

Cálculo de Programas

2.º Ano de LCC (Universidade do Minho)
Ano Lectivo de 2023/24

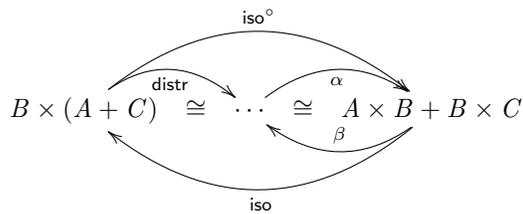
2º Teste — 15 de Maio de 2024, 16h00–18h00
Salas 0.03 + 0.05 do Edifício 2

PROVA PRESENCIAL INDIVIDUAL SEM CONSULTA (2h)

Importante — *Ler antes de iniciar a prova:*

- *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.*

Questão 1 Considere o seguinte diagrama:



(a) Defina o isomorfismo β ; (b) Infira a propriedade grátis de α sem o definir.

Questão 2 Demonstrar

$$(p \rightarrow g, h) \times f = p \cdot \pi_1 \rightarrow g \times f, h \times f$$

a partir das leis do condicional de McCarthy e do facto seguinte:

$$q \rightarrow f, f = f \tag{E1}$$

Questão 3 A função $\pi_2 : A \times B \rightarrow B$ é binária e, como tal, faz sentido a sua versão “curried” $\overline{\pi}_2 : A \rightarrow B^B$. Usando as leis da exponenciação mostre que $\overline{\pi}_2$ é uma função constante, isto é, que qualquer que seja f se tem:

$$\overline{\pi}_2 \cdot f = \overline{\pi}_2 \tag{E2}$$

Que função constante é essa? Justifique.

Questão 4 As chamadas “rose trees”, que se podem definir em Haskell por

```
data Rose a = Rose a [Rose a]
```

são árvores generalizadas em que cada nó tem um número arbitrário (mas finito) de sub-árvores. Defina para este tipo

- os isomorfismos in e out que o caracterizam;
- o functor de base B e o da recursividade F ;
- o gene g do catamorfismo $\llbracket g \rrbracket$ que conta o número de nós de uma “rose tree”.

Questão 5 Recorra à lei de absorção-cata para demonstrar a propriedade

$$\text{count} \cdot (\text{BTree } f) = \text{count} \tag{E3}$$

onde $\text{BTree } A \xrightarrow{\text{count}} \mathbb{N}_0$ é o catamorfismo

$$\text{count} = \llbracket [\text{zero}, \text{succ} \cdot \text{add} \cdot \pi_2] \rrbracket$$

para $\text{zero } _ = 0$, $\text{succ } n = n + 1$ e $\text{add } (a, b) = a + b$. **NB:** recorda-se que a base do tipo BTree é $B(f, g) = \text{id} + f \times (g \times g)$.

Questão 6 Considere a seguinte generalização da lei de absorção-cata,

$$\llbracket g \rrbracket \cdot \llbracket \text{in}_2 \cdot \alpha \rrbracket = \llbracket g \cdot \alpha \rrbracket \iff G f \cdot \alpha = \alpha \cdot F f \tag{E4}$$

a que corresponde o diagrama que se dá ao lado. Use (E4) para demonstrar a igualdade

$$\text{length} \cdot \text{zeros} = \text{id} \tag{E5}$$

onde

$$\begin{cases} \text{length} : A^* \rightarrow \mathbb{N}_0 \\ \text{length} = \llbracket \text{in}_{\mathbb{N}_0} \cdot (\text{id} + \pi_2) \rrbracket \end{cases} \tag{E6}$$

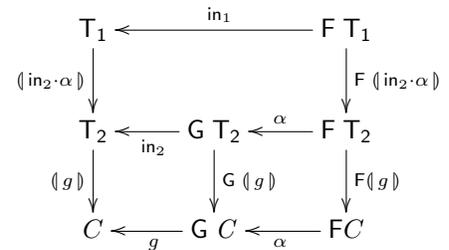
é uma função que conhece bem e onde $\text{zeros } n$ é a lista finita de n zeros,

$$\begin{aligned} \text{zeros } 0 &= [] \\ \text{zeros } (n + 1) &= 0 : \text{zeros } n \end{aligned}$$

isto é

$$\begin{cases} \text{zeros} : \mathbb{N}_0 \rightarrow \mathbb{N}_0^* \\ \text{zeros} = \llbracket \text{in}_* \cdot (\text{id} + \langle \underline{0}, \text{id} \rangle) \rrbracket \end{cases} \tag{E7}$$

para $\text{in}_* = [\text{nil}, \text{cons}]$ e $\text{in}_{\mathbb{N}_0} = [\underline{0}, \text{succ}]$. **NB:** recordar das aulas $F f = \text{id} + f$ e $G f = \text{id} + \text{id} \times f$, a usar em (E4).



Questão 7 Considere o anamorfismo $r = \llbracket g \rrbracket$ onde

$$\begin{aligned} g [] &= i_1 () \\ g x &= i_2 (\text{last } x, \text{init } x) \end{aligned}$$

$\text{last } x$ dá o último elemento da lista x e $\text{init } x$ dá x sem esse último elemento. O que faz a função r ? Acompanhe a sua resposta com o diagrama de r .

Questão 8 Considere a seguinte extensão da lei de recursividade mútua a hilomorfismos que partilham o mesmo anamorfismo:

$$\langle f, g \rangle = \llbracket \langle h, k \rangle \rrbracket \cdot \llbracket q \rrbracket \equiv \begin{cases} f = h \cdot F \langle f, g \rangle \cdot q \\ g = k \cdot F \langle f, g \rangle \cdot q \end{cases} \quad (\text{E8})$$

- Para uma dada função q , (E8) reduz-se à lei de recursividade mútua do formulário. Identifique essa função q , justificando.
- Apresente as justificações em falta no seguinte cálculo da lei (E8):

$$\begin{aligned} & \langle f, g \rangle = \llbracket \langle h, k \rangle \rrbracket \cdot \llbracket q \rrbracket \\ \equiv & \quad \{ \text{seja } \langle \alpha, \beta \rangle = \llbracket \langle h, k \rangle \rrbracket \} \\ & \langle f, g \rangle = \langle \alpha, \beta \rangle \cdot \llbracket q \rrbracket \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \langle f, g \rangle = \langle \alpha \cdot \llbracket q \rrbracket, \beta \cdot \llbracket q \rrbracket \rangle \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \begin{cases} f = \alpha \cdot \llbracket q \rrbracket \\ g = \beta \cdot \llbracket q \rrbracket \end{cases} \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \begin{cases} f = h \cdot F \langle \alpha, \beta \rangle \cdot \text{out} \cdot \llbracket q \rrbracket \\ g = k \cdot F \langle \alpha, \beta \rangle \cdot \text{out} \cdot \llbracket q \rrbracket \end{cases} \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \begin{cases} f = h \cdot F \langle \alpha, \beta \rangle \cdot F \llbracket q \rrbracket \cdot q \\ g = k \cdot F \langle \alpha, \beta \rangle \cdot F \llbracket q \rrbracket \cdot q \end{cases} \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \begin{cases} f = h \cdot F \langle \alpha \cdot \llbracket q \rrbracket, \beta \cdot \llbracket q \rrbracket \rangle \cdot q \\ g = k \cdot F \langle \alpha \cdot \llbracket q \rrbracket, \beta \cdot \llbracket q \rrbracket \rangle \cdot q \end{cases} \\ \equiv & \quad \{ \dots\dots\dots \} \\ & \begin{cases} f = h \cdot F \langle f, g \rangle \cdot q \\ g = k \cdot F \langle f, g \rangle \cdot q \end{cases} \\ \square & \end{aligned}$$