Cálculo de Programas

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

2016/17 - Ficha nr.° 4

1. Considere a função $\mathsf{coassocr} = [id + i_1 \ , i_2 \cdot i_2]$ que é uma das testemunhas do isomorfismo

   $$(A + B) + C \cong A + (B + C)$$

   da esquerda para a direita. Calcule a sua conversa $\mathsf{coassocl}$ a partir da equação

   $$\mathsf{coassocl} \cdot \mathsf{coassocr} \quad = \quad id$$

   (**Sugestão**: Faça-o resolvendo em ordem a x,y e z a seguinte versão

   $$\underbrace{[x \ , [y \ , z]]}_{\mathsf{coassocl}} \cdot \underbrace{[id + i_1 \ , i_2 \cdot i_2]}_{\mathsf{coassocr}} = id$$

   dessa equação.) Finalmente, calcule da equação $\mathsf{coassocl} = [x \ , [y \ , z]]$ que tiver obtido um programa em Haskell que implemente $\mathsf{coassocl}$ sem recurso ao combinador "either".

2. Sejam dadas, em geral, duas funções f e g satisfazendo a propriedade

   $$f \ y = x \quad \equiv \quad y = g \ x \tag{F1}$$

   (a) Mostre que f e g são inversas uma da outra — isto é, que as igualdades

   $$\begin{cases} id = g \cdot f \\ f \cdot g = id \end{cases}$$

   se verificam — e que, portanto, são ambas isomorfismos.

   (b) Escreva a equivalência (F1) para o caso $f = \mathsf{assocr}$ (identifique g).
   NB: todos estes isomorfismos célebres encontram-se definidos em Haskell na biblioteca `Cp.hs` disponível no material pedagógico da disciplina.

3. Deduza o tipo mais geral da função $f = (id + \pi_1) \cdot i_2 \cdot \pi_2$ e infira a respectiva propriedade *grátis* (*natural*) através de um diagrama.

4. Considere a função $\mathsf{iso} = \langle ! + !, [id \ , id] \rangle$.

   (a) Identifique o isomorfismo que ela testemunha, desenhando-a sob a forma de um diagrama de tipos.

   (b) Derive a partir desse diagrama a propriedade (dita *grátis*) de iso,

   $$(id \times f) \cdot \mathsf{iso} \quad = \quad \mathsf{iso} \cdot (f + f) \tag{F2}$$

   (c) Confirme, por cálculo analítico, essa propriedade.

   (d) Derive uma definição em Haskell *pointwise* de iso.

5. Identifique, apoiando a sua resolução num diagrama, qual é a definição da função polimórfica α cuja propriedade natural ("grátis") é

$$(f + h) \cdot \alpha \quad = \quad \alpha \cdot (f + g \times h)$$

6. Para o caso de um *isomorfismo* h, a lei de Leibniz (identifique-a no formulário) transforma-se numa equivalência:

$$f \cdot h = g \cdot h \quad \equiv \quad f = g \tag{F3}$$

Recorra a esta propriedade para mostrar que a igualdade

$$h \cdot \mathsf{distr} \cdot (g \times (id + f)) = k$$

é equivalente à igualdade

$$h \cdot (g \times id + g \times f) = k \cdot \mathsf{undistr}$$

(**Sugestão:** não ignore a propriedade natural (i.e. *grátis*) do isomorfismo distr.)

7. Um **índice remissivo** de autores é um apêndice de um texto (e.g. livro) indicando, por ordem alfabética de nomes de autores referidos e para cada nome de autor, a lista ordenada das páginas em que vem citados trabalhos seus, por exemplo:

Arbib, M. A. – 10, 11

Bird, R. – 28

Horowitz, E. – 2, 3, 15, 16, 19

Hudak, P. – 11, 12, 29

Jones, C. B. – 3, 7, 28

Manes, E. G. – 10, 11

Sahni, S. – 2, 3, 15, 16, 19

Spivey, J.M. – 3, 7

Wadler, P. – 2, 3

A estrutura acima pode representar-se pelo tipo $Ind = (Aut \times Pag^*)^*$, listando autores ($Aut$) e as respectivas páginas onde são referidos (Pag).

A geração de bibliografias (e.g. em LaTeX) baseia-se numa base de dados bibliográfica do qual se pode extrair a informação

$$Bib = (Key \times Aut^*)^*$$

que associa a cada chave de citação (Key) os respectivos autores.

Por outro lado, sempre que o LaTeX processa um documento, compila todas as ocorrências de chaves de citação num ficheiro auxiliar:

$$Aux = (Pag \times Key^*)^*$$

Pretendendo-se gerar Ind a partir de Bib e de Aux, mostre que nada tem que fazer pois... já resolveu este problema numa ficha anterior! [1]

[1] Chama-se *programação genérica* a este tipo de abordagem à programação.