$$\begin{array}{cl}
 & tri\ \mathsf{succ}\ [1\,..\,5] \\
= & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & 1 : (\mathsf{map}\ \mathsf{succ}\ (tri\ \mathsf{succ}\ [2\,..\,5])) \\
= & \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
 & 1 : (\mathsf{map}\ \mathsf{succ}\ (2 : \mathsf{map}\ \mathsf{succ}\ (tri\ \mathsf{succ}\ [3\,..\,5])))
\end{array}$$

$$
\begin{array}{ll}
= & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& 1 : (\mathsf{map\ succ}\ (2 : \mathsf{map\ succ}\ (3 : \mathsf{map\ succ}\ (\mathit{tri}\ \mathsf{succ}\ [4\mathinner{.\,.}5])))) \\
= & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& 1 : (\mathsf{map\ succ}\ (2 : \mathsf{map\ succ}\ (3 : \mathsf{map\ succ}\ (4 : \mathsf{map\ succ}\ (\mathit{tri}\ \mathsf{succ}\ [5]))))) \\
= & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& 1 : (\mathsf{map\ succ}\ (2 : \mathsf{map\ succ}\ (3 : \mathsf{map\ succ}\ (4 : [6])))) \\
= & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& 1 : (\mathsf{map\ succ}\ [2, 4, 6, 8]) \\
= & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& [1, 3, 5, 7, 9]
\end{array}
$$

□

□

Questão 6 Se uma lista tiver pelo menos um elemento a, então para contar os seus elementos (length) basta contar os da cauda, começando a contagem em 1 e não 0. Sendo $\mathsf{length} = (\![\mathsf{zero}\ , \mathsf{succ} \cdot \pi_2]\!)$ ter-se-á:

$$\mathsf{length} \cdot (a:) = (\![\mathsf{one}\ , \mathsf{succ} \cdot \pi_2]\!) \tag{E6}$$

Demonstre (E6) sabendo que

$$\mathsf{length} \cdot (a:) = \mathsf{succ} \cdot \mathsf{length} \tag{E7}$$

se verifica. (**NB:** assuma $\mathsf{zero}\ \_ = 0$ e $\mathsf{one}\ \_ = 1$.)

RESOLUÇÃO: Tem-se (a justificar):

$$
\begin{array}{ll}
& \mathsf{length} \cdot (a:) = (\![\mathsf{one}\ , \mathsf{succ} \cdot \pi_2]\!) \\
\equiv & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathsf{succ} \cdot (\![\mathsf{zero}\ , \mathsf{succ} \cdot \pi_2]\!) = (\![\mathsf{one}\ , \mathsf{succ} \cdot \pi_2]\!) \\
\Leftarrow & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathsf{succ} \cdot [\mathsf{zero}\ , \mathsf{succ} \cdot \pi_2] = [\mathsf{one}\ , \mathsf{succ} \cdot \pi_2] \cdot (id + id \times \mathsf{succ}) \\
\equiv & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \left\{ \begin{array}{l} \mathsf{succ} \cdot \mathsf{zero} = \mathsf{one} \\ (\mathsf{succ} \cdot \mathsf{succ} \cdot \pi_2) = \mathsf{succ} \cdot \pi_2 \cdot (id \times \mathsf{succ}) \end{array} \right. \\
\equiv & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \left\{ \begin{array}{l} \mathit{true} \\ (\mathsf{succ} \cdot \mathsf{succ} \cdot \pi_2) = \mathsf{succ} \cdot \mathsf{succ} \cdot \pi_2 \end{array} \right. \\
\equiv & \{ \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \quad \} \\
& \mathit{true}
\end{array}
$$

□

□

Questão 7 O algoritmo de subtracção nos números naturais

$$\begin{array}{l} m \ominus n \\ \quad \mid m \leqslant n = 0 \\ \quad \mid \mathit{otherwise} = 1 + ((m-1) \ominus n) \end{array}$$

tem de ser truncado a 0 para evitar números negativos, por exemplo $7 \ominus 5 = 2$ mas $5 \ominus 7 = 0$. É, contudo, de esperar que a propriedade $(n+m) \ominus n = m$ permaneça válida, isto é,

$$(\ominus n) \cdot (n+) = id \tag{E8}$$

Sabendo que $(\ominus n)$ se pode escrever como o anamorfismo de naturais

$$(\ominus n) = [\![g\ n]\!] \ \textbf{where}\ g\ n = (\leqslant n) \rightarrow (i_1 \cdot !), (i_2 \cdot \mathit{pred}) \tag{E9}$$

onde $\mathit{pred}\ n = n - 1$ (para $n > 0$) apresente justificações para os passos da seguinte prova de (E8):

$$\begin{array}{ll}
& (\ominus n) \cdot (n+) = 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 \ \} \\
& [\![g\ n]\!] \cdot (n+) = [\![\mathsf{out}]\!] \\
\Leftarrow & \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 \ \} \\
& (g\ n) \cdot (n+) \cdot \mathbf{in} = id + (n+) \\
\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 \ \} \\
& (g\ n) \cdot [\underline{n}\ , \mathsf{succ} \cdot (n+)] = id + (n+) \\
\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} (g\ n) \cdot \underline{n} = i_1 \\ (g\ n) \cdot \mathsf{succ} \cdot (n+) = i_2 \cdot (n+) \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} (\leqslant n) \cdot \underline{n} \rightarrow (i_1 \cdot !), (i_2 \cdot \mathit{pred} \cdot \underline{n}) = i_1 \cdot ! \\ ((\leqslant n) \cdot \mathsf{succ} \rightarrow (i_1 \cdot !), i_2) \cdot (n+) = i_2 \cdot (n+) \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} i_1 \cdot ! = i_1 \cdot ! \\ ((\leqslant n) \cdot \mathsf{succ} \rightarrow (i_1 \cdot !), i_2) \cdot (n+) = i_2 \cdot (n+) \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} \mathit{true} \\ (\leqslant n) \cdot ((n+1)+) \rightarrow (i_1 \cdot !), (i_2 \cdot (n+)) = i_2 \cdot (n+) \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 \ \} \\
& i_2 \cdot (n+) = i_2 \cdot (n+) \\
\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 \ \} \\
& \mathit{true}
\end{array}$$

RESOLUÇÃO: Tem-se:

$$\begin{array}{ll}
& (\ominus n) \cdot (n+) = id \\
\equiv & \quad \{ \ \text{def } (\ominus n) \text{ (E9) ; reflexão-ana (47)} \ \} \\
& [\![g\ n]\!] \cdot (n+) = [\![\mathsf{out}]\!]
\end{array}$$

$\Leftarrow \qquad \{ \text{ fusão-ana (48) ; } (\mathsf{in}_{\mathbb{N}_0})^\circ = \mathsf{out}_{\mathbb{N}_0} \ \}$

$(g\ n) \cdot (n+) \cdot \mathbf{in} = id + (n+)$

$\equiv \qquad \{ \ \mathbf{in} = [\mathsf{zero}\ , \mathsf{succ}] \text{ ; fusão-+ (20) ;} n + 0 = n \text{ ; } n + (x+1) = 1 + (n+x) \ \}$

$(g\ n) \cdot [\underline{n}\ , \mathsf{succ} \cdot (n+)] = id + (n+)$

$\equiv \qquad \{ \text{ fusão-+ (20) ; def-+ (22) ; universal-+ (17) ou Eq-+ (27) } \}$

$$\begin{cases} (g\ n) \cdot \underline{n} = i_1 \\ (g\ n) \cdot \mathsf{succ} \cdot (n+) = i_2 \cdot (n+) \end{cases}$$

$\equiv \qquad \{ \text{ def } g\ n \text{ (E9) ; fusão-McCarthy (56) ; } ! \cdot k = ! \text{ (3) ; } 1 \xleftarrow{\ !\ } 1 = id \ \}$

$$\begin{cases} (\leqslant n) \cdot \underline{n} \to (i_1 \cdot !), (i_2 \cdot pred \cdot \underline{n}) = i_1 \cdot ! \\ ((\leqslant n) \cdot \mathsf{succ} \to (i_1 \cdot !), i_2) \cdot (n+) = i_2 \cdot (n+) \end{cases}$$

$\equiv \qquad \{ \ n \leqslant n = true ; true \to f, g = f \ \}$

$$\begin{cases} i_1 \cdot ! = i_1 \cdot ! \\ ((\leqslant n) \cdot \mathsf{succ} \to (i_1 \cdot !), i_2) \cdot (n+) = i_2 \cdot (n+) \end{cases}$$

$\equiv \qquad \{ \text{ fusão-McCarthy ; } \mathsf{succ}\ n = n + 1 \text{ ; } ! \cdot k = ! \ \}$

$$\begin{cases} true \\ (\leqslant n) \cdot ((n+1)+) \to (i_1 \cdot !), (i_2 \cdot (n+)) = i_2 \cdot (n+) \end{cases}$$

$\equiv \qquad \{ \ (n+1) + x \leqslant n = false \text{ em } \mathbb{N}_0 ; false \to f, g = f \ \}$

$i_2 \cdot (n+) = i_2 \cdot (n+)$

$\equiv \qquad \{ \text{ toda a função é igual a si própria } \}$

$true$

$\square$

$\square$

Questão 8 A função

$$\begin{aligned} &\mathsf{mirror}\ (Leaf\ a) = Leaf\ a \\ &\mathsf{mirror}\ (Fork\ (x, y)) = Fork\ (\mathsf{mirror}\ y, \mathsf{mirror}\ x) \end{aligned}$$

que "espelha" árvores binárias do tipo

$$\mathbf{data}\ \mathsf{LTree}\ a = Leaf\ a \mid Fork\ (\mathsf{LTree}\ a, \mathsf{LTree}\ a)$$

pode definir-se como o catamorfismo

$$\mathsf{mirror} = (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) \tag{E10}$$

Demonstre, por fusão e absorção cata, a propriedade natural de mirror:

$$\mathsf{LTree}\ f \cdot \mathsf{mirror} = \mathsf{mirror} \cdot (\mathsf{LTree}\ f) \tag{E11}$$

onde, como sabe,

$$\mathsf{LTree}\ f = (\!|\mathsf{in} \cdot (f + id)|\!) \tag{E12}$$

RESOLUÇÃO: Tem-se (a justificar):

$$
\begin{array}{ll}
& \mathsf{LTree}\ f \cdot \mathsf{mirror} = \mathsf{mirror} \cdot (\mathsf{LTree}\ f) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot \mathsf{mirror} = (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) \cdot (\mathsf{LTree}\ f) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot \mathsf{mirror} = (\!|\mathsf{in} \cdot (id + \mathsf{swap}) \cdot (f + id)|\!) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) = (\!|\mathsf{in} \cdot (f + id) \cdot (id + \mathsf{swap})|\!) \\
\Leftarrow & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot \mathsf{in} \cdot (id + \mathsf{swap}) = \mathsf{in} \cdot (f + id) \cdot (id + \mathsf{swap}) \cdot (id + ((\mathsf{LTree}\ f) \times (\mathsf{LTree}\ f))) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f + id) \cdot (id + \mathsf{swap}) \cdot (id + ((\mathsf{LTree}\ f) \times (\mathsf{LTree}\ f))) \cdot (id + \mathsf{swap}) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathsf{LTree}\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f + id) \cdot (id + ((\mathsf{LTree}\ f) \times (\mathsf{LTree}\ f))) \cdot (id + \mathsf{swap}) \cdot (id + \mathsf{swap}) \\
\equiv & \qquad \{ \ id + \mathsf{swap} \text{ é isomorfismo e o seu próprio converso } \} \\
& \mathsf{LTree}\ f \cdot \mathsf{in} = \mathsf{in} \cdot (f + id) \cdot (id + ((\mathsf{LTree}\ f) \times (\mathsf{LTree}\ f))) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \mathit{true} \\
\Box &
\end{array}
$$

$\Box$

Questão 9 O seguinte catamorfismo de números naturais

$$f = (\!|[\langle \underline{1}, \underline{1} \rangle\ , \langle \mathsf{succ} \cdot \pi_1, \mathsf{mul} \rangle]|\!) \tag{E13}$$

pode, pela lei da recursividade múltipla, ser desdobrado em duas funções mutuamente recursivas f_1 e f_2,

$$
\begin{cases} f_1\ 0 = 1 \\ f_1\ (n+1) = f_1\ n + 1 \end{cases}
\begin{cases} f_2\ 0 = 1 \\ f_2\ (n+1) = (f_1\ n) * (f_2\ n) \end{cases}
$$

tais que $f_1 = \pi_1 \cdot f$ e $f_2 = \pi_2 \cdot f$. Demonstre que assim é de facto e identifique o que faz a função f_2.
NB: relembra-se que, neste exercício, se tem $\mathsf{F}\, f = id + f$ e $\mathbf{in} = [\underline{0}\ , \mathsf{succ}]$.

RESOLUÇÃO: $f_1 = \pi_1 \cdot f$ e $f_2 = \pi_2 \cdot f$ equivalem a $f = \langle f_1, f_2 \rangle$. Logo (justificar cada passo):

$$
\begin{array}{ll}
& \langle f_1, f_2 \rangle = (\!|[\langle \underline{1}, \underline{1} \rangle\ , \langle \mathsf{succ} \cdot \pi_1, \mathsf{mul} \rangle]|\!) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \langle f_1, f_2 \rangle = (\!|\langle [\underline{1}\ , \mathsf{succ} \cdot \pi_1], [\underline{1}\ , \mathsf{mul}] \rangle|\!) \\
\equiv & \qquad \{ \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \ \} \\
& \begin{cases} f_1 \cdot \mathbf{in} = [\underline{1}\ , \mathsf{succ} \cdot \pi_1] \cdot (id + \langle f_1, f_2 \rangle) \\ f_2 \cdot \mathbf{in} = [\underline{1}\ , \mathsf{mul}] \cdot (id + \langle f_1, f_2 \rangle) \end{cases}
\end{array}
$$

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

$$\left\{ \begin{array}{l} f_1 \cdot \mathbf{in} = [\underline{1}\;, \mathsf{succ} \cdot f_1] \\ f_2 \cdot \mathbf{in} = [\underline{1}\;, \mathsf{mul} \cdot \langle f_1, f_2 \rangle] \end{array} \right.$$

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

$$\left\{ \begin{array}{l} \left\{ \begin{array}{l} f_1\ 0 = 1 \\ f_1\ (n+1) = f_1\ n + 1 \end{array} \right. \\ \left\{ \begin{array}{l} f_2\ 0 = 1 \\ f_2\ (n+1) = (f_1\ n) \times (f_2\ n) \end{array} \right. \end{array} \right.$$

$\square$

Reparando que $f_1 = (\![[\underline{1}\;, \mathsf{succ}]]\!) = \mathsf{succ}$, f_2 fica:

$$\left\{ \begin{array}{l} f_2\ 0 = 1 \\ f_2\ (n+1) = (n+1) \times (f_2\ n) \end{array} \right.$$

que é a função que calcula o factorial de um natural. $\square$

Questão 10 Qualquer algoritmo h é um hilomorfismo da forma $h = (\![g]\!) \cdot [\!(f)\!]$ satisfazendo a propriedade:

$$h = g \cdot (\mathsf{F}\ h) \cdot f \tag{E14}$$

Considere o hilomorfismo de listas ($\mathsf{F}\ f = id + id \times f$)

$$sq = [\![\mathsf{sum}, odds]\!] \tag{E15}$$

onde

$$\begin{array}{l} \mathsf{sum} = [\underline{0}\;, \mathsf{add}] \\ odds = (id + \langle impar, id \rangle) \cdot \mathsf{out}_{\mathbb{N}_0} \\ impar\ n = 2 * n + 1 \\ \mathsf{out}_{\mathbb{N}_0} \text{ é o converso de } \mathsf{in}_{\mathbb{N}_0} = [\underline{0}\;, \mathsf{succ}]. \end{array}$$

Mostre que sq (E15) é a função

$$\begin{array}{l} sq\ 0 = 0 \\ sq\ (n+1) = 2 * n + 1 + sq\ n \end{array}$$

que calcula o quadrado de um número natural.

RESOLUÇÃO: Tem-se (a justificar):

$$sq = [\![\mathsf{sum}, odds]\!]$$

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

$$sq = \mathsf{sum} \cdot (id + id \times sq) \cdot odds$$

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

$$sq = [\underline{0}\;, \mathsf{add}] \cdot (id + id \times sq) \cdot (id + \langle impar, id \rangle) \cdot \mathsf{out}_{\mathbb{N}_0}$$

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

$$sq \cdot \mathsf{in}_{\mathbb{N}_0} = [\underline{0}\;, \mathsf{add} \cdot (id \times sq)] \cdot (id + \langle impar, id \rangle)$$

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

$$sq \cdot [\mathsf{zero}\;, \mathsf{succ}] = [\underline{0}\;, \mathsf{add} \cdot \langle impar, sq \rangle]$$

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

$$\left\{\begin{array}{l} sq\ \underline{0} = \underline{0} \\ sq \cdot \mathsf{succ} = \mathsf{add} \cdot \langle impar, sq \rangle \end{array}\right.$$

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

$$\left\{\begin{array}{l} sq\ 0 = 0 \\ sq\ (n+1) = \mathsf{add}\ (impar\ n, sq\ n) \end{array}\right.$$

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

$$\left\{\begin{array}{l} sq\ 0 = 0 \\ sq\ (n+1) = (2\ n+1) + sq\ n \end{array}\right.$$

$\square$

$\square$
