# Cálculo de Programas

# Resolução - Ficha 07

Eduardo Freitas Fernandes

2026

## Exercício 1

$$\begin{tikzcd}
{\mathbb{N}_0}^* \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "prod"'] & & 1 + \mathbb{N}_0 \times {\mathbb{N}_0}^* \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times prod"] \\
\mathbb{N}_0 & & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \arrow[ll, bend left, "{[\underline{1}, mul]}"]
\end{tikzcd}$$

$$prod = \llparenthesis\, [\underline{1}, mul] \,\rrparenthesis$$

$$\begin{tikzcd}
A^* \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "reverse"'] & & 1 + A \times A^* \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times reverse"] \\
A^* & & 1 + A \times A^* \arrow[ll, bend left, "{[nil, aux]}"]
\end{tikzcd}$$

$$reverse' = \llparenthesis\, [nil, aux] \,\rrparenthesis$$
$$\quad \textbf{where } aux\ (h, t) = t \mathbin{+\!\!+} [h]$$

$$\begin{tikzcd}
A^{**} \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "concat"'] & & 1 + A^* \times A^{**} \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times concat"] \\
A^* & & 1 + A^* \times A^* \arrow[ll, bend left, "{[nil, \mathsf{conc}]}"]
\end{tikzcd}$$

$$concat' = \llparenthesis\, [nil, \mathsf{conc}] \,\rrparenthesis$$

$$\begin{tikzcd}
A^* \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "\mathsf{map}\ f"'] & & 1 + A \times A^* \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times \mathsf{map}\ f"] \\
B^* & & 1 + A \times B^* \arrow[ll, bend left, "{[nil, cons \cdot (f \times id)]}"]
\end{tikzcd}$$

$$map'\ f = \llparenthesis\, [nil, cons \cdot (f \times id)] \,\rrparenthesis$$

$$\begin{tikzcd}
A^* \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "maximum"'] & & A + A \times A^* \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times maximum"] \\
A & & A + A \times A \arrow[ll, bend left, "{[id, umax]}"]
\end{tikzcd}$$

$$maximum' = \llparenthesis\, [id, umax] \,\rrparenthesis$$

$$
\begin{tikzcd}
A^* \arrow[rr, bend left, "\mathsf{out}"] \arrow[d, "filter\ p"'] & & 1 + A \times A^* \arrow[ll, bend left, "\mathsf{in}"] \arrow[d, "id + id \times filter\ p"] \\
A^* & & 1 + A \times A^* \arrow[ll, bend left, "{[nil, \mathsf{conc} \cdot (p \to singl, nil \times id)]}"]
\end{tikzcd}
\qquad
filter'\ p = (\![ [nil, \mathsf{conc} \cdot (p \to singl, nil \times id)] ]\!)
$$

**Exercício 2**

$$
\begin{array}{ll}
& sumprod\ a = (a*) \cdot sum \\
\equiv & \quad \{ \text{ def. } sum\text{, def. } sumprod\ a \ \} \\
& (\![ [\underline{0}, add \cdot ((a*) \times id)] ]\!) = (a*) \cdot (\![ [\underline{0}, add] ]\!) \\
\Longleftarrow & \quad \{ \text{ fusão-cata } \} \\
& (a*) \cdot [\underline{0}, add] = [\underline{0}, add \cdot ((a*) \times id)] \cdot (id + id \times (a*)) \\
\equiv & \quad \{ \text{ fusão-}+\text{, absorção-}+ \ \} \\
& [(a*) \cdot \underline{0}, (a*) \cdot add] = [\underline{0}, add \cdot ((a*) \times id) \cdot (id \times (a*))] \\
\equiv & \quad \{ \text{ eq-}+ \ \} \\
& \left\{ \begin{array}{l} (a*) \cdot \underline{0} = \underline{0} \\ (a*) \cdot add = add \cdot ((a*) \times id) \cdot (id \times (a*)) \end{array} \right. \\
\equiv & \quad \{ \text{ functor-}\times \ \} \\
& \left\{ \begin{array}{l} (a*) \cdot \underline{0} = \underline{0} \\ (a*) \cdot add = add\ ((a*) \times (a*)) \end{array} \right. \\
\equiv & \quad \{ \text{ pointwise, def. } add\text{, def. } \underline{0} \ \} \\
& \left\{ \begin{array}{l} a * 0 = 0 \\ a * (x + y) = (a * x) + (a * y) \end{array} \right.
\end{array}
$$

**Exercício 3**

$$
\begin{array}{ll}
& f \cdot length = (\![ [\underline{0}, 2+] \cdot \pi_2 ]\!) \\
\equiv & \quad \{ \text{ def. } length \ \} \\
& f \cdot (\![ [\underline{0}, \mathsf{succ} \cdot \pi_2] ]\!) = (\![ [\underline{0}, 2+] \cdot \pi_2 ]\!) \\
\Longleftarrow & \quad \{ \text{ fusão-cata } \} \\
& f \cdot [\underline{0}, \mathsf{succ} \cdot \pi_2] = [\underline{0}, (2+) \cdot \pi_2] \cdot (id + id \times f) \\
\equiv & \quad \{ \text{ fusão-}+\text{, absorção-}+\text{, natural-}id \ \} \\
& [f \cdot \underline{0}, f \cdot \mathsf{succ} \cdot \pi_2] = [\underline{0}, (2+) \cdot \pi_2 \cdot (id \times f)] \\
\equiv & \quad \{ \text{ eq-}+\text{, natural-}\pi_2 \ \} \\
& \left\{ \begin{array}{l} f \cdot \underline{0} = \underline{0} \\ f \cdot \mathsf{succ} \cdot \pi_2 = (2+) \cdot f \cdot \pi_2 \end{array} \right. \\
\equiv & \quad \{ \text{ pointwise } \} \\
& \left\{ \begin{array}{l} (f \cdot \underline{0})\ n = \underline{0}\ n \\ (f \cdot \mathsf{succ} \cdot \pi_2)\ (x, y) = ((2+) \cdot f \cdot \pi_2)\ (x, y) \end{array} \right.
\end{array}
$$

$$\equiv \qquad \{ \text{ def. comp, def. } \pi_2 \ \}$$

$$\begin{cases} f\ 0 = 0 \\ f\ (y+1) = 2 + f\ y \end{cases}$$

Dado que $2 + 0 = 2$ e $2 * 0 = 0$, concluimos que $f = (2*)$.

## Exercício 4

$$foldr\ \overline{\pi_2}\ i = f$$

$$\equiv \qquad \{ \text{ def. } foldr \ \}$$

$$(\!|\, [\underline{i}, \widehat{\overline{\pi_2}}]\, |\!) = f$$

$$\equiv \qquad \{ \text{ universal-cata, } \widehat{\overline{f}} = f \ \}$$

$$f \cdot \mathsf{in} = [\underline{0}, \pi_2] \cdot (id + id \times f)$$

$$\equiv \qquad \{ \text{ def. in, fusão-}+\text{, absorção-}+\text{, eq-}+ \ \}$$

$$\begin{cases} f \cdot nil = \underline{i} \\ f \cdot cons = \pi_2 \cdot (id \times f) \end{cases}$$

$$\equiv \qquad \{ \text{ pointwise } \}$$

$$\begin{cases} f\ [\,] = i \\ f\ (h : t) = f\ t \end{cases}$$

Podemos concluir que $foldr\ \bar{\cdot}\ \pi_2\ i$ é a função constante $i$.

## Exercício 5

$$f \cdot \mathsf{for}\ f\ i = \mathsf{for}\ f\ f\ i$$

$$\equiv \qquad \{ \text{ def. for } b\ i \ \}$$

$$f \cdot (\!|\, [\underline{i}, f]\, |\!) = (\!|\, [\underline{f\ i}, f]\, |\!)$$

$$\Longleftarrow \qquad \{ \text{ fusão-cata } \}$$

$$f \cdot [\underline{i}, f] = [\underline{f\ i}, f] \cdot (id + f)$$

$$\equiv \qquad \{ \text{ fusão-}+\text{, absorção-}+ \ \}$$

$$[f \cdot \underline{i}, f \cdot f] = [\underline{f\ i} \cdot id, f \cdot f]$$

$$\equiv \qquad \{ \text{ eq-}+\text{, natural-}id\text{, absorção-const } \}$$

$$\begin{cases} \underline{f\ i} = \underline{f\ i} \\ f \cdot f = f \cdot f \end{cases}$$

$\square$

## Exercício 6

$$\mathsf{for}\ id\ i = \mathsf{for}\ \underline{i}\ i$$

$$\equiv \qquad \{ \text{ def. for } \cdot\ \cdot \text{ (twice) } \}$$

$$(\!|[\underline{i}, id]|\!) = (\!|[\underline{i}, \underline{i}]|\!)$$

$$\equiv \qquad \{ \text{ universal-cata } \}$$

$$(\!|[\underline{i}, id]|\!) \cdot \mathsf{in} = [\underline{i}, \underline{i}] \cdot (id + (\!|[\underline{i}, id]|\!))$$

$$
\begin{aligned}
&\equiv \quad \{ \text{ cancelamento-cata, absorção-}+ \ \} \\
&\quad [\underline{i}, id] \cdot (id + \llparenthesis [\underline{i}, id] \rrparenthesis) = [\underline{i}, \underline{i}] \\
&\equiv \quad \{ \text{ absorção-}+ \ \} \\
&\quad [\underline{i}, \llparenthesis [\underline{i}, id] \rrparenthesis] = [\underline{i}, \underline{i}] \\
&\equiv \quad \{ \text{ eq-}+ \ \} \\
&\quad \begin{cases} \underline{i} = \underline{i} \\ \llparenthesis [\underline{i}, id] \rrparenthesis = \underline{i} \end{cases} \\
&\equiv \quad \{ \text{ universal-cata } \} \\
&\quad \begin{cases} true \\ \underline{i} \cdot \mathsf{in} = [\underline{i}, id] \cdot (id + \underline{i}) \end{cases} \\
&\equiv \quad \{ \text{ absorção-}+ \ \} \\
&\quad \underline{i} = [\underline{i}, \underline{i}] \\
&\Box
\end{aligned}
$$

A última expressão é verdadeira, pois o resultado de $[\underline{i}, \underline{i}]$ será sempre $i$.

## Exercício 7

$$
\begin{aligned}
\underline{id} &: B \to A^A \\
f &: C \to D \\
(f \cdot) &: C^E \to D^E
\end{aligned}
$$

Temos de ter igualdade de tipos, out seja:

$$
A^A = D^E \implies \begin{cases} A = E \\ A = D \end{cases}
$$

Logo, temos:

$$
\begin{aligned}
\underline{id} &: B \to A^A \\
f &: C \to A \\
(f \cdot) &: C^A \to A^A
\end{aligned}
$$

$$
\begin{tikzcd}
\mathbb{N}_0 \arrow[r, bend left, "\mathsf{out}"] \arrow[d, "rep\ f"'] & 1 + \mathbb{N}_0 \arrow[l, bend left, "\mathsf{in}"] \arrow[d, "id + rep\ f"] \\
A^A & 1 + C^A \arrow[l, "{[\underline{i}, (f \cdot)]}"]
\end{tikzcd}
$$

## Exercício 9

$$
\begin{aligned}
&\mathbf{type}\ Date = String \\
&\mathbf{type}\ Player = String \\
&\mathbf{type}\ Game = String \\
&db1 = [ \\
&\quad (\texttt{"2023-10-01"}, [\texttt{"Game1"}, \texttt{"Game2"}]), \\
&\quad (\texttt{"2023-10-02"}, [\texttt{"Game2"}, \texttt{"Game3"}])
\end{aligned}
$$

```haskell
  ]
db2 = [
  ("Game1", ["PlayerA", "PlayerB"]),
  ("Game2", ["PlayerA", "PlayerC"]),
  ("Game3", ["PlayerB", "PlayerC"])
  ]
f :: [(Date, [Game])] → [(Game, [Player])] → [(Player, [Date])]
f = ⊥
main :: IO ()
main = do
  let result = Main.f db1 db2
  print result
```