

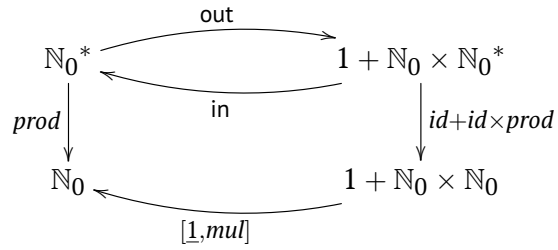
Cálculo de Programas

Resolução - Ficha 07

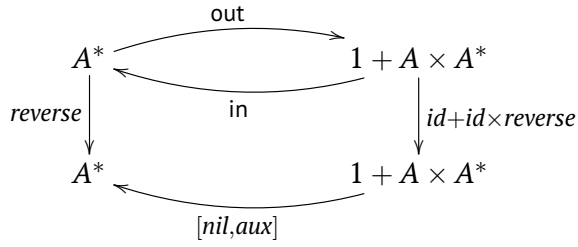
Eduardo Freitas Fernandes

2026

Exercício 1

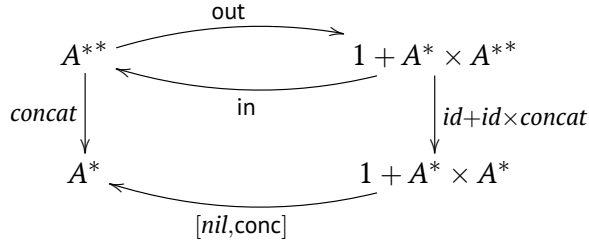


$$prod = \llbracket [1, mul] \rrbracket$$

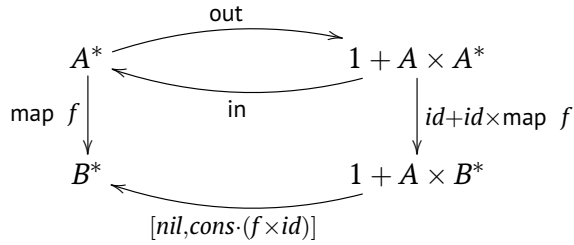


$$reverse' = \llbracket [nil, aux] \rrbracket$$

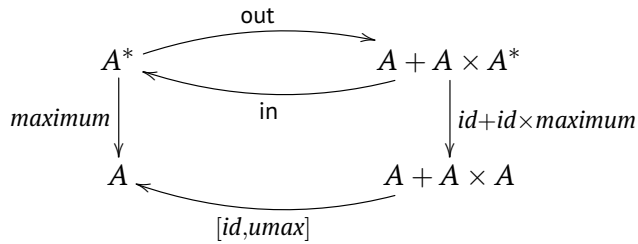
where $aux(h, t) = t ++ [h]$



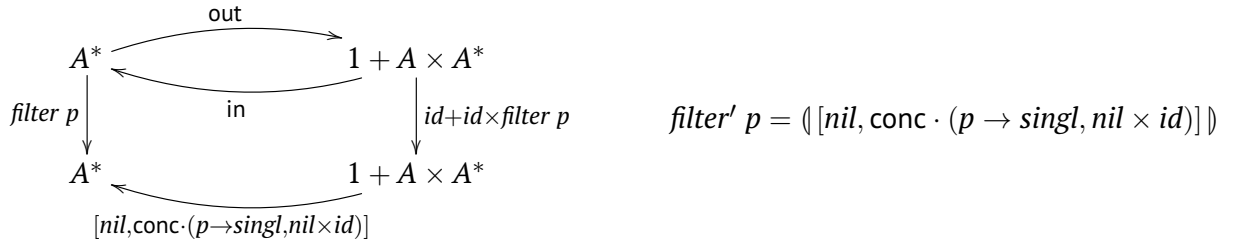
$$concat' = \llbracket [nil, conc] \rrbracket$$



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



$$maximum' = \llbracket [id, umax] \rrbracket$$



Exercício 2

$$\begin{aligned}
 & sumprod\ a = (a^*) \cdot sum \\
 \equiv & \{ \text{def. } sum, \text{def. } sumprod\ a \} \\
 & \llbracket [0, add \cdot ((a^*) \times id)] \rrbracket = (a^*) \cdot \llbracket [0, add] \rrbracket \\
 \Leftarrow & \{ \text{fusão-cata} \} \\
 & (a^*) \cdot [0, add] = [0, add \cdot ((a^*) \times id)] \cdot (id + id \times (a^*)) \\
 \equiv & \{ \text{fusão-+}, \text{absorção-+} \} \\
 & [(a^*) \cdot 0, (a^*) \cdot add] = [0, add \cdot ((a^*) \times id) \cdot (id \times (a^*))] \\
 \equiv & \{ \text{eq-+} \} \\
 & \begin{cases} (a^*) \cdot 0 = 0 \\ (a^*) \cdot add = add \cdot ((a^*) \times id) \cdot (id \times (a^*)) \end{cases} \\
 \equiv & \{ \text{functor-}\times \} \\
 & \begin{cases} (a^*) \cdot 0 = 0 \\ (a^*) \cdot add = add \cdot ((a^*) \times (a^*)) \end{cases} \\
 \equiv & \{ \text{pointwise, def. } add, \text{def. } 0 \} \\
 & \begin{cases} a * 0 = 0 \\ a * (x + y) = (a * x) + (a * y) \end{cases}
 \end{aligned}$$

Exercício 3

$$\begin{aligned}
 & f \cdot length = \llbracket [0, 2+] \cdot \pi_2 \rrbracket \\
 \equiv & \{ \text{def. } length \} \\
 & f \cdot \llbracket [0, succ \cdot \pi_2] \rrbracket = \llbracket [0, 2+] \cdot \pi_2 \rrbracket \\
 \Leftarrow & \{ \text{fusão-cata} \} \\
 & f \cdot [0, succ \cdot \pi_2] = [0, (2+) \cdot \pi_2] \cdot (id + id \times f) \\
 \equiv & \{ \text{fusão-+}, \text{absorção-+}, \text{natural-id} \} \\
 & [f \cdot 0, f \cdot succ \cdot \pi_2] = [0, (2+) \cdot \pi_2 \cdot (id \times f)] \\
 \equiv & \{ \text{eq-+}, \text{natural-}\pi_2 \} \\
 & \begin{cases} f \cdot 0 = 0 \\ f \cdot succ \cdot \pi_2 = (2+) \cdot f \cdot \pi_2 \end{cases} \\
 \equiv & \{ \text{pointwise} \} \\
 & \begin{cases} (f \cdot 0)\ n = 0\ n \\ (f \cdot succ \cdot \pi_2)(x, y) = ((2+) \cdot f \cdot \pi_2)(x, y) \end{cases}
 \end{aligned}$$

$$\begin{aligned} &\equiv \{ \text{def. comp, def. } \pi_2 \} \\ &\quad \begin{cases} f \ 0 = 0 \\ f \ (y + 1) = 2 + f \ y \end{cases} \end{aligned}$$

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

Exercício 4

$$\begin{aligned} &\text{foldr } \pi_2 \ i = f \\ &\equiv \{ \text{def. foldr} \} \\ &\quad \llbracket [i, \widehat{\pi_2}] \rrbracket = f \\ &\equiv \{ \text{universal-cata, } \widehat{f} = f \} \\ &\quad f \cdot \text{in} = [0, \pi_2] \cdot (id + id \times f) \\ &\equiv \{ \text{def. in, fusão-+, absorção-+, eq-+} \} \\ &\quad \begin{cases} f \cdot nil = i \\ f \cdot cons = \pi_2 \cdot (id \times f) \end{cases} \\ &\equiv \{ \text{pointwise} \} \\ &\quad \begin{cases} f \ [] = i \\ f \ (h : t) = f \ t \end{cases} \end{aligned}$$

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

Exercício 5

$$\begin{aligned} &f \cdot \text{for } f \ i = \text{for } f \ f \ i \\ &\equiv \{ \text{def. for } b \ i \} \\ &\quad f \cdot \llbracket [i, f] \rrbracket = \llbracket [f \ i, f] \rrbracket \\ &\Leftarrow \{ \text{fusão-cata} \} \\ &\quad f \cdot [i, f] = [f \ i, f] \cdot (id + f) \\ &\equiv \{ \text{fusão-+, absorção-+} \} \\ &\quad [f \cdot i, f \cdot f] = [f \ i \cdot id, f \cdot f] \\ &\equiv \{ \text{eq-+, natural-id, absorção-const} \} \\ &\quad \begin{cases} f \ i = f \ i \\ f \cdot f = f \cdot f \end{cases} \\ &\square \end{aligned}$$

Exercício 6

$$\begin{aligned} &\text{for } id \ i = \text{for } i \ i \\ &\equiv \{ \text{def. for } \cdot \cdot \text{ (twice)} \} \\ &\quad \llbracket [i, id] \rrbracket = \llbracket [i, i] \rrbracket \\ &\equiv \{ \text{universal-cata} \} \\ &\quad \llbracket [i, id] \rrbracket \cdot \text{in} = [i, i] \cdot (id + \llbracket [i, id] \rrbracket) \end{aligned}$$

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

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

Exercício 7

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

Temos de ter igualdade de tipos, out seja:

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

Logo, temos:

$$\begin{array}{ccc}
\mathbb{N}_0 & \xrightarrow{\text{out}} & 1 + \mathbb{N}_0 \\
\downarrow \text{rep } f & \swarrow \text{in} & \downarrow id + \text{rep } f \\
A^A & \xleftarrow{[i, (f \cdot)]} & 1 + C^A
\end{array}$$

Exercício 9

```

type Date = String
type Player = String
type Game = String
db1 = [
  ("2023-10-01", ["Game1", "Game2"]),
  ("2023-10-02", ["Game2", "Game3"])
]

```

```

]
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

```