Cálculo de Programas

2.° ano das Licenciaturas em
Engenharia Informática e Ciências da Computação
UNIVERSIDADE DO MINHO

2012/13 - Ficha nr.° 8

1. Considere o diagrama que representa a propriedade universal dos catamorfismos, instanciada para listas em Haskell ($\mathsf{F}\ f = id + id \times f$):

$$\begin{tikzcd} [\,a\,] \arrow[d, "(\!|g|\!)"'] & 1 + a \times [\,a\,] \arrow[l, "\mathsf{in}"'] \arrow[d, "id+id\times(\!|g|\!)"] \\ b & 1 + a \times b \arrow[l, "g"] \end{tikzcd} \qquad f = (\!|g|\!) \ \equiv\ f \cdot \mathsf{in} = g \cdot (id + id \times f) \tag{1}$$

Tem-se, neste caso, $\mathsf{in} = [nil\ , \mathsf{cons}]$, $nil\ \_ = [\,]$ e $\mathsf{cons}\ (a, x) = a : x$.

   (a) Calcule a função out tal que $\mathsf{out} \cdot \mathsf{in} = id$.

   (b) Mostre que

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

   se reduz à equação

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

   e que, portanto, $\mathsf{map}\ f = (\!|\mathsf{in} \cdot (id + f \times id)|\!)$.

2. Identifique como catamorfismos de listas as funções seguintes, indicando o gene g para cada caso:

   - $f = \mathsf{reverse}$
   - f é a função que implementa o algoritmo de ordenação de listas por inserção ('insertion sort').
   - A função $f = \mathsf{filter}\ p$ tal que

$$\begin{array}{l} \mathsf{filter}\ p\ [\,] = [\,] \\ \mathsf{filter}\ p\ (h : t) = x +\!\!+\ \mathsf{filter}\ p\ t\ \mathbf{where} \\ \quad x = \mathbf{if}\ (p\ h)\ \mathbf{then}\ [h]\ \mathbf{else}\ [\,] \end{array}$$

   Apoie as suas resoluções com diagramas.

3. O diagrama que se segue representa a lei de fusão de catamorfismos

$$\begin{tikzcd} \mathsf{T} \arrow[d, "(\!|g|\!)"'] & \mathsf{F\,T} \arrow[l, "\mathsf{in}"'] \arrow[d, "\mathsf{F}\,(\!|g|\!)"] \\ A \arrow[d, "f"'] & \mathsf{F}\,A \arrow[l, "g"] \arrow[d, "\mathsf{F}\,f"] \\ B & \mathsf{F}\,B \arrow[l, "h"] \end{tikzcd} \qquad f \cdot (\!|g|\!) = (\!|h|\!) \ \Leftarrow\ f \cdot g = h \cdot \mathsf{F}\,f \tag{2}$$

em que T é um tipo indutivo (eg. listas, $\mathbb{N}_0$) e in é a sua álgebra de construção (com inversa out, não representada no diagrama). Demonstre essa lei. (**Sugestão:** generalize para o caso geral a demonstração que fez da mesma propriedade para o caso particular do combinador ciclo-for.)

4. Considere o diagrama

$$\begin{tikzcd} \mathsf{NTree} \arrow[d, "(\!|g|\!)"'] & \mathbb{N}_0 + \mathsf{NTree} \times \mathsf{NTree} \arrow[l, "\mathsf{in}=[\mathsf{Leaf}\ ,\mathsf{Fork}]"'] \arrow[d, "id+(\!|g|\!)\times(\!|g|\!)"] & (= \mathsf{F\ NTree}) \arrow[d, "(=\mathsf{F}(\!|g|\!))"] \\ A & \mathbb{N}_0 + A \times A \arrow[l, "g"] & (= \mathsf{F\ A}) \end{tikzcd}$$

que instancia T da questão 3 com o tipo das árvores binárias de números naturais:

$$\mathbf{data}\ \mathsf{NTree} = \mathsf{Leaf}\ \mathbb{N}_0 \mid \mathsf{Fork}\ (\mathsf{NLTree}, \mathsf{NTree})$$

para o qual $\mathsf{F}X = \mathbb{N}_0 + X \times X$.

Exprima sob a forma de catamorfismos deste tipo as funções seguintes: (a) função que soma todos os números que estão na árvore; (b) função que identifica o maior desses números; (c) função que substitui todos os números por zero; (d) função que conta quantos zeros estão na árvore. Em cada caso identifique o gene g do respectivo catamorfismo.

5. Resolva a equação $(\!|x|\!) = id$ em ordem a x e demonstre assim a lei de *relexão-cata*, válida para qualquer tipo de dados (naturais, listas, árvores, etc). Faça um diagrama que ilustre esta situação bem particular do cálculo de catamorfismos.

6. Considere o seguinte par de funções mutuamente recursivas que testam a paridade de um número:

$$\left\{ \begin{array}{l} impar\ 0 = \mathsf{False} \\ impar\ (n+1) = par\ n \end{array} \right. \qquad \left\{ \begin{array}{l} par\ 0 = \mathsf{True} \\ par\ (n+1) = impar\ n \end{array} \right.$$

   (a) Mostre que esse par de definições é equivalente ao sistema de equações

$$\left\{ \begin{array}{l} impar \cdot \mathsf{in} = [\underline{\mathsf{False}}\ , \pi_2] \cdot (id + \langle impar, par \rangle) \\ par \cdot \mathsf{in} = [\underline{\mathsf{True}}\ , \pi_1] \cdot (id + \langle impar, par \rangle) \end{array} \right.$$

   onde $\mathsf{in} = [\underline{0}\ , \mathsf{succ}]$ e $\mathsf{succ}\ n = n + 1$.

   (b) Mostre, recorrendo às leis da recursividade múltipla e da troca, que par e $impar$ se podem combinar num único ciclo-for com duas variáveis,

$$\begin{array}{l} impar = \pi_1 \cdot imparpar \\ par = \pi_2 \cdot imparpar \\ imparpar = \mathsf{for\ swap}\ (\mathsf{False}, \mathsf{True}) \end{array}$$

   sabendo que, como se viu nas aulas teóricas, catamorfismos de naturais são ciclos-for.