Cálculo de Programas

2.° ano
Lic. Ciências da Computação e Mestrado Integrado em Engenharia Informática
UNIVERSIDADE DO MINHO

2016/17 - Ficha nr.° 9

1. Considere a função

$$\begin{array}{l} \mathsf{mirror}\ (\mathit{Leaf}\ \ a) = \mathit{Leaf}\ \ a \\ \mathsf{mirror}\ (\mathit{Fork}\ (x, y)) = \mathit{Fork}\ (\mathsf{mirror}\ y, \mathsf{mirror}\ x) \end{array}$$

que "espelha" árvores binárias do tipo

$$\mathsf{T} = \mathsf{LTree}\ A \qquad \left\{ \begin{array}{l} \mathsf{F}\ X = A + X^2 \\ \mathsf{F}\ f = id + f^2 \end{array} \right. \qquad \mathsf{in} = [\mathit{Leaf}\ , \mathit{Fork}]$$

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

Comece por mostrar que

$$\mathsf{mirror} = (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) \tag{F1}$$

desenhando o digrama que representa este catamorfismo.

Tal como swap, mirror é um isomorfismo de árvores pois é a sua própria inversa:

$$\mathsf{mirror} \cdot \mathsf{mirror} = id \tag{F2}$$

Complete a seguinte demonstração de (F2):

$$\begin{array}{ll} & \mathsf{mirror} \cdot \mathsf{mirror} = id \\ \equiv & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\ & \mathsf{mirror} \cdot (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) = (\!|\mathsf{in}|\!) \\ \Leftarrow & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\ & \mathsf{mirror} \cdot \ldots = \ldots \\ \ldots & \qquad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\ & (etc) \end{array}$$

2. Assumindo as definições $\mathsf{in}_{\mathbb{N}_0} = [\underline{0}\ , \mathsf{succ}]$, $\mathsf{in} = [\mathsf{nil}\ , \mathsf{cons}]$, $\mathsf{succ}\ n = n + 1$, $\mathsf{nil} = \underline{[\,]}$ e $\mathsf{cons}\ (h, t) = h : t$, mostre que o diagrama

$$\begin{tikzcd}
{[\mathbb{N}_0]} \arrow[rrr, bend left, "\mathsf{out}_{\mathsf{List}}"] \arrow[rrr, phantom, "\cong"] & & & 1 + \mathbb{N}_0 \times [\mathbb{N}_0] \arrow[lll, bend left, "\mathsf{in}"] \\
\mathbb{N}_0 \arrow[u, "f"] \arrow[rr, bend left, "\mathsf{out}_{\mathbb{N}_0}"] \arrow[rr, phantom, "\cong"] & & 1 + \mathbb{N}_0 \arrow[ll, bend left, "\mathsf{in}_{\mathbb{N}_0}"] \arrow[r, "id + \langle g, id \rangle"'] & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \arrow[u, "id + id \times f"']
\end{tikzcd}$$

define o *anamorfismo* de listas

$$f \quad = \quad [\!((id + \langle g, id \rangle) \cdot \mathsf{out}_{\mathbb{N}_0})\!]$$

e diga o que faz este anamorfismo para $g = \mathsf{succ} \cdot (2*)$.

3. A função

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

é o catamorfismo de listas $\mathsf{map}\ f = (\!| \mathsf{in} \cdot (id + f \times id) |\!)$, para $\mathsf{in} = [\mathsf{nil}\ , \mathsf{cons}]$. Mostre que a mesma função se pode escrever como um anamorfismo:

$$\mathsf{map}\ f = [\!((id + f \times id) \cdot \mathsf{out})\!] \tag{F3}$$

4. Mostre que o anamorfismo de listas

$$suffixes = [\!(g)\!]\ \mathbf{where}\ g = (id + \langle \mathsf{cons}, \pi_2 \rangle) \cdot \mathsf{out}$$

é a função

$$\begin{array}{l} suffixes\ [\,] = [\,] \\ suffixes\ (h : t) = (h : t) : suffixes\ t \end{array}$$

5. O algoritmo da divisão inteira,

$$\begin{array}{l} m \div n \\ \quad \mid m < n = 0 \\ \quad \mid \mathsf{otherwise} = 1 + (m - n) \div n \end{array}$$

corresponde ao anamorfismo de naturais $(\div n) = [\!(g\ n)\!]$ em que $g\ n = (<n) \rightarrow (i_1 \cdot !), (i_2 \cdot (-n))$. É de esperar que o algoritmo dado satisfaça a propriedade $(m * n) \div n = m$, isto é, $(\div n) \cdot (*n) = id$. Complete a prova seguinte dessa propriedade, que se baseia na lei de fusão dos anamorfismos (identifique-a no formulário):

$$\begin{array}{cl}
& (\div n) \cdot (*n) = id \\
\equiv & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& [\!(g\ n)\!] \cdot (*n) = [\!(\mathsf{out})\!] \\
\Leftarrow & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (g\ n) \cdot (*n) = (id + (*n)) \cdot \mathsf{out} \\
\equiv & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& ((<n) \cdot (*n)) \rightarrow (i_1 \cdot !), (i_2 \cdot (-n) \cdot (*n)) = (id + (*n)) \cdot \mathsf{out} \\
\equiv & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (= 0) \rightarrow (i_1 \cdot !), (i_2 \cdot (*n) \cdot \mathsf{pred}) = (id + (*n)) \cdot \mathsf{out} \\
\equiv & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& (id + (*n)) \cdot \mathsf{out} = (id + (*n)) \cdot \mathsf{out} \\
\equiv & \quad \{ \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad \} \\
& true
\end{array}$$