Cálculo de Programas

2.º Ano de LCC+MiEI (Universidade do Minho)
Ano Lectivo de 2020/21

Exame da época especial — 14 de Julho de 2021
14h00–16h00 - Sala CP1-0.08

- *Esta prova consta de **8** questões que valem, cada uma, 2.5 valores. O tempo médio estimado para resolução de cada questão é de 15 min.*
- *Recomenda-se que os alunos leiam a prova antes de decidirem por que ordem querem responder às questões que são colocadas.*

PROVA PRESENCIAL INDIVIDUAL SEM CONSULTA (2h)

Questão 1 Deduza o tipo mais geral da função $\mathsf{undistr} = [id \times i_1 \, , id \times i_2]$ e verifique analiticamente a respectiva propriedade *grátis* (*natural*), inferindo-a primeiro através de um diagrama.

Questão 2 Demonstre a lei do condicional de McCarthy que se segue

$$p \rightarrow (q \rightarrow a \ , \ b) \ , \ b \quad = \quad (p \wedge q) \rightarrow a \ , \ b \tag{E1}$$

sabendo que

$$(p \wedge q)? \quad = \quad p \rightarrow q? \ , \ i_2 \tag{E2}$$

é uma propriedade da conjunção de predicados.

Questão 3 Provar a igualdade $\overline{f \cdot (g \times h)} = \overline{\mathsf{ap} \cdot (id \times h)} \cdot \overline{f} \cdot g$ usando as leis das exponenciais e dos produtos.

Questão 4 Demonstre a seguinte propriedade do combinador catamorfismo: $(\!|\, g \,|\!) \cdot (\!|\, \mathsf{in} \cdot k \,|\!) = (\!|\, g \cdot k \,|\!)$ desde que $k \cdot \mathsf{F}\, f = \mathsf{F}\, f \cdot k$ se verifique.

Questão 5 Considere uma série numérica definida por recorrência da seguinte forma:

$$s_0 = 1$$
$$s_{n+1} = 2 * s_n + n + 2$$

Assim, a lista $[1, 4, 11, 26, 57, 120, 247, 502, \ldots]$ mostra os primeiros termos da série. Demonstre, por aplicação da lei de recursividade mútua, que a seguinte função

$$s = \pi_1 \cdot \mathsf{for}\ loop\ \mathsf{init}\ \mathbf{where}$$
$$\quad loop\ (x, y) = (2 * x + y, y + 1)$$
$$\quad \mathsf{init} = (1, 2)$$

calcula o n-ésimo termo da série.

Questão 6 Considere, definido em Haskell, o tipo

$$\mathbf{data}\ \mathsf{RTree}\ a = Ros\ a\ [\mathsf{RTree}\ a]$$

das habitualmente designadas "rose trees", que tem bifunctor de base $\mathsf{B}\ (X, Y) = X \times Y^*$ e

$$\mathsf{in} = \widehat{Ros}$$
$$\mathsf{out}\ (Ros\ a\ x) = (a, x)$$

É dada a definição da função

$$count = (\!|\, \mathsf{succ} \cdot \mathsf{sum} \cdot \pi_2 \,|\!) \tag{E3}$$

que conta o número de nós de uma "rose tree", onde sum é o catamorfismo que soma listas de números naturais. Mostre que

$$count \cdot (\mathsf{RTree}\ f) = count \tag{E4}$$

se verifica, onde $\mathsf{RTree}\ f$ designa o functor do tipo RTree que aplica f a todos os nós de uma árvore.

Questão 7 O facto de haver tantos números pares como ímpares permite-nos pensar noutra forma de construir e manipular números naturais, nomeadamente usando — em vez de a habitual álgebra $\mathsf{in} = [\mathsf{zero}, \mathsf{succ}]$ — a alternativa $\mathsf{in}^\bullet$ que se segue

$$\mathsf{in}^\bullet : \mathbb{N}_0 \leftarrow 1 + (\mathbb{N}_0 + \mathbb{N}_0)$$
$$\mathsf{in}^\bullet = [\mathsf{zero}, [\mathsf{par}, \mathsf{impar}]]\ \mathbf{where}$$
$$\quad \mathsf{par}\ n = 2 * n$$
$$\quad \mathsf{impar}\ n = 2 * n + 1$$
$$\quad \mathsf{zero}\ \_ = 0$$

cujo functor é $\mathsf{F}\ f = id + (f + f)$. No exame de recurso viu-se como exprimir a conversão de um número natural para base 2

[1] Completar com as justificações.

$$\begin{cases} base_2\ 0 = [\,] \\ base_2\ (2\ n) = base_2\ n \mathbin{+\!\!+} [0] \\ base_2\ (2\ n+1) = base_2\ n \mathbin{+\!\!+} [1] \end{cases}$$

como um catamorfismo deste tipo:

$$base_2 = (\!|\, [\mathsf{nil}, [g\ 0, g\ 1]] \,|\!)\ \textbf{where}\ g\ b\ w = w \mathbin{+\!\!+} [b] \tag{E5}$$

Mostre agora que a operação inversa

$$\begin{array}{l} base_{10}\ [\,] = 0 \\ base_{10}\ x = g\ (\mathsf{init}\ x, \mathsf{last}\ x)\ \textbf{where} \\ \quad g\ (i, 0) = 2 * base_{10}\ i \\ \quad g\ (i, 1) = 2 * base_{10}\ i + 1 \end{array}$$

é um anamorfismo do mesmo tipo, isto é identifique g em $base_{10} = [\!(g)\!]$. Justifique convenientemente a sua resposta, eg. sob a forma de um diagrama.

Questão 8 Com base em

$$\begin{array}{l} a \bullet x = \mathsf{map}\ (a*)\ x \\ x \diamond y = zipWith\ (+)\ x\ y \\ (k \otimes f)\ t = k\ t \bullet f\ t \\ (f \oplus g)\ t = f\ t \diamond g\ t \end{array}$$

defina-se a função:

$$\begin{array}{l} \alpha\ [p] = \underline{p} \\ \alpha\ x = (1-) \otimes \alpha\ (\mathsf{init}\ x) \oplus id \otimes \alpha\ (\mathsf{tail}\ x) \end{array}$$

Pretende demostrar-se que esta função é um hilomorfismo $\alpha = [\![c, a]\!]$. Sabendo que o seu tipo de entrada é paramétrico num tipo numérico genérico A, identifique: (a) o tipo de saída e de entrada de α; (b) a estrutura de dados intermédia (virtual) desse hilomorfismo; (c) os respectivos genes c e a. (Justifique convenientemente a sua resposta, eg. sob a forma de um diagrama.)

$$\alpha\ [p] = \underline{p}$$

ficamos a saber que $\alpha : P^* \to P^X$ (P e X a apurar), pois a entrada é uma lista e a saída é uma função constante em P. Quanto à outra cláusula, faça-se:

$$\alpha\ x = (1-) \otimes \underbrace{\alpha\ (\mathsf{init}\ x)}_{a} \oplus id \otimes \underbrace{\alpha\ (\mathsf{tail}\ x)}_{b}$$

e

$$\beta\ (a, b) = (1-) \otimes a \oplus id \otimes b$$

isto é

$$\begin{aligned} \alpha\ x &= \beta\ (\alpha\ (\mathsf{init}\ x), \alpha\ (\mathsf{tail}\ x)) \\ \alpha &= \beta \cdot (\alpha \times \alpha) \cdot \langle \mathsf{init}, \mathsf{tail} \rangle \end{aligned}$$

Tipo de β:

$$\beta : {A^*}^{A^*} \times {A^*}^{A^*} \to {A^*}^{A^*}$$

Logo, para um dado P:

$$\begin{tikzcd} P^* \arrow[d, "\alpha"'] & ? + P^* \times P^* \arrow[d, "? + \alpha \times \alpha"] \\ {A^*}^{A^*} & ? + {A^*}^{A^*} \times {A^*}^{A^*} \arrow[l, "{c = [?\ , \beta]}"] \end{tikzcd}$$

A outra cláusula, $\alpha\ [p] = \underline{p}$, diz-nos que $P = A^*$. Logo:

$$\begin{tikzcd} A^{**} \arrow[r, "a"] \arrow[d, "\alpha"'] & A^* + A^{**} \times A^{**} \arrow[d, "id + \alpha \times \alpha"] \\ {A^*}^{A^*} & A^* + {A^*}^{A^*} \times {A^*}^{A^*} \arrow[l, "{c = [\underline{\cdot}\ , \beta]}"] \end{tikzcd}$$

Daqui apuramos $\mathsf{B}\ (X, Y) = X + Y^2$. Logo o tipo intermédio será $\mathsf{LTree}\ A^*$. Finalmente, assumindo listas não vazias à entrada:

$$\begin{aligned} &a\ [p] = i_1\ p \\ &a\ x = (i_2 \cdot \langle \mathsf{init}, \mathsf{tail} \rangle)\ x \end{aligned}$$

$\square$
