Cálculo de Programas

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

2018/19 - Ficha nr.° 7

1. Sabendo que $\mathsf{for}\ f\ i = (\![\underline{i}\ , f]\!)$ para $\mathsf{F}\ f = id + f$ (naturais), recorra à lei de fusão-cata para demonstrar a propriedade:[1]

$$f \cdot (\mathsf{for}\, f\ i) = \mathsf{for}\, f\ (f\ i) \tag{F1}$$

2. Mostre que a lei da recursividade mútua generaliza a mais do que duas funções, neste caso três:

$$\left\{\begin{array}{l} f \cdot in = h \cdot \mathsf{F}\,\langle f, \langle g, j\rangle\rangle \\ g \cdot in = k \cdot \mathsf{F}\,\langle f, \langle g, j\rangle\rangle \\ j \cdot in = l \cdot \mathsf{F}\,\langle f, \langle g, j\rangle\rangle \end{array}\right. \quad\equiv\quad \langle f, \langle g, j\rangle\rangle = (\!|\langle h, \langle k, l\rangle\rangle|\!) \tag{F2}$$

3. As seguintes funções mutuamente recursivas testam a paridade de um número natural:

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

Assumindo o functor $\mathsf{F}\ f = id + f$, mostre que esse par de definições é equivalente ao sistema de equações

$$\left\{\begin{array}{l} impar \cdot \mathsf{in} = h \cdot \mathsf{F}\ \langle impar, par\rangle \\ par \cdot \mathsf{in} = k \cdot \mathsf{F}\ \langle impar, par\rangle \end{array}\right.$$

para um dado h e k (deduza-os). De seguida, recorra às leis da recursividade mútua e da troca para mostrar que

$$imparpar = \langle impar, par\rangle = \mathsf{for}\ \mathsf{swap}\ (\mathsf{False}, \mathsf{True})$$

4. Recorrendo à lei de recursividade mútua, mostre que a função factorial[2] pode ser implementada como um ciclo-for:

$$fac = \pi_2 \cdot aux\ \mathbf{where}\ aux = \mathsf{for}\ \langle \mathsf{succ} \cdot \pi_1, \mathsf{mul}\rangle\ (1, 1)$$

Converta essa implementação numa versão em Haskell que não recorra ao combinador for .

5. Considere o par de funções mutuamente recursivas

$$\left\{\begin{array}{l} f_1\ [\,] = [\,] \\ f_1\ (h : t) = h : (f_2\ t) \end{array}\right. \qquad \left\{\begin{array}{l} f_2\ [\,] = [\,] \\ f_2\ (h : t) = f_1\ t \end{array}\right.$$

Use a lei de recursividade mútua para definir $\langle f_1, f_2\rangle$ como um catamorfismo de listas (onde o functor de trabalho é $\mathsf{F}\ f = id + id \times f$) e desenhe o respectivo diagrama. Que faz cada uma destas funções f_1 e f_2?

[1] Como complemento desta questão, escreva em sintaxe C os programas correspondentes aos dois lados da igualdade e compare-os informalmente.

[2] Recorde $fac \cdot [\underline{0}\ , \mathsf{succ}] = [\underline{1}\ , \mathsf{mul} \cdot \langle \mathsf{succ}, fac\rangle]$ – de uma ficha anterior.

6. Sejam dados os functores elementares seguintes:

$$\left\{\begin{array}{l}\mathsf{F}\ X = \mathbb{Z}\\ \mathsf{F}\ f = id\end{array}\right. \quad \text{e} \quad \left\{\begin{array}{l}\mathsf{G}\ X = X\\ \mathsf{G}\ f = f\end{array}\right.$$

Calcule $\mathsf{H}\ f$ e $\mathsf{K}\ f$ para

$$\mathsf{H}\ X = \mathsf{F}\ X + \mathsf{G}\ X \quad \text{e} \quad \mathsf{K}\ X = \mathsf{G}\ X \times \mathsf{F}\ X$$

7. Mostre que, se F e G são functores, então também o serão $\mathsf{F} + \mathsf{G}$ e $\mathsf{F} \times \mathsf{G}$ que a seguir se definem:

$$\begin{array}{l}(\mathsf{F} + \mathsf{G})\ X = (\mathsf{F}\ X) + (\mathsf{G}\ X)\\ (\mathsf{F} \times \mathsf{G})\ X = (\mathsf{F}\ X) \times (\mathsf{G}\ X)\end{array}$$

8. Considere o functor

$$\begin{array}{l}\mathsf{T}\ X = X \times X\\ \mathsf{T}\ f = f \times f\end{array}$$

e as funções

$$\begin{array}{l}\mu = \pi_1 \times \pi_2\\ u = \langle id, id\rangle.\end{array}$$

Mostre que a propriedade $\mu \cdot \mathsf{T}\ u = id = \mu \cdot u$ se verifica.

9. Considere a função

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

que "espelha" árvores binárias do tipo

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

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

Comece por mostrar que

$$\mathsf{mirror} = (\!|\mathsf{in} \cdot (id + \mathsf{swap})|\!) \tag{F3}$$

desenhando o digrama que representa este catamorfismo.

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

$$\mathsf{mirror} \cdot \mathsf{mirror} = id \tag{F4}$$

Complete a seguinte demonstração de (F4):

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