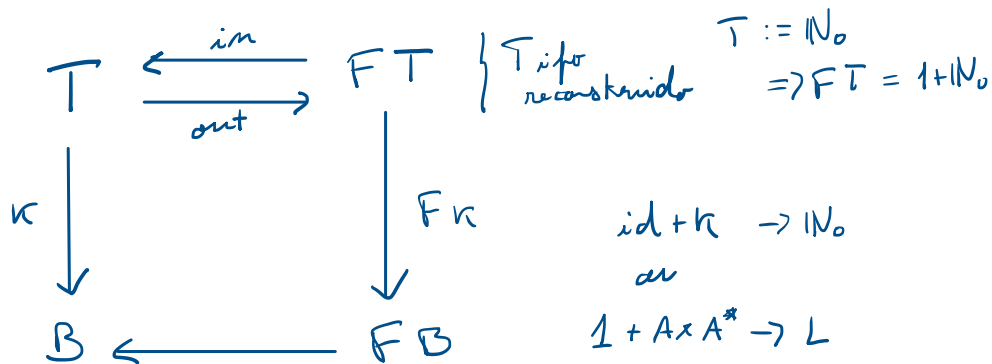


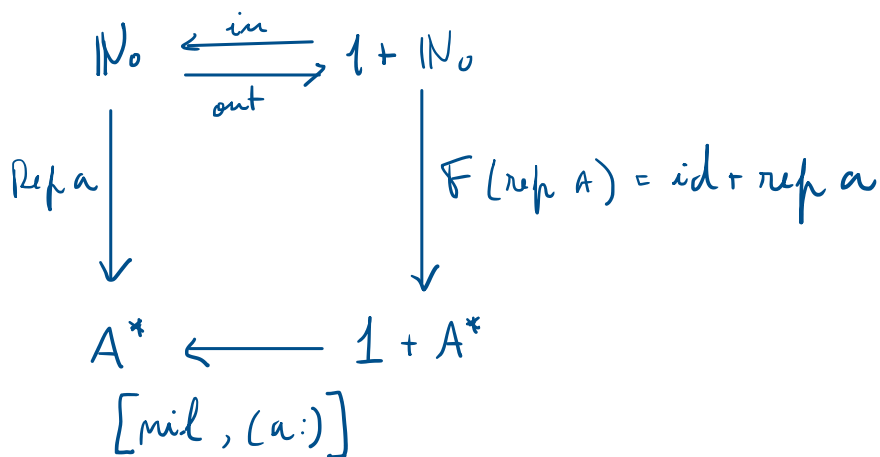
# Aula Pl #07

31 de outubro de 2023 18:02



$\gamma$   $\leftarrow$  Catamorfismo com gene  $\gamma$

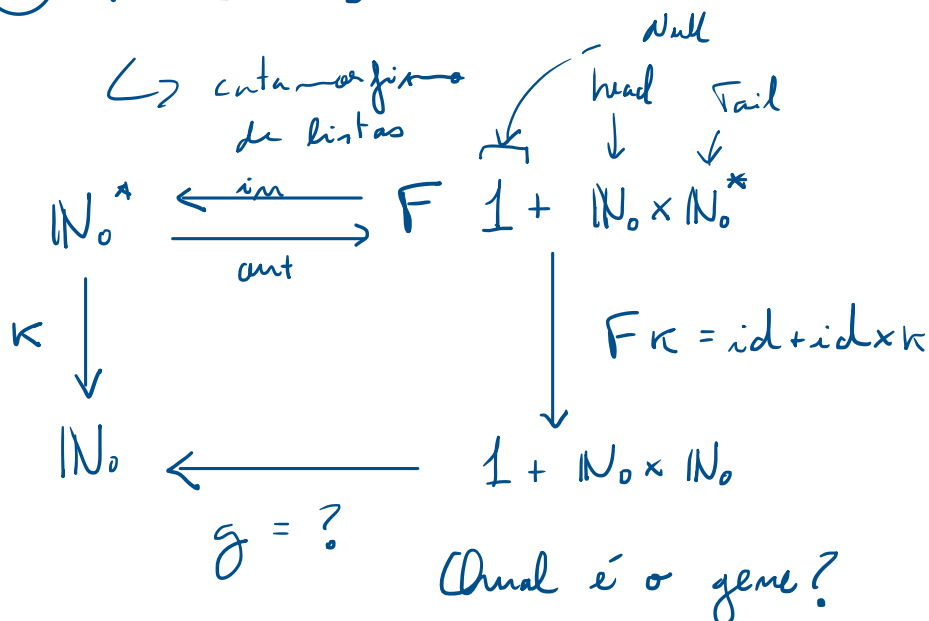
$$\kappa = [g] \Leftrightarrow \kappa \cdot \text{in} = g \cdot F\kappa$$

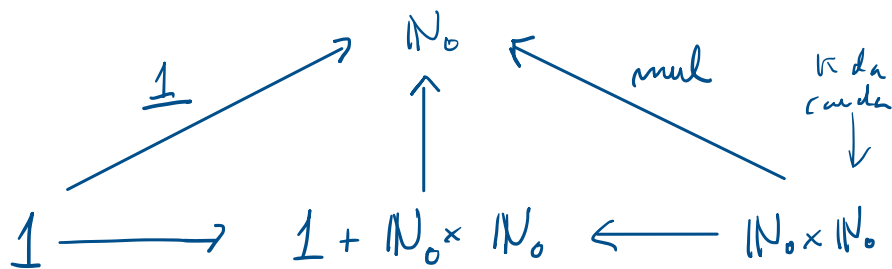


$$\text{rep True } 0 = []$$

$$\text{rep True } 1 = [\text{True}]$$

③ a)  $\kappa [1,2,3] = 6$



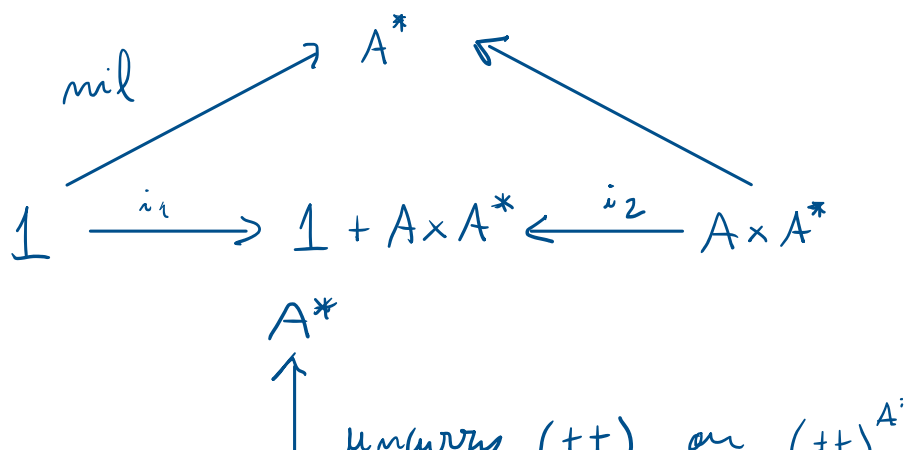
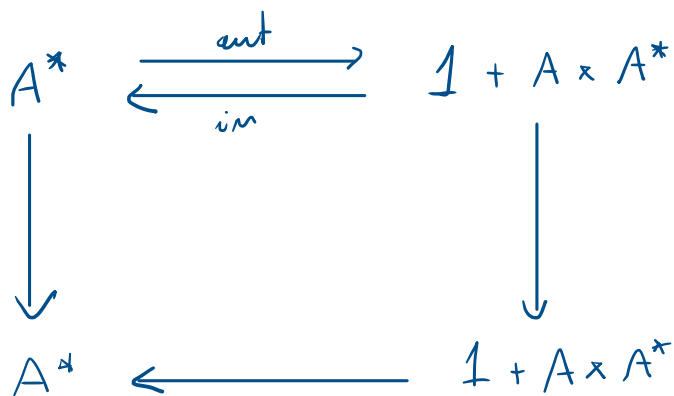


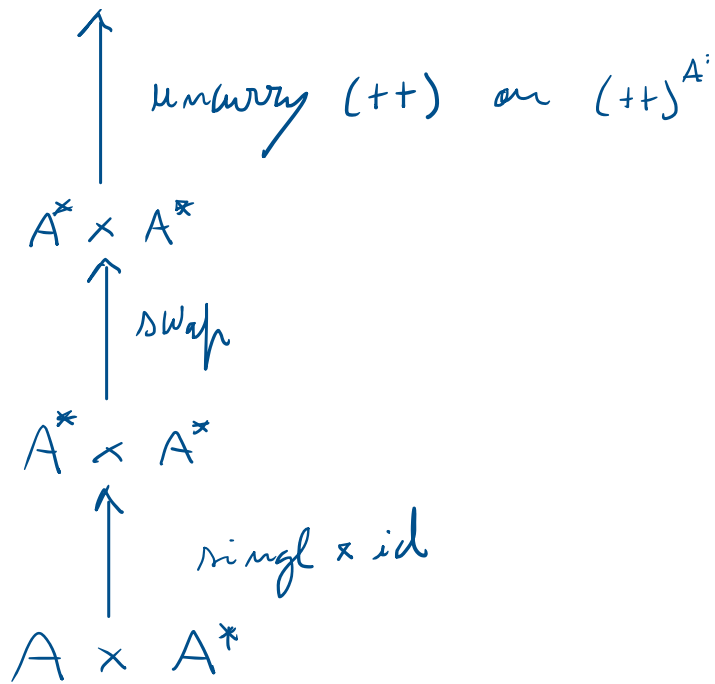
```
ghci> mycata g = g . (id -|- (mycata g)) . outNat
ghci> rep a = mycata (either nil (a:))
ghci> :t rep
rep :: Integral c => a -> c -> [a]
ghci> rep 5 2
[5,5]
ghci> rep 5 5
[5,5,5,5,5]
ghci> :l List.hs
[1 of 3] Compiling Cp                ( Cp.hs, interpreted )
[2 of 3] Compiling Nat                ( Nat.hs, interpreted )
[3 of 3] Compiling List               ( List.hs, interpreted )
Ok, three modules loaded.
ghci> mycataL g = g . (id -|- (id >< (mycataL g))) . outList
ghci> k = mycataL (either (const 1) mul)
ghci> k [1,2,3,4]
24
```

Natsumis

Lintas

b) Reverse



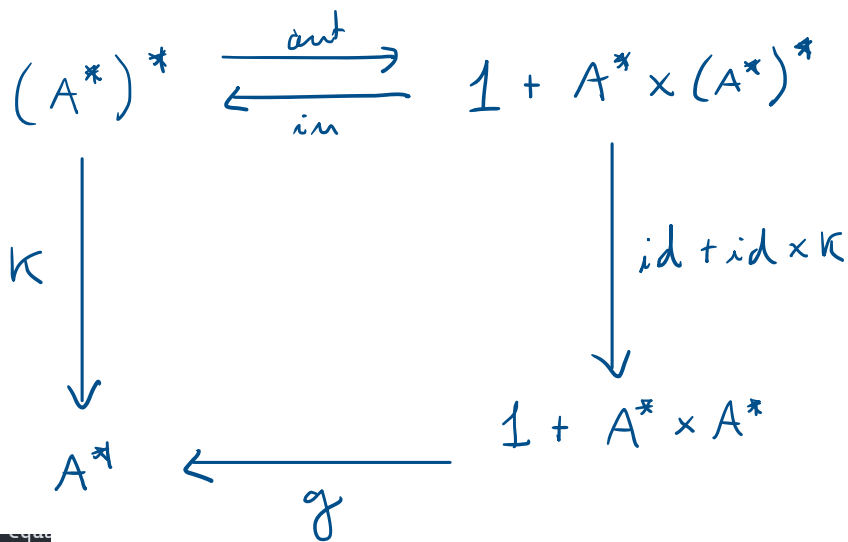


```

ghci> myReverse = mycataL (either nil (uncurry (++)) . swap . (singl >< id)))
ghci> myReverse [1,2,3,4]
[4,3,2,1]
ghci>

```

c) concat  $[[1,2]] = [1,2]$



```

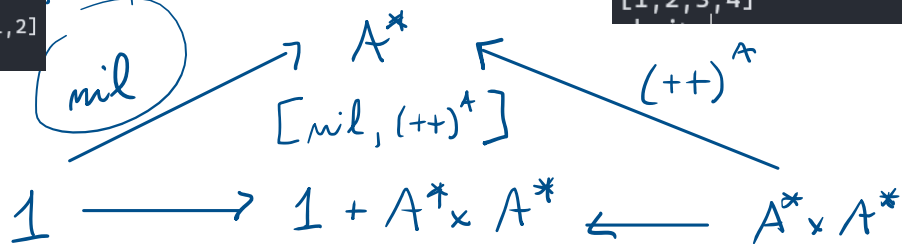
ghci> nil 1
[]
ghci> nil 123
[]
ghci> nil [1,2]
[]

```

```

ghci> uncurry (++) ([1,2], [3,4])
[1,2,3,4]

```



$$ds \text{ map } f :: A^* \rightarrow B^*$$

$$f: A \rightarrow B$$

$$\begin{array}{ccc} A^* & \begin{array}{c} \xleftarrow{\text{in}} \\ \xrightarrow{\text{out}} \end{array} & 1 + A \times A^* \\ \text{map } f \downarrow & & \downarrow \text{id} + \text{id} \times (\text{map } f) \\ B^* & \xleftarrow{g} & 1 + A \times B^* \end{array}$$

Vai até aos  
casos mais atômicos  
e esses resultados  
servirão como input  
para as chamadas  
anteriores

$$\begin{array}{c} B^* \\ \uparrow \text{cons} \\ (\hat{\cdot}) \\ B \times B^* \\ \uparrow f \times \text{id} \\ A \times B^* \end{array}$$

"colapsar até ao  
início depois de  
atingir o fim"

(pode não estar  
correto, é a minha  
interpretação)

↳ CATA M O R F I S M O !

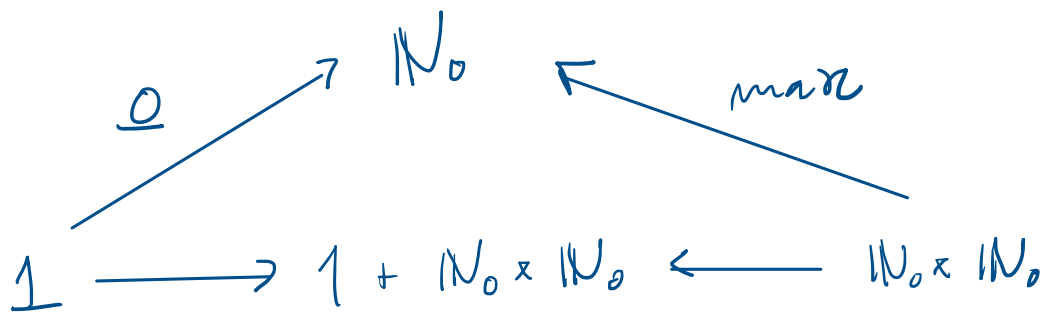
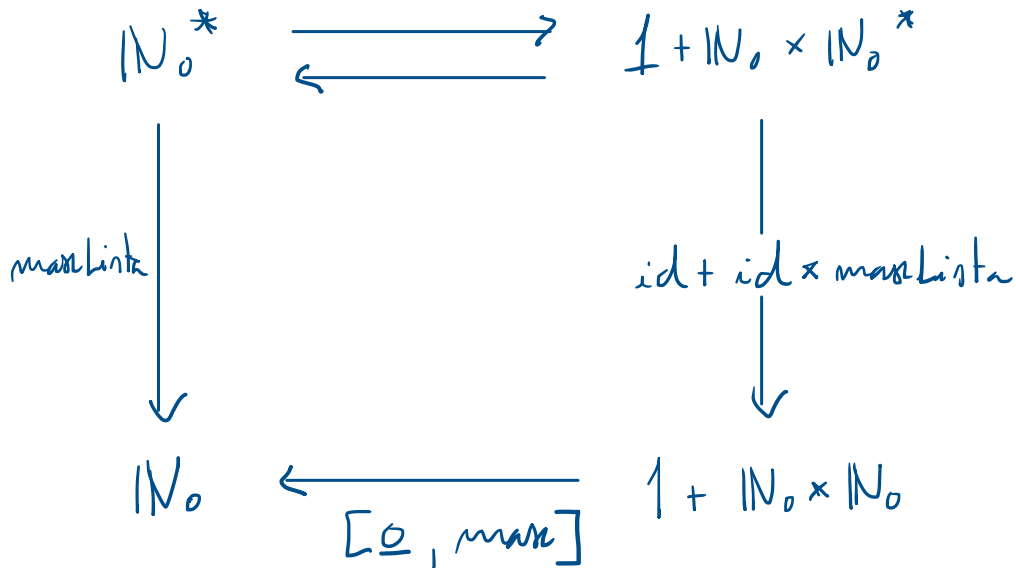
$$\begin{array}{ccccc} & & B^* & & \\ & \nearrow & & \nwarrow & \\ \text{nil} & & & & (\hat{\cdot})^A \cdot (f \times \text{id}) \\ & \searrow & & \swarrow & \\ & & [ \dots ] & & \\ 1 & \longrightarrow & 1 + A \times B^* & \longleftarrow & A \times B^* \end{array}$$

```

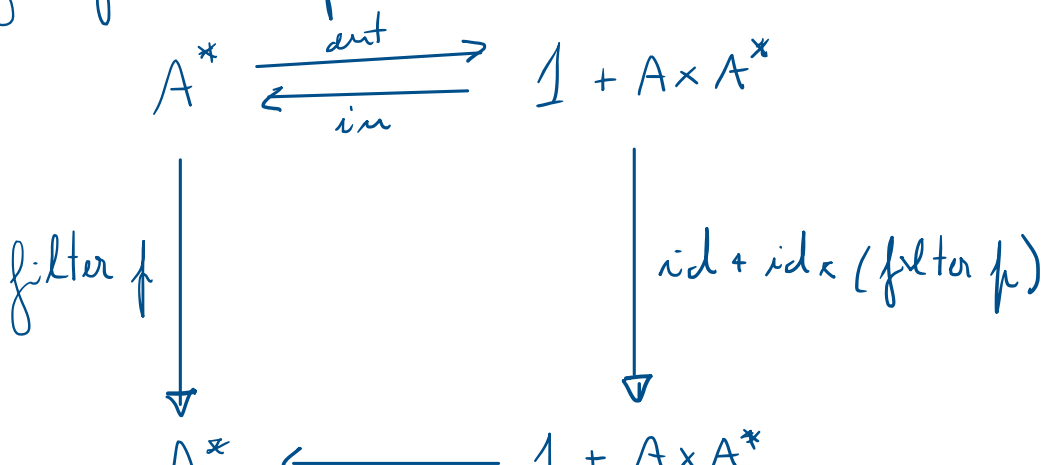
ghci> mymap f = mycataL (either nil (uncurry (:)) . (f >< id)))
ghci> f x = x + 1
ghci> mymap f [1,2,3]
[2,3,4]
ghci> |

```

i)  $\text{maxListL}$



f)  $\text{filter } f$



$$\begin{array}{ccc} \Downarrow & & \Downarrow \\ A^* & \longleftarrow & 1 + A \times A^* \\ & & [\text{nil}, [\pi_2, (\cdot)^A] \cdot \text{distl} \cdot (p? \times \text{id})] \end{array}$$

$$\begin{array}{c} A^* \\ \uparrow [\pi_2, (\cdot)^A] \\ (A \times A^*) + (A \times A^*) \\ \uparrow \text{distl} \\ (A+A) \times A^* \\ p? \uparrow \times \text{id} \\ A \times A^* \end{array}$$

```
filter = mycata (either nil ((either p2 (uncurry (·))) . distl . (p? >< id)))
```

4

$$\begin{array}{l} \frac{\text{função}}{f \cdot \langle g \rangle} = \frac{\text{cata}}{\langle h \rangle} \\ \equiv \{ \quad \text{46 (Universal - cata)} \quad \} \\ f \cdot \langle g \rangle \cdot \text{in} = h \cdot F(f \cdot \langle g \rangle) \\ \equiv \{ \quad \text{47, 50} \quad \} \\ f \cdot g \cdot (\text{id} + \langle g \rangle) = h \cdot (\text{id} + f \cdot \langle g \rangle) \\ \equiv \{ \quad \text{1} \quad \} \\ f \cdot g \cdot (\text{id} + \langle g \rangle) = h \cdot (\text{id} \cdot \text{id} + f \cdot \langle g \rangle) \\ \equiv \{ \quad \text{25} \quad \} \\ f \cdot g \cdot (\text{id} + \langle g \rangle) = h \cdot (\text{id} + f) \cdot (\text{id} + \langle g \rangle) \\ \Leftarrow \{ \quad \text{5} \quad \} \\ f \cdot g = h \cdot (\text{id} + f) \end{array}$$

$$f \cdot \langle g \rangle \cdot \text{in} = h \cdot F(f \cdot \langle g \rangle)$$

⑤

sumprod  $a = (a^*) \cdot \text{sum}$

$$\Rightarrow ([Zero, add \cdot ((a^*) \times id)]) = (a^*) \cdot ([Zero, add])$$

$$49 \Rightarrow (a^*) \cdot [Zero, add] = [Zero, add \cdot ((a^*) \times id)] \cdot F(add)$$

$$\Rightarrow (a^*) \cdot [Zero, add] = [Zero, add \cdot ((a^*) \times id)] \cdot (id + id \times y)$$

$$20, 21, 1 \Rightarrow [(a^*) \cdot Zero, (a^*) \cdot add] = [Zero, add \cdot ((a^*) \times id) \cdot ((id) \times (a^*))]$$

$$27 \Rightarrow \begin{cases} (a^*) \cdot Zero = Zero \\ (a^*) \cdot add = add \cdot ((a^*) \times id) \cdot (id \times (a^*)) \end{cases}$$

$$\Rightarrow \begin{cases} \forall x \mid ((a^*) \cdot add)(x, y) = (add \cdot ((a^*) \times id))(x, y) \\ \forall x, y \mid ((a^*) \cdot add)(x, y) = (add \cdot ((a^*) \times (a^*))) (x, y) \end{cases}$$

$$\Rightarrow \begin{cases} \forall x \mid a \times 0 = 0 \\ \forall x, y \mid a \times (x + y) = add(a \times x, a \times y) \end{cases}$$

$$\Rightarrow \text{TRUE}$$