Cálculo de Programas
Resolução - Ficha 03

Eduardo Freitas Fernandes

2025

Exercício 1

$$
\begin{aligned}
&\equiv \{\text{def. assocl, Fusão-}\times\text{, Reflexão-}\times\text{, Eq-}\times\} \\
&\equiv \{\text{Def-}\times\text{, Universal-}\times\} \\
&\equiv \{\text{associação à direita}\} \\
&\equiv \{\text{Universal-}\times\} \\
&\equiv \{\text{Natural-id, Universal-}\times\}
\end{aligned}
$$

Exercício 2

```
ghci> assocr = split (p1 . p1) (p2 >< id)
ghci> :t assocr
assocr :: ((b1, b2), d) -> (b1, (b2, d))
ghci> assocr ((3,4), 5)
(3,(4,5))
ghci> assocr ((x,y), z) = (x, (y,z))
ghci> :t assocr
assocr :: ((a1, a2), b) -> (a1, (a2, b))
ghci> assocr ((3,4), 5)
(3,(4,5))
```

Exercício 3

$$\frac{f : A \to B \qquad g : C \to D}{f \times g : A \times C \to B \times D}$$

$$\frac{f : A \to B \qquad g : A \to C}{\langle f, g \rangle : A \to B \times C}$$

$$\frac{f : A \to B \qquad g : B \to C}{f \cdot g : A \to C}$$

$$\frac{\pi_2 : A \times B \to B \qquad \pi_1 : A \times B \to A}{\langle \pi_2, \pi_1 \rangle : A \times B \to B \times A}$$

$$\frac{id : A \to A \qquad swap : B \times C \to C \times B}{id \times swap : A \times (B \times C) \to A \times (C \times B)}$$

$$\frac{swap : D \times E \to E \times D \qquad id \times swap : A \times (B \times C) \to A \times (C \times B)}{swap \cdot (id \times swap) : A \times (B \times C) \to (C \times B) \times A}$$

$$(F0) : \quad (g \times f) \cdot swap = swap \cdot (f \times g)$$

$$
\begin{aligned}
& \beta \cdot (f \times (g \times h)) \\
= & \{\text{def. } \beta\} \\
& swap \cdot (id \times swap) \cdot (f \times (g \times h)) \\
= & \{(\text{F0})\} \\
& (id \times swap) \cdot swap \cdot (f \times (g \times h)) \\
= & \{(\text{F0})\} \\
& (swap \times id) \cdot ((g \times h) \times f) \cdot swap \\
= & \{\text{Functor-}\times\} \\
& ((swap \cdot (g \times h)) \times (id \cdot f)) \cdot swap \\
= & \{(\text{F0})\} \\
& (((h \times g) \cdot swap) \times (f \cdot id)) \cdot swap \\
= & \{\text{Functor-}\times\} \\
& ((h \times g) \times f) \cdot (swap \times id) \cdot swap \\
= & \{(\text{F0})\} \\
& ((h \times g) \times f) \cdot swap \cdot (id \times swap) \\
= & \{\text{def. } \beta\} \\
& ((h \times g) \times f) \cdot \beta
\end{aligned}
$$

Exercício 4

$$\underline{k}\ x = \underline{k}\ (x) = \underline{k}\ (id\ x) = \underline{k} \cdot id = \underline{k} = k$$

Exercício 5

```
ghci> data X = B Bool | P (Bool,Int)
ghci> :t B
B :: Bool -> X
ghci> :t P
P :: (Bool, Int) -> X
ghci> f = either B P
ghci> :t f
f :: Either Bool (Bool, Int) -> X
```

Exercício 6

$$\frac{\underline{False} : A \to Bool \qquad id : A \to A}{\langle \underline{False}, id \rangle : A \to Bool \times A} \qquad\qquad \frac{f : A \to C \qquad g : B \to C}{[f, g] : A + B \to C}$$

$$\frac{\langle \underline{False}, id \rangle : A \to Bool \times A \qquad \langle \underline{True}, id \rangle : A \to Bool \times A}{[\langle \underline{False}, id \rangle, \langle \underline{True}, id \rangle] : A + A \to Bool \times A}$$

Exercício 7

$$\alpha = [\langle \underline{False}, id\rangle, \langle \underline{True}, id\rangle]$$
$$\equiv \{\text{Universal-}+\}$$
$$\begin{cases} \alpha \cdot i_1 = \langle \underline{False}, id\rangle \\ \alpha \cdot i_2 = \langle \underline{True}, id\rangle \end{cases}$$
$$\equiv \{\text{point wise}\}$$
$$\begin{cases} (\alpha \cdot i_1)\ a = \langle \underline{False}, id\rangle\ a \\ (\alpha \cdot i_2)\ a = \langle \underline{True}, id\rangle\ a \end{cases}$$
$$\equiv \{\text{def. composição, def. split}\}$$
$$\begin{cases} \alpha(i_1\ a) = (\underline{False}\ a, id\ a) \\ \alpha(i_2\ a) = (\underline{True}\ a, id\ a) \end{cases}$$
$$\equiv \{\text{def. const, def. id}\}$$
$$\begin{cases} \alpha(i_1\ a) = (False, a) \\ \alpha(i_2\ a) = (True, a) \end{cases}$$

```
ghci> alpha (Left a) = (False, a)
ghci> alpha (Right a) = (True, a)
```

Exercício 8

$$\frac{\pi_1 : A \times B \to A \qquad id : C \to C}{\pi_1 \times id : (A \times B) \times C \to A \times C}$$

$$\frac{\pi_2 : A \times B \to B \qquad \pi_1 : (A \times B) \times C \to A \times B}{\pi_2 \cdot \pi_1 : (A \times B) \times C \to B}$$

$$\frac{\pi_1 \times id : (A \times B) \times C \to A \times C \qquad \pi_2 \cdot \pi_1 : (A \times B) \times C \to B}{\langle \pi_1 \times id, \pi_2 \cdot \pi_1\rangle : (A \times B) \times C \to (A \times C) \times B}$$

$$xr \cdot \langle\langle f, g\rangle, h\rangle = \langle\langle f, h\rangle, g\rangle$$
$$\equiv \{\text{Universal-}\times\}$$
$$\begin{cases} \pi_1 \cdot xr \cdot \langle\langle f, g\rangle, h\rangle = \langle f, h\rangle \\ \pi_2 \cdot xr \cdot \langle\langle f, g\rangle, h\rangle = g \end{cases}$$
$$\equiv \{\text{def. xr, Cancelamento}\}$$
$$\begin{cases} (\pi_1 \times id) \cdot \langle\langle f, g\rangle, h\rangle = \langle f, h\rangle \\ \pi_2 \cdot \pi_1 \cdot \langle\langle f, g\rangle, h\rangle = g \end{cases}$$
$$\equiv \{\text{Absorção-}\times\text{, Cancelamento-}\times\}$$
$$\begin{cases} \langle \pi_1 \cdot \langle f, g\rangle, id \cdot h\rangle = \langle f, h\rangle \\ \pi_2 \cdot \langle f, g\rangle = g \end{cases}$$
$$\equiv \{\text{Cancelamento-}\times\}$$
$$\begin{cases} \langle f, h\rangle = \langle f, h\rangle \\ g = g \end{cases}$$

__Exercício 9__

```
mkInd :: (Bib, Aux) -> Ind
mkInd = map (id >< sort)
        . uncurry applyMapping
        . swap
        . (invertRelation >< invertRelation)

invertRelation :: (Ord a, Ord b) => [(a, [b])] -> [(b, [a])]
invertRelation = groupPairs . sortOn p1 . map swap . expandPairs

expandPairs :: [(a, [b])] -> [(a, b)]
expandPairs = concat . map (\(x, l) -> map (x,) l)

groupPairs :: (Eq a) => [(a, b)] -> [(a, [b])]
groupPairs = uncurry zip . split
                            (nub . map p1)
                            (map (map p2)
                             . groupBy (curry (uncurry (==)
                                                 . (p1 >< p1))))

applyMapping :: (Eq b) => [(b, [c])] -> [(a, [b])] -> [(a, [c])]
applyMapping a = map (applyMappingOne a)

applyMappingOne :: (Eq b) => [(b, [c])] -> (a, [b]) -> (a, [c])
applyMappingOne d = id >< (concat . map ( \x -> case lookup x d of
                                                    Just xs -> xs
                                                    _ -> []))
```

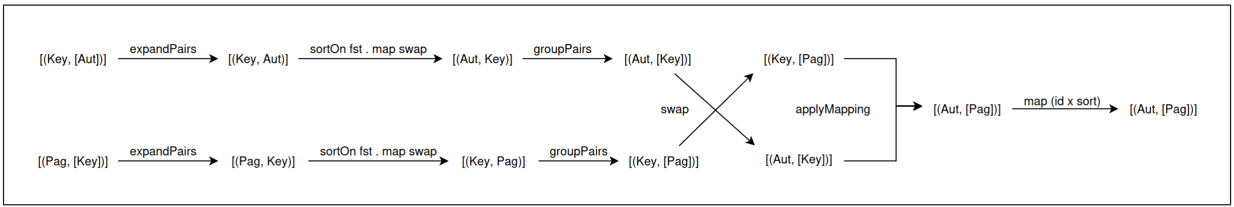

Figura 1: mkInd