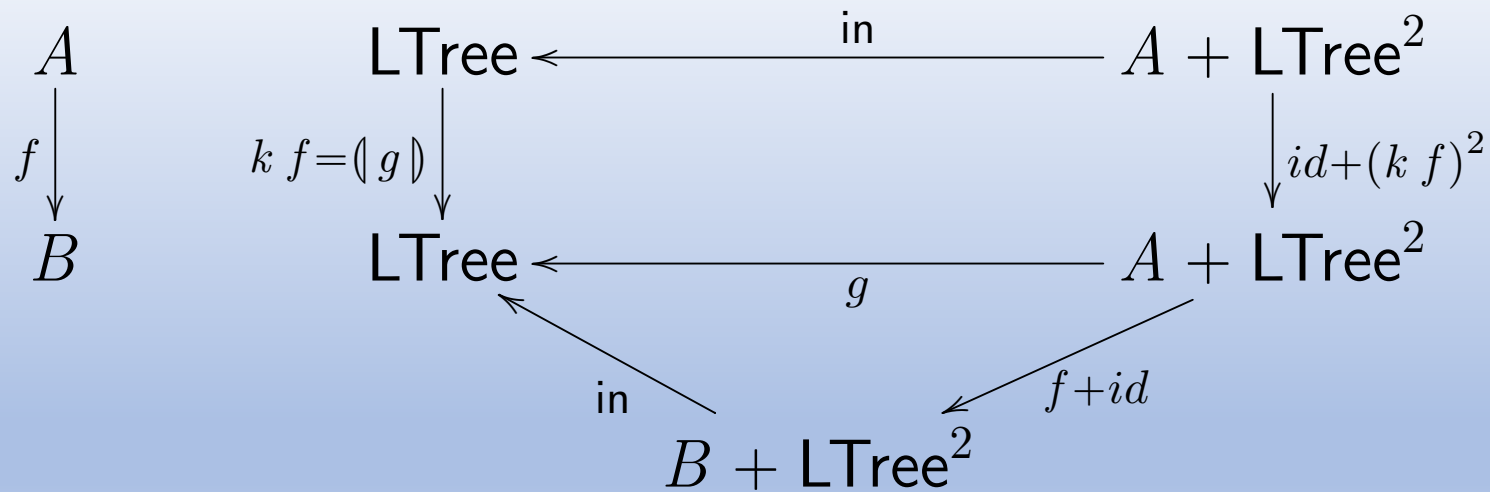


Cálculo de Programas T10

LTREE TYPE FUNCTOR

$$\begin{array}{ccc} A & \cdots & \text{LTree } A \\ \downarrow f & & \downarrow \text{LTree } f = (\text{in} \cdot (f + id)) \\ B & \cdots & \text{LTree } B \end{array}$$

$$k f \cdot \text{in} = \text{in} \cdot (f + id) \cdot (id + k f \times k f)$$



LTREE TYPE FUNCTOR

$$\begin{array}{ccc} \text{LTree } A & \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} & \underbrace{A + (\text{LTree } A)^2}_{\mathbf{F}(\text{LTree } A)} \end{array}$$

$$\begin{array}{ccc} \text{LTree } B & \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} & \underbrace{B + (\text{LTree } B)^2}_{\mathbf{F}(\text{LTree } B)} \end{array}$$

LTREE TYPE FUNCTOR

$$\mathbf{F} X = A + X^2 \quad ?$$

$$\mathbf{F} X = B + X^2 \quad ?$$

$$\begin{array}{ccc} \text{LTree } A & \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} & \underbrace{A + (\text{LTree } A)^2}_{\mathbf{F}(\text{LTree } A)} \\ \\ \text{LTree } B & \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} & \underbrace{B + (\text{LTree } B)^2}_{\mathbf{F}(\text{LTree } B)} \end{array}$$

LTREE base bifunctor **B**

$$\text{LTree } X \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} \underbrace{X + (\text{LTree } X)^2}_{\mathbf{B}(X, \text{LTree } X)}$$

$$\mathbf{B}(X, Y) = X + Y^2$$

BIFUNCTORS

$$\begin{array}{ccccc} A & \cdots & C & \cdots & \mathbf{B} (A, C) \\ f \downarrow & & g \downarrow & & \downarrow \mathbf{B} (f, g) \\ D & \cdots & E & \cdots & \mathbf{B} (D, E) \end{array}$$

BIFUNCTORS (Laws)

$$\begin{array}{ccccc} A & \cdots & C & \cdots & \mathbf{B} (A, C) \\ f \downarrow & & g \downarrow & & \downarrow \mathbf{B} (f, g) \\ D & \cdots & E & \cdots & \mathbf{B} (D, E) \end{array}$$

$$\mathbf{B} (id, id) = id$$

$$\mathbf{B} (h \cdot f, k \cdot g) = \mathbf{B} (h, k) \cdot \mathbf{B} (f, g)$$

BIFUNCTORS (Example)

$$\mathbf{B} (X, Y) = X \times Y$$

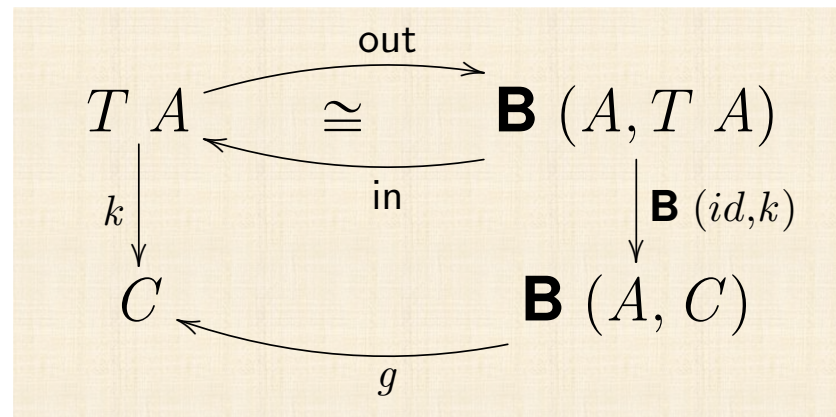
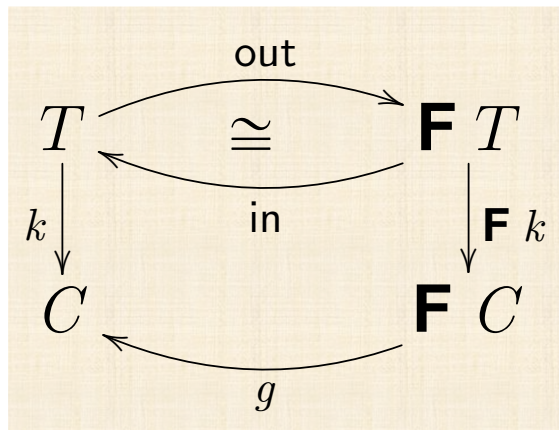
$$\mathbf{B} (f, g) = f \times g$$

$$\begin{array}{ccccc} A & \cdots & C & \cdots & A \times C \\ f \downarrow & & g \downarrow & & \downarrow f \times g \\ D & \cdots & E & \cdots & D \times E \end{array}$$

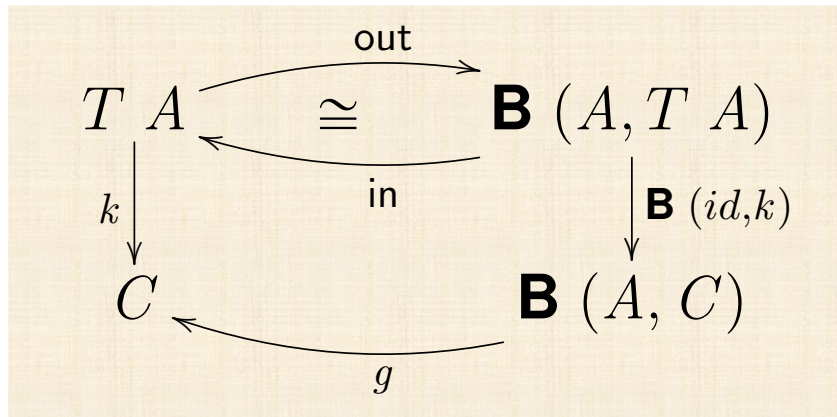
$$id \times id = id$$

$$h \cdot f \times k \cdot g = h \times k \cdot f \times g$$

CATAMORPHISMS (parameterization)



CATAMORPHISMS (in general)



Universal property

$$k = \langle g \rangle \Leftrightarrow k \cdot \text{in} = g \cdot \mathbf{B}(id, k)$$

Type functor:

$$T f = \langle \text{in} \cdot \mathbf{B}(f, id) \rangle$$

Abbreviation:

$$\mathbf{F} k = \mathbf{B}(id, k)$$

CATAMORPHISMS (Laws)

Universal-cata	$k = \llbracket g \rrbracket \Leftrightarrow k \cdot \text{in} = g \cdot F k$	(46)
----------------	-------------------------------------------------------------------------------	------

Cancelamento-cata	$\llbracket g \rrbracket \cdot \text{in} = g \cdot F \llbracket g \rrbracket$	(47)
-------------------	-------------------------------------------------------------------------------	------

Reflexão-cata	$\llbracket \text{in} \rrbracket = id_{\top}$	(48)
---------------	-----------------------------------------------	------

Fusão-cata	$f \cdot \llbracket g \rrbracket = \llbracket h \rrbracket \Leftarrow f \cdot g = h \cdot F f$	(49)
------------	------------------------------------------------------------------------------------------------	------

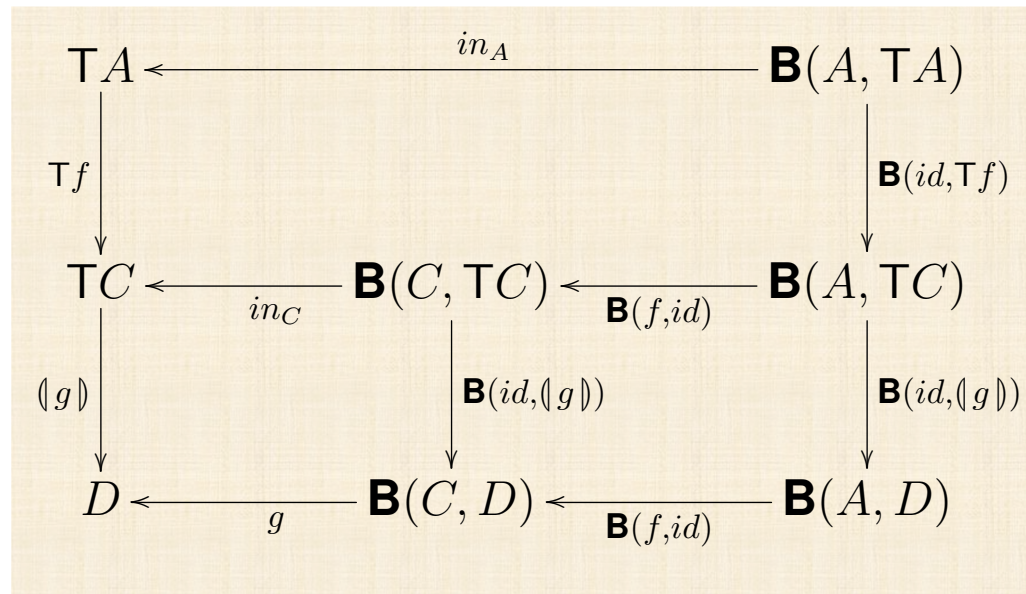
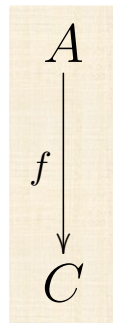
Base-cata	$F f = B(id, f)$	(50)
-----------	------------------	------

Def-map-cata	$\top f = \llbracket \text{in} \cdot B(f, id) \rrbracket$	(51)
--------------	-----------------------------------------------------------	------

Absorção-cata	$\llbracket g \rrbracket \cdot \top f = \llbracket g \cdot B(f, id) \rrbracket$	(52)
---------------	---------------------------------------------------------------------------------	------

Cata-absorption

$$\llbracket g \rrbracket \cdot \top f = \llbracket g \cdot \mathbf{B}(f, id) \rrbracket$$



Cata-absorption

$$\langle g \rangle \cdot \top f = \langle g \cdot \mathbf{B}(f, id) \rangle$$

$$\begin{aligned} & \langle g \rangle \cdot \top f = \langle g \cdot \mathbf{B}(f, id) \rangle \\ \equiv & \quad \{ \text{type-functor definition} \} \\ & \langle g \rangle \cdot \langle \text{in} \cdot \mathbf{B}(f, id) \rangle = \langle g \cdot \mathbf{B}(f, id) \rangle \\ \Leftarrow & \quad \{ \text{cata-fusion} \} \\ & \langle g \rangle \cdot \text{in} \cdot \mathbf{B}(f, id) = g \cdot \mathbf{B}(f, id) \cdot \mathbf{B}(id, \langle g \rangle) \\ \equiv & \quad \{ \mathbf{B} \text{ is a bi-functor} \} \\ & \langle g \rangle \cdot \text{in} \cdot \mathbf{B}(f, id) = g \cdot \mathbf{B}(id, \langle g \rangle) \cdot \mathbf{B}(f, id) \\ \Leftarrow & \quad \{ \text{Leibniz} \} \\ & \langle g \rangle \cdot \text{in} = g \cdot \mathbf{B}(id, \langle g \rangle) \\ \equiv & \quad \{ \text{cata-cancellation} \} \\ & \text{TRUE} \end{aligned}$$

Type-functor

$$\mathbf{T} f = (\text{in} \cdot \mathbf{B} (f, id))$$

$$\begin{aligned} & \mathbf{T} (h \cdot f) \\ = & \quad \{ \text{definition of } \mathbf{T} \} \\ & (\text{in} \cdot \mathbf{B} (h \cdot f, id)) \\ = & \quad \{ \text{bifunctor } \mathbf{B} \} \\ & (\text{in} \cdot \underbrace{\mathbf{B} (h, id) \cdot \mathbf{B} (f, id)}_g) \\ = & \quad \{ \text{cata-absorption: } (\text{in } g) \cdot \mathbf{T} f = (\text{in } g \cdot \mathbf{B}(f, id)) \} \\ & (\text{in} \cdot \mathbf{B} (h, id)) \cdot \mathbf{T} f \\ = & \quad \{ \text{definition of } \mathbf{T} \} \\ & \mathbf{T} h \cdot \mathbf{T} f \end{aligned}$$

Type-functor

$$\mathbf{T} f = (\text{in} \cdot \mathbf{B} (f, id))$$

$$\begin{aligned}
 & \mathbf{T} (h \cdot f) \\
 = & \quad \{ \text{definition of } \mathbf{T} \} \\
 & (\text{in} \cdot \mathbf{B} (h \cdot f, id)) \\
 = & \quad \{ \text{bifunctor } \mathbf{B} \} \\
 & (\text{in} \cdot \underbrace{\mathbf{B} (h, id) \cdot \mathbf{B} (f, id)}_g) \\
 = & \quad \{ \text{cata-absorption: } (\text{in } g) \cdot \mathbf{T} f = (\text{in } g \cdot \mathbf{B} (f, id)) \} \\
 & (\text{in} \cdot \mathbf{B} (h, id)) \cdot \mathbf{T} f \\
 = & \quad \{ \text{definition of } \mathbf{T} \} \\
 & \mathbf{T} h \cdot \mathbf{T} f
 \end{aligned}$$

(a) Trees with data in their leaves :

$$T = \text{LTree } A \quad \begin{cases} F X = A + X^2 \\ F f = id + f^2 \end{cases} \quad \text{in} = [\text{Leaf}, \text{Fork}]$$

Haskell: `data LTree a = Leaf a | Fork (LTree a, LTree a)`

(b) Trees whose data of type A are stored in their nodes:

$$T = \text{BTree } A \quad \begin{cases} F X = 1 + A \times X^2 \\ F f = id + id \times f^2 \end{cases} \quad \text{in} = [\underline{\text{Empty}}, \text{Node}]$$

Haskell: `data BTree a = Empty | Node (a, (BTree a, BTree a))`

(c) Full trees — data in both leaves and nodes:

$$T = \text{FTree } B A \quad \begin{cases} F X = B + A \times X^2 \\ F f = id + id \times f^2 \end{cases} \quad \text{in} = [\text{Unit}, \text{Comp}]$$

Haskell: `data FTree b a = Unit b | Comp (a, (FTree b a, FTree b a))`

(d) Expression trees:

$$T = \text{Expr } V O \quad \begin{cases} F X = V + O \times X^* \\ F f = id + id \times \text{map } f \end{cases} \quad \text{in} = [\text{Var}, \text{Term}]$$

Haskell: `data Expr v o = Var v | Term (o, [Expr v o])`

(a) Trees with data in their leaves :

$$T = \text{LTree } A \quad \left\{ \begin{array}{l} \text{B } (X, Y) = X + Y^2 \\ \text{B } (g, f) = g + f^2 \end{array} \right. \quad \text{in} = [\text{Leaf}, \text{Fork}]$$

Haskell: `data LTree a = Leaf a | Fork (LTree a, LTree a)`

(b) Trees whose data of type A are stored in their nodes:

$$T = \text{BTree } A \quad \left\{ \begin{array}{l} \text{B } (X, Y) = 1 + X \times Y^2 \\ \text{B } (g, f) = id + g \times f^2 \end{array} \right. \quad \text{in} = [\underline{\text{Empty}}, \text{Node}]$$

Haskell: `data BTree a = Empty | Node (a, (BTree a, BTree a))`

(c) Full trees — data in both leaves and nodes:

$$T = \text{FTree } B \ A \quad \left\{ \begin{array}{l} \text{B } (Z, X, Y) = Z + X \times Y^2 \\ \text{B } (h, g, f) = h + g \times f^2 \end{array} \right. \quad \text{in} = [\text{Unit}, \text{Comp}]$$

Haskell: `data FTree b a = Unit b | Comp (a, (FTree b a, FTree b a))`

(d) Expression trees:

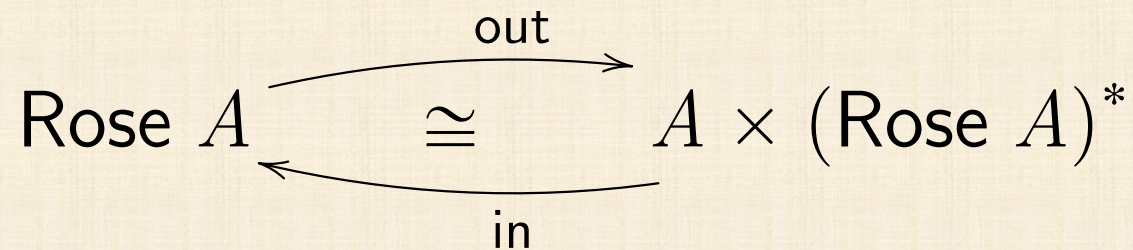
$$T = \text{Expr } V \ O \quad \left\{ \begin{array}{l} \text{B } (Z, X, Y) = Z + X \times Y^* \\ \text{B } (h, g, f) = h + g \times \text{map } f \end{array} \right. \quad \text{in} = [\text{Var}, \text{Term}]$$

Haskell: `data Expr v o = Var v | Term (o, [Expr v o])`

`data` Rose a = Rose a [Rose a] `deriving` Show

`inRose` = `uncurry` Rose

`outRose` (Rose a x) = (a,x)



$$\text{Rose } A \begin{array}{c} \xrightarrow{\text{out}} \\ \cong \\ \xleftarrow{\text{in}} \end{array} A \times (\text{Rose } A)^*$$

$$\mathbf{B} (X, Y) = X \times Y^*$$

$$\mathbf{B} (f, g) = f \times g^*$$

$$\begin{array}{c} A \times Y^* \\ X \times Y^* \end{array}$$



```
data Rose a = Rose a [Rose a] deriving Show
```

```
inRose = uncurry Rose
```

```
outRose (Rose a x) = (a,x)
```

$$\mathbf{B} (X, Y) = X \times Y^*$$

$$\mathbf{B} (f, g) = f \times g^*$$

f >< map g





```
data Rose a = Rose a [Rose a] deriving Show
```

```
inRose = uncurry Rose
```

```
outRose (Rose a x) = (a,x)
```



```
data Rose a = Rose a [Rose a] deriving Show
```

```
inRose = uncurry Rose
```

```
outRose (Rose a x) = (a,x)
```

```
baseRose f g = f >< map g
```

```
recRose g = baseRose id g
```



```
data Rose a = Rose a [Rose a] deriving Show
```

```
inRose = uncurry Rose
```

```
outRose (Rose a x) = (a,x)
```

```
baseRose f g = f >< map g
```

```
recRose g = baseRose id g
```

```
cataRose g = g . (recRose (cataRose g)) . outRose
```



```
data Rose a = Rose a [Rose a] deriving Show
```

```
inRose = uncurry Rose
```

```
outRose (Rose a x) = (a,x)
```

```
baseRose f g = f >< map g
```

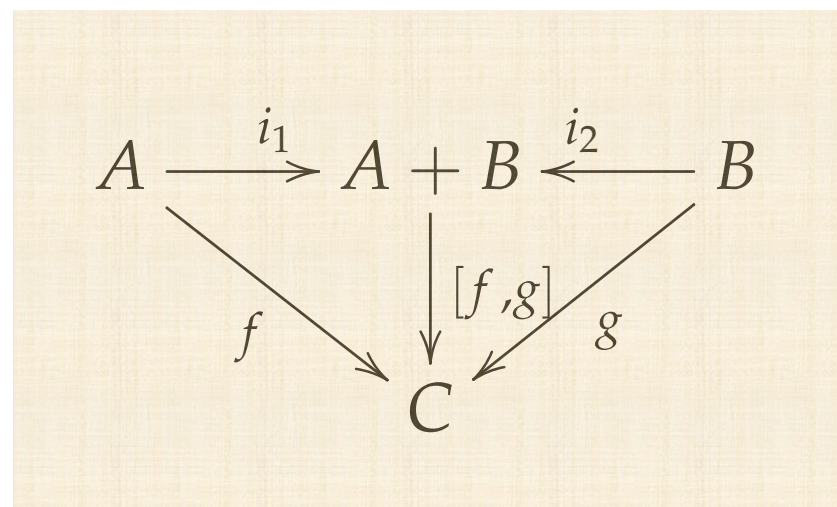
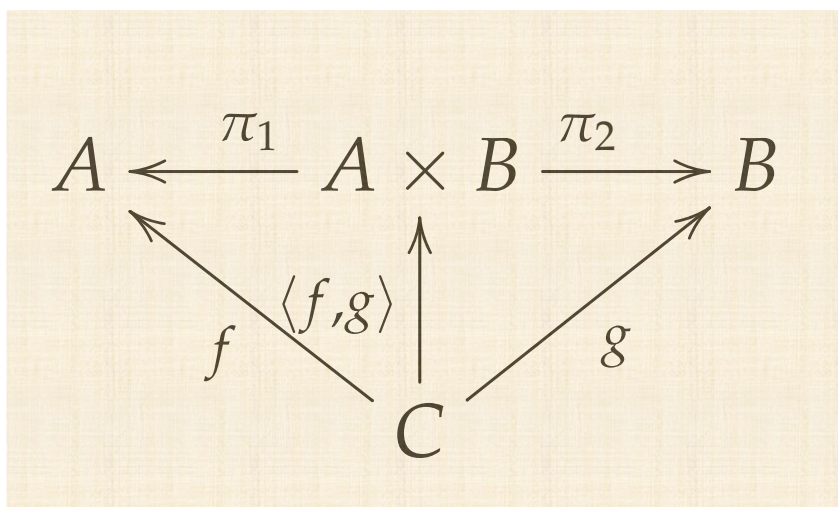
```
recRose g = baseRose id g
```

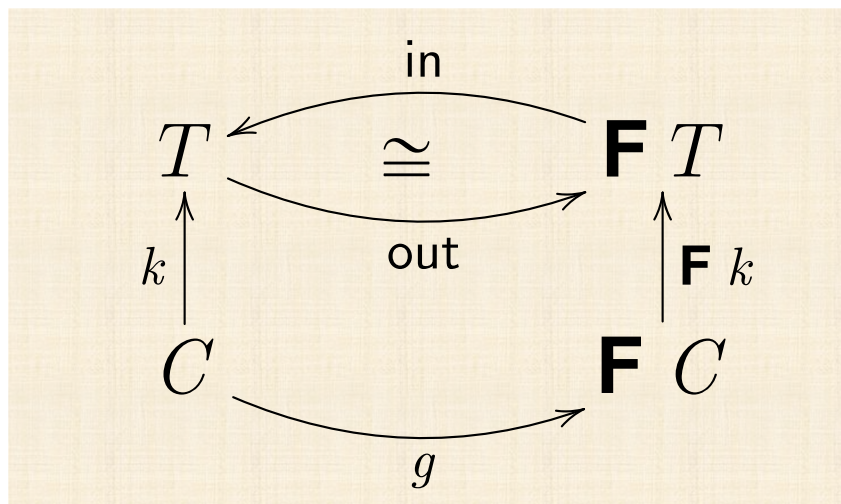
```
cataRose g = g . (recRose (cataRose g)) . outRose
```

```
instance Functor Rose
```

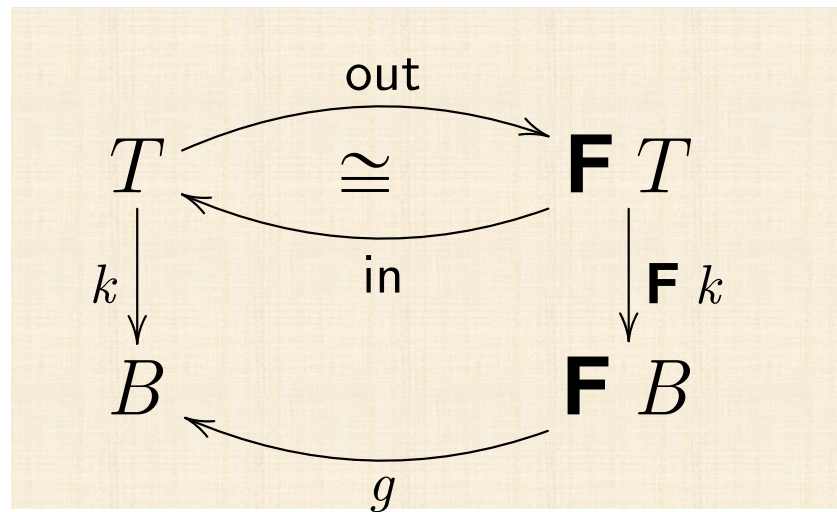
```
  where fmap f = cataRose ( inRose . baseRose f id )
```

Anamorphisms

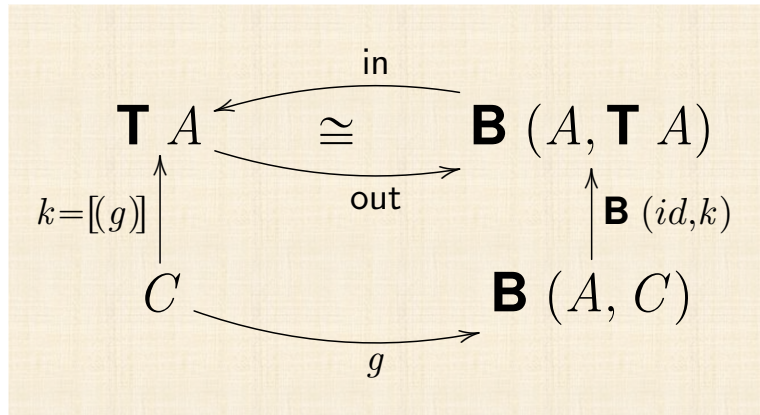




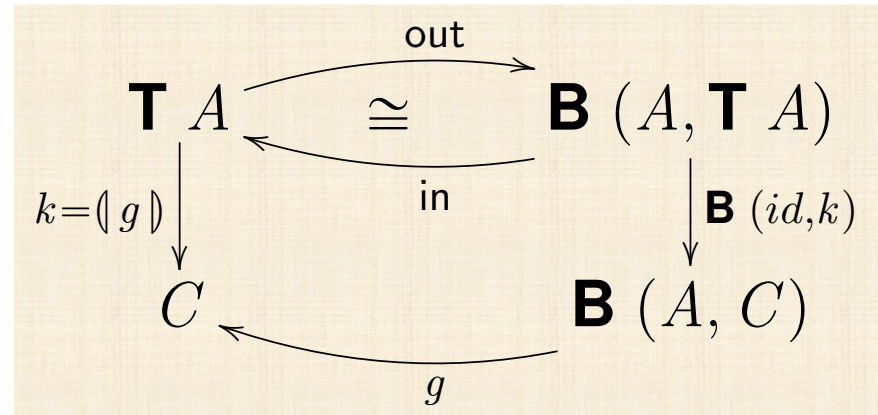
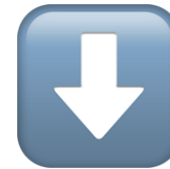
$\alpha\upsilon\alpha$ (ana)



$\kappa\alpha\tau\alpha$ (cata)

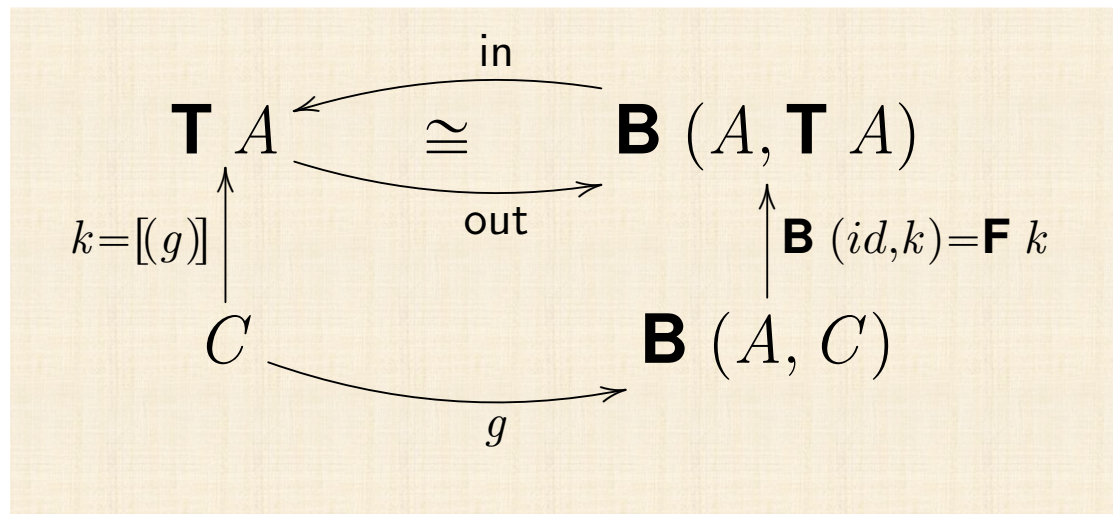


$\alpha\eta\alpha$ (ana)



$\kappa\alpha\tau\alpha$ (cata)

ANAMORPHISMS



$$k = \llbracket g \rrbracket \iff \text{in} \cdot \mathbf{F} k \cdot g$$

ANAMORPHISM LAWS

Universal-ana

$$k = \llbracket g \rrbracket \Leftrightarrow \text{out} \cdot k = (\mathbf{F} k) \cdot g$$

Cancelamento-ana

$$\text{out} \cdot \llbracket g \rrbracket = \mathbf{F} \llbracket g \rrbracket \cdot g$$

Reflexão-ana

$$\llbracket \text{out} \rrbracket = id_{\top}$$

Fusão-ana

$$\llbracket g \rrbracket \cdot f = \llbracket h \rrbracket \Leftarrow g \cdot f = (\mathbf{F} f) \cdot h$$

Base-ana

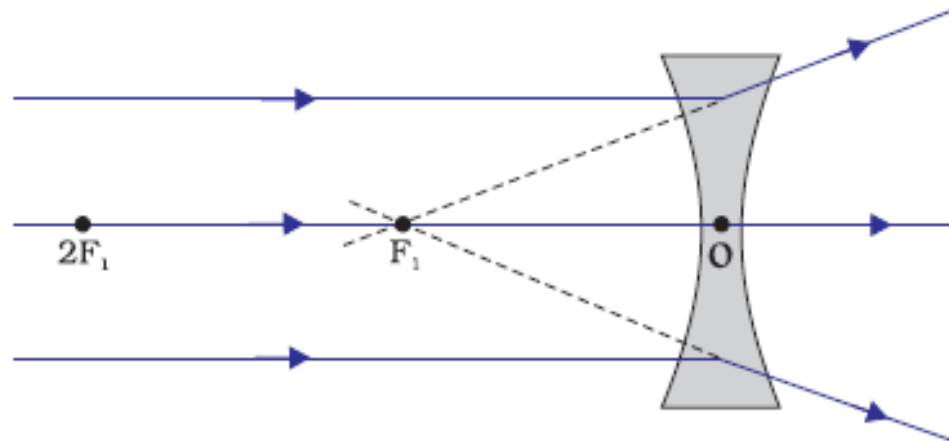
$$\mathbf{F} f = \mathbf{B} (id, f)$$

Def-map-ana

$$\top f = \llbracket (\mathbf{B}(f, id) \cdot \text{out}) \rrbracket$$

Absorção-ana

$$\top f \cdot \llbracket g \rrbracket = \llbracket (\mathbf{B}(f, id) \cdot g) \rrbracket$$



$[(-)]$

Example

$$\begin{array}{ccccc}
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* \\
 \uparrow k & & \uparrow id + id \times k \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} 1 + \mathbb{N}_0 & \xrightarrow{id + \langle \text{succ}, id \rangle} & 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{aligned}
& k = [(id + \langle succ, id \rangle) \cdot out_{\mathbb{N}_0}] \\
\equiv & \quad \{ \text{ana-universal} \} \\
& k = in \cdot (id + id \times k) \cdot (id + \langle succ, id \rangle) \cdot out_{\mathbb{N}_0} \\
\equiv & \quad \{ \text{isomorphism } in_{\mathbb{N}_0} / out_{\mathbb{N}_0} \} \\
& k \cdot in_{\mathbb{N}_0} = in \cdot (id + id \times k) \cdot (id + \langle succ, id \rangle) \\
\equiv & \quad \{ \text{functor-+; absorption-}\times \} \\
& k \cdot in_{\mathbb{N}_0} = in \cdot (id + \langle succ, k \rangle)
\end{aligned}$$

$$\equiv \quad \{ \text{definitions of } \text{in} \text{ and } \text{in}_{\mathbb{N}_0}; \text{ fusion and absorption-+ } \}$$

$$[k \cdot \underline{0}, k \cdot \text{succ}] = [\text{nil}, \text{cons} \cdot \langle \text{succ}, k \rangle]$$

$$\equiv \quad \{ \text{eq-+} \}$$

$$\begin{cases} k \cdot \underline{0} = \text{nil} \\ k \cdot \text{succ} = \text{cons} \cdot \langle \text{succ}, k \rangle \end{cases}$$

$$\equiv \quad \{ \text{going pointwise} \}$$

$$\begin{cases} k \ 0 = [] \\ k \ (n + 1) = (n + 1) : k \ n \end{cases}$$

$$\begin{array}{ccccc}
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* \\
 \uparrow k & & \uparrow id + id \times k \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} 1 + \mathbb{N}_0 & \xrightarrow{id + \langle \text{succ}, id \rangle} & 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{cases} k\ 0 = [] \\ k\ (n + 1) = (n + 1) : k\ n \end{cases}$$

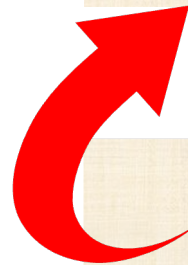
Hylomorphisms

$$\begin{array}{ccc}
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* \\
 \uparrow k & & \uparrow id + id \times k \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} 1 + \mathbb{N}_0 \xrightarrow{id + \langle \text{succ}, id \rangle} & 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{array}{ccc}
 \mathbb{N}_0 & \xleftarrow{[\underline{1}, \text{mul}]} & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \\
 \uparrow m & & \uparrow id + id \times m \\
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^*
 \end{array}$$

$$\begin{array}{ccccc}
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* & & \\
 \uparrow k & & \uparrow id + id \times k & & \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} & 1 + \mathbb{N}_0 & \xrightarrow{id + \langle \text{succ}, id \rangle} & 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{array}{ccc}
 \mathbb{N}_0 & \xleftarrow{[\underline{1}, \text{mul}]} & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \\
 \uparrow m & & \uparrow id + id \times m \\
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^*
 \end{array}$$



$$\begin{array}{ccccc}
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* & & \\
 \uparrow k & & \uparrow id + id \times k & & \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} & 1 + \mathbb{N}_0 & \xrightarrow{id + \langle \text{succ}, id \rangle} & 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{array}{ccc}
 \mathbb{N}_0 & \xleftarrow{[\underline{1}, \text{mul}]} & 1 + \mathbb{N}_0 \times \mathbb{N}_0 \\
 \uparrow m & & \uparrow id + id \times m \\
 \mathbb{N}_0^* & \xleftarrow{\text{in}} & 1 + \mathbb{N}_0 \times \mathbb{N}_0^* \\
 \uparrow k & & \uparrow id + id \times k \\
 \mathbb{N}_0 & \xrightarrow{\text{out}_{\mathbb{N}_0}} 1 + \mathbb{N}_0 & \xrightarrow{id + \langle \text{succ}, id \rangle} 1 + \mathbb{N}_0 \times \mathbb{N}_0
 \end{array}$$

$$\begin{aligned}
& f = m \cdot k \\
\equiv & \quad \left\{ m = \llbracket \underline{1}, \text{mul} \rrbracket \text{ and } k = \llbracket (id + \langle \text{succ}, id \rangle) \cdot \text{out}_{\mathbb{N}_0} \rrbracket \right\} \\
& f = \llbracket \underline{1}, \text{mul} \rrbracket \cdot \llbracket (id + \langle \text{succ}, id \rangle) \cdot \text{out}_{\mathbb{N}_0} \rrbracket \\
\equiv & \quad \left\{ \text{cancellations (ana and cata)} \right\} \\
& f = \llbracket \underline{1}, \text{mul} \rrbracket \cdot \mathbf{F} \, m \cdot \text{out} \cdot \text{in} \cdot \mathbf{F} \, k \cdot (id + \langle \text{succ}, id \rangle) \cdot \text{out}_{\mathbb{N}_0} \\
\equiv & \quad \left\{ \text{in} \cdot \text{out} = id ; \text{ functor } \mathbf{F}: (\mathbf{F} \, m) \cdot (\mathbf{F} \, k) = \mathbf{F} \, (m \cdot k) \right\} \\
& f = \llbracket \underline{1}, \text{mul} \rrbracket \cdot \mathbf{F} \, (m \cdot k) \cdot (id + \langle \text{succ}, id \rangle) \cdot \text{out}_{\mathbb{N}_0} \\
\equiv & \quad \left\{ \text{isomorphism } \text{in}_{\mathbb{N}_0} / \text{out}_{\mathbb{N}_0}; m \cdot k = f ; \mathbf{F} \, f = id + id \times f \right\} \\
& f \cdot \text{in}_{\mathbb{N}_0} = \llbracket \underline{1}, \text{mul} \rrbracket \cdot (id + id \times f) \cdot (id + \langle \text{succ}, id \rangle)
\end{aligned}$$

$$f \cdot \text{in}_{\mathbb{N}_0} = [\underline{1}, \text{mul}] \cdot (id + id \times f) \cdot (id + \langle \text{succ}, id \rangle)$$

$$\equiv \{ \text{+-absorption ; } \times\text{-absorption ; etc } \}$$

$$f \cdot \text{in}_{\mathbb{N}_0} = [\underline{1}, \text{mul} \cdot \langle \text{succ}, f \rangle]$$

$$\equiv \{ \text{Eq-+ ; } \text{in}_{\mathbb{N}_0} = [\underline{0}, \text{succ}] \}$$

$$\begin{cases} f \cdot \underline{0} = \underline{1} \\ f \cdot \text{succ} = \text{mul} \cdot \langle \text{succ}, f \rangle \end{cases}$$

$$\equiv \{ \text{go pointwise} \}$$

$$\begin{cases} f \ 0 = 1 \\ f \ (n + 1) = (n + 1) \times f \ n \end{cases}$$

Factorial (“rediscovered”)

$$\text{fac} = ([1, \text{mul}]) \cdot [((id + \langle \text{succ}, id \rangle) \cdot \text{out}_{N_0})]$$

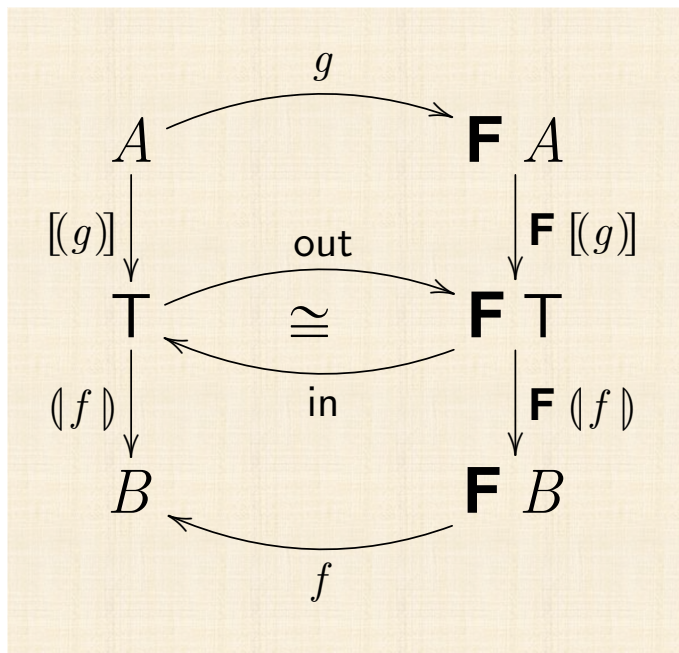
HILOMORPHISM

“Hylo + morphism”

ξύλο = matter, thing

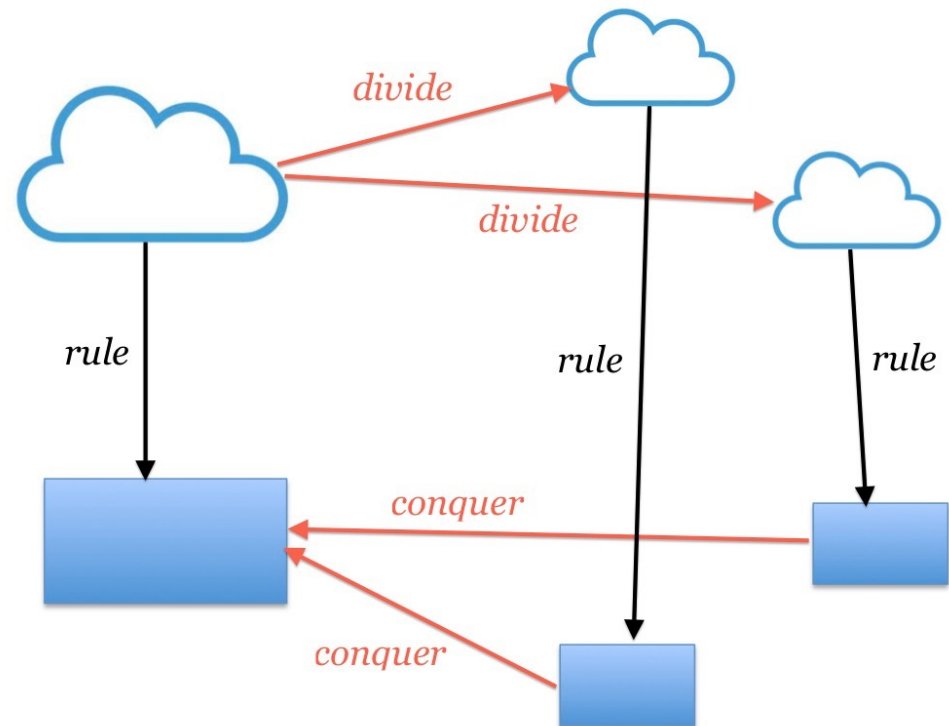
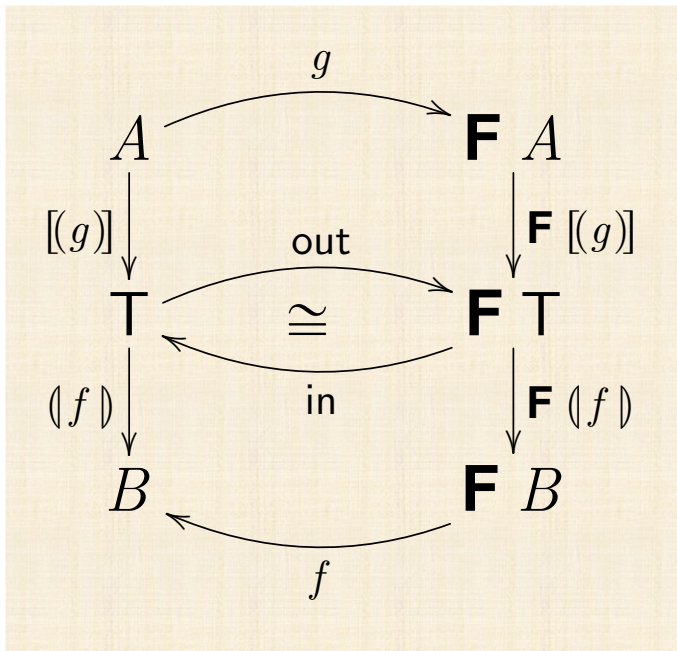
$$[[f, g]] = ([f]) \cdot [(g)]$$

HILOMORPHISM

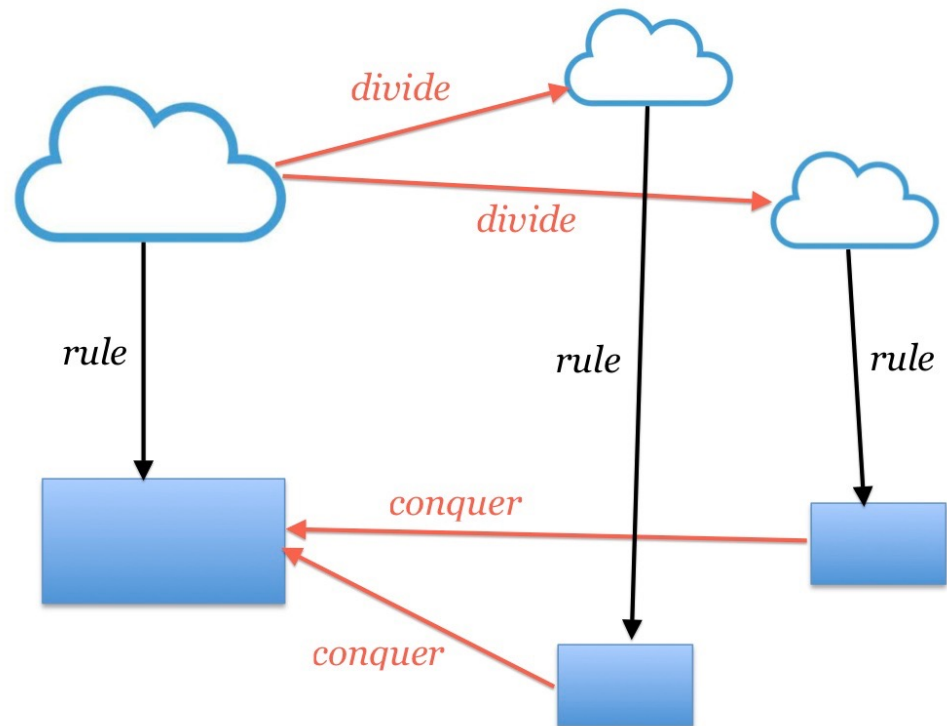
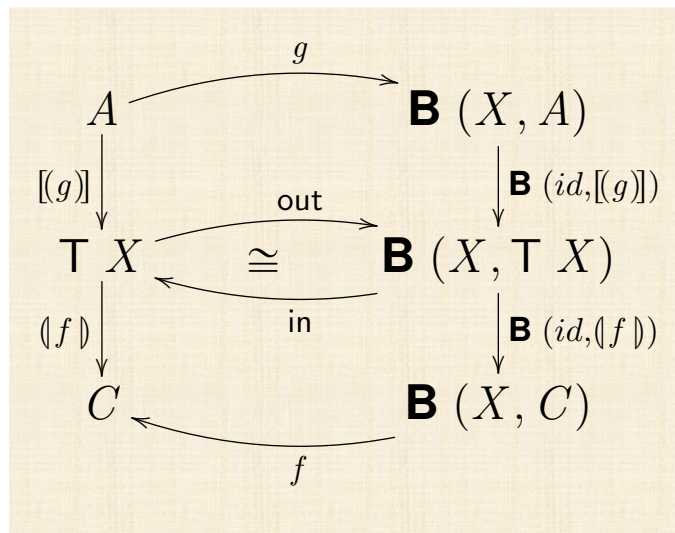


$$[[f, g]] = (f) \cdot [(g)]$$

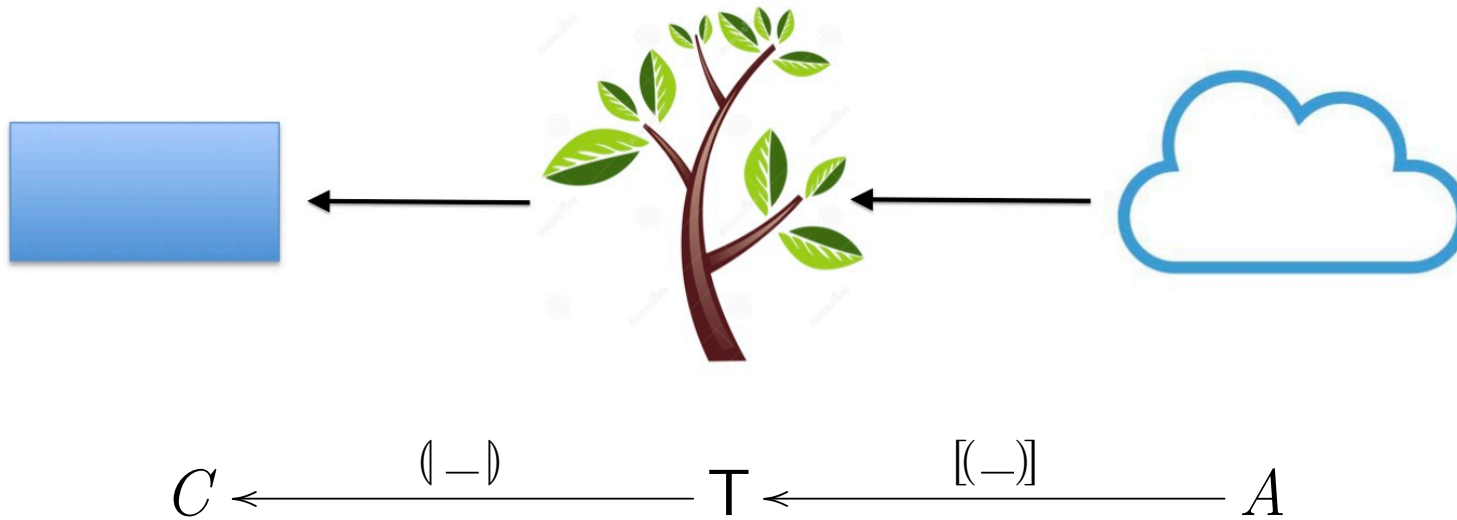
'DIVIDE & CONQUER'




'DIVIDE & CONQUER'



'DIVIDE & CONQUER'



ANA, CATA & HYLO


$$\begin{aligned} \llbracket \text{in}, g \rrbracket &= \llbracket (g) \rrbracket \\ \llbracket f, \text{out} \rrbracket &= \langle f \rangle \end{aligned}$$

Reflexion laws:

$$\langle \text{in} \rangle = id$$

$$\llbracket \text{out} \rrbracket = id$$

$$C \xleftarrow{\langle f \rangle} \top \xleftarrow{\llbracket g \rrbracket} A$$

$$\llbracket f, g \rrbracket = \langle f \rangle \cdot \llbracket g \rrbracket$$

Cálculo de Programas T11