Ficha 5

- Polimorfismo $\begin{cases} \text{Economia} \\ \text{Propriedade "grátis"} \end{cases}$
- Ordem superior

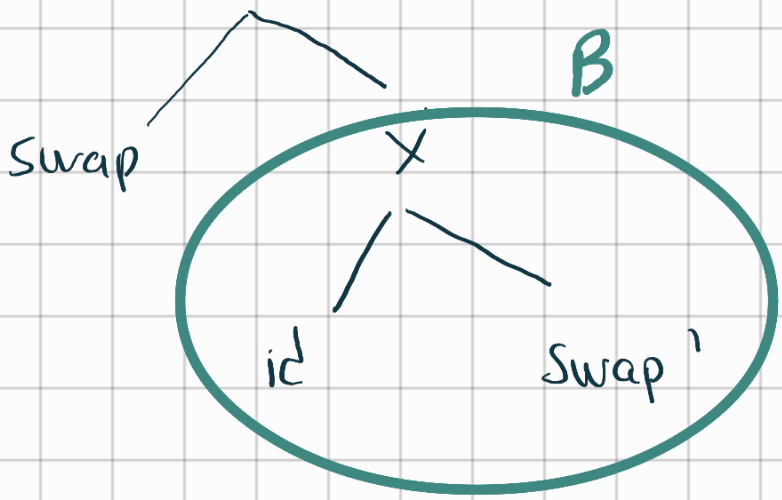
1. Considere a função

$$\alpha = \mathsf{swap} \cdot (id \times \mathsf{swap})$$

Calcule o tipo mais geral de α e formule a sua propriedade natural (grátis) a inferir através de um diagrama, como se explicou na aula teórica.

Let $\alpha = \mathsf{swap} \cdot (id \times \mathsf{swap})$ *be given. Infer the most general type of* α *and the associated natural ("free") property using a diagram, as shown in the theory class.*

$\alpha =$



$$B \times A \xleftarrow{\ swap\ } A \times B \qquad\qquad D \times C \xleftarrow{\ swap'\ } C \times D$$

$$\begin{array}{ccc} E & C \times D & E \times (C \times D) \\ \downarrow id & \downarrow swap' & \downarrow B = id \times swap' \\ E & D \times C & E \times (D \times C) \end{array}$$

Tipo mais geral do α

$$\alpha : (D\times C)\times E \xleftarrow{\ swap\ } E\times(D\times C) \xleftarrow{\ B\ } E\times(C\times D)$$

Teorema grátis

$E \to f$
$C \to g$
$D \to h$

$$\begin{tikzcd}
(D\times C)\times E \arrow[d, "(h\times g)\times f"'] & E\times(C\times D) \arrow[l, "\alpha"'] \arrow[d, "f\times(g\times h)"] \\
(D'\times C')\times E' & E'\times(C'\times D') \arrow[l, "\alpha"']
\end{tikzcd}$$

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

* Caso C fosse $\mathbb{N}$, g iria ser id ($C \to id$)

2. Recorde as seguintes funções elementares que respectivamente juntam ou duplicam informação:

Recall the following basic functions that respectively gather or duplicate information:

$$join = [id\ , id] \tag{F1}$$
$$dup = \langle id, id\rangle \tag{F2}$$

Calcule (justificando) a propriedade grátis da função $\alpha = dup \cdot join$ e indique por que razão não pode calcular essa propriedade para $join \cdot dup$.

Calculate (justifying) the free property of the function $\alpha = dup\cdot join$ and indicate why you cannot calculate this property for $join \cdot dup$.

$\alpha = dup \cdot join$

$$\begin{tikzcd}
A \arrow[r, "i_1"] \arrow[dr, "id"'] & A+A \arrow[d, "{[id, id]}"] & A \arrow[l, "i_2"'] \arrow[dl, "id"] \\
& A \arrow[dl, "id"'] \arrow[d, "{\langle id, id \rangle}"] \arrow[dr, "id"] & \\
A & A \times A \arrow[l] \arrow[r] & A
\end{tikzcd}$$

$$\begin{tikzcd}
A \times A \arrow[d, "f \times f"'] & A+A \arrow[l, "\alpha"'] \arrow[d, "f+f"] \\
A' \times A' & A'+A' \arrow[l, "\alpha"']
\end{tikzcd}$$

$$((f \times f) \cdot \alpha = \alpha \cdot (f+f))$$

$$A \times A \xleftarrow{dup} A$$

$$=$$

$$A \xleftarrow[join]{} A+A$$

→ ñ acontece por isso ñ se pode calcular para dup.join.

3. Considere a função

Assuming join defined above (F1), consider

$$\mathsf{iso} = \langle ! + !, join \rangle$$

onde $join$ está definida acima (F1) e $! : A \to 1$ designa a única função (constante) que habita o tipo $A \to 1$, habitualmente designada por "*bang*".
Após identificar o isomorfismo que ela testemunha, derive a partir do correspondente diagrama a propriedade (dita *grátis*) de iso:

where $! : A \to 1$ is the "bang" function (the unique polymorphic constant function of its type).
After identifying the isomorphism witnessed by iso*, derive its free (natural) property using a diagram:*

$$(id \times f) \cdot \mathsf{iso} \quad = \quad \mathsf{iso} \cdot (f + f) \tag{F3}$$

De seguida confirme, por cálculo analítico, essa propriedade. Finalmente, derive uma definição de iso em Haskell *pointwise* sem recurso a combinadores.

As a way of confirming (F3), give an analytic proof of this result. Finally, derive a pointwise definition of iso.

$$iso = \langle ! + !, join \rangle$$

$$join = [id, id]$$

$$iso = \langle ! + !, join \rangle$$

$$iso : A + A \longrightarrow (1+1) \times A$$

$$\begin{tikzcd}
(1+1) \times A \arrow[d, "id \times f"'] & A + A \arrow[l, "iso"'] \arrow[d, "f+f"] \\
(1+1) \times A' & A' + A' \arrow[l, "iso"]
\end{tikzcd}$$

$$(id \times f) \cdot iso = iso \cdot (f + f) \Leftrightarrow$$

$$\Leftrightarrow (id \times f) \cdot \langle ! + !, join \rangle = \langle ! + !, join \rangle \cdot (f + f) \Leftrightarrow$$

(11)

$\Leftrightarrow \langle id \cdot (!+!), f \cdot join \rangle = \langle !+!, join \rangle \cdot (f+f) \Leftrightarrow$

$\Leftrightarrow \langle (!+!), f \cdot [id, id] \rangle = \langle !+!, join \rangle \cdot (f+f) \Leftrightarrow$

(1, 9)

$\Leftrightarrow \langle (!+!), [f, f] \rangle = \langle \underbrace{(!+!) \cdot (f+f)}_{*}, join \cdot (f+f) \rangle \Leftrightarrow$

$\Leftrightarrow \langle (!+!), [f, f] \rangle = \langle (!+!), [f, f] \rangle$

* ! é uma função constante

4. Seja dada uma função ∇ da qual só sabe duas propriedades: $\nabla \cdot i_1 = id$ e $\nabla \cdot i_2 = id$. Mostre que, necessariamente, ∇ satisfaz também a propriedade natural

Suppose that, about a function ∇, you only know two properties: $\nabla \cdot i_1 = id$ and $\nabla \cdot i_2 = id$. Show that, necessarily, ∇ also satisfies the natural property

$$f \cdot \nabla = \nabla \cdot (f + f) \tag{F4}$$

4-

$$\begin{cases} \nabla \cdot i_1 = id \\ \nabla \cdot i_2 = id \end{cases} \Leftrightarrow \nabla = [id, id] = join$$

$f \cdot [id, id] = [id, id] \cdot (f+f) \Leftrightarrow$

(20, 22)

$\Leftrightarrow [f \cdot id, f \cdot id] = [id \cdot f, id \cdot f] \Leftrightarrow$

(1)

$\Leftrightarrow [f, f] = [f, f]$

$A \xleftarrow{\nabla} A + A$

igualdades envolvendo co-produtos, normalmente usa-se fusão de um lado e absorção do outro

5. Seja dada uma função α cuja propriedade grátis é:

Let α be a polymorphic function with free property:

$$(f + h) \cdot \alpha = \alpha \cdot (f + g \times h) \tag{F5}$$

Será esta propriedade suficiente para deduzir a definição de α? Justifique analiticamente.

Can a definition of α be inferred from (F5)? Justify.

$$\begin{tikzcd}
A+C \arrow[d, "f+h"'] & & A+B\times C \arrow[ll, "\alpha"'] \arrow[d, "f+g\times h"] \\
A'+C' & & A'+B'\times C' \arrow[ll, "\alpha"]
\end{tikzcd}$$

$$\alpha : A+B\times C \longrightarrow A+C$$

$$\begin{tikzcd}
A \arrow[r, "i_1"] \arrow[d, "id"] & A+B\times C \arrow[d, "\alpha"] & B\times C \arrow[l, "i_2"'] \arrow[d, "\pi_2"] \\
A \arrow[r, "i_1"] & A+C & C \arrow[l, "i_2"']
\end{tikzcd}$$

$$\alpha = id + \pi_2$$

6. O formulário inclui as duas equivalências seguintes, válidas para qualquer isomorfismo *alpa*:

Any isomorphism α satisfies the following equivalences (also given in the reference sheet),

distr

$$\alpha \cdot g = h \equiv g = \alpha^\circ \cdot h \tag{F6}$$

∀ ✓

$$g \cdot \alpha = h \equiv g = h \cdot \alpha^\circ \tag{F7}$$

undistr

Recorra a essas propriedades para mostrar que a igualdade

which can be useful to show that the equality

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

é equivalente à igualdade

is equivalent to:

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

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

Prove this equivalenvce. (**Hint:** *the free-property of* distr *shoudn't be ingored in the reasoning.*)

$$\begin{tikzcd}
A \times C + B \times C \arrow[rr, phantom, "\cong"] \arrow[rr, bend right, "\text{undistl}"'] & & (A+B) \times C \arrow[ll, bend right, "\text{distl}"']
\end{tikzcd}$$

undistl ("pôr" c em evidência)

$$\begin{tikzcd}
A \times C + B \times C \arrow[d, "(f \times h) + (g \times h)"] & (A+B) \times C \arrow[l, "\text{distl}"'] \arrow[d, "(f+g) \times h"] \\
A' \times C' + B' \times C' & (A'+B') \times C \arrow[l, "\text{distl}"']
\end{tikzcd}$$

$$(f \times h + g \times h) \cdot \text{distl} = \text{distl} \cdot ((f+g) \times h)$$

//

$$\begin{tikzcd}
C \times A + C \times B \arrow[rr, phantom, "\cong"] \arrow[rr, bend right, "\text{undistr}"'] & & C \times (A+B) \arrow[ll, bend right, "\text{distr}"']
\end{tikzcd}$$

undistr ("pôr" c em evidência)

$$\begin{tikzcd}
C \times A + C \times B \arrow[d, "(h \times f) + (h \times g)"] & C \times (A+B) \arrow[l, "dist_r"'] \arrow[d, "h \times (f+g)"] \\
C' \times A' + C' \times B' & C' \times (A'+B') \arrow[l, "distr"']
\end{tikzcd}$$

$$(h \times f + h \times g) \cdot dist_r = dirtr \, (h \times (f+g))$$

7. Considere o combinador *comb f* definido por

Let comb f be a combinator defined by:

$$comb\ f = [id\,, f] \cdot (i_1 + i_2) \cdot f \tag{F8}$$

Mostre que o tipo mais geral de *comb* é

Show that its most general type is

$$comb : (C+B)^{A+B} \to (C+B)^{A+B}$$

e demonstre analiticamente que:

and prove that:

$$comb\ id = id$$

$f : A+B \longrightarrow C+B$

$f : ((C+B))^{A+B}$

$$\begin{tikzcd}
 & A+B \arrow[d, "f"] & \\
 & C+B \arrow[d, "i_1+i_2"] & \\
C+B \arrow[r, "i_1"] \arrow[dr, "id"'] & C+B+A+B \arrow[d, "{[id,f]}"] & A+B \arrow[l, "i_2"'] \arrow[dl, "f"] \\
 & C+B &
\end{tikzcd}$$