

# CP - Ficha 5

---

## Exercício 1

---

No cálculo de programas, as definições condicionais do tipo

$$h\ x = \mathbf{if}\ p\ x\ \mathbf{then}\ f\ x\ \mathbf{else}\ g\ x \quad (\text{F1})$$

são escritas usando o combinador ternário

$$p \rightarrow f, g$$

conhecido pelo nome de *condicional de McCarthy*, cuja definição

$$p \rightarrow f, g = [f, g] \cdot p? \quad (\text{F2})$$

vem no [formulário](#). Baseie-se em leis desse [formulário](#) para demonstrar a chamada 2ª-lei de fusão do condicional:

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

## Resolução 1

$$\begin{aligned} & (p \rightarrow f, g) \cdot h \\ &= \quad \quad \quad (30: \text{Cond. McCharty (F2)}, 2: \text{Assoc-comp}) \\ & [f, g] \cdot (p? \cdot h) \\ &= \quad \quad \quad (29: \text{Natural-guarda}) \\ & [f, g] \cdot (h + h) \cdot (p \cdot h)? \\ &= \quad \quad \quad (22: \text{Absorção-+}) \\ & [f \cdot h, g \cdot h] \cdot (p \cdot h)? \\ &= \quad \quad \quad (30: \text{Cond. McCharty (F2)}) \\ & (p \cdot h) \rightarrow f \cdot h, g \cdot h \end{aligned}$$

## Exercício 2

Numa máquina paralela pode fazer sentido, em (F1), não esperar por  $p \ x$  para avaliar  $f \ x$  ou  $g \ x$ , mas sim correr tudo em paralelo,

$$\text{parallel } p \ f \ g = \langle \langle f, g \rangle, p \rangle$$

e depois fazer a escolha do resultado:

$$\text{choose} = \pi_2 \rightarrow \pi_1 \cdot \pi_1, \pi_2 \cdot \pi_1$$

Mostre que, de facto:

$$\text{choose} \cdot \text{parallel } p \ f \ g = p \rightarrow f, g$$

## Resolução 2

$$\text{choose} \cdot \text{parallel } p \ f \ g$$

=

(Def. *parallel*, Def. *choose*)

$$(\pi_2 \rightarrow \pi_1 \cdot \pi_1, \pi_2 \cdot \pi_1) \cdot \langle \langle f, g \rangle, p \rangle$$

=

(32: 2ª lei de fusão condicional)

$$\pi_2 \cdot \langle \langle f, g \rangle, p \rangle \rightarrow \pi_1 \cdot \pi_1 \cdot \langle \langle f, g \rangle, p \rangle, \pi_2 \cdot \pi_1 \cdot \langle \langle f, g \rangle, p \rangle$$

=

(7: Cancelamento- $\times$ )

$$p \rightarrow f, g$$

## Exercício 3

Sabendo que as igualdades

$$p \rightarrow k, k = k \quad (\text{F3})$$

$$(p? + p?) \cdot p? = (i_1 + i_2) \cdot p? \quad (\text{F4})$$

se verificam, demonstre as seguintes propriedades do mesmo combinador:

$$\langle (p \rightarrow f, h), (p \rightarrow g, i) \rangle = p \rightarrow \langle f, g \rangle, \langle h, i \rangle \quad (\text{F5})$$

$$\langle f, (p \rightarrow g, h) \rangle = p \rightarrow \langle f, g \rangle, \langle f, h \rangle \quad (\text{F6})$$

$$p \rightarrow (p \rightarrow a, b), (p \rightarrow c, d) = p \rightarrow a, d \quad (\text{F7})$$

## Resolução 3

(F5)

$$\begin{aligned} & \langle (p \rightarrow f, h), (p \rightarrow g, i) \rangle \\ &= \quad (30: \text{Cond. McCharty}) \\ & \langle [f, h] \cdot p?, [g, i] \cdot p? \rangle \\ &= \quad (9: \text{Fusão-}\times) \\ & \langle [f, h], [g, i] \rangle \cdot p? \\ &= \quad (28: \text{Lei da troca}) \\ & [\langle f, g \rangle, \langle h, i \rangle] \cdot p? \\ &= \quad (30: \text{Cond. McCharty}) \\ & p \rightarrow \langle f, g \rangle, \langle h, i \rangle \end{aligned}$$

(F6)

$$\begin{aligned} & \langle f, (p \rightarrow g, h) \rangle \\ &= \quad (\text{F3}) \\ & \langle (p \rightarrow f, f), (p \rightarrow g, h) \rangle \\ &= \quad (\text{F5}) \\ & p \rightarrow \langle f, g \rangle, \langle f, h \rangle \end{aligned}$$

(F7)

$$\begin{aligned} & p \rightarrow (p \rightarrow a, b), (p \rightarrow c, d) \\ & = \quad \quad \quad (30: \text{Cond. McCharty } (3 \times)) \\ & [[a, b] \cdot p?, [c, d] \cdot p?] \cdot p? \\ & = \quad \quad \quad (22: \text{Absorção-+}) \\ & [[a, b], [c, d]] \cdot (p? \cdot p?) \cdot p? \\ & = \quad \quad \quad (F4) \\ & [[a, b], [c, d]] \cdot (i_1 + i_2) \cdot p? \\ & = \quad \quad \quad (22: \text{Absorção-+}, 18: \text{Cancelamento-+}) \\ & [a, d] \cdot p? \\ & = \quad \quad \quad (30: \text{Cond. McCharty}) \\ & p \rightarrow a, d \end{aligned}$$

## Exercício 4

---

Mostre que a propriedade de cancelamento da exponenciação

$$\text{ap} \cdot (\bar{f} \times id) = f \quad (F8)$$

corresponde à definição

$$\text{curry } f \ a \ b = f \ (a, b)$$

quando se escreve  $\text{curry } f$  em lugar de  $\bar{f}$ .

## Resolução 4

$$\begin{aligned} & \text{ap} \cdot (\bar{f} \times id) = f \\ & \equiv \quad \quad \quad (72: \text{Ig. Ext.}, 73: \text{Def-comp}) \\ & \text{ap} \ ((\bar{f} \times id) \ (a, b)) = f \ (a, b) \\ & \equiv \quad \quad \quad (78: \text{Def-}\times) \\ & \text{ap} \ (\bar{f} \ a, id) = f \ (a, b) \\ & \equiv \quad \quad \quad (84: \text{Def-ap}, 1: \text{Natural-id}) \\ & \bar{f} \ a \ b = f \ (a, b) \\ & \equiv \quad \quad \quad (85: \text{Curry}, \text{Def. } \text{curry}) \\ & \text{curry } f \ a \ b = f \ (a, b) \end{aligned}$$

## Exercício 5

Mostre que a definição de `uncurry` se pode obter também de (F8) fazendo  $f := \text{uncurry } g$ , introduzindo variáveis e simplificando.

## Resolução 5

$$\begin{aligned}
f &:= \text{uncurry } g = \widehat{g} \\
&\equiv \quad (\text{F8}) \\
\text{ap} \cdot (\overline{\widehat{g}} \times id) &= \widehat{g} \\
&\equiv \quad (72: \text{Ig. Ext.}, 73: \text{Def-comp}, \text{Iso.: } \overline{\widehat{g}} = g) \\
\text{ap} ((g \times id) (a, b)) &= \widehat{g} (a, b) \\
&\equiv \quad (78: \text{Def-}\times, 1: \text{Natural-id}) \\
\text{ap} (g \ a, b) &= \widehat{g} (a, b) \\
&\equiv \quad (84: \text{Def-ap}, \widehat{g} = \text{uncurry } g) \\
g \ a \ b &= \text{uncurry } g \ (a, b)
\end{aligned}$$

## Exercício 6

---

Prove a igualdade

$$\overline{f \cdot (g \times h)} = \overline{ap \cdot (\text{id} \times h)} \cdot \bar{f} \cdot g \quad (\text{F9})$$

usando as leis das exponenciais e dos produtos.

## Resolução 6

TODO

## Exercício 7

É dada a definição

$$\text{flip } f = \overline{\hat{f} \cdot \text{swap}} \quad (\text{F10})$$

de acordo com:

$$\begin{array}{ccccccc} (C^B)^A & \cong & C^{A \times B} & \cong & C^{B \times A} & \cong & (C^A)^B \\ f & \mapsto & \hat{f} & \mapsto & \hat{f} \cdot \text{swap} & \mapsto & \overline{\hat{f} \cdot \text{swap}} = \text{flip } f \end{array}$$

Mostre que **flip** é um isomorfismo por ser a sua própria inversa:

$$\text{flip } (\text{flip } f) = f \quad (\text{F11})$$

Mostre ainda que:

$$\text{flip } f \ x \ y = f \ y \ x$$

## Resolução 7

$$\begin{aligned} & \text{flip } (\text{flip } f) \\ &= \quad \quad \quad (\text{Def. flip } (2 \times)) \\ & \overline{\overline{\hat{f} \cdot \text{swap} \cdot \text{swap}}} \\ &= \quad \quad \quad (\text{Isomorfismo curry/uncurry}) \\ & \overline{\hat{f} \cdot \text{swap} \cdot \text{swap}} \\ &= \quad \quad \quad (\text{swap} \cdot \text{swap} = id, 1: \text{Natural-id}) \\ & \overline{\hat{f}} \\ &= \quad \quad \quad (\text{Isomorfismo curry/uncurry}) \\ & f \quad \text{c.q.m.} \end{aligned}$$

## Exercício 8

Mostre que

$$junc \cdot unjunc = id \quad (F12)$$

$$unjunc \cdot junc = id \quad (F13)$$

se verificam, onde

$$\begin{array}{ccc} A^{B+C} & \xrightarrow{unjunc} & A^B \times A^C \\ & \cong & \\ & \xleftarrow{junc} & \end{array}$$

$$\begin{cases} junc(f, g) = [f, g] \\ unjunc k = (k \cdot i_1, k \cdot i_2) \end{cases} \quad (F14)$$

## Resolução 8

$$junc \cdot unjunc = id$$

$$\equiv \quad (72: \text{Ig. Ext.}, 73: \text{Def-comp})$$

$$junc(unjunc k) = id k$$

$$\equiv \quad (\text{Def. } unjunc, 74: \text{Def-id})$$

$$junc(k \cdot i_1, k \cdot i_2) = k$$

$$\equiv \quad (\text{Def. } junc)$$

$$[k \cdot i_1, k \cdot i_2] = k$$

$$\equiv \quad (20: \text{Fusão-+}, 19: \text{Reflexão-+}, 1: \text{Natural-id})$$

$$k = k$$

$$unjunc \cdot junc = id$$

$$\equiv \quad (72: \text{Ig. Ext.}, 73: \text{Def-comp}, 74: \text{Def-id})$$

$$unjunc(junc(f, g)) = (f, g)$$

$$\equiv \quad (\text{Def. } junc)$$

$$unjunc[f, g] = (f, g)$$

$$\equiv \quad (\text{Def. } unjunc)$$

$$(f \cdot i_1, g \cdot i_2) = (f, g)$$

$$\equiv \quad (18: \text{Cancelamento-+})$$

$$(f, g) = (f, g)$$



## Exercício 9

Considere a seguinte sintaxe concreta em Haskell para um tipo que descreve pontos no espaço tridimensional:

```
data Point a = Point {x :: a, y :: a, z :: a} deriving (Eq, Show)
```

Pelo GHCi apura-se:

```
Point :: a -> a -> a -> Point a
```

Raciocinando apenas em termos de tipos, conjecture a definição de `in` na seguinte conversão dessa sintaxe concreta para abstracta:

$$\begin{array}{ccc} & \xrightarrow{\text{out}=\langle\langle x,y\rangle,z\rangle} & \\ \text{Point } A & \cong & (A \times A) \times A \\ & \xleftarrow{\text{in}=\dots} & \end{array}$$

## Resolução 9

$$\begin{aligned} \text{in} \cdot \langle x, \langle y, z \rangle \rangle &= id \\ &\equiv (72: \text{Ig. Ext.}, 74: \text{Def-id}) \\ \text{in} \cdot \langle x, \langle y, z \rangle \rangle (\text{Point } a \ b \ c) &= \text{Point } a \ b \ c \\ &\equiv (p = \text{Point } a \ b \ c, 73: \text{Def-comp}, 77: \text{Def-split } (2\times)) \\ \text{in } (x \ p, (y \ p, z \ p)) &= p \\ &\equiv (\text{Def. } x, y \text{ e } z) \\ \text{in } (a, (b, c)) &= \text{Point } a \ b \ c \\ &\equiv (86: \text{Uncurry } (2\times)) \\ \text{in } (a, (b, c)) &= \overline{\overline{\text{Point}}} ((a, b), c) \\ &\equiv (72: \text{Ig. Ext.}) \\ \text{in} &= \overline{\overline{\text{Point}}} \end{aligned}$$

# Exercício 10

---

## Questão prática

**Problem requirements:** The solution given for a previous problem,

$$\text{store } c = \text{take } 10 \cdot \text{nub} \cdot (c :) \quad (\text{F15})$$

calls the standard function

**nub** :: (Eq a) => [a] -> [a]

available from the [Data.List](#) library in Haskell.

After inspecting the standard implementation of this function, define  $f$  so that

$$\text{nub} = [\text{nil}, \text{cons}] \cdot f$$

is an alternative to the standard definition, where  $\text{nil } \_ = []$  and  $\text{cons } (h, t) = h : t$ .

Check that *store c* (F15) works properly once the standard *nub* is replaced by yours.

**Important:** Structure your solution across the  $f \cdot g$ ,  $\langle f, g \rangle$ ,  $f \times g$ ,  $[f, g]$  and  $f + g$  combinators that can be found in library [Cp.hs](#). Use **diagrams** to plan your proposed solution, which should avoid re-inventing functions over lists already available in the Haskell [standard libraries](#).

## Resolução 10

TODO

