

Cálculo de Programas

2.º ano

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

2016/17 - Ficha nr.º 9

1. Considere a função

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

que “espelha” árvores binárias do tipo

$$\begin{aligned} T &= \text{LTree } A && \begin{cases} F X = A + X^2 \\ F f = id + f^2 \end{cases} && \text{in} = [\text{Leaf}, \text{Fork}] \\ \text{Haskell: } \text{data LTree } a &= \text{Leaf } a \mid \text{Fork} (\text{LTree } a, \text{LTree } a) \end{aligned}$$

Comece por mostrar que

$$\text{mirror} = (\text{in} \cdot (id + \text{swap})) \tag{F1}$$

desenhando o digrama que representa este catamorfismo.

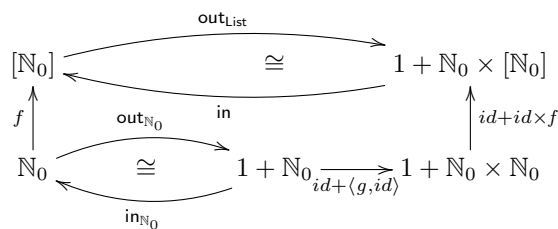
Tal como *swap*, *mirror* é um isomorfismo de árvores pois é a sua própria inversa:

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

Complete a seguinte demonstração de (F2):

$$\begin{aligned} &\text{mirror} \cdot \text{mirror} = id \\ \equiv &\quad \{ \dots\dots\dots \} \\ &\text{mirror} \cdot (\text{in} \cdot (id + \text{swap})) = (\text{in}) \\ \Leftarrow &\quad \{ \dots\dots\dots \} \\ &\text{mirror} \dots = \dots \\ \dots &\quad \{ \dots\dots\dots \} \\ &(\text{etc}) \end{aligned}$$

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



define o *anamorfismo* de listas

$$f = \llbracket (id + \langle g, id \rangle) \cdot \text{out}_{\mathbb{N}_0} \rrbracket$$

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

3. A função

$$\begin{aligned} \text{map } f \ [] &= [] \\ \text{map } f \ (h : t) &= (f \ h) : \text{map } f \ t \end{aligned}$$

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

$$\text{map } f = \llbracket (id + f \times id) \cdot \text{out} \rrbracket \tag{F3}$$

4. Mostre que o anamorfismo de listas

$$\text{suffixes} = \llbracket g \rrbracket \textbf{ where } g = (id + \langle \text{cons}, \pi_2 \rangle) \cdot \text{out}$$

é a função

$$\begin{aligned} \text{suffixes} \ [] &= [] \\ \text{suffixes} \ (h : t) &= (h : t) : \text{suffixes} \ t \end{aligned}$$

5. O algoritmo da divisão inteira,

$$\begin{aligned} m \div n \\ | \ m < n = 0 \\ | \ \text{otherwise} = 1 + (m - n) \div n \end{aligned}$$

corresponde ao anamorfismo de naturais $(\div n) = \llbracket g \ n \rrbracket$ em que $g \ n = (\langle n \rangle \rightarrow (i_1 \cdot !), (i_2 \cdot (-n)))$. 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):

$$\begin{aligned} & (\div n) \cdot (*n) = id \\ \equiv & \quad \{ \dots \} \\ & \llbracket g \ n \rrbracket \cdot (*n) = \llbracket \text{out} \rrbracket \\ \Leftarrow & \quad \{ \dots \} \\ & (g \ n) \cdot (*n) = (id + (*n)) \cdot \text{out} \\ \equiv & \quad \{ \dots \} \\ & ((\langle n \rangle \cdot (*n)) \rightarrow (i_1 \cdot !), (i_2 \cdot (-n) \cdot (*n))) = (id + (*n)) \cdot \text{out} \\ \equiv & \quad \{ \dots \} \\ & (= 0) \rightarrow (i_1 \cdot !), (i_2 \cdot (*n) \cdot \text{pred}) = (id + (*n)) \cdot \text{out} \\ \equiv & \quad \{ \dots \} \\ & (id + (*n)) \cdot \text{out} = (id + (*n)) \cdot \text{out} \\ \equiv & \quad \{ \dots \} \\ & \text{true} \end{aligned}$$