Aula T2 - Anotações

25 de setembro de 2023 20:40

$(f \cdot g)\, a = f(g\, a)$ (2.6) $a \longrightarrow \boxed{g} \xrightarrow{g a} \boxed{f} \longrightarrow f(g\, a)$

Composição de funções

Split de funções: $\langle f, g \rangle \overset{(2.20)}{=} (f\, a, g\, a)$ $a \begin{cases} \to \boxed{f} \to f a \\ \to \boxed{g} \to g a \end{cases}$

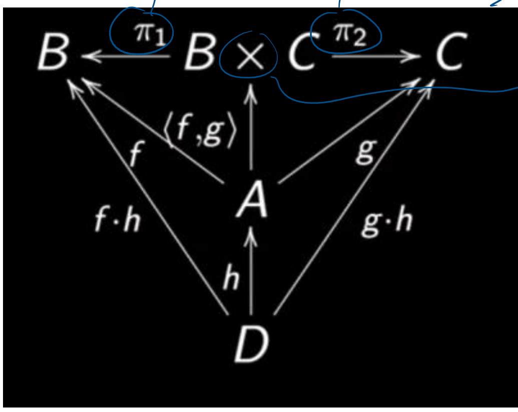
$f \times g = \langle f \cdot \pi_1, g \cdot \pi_2 \rangle$

par conceptual $\begin{cases} c \to \boxed{f} \to f c \\ d \to \boxed{g} \to g d \end{cases}$

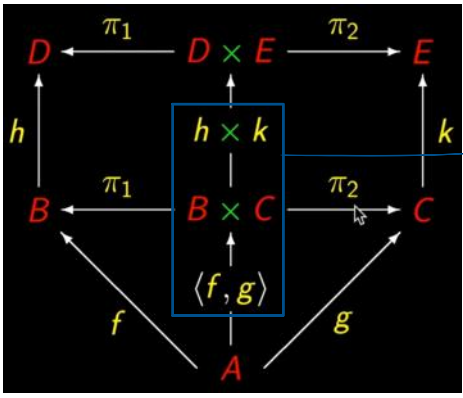
$\langle f, g \rangle \cdot h \Leftrightarrow \langle f \cdot h, g \cdot h \rangle$ (2.26) FUSÃO-X

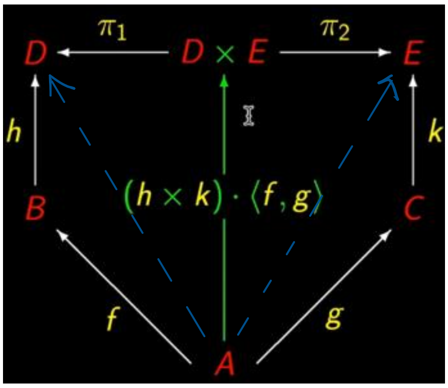
Projeções

Produto cartesiano

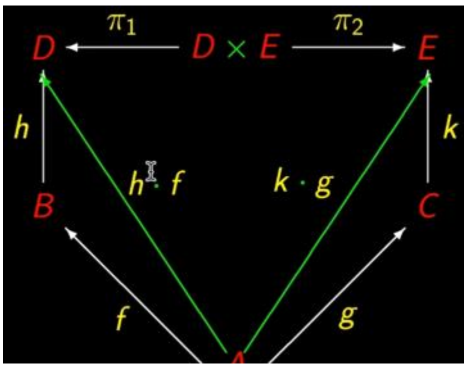


$\{1,2,3\} \times \{1,2\}$

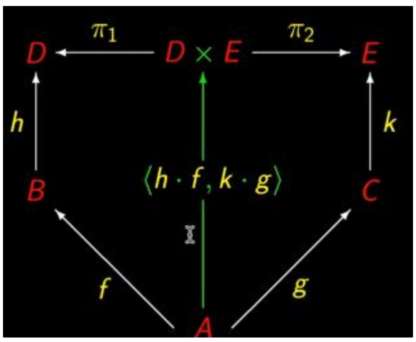


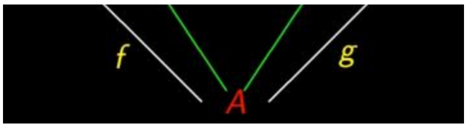


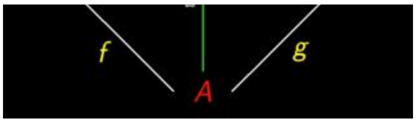
Produto cartesiano de (funções) $h \times k$ após split $\langle f, g \rangle$



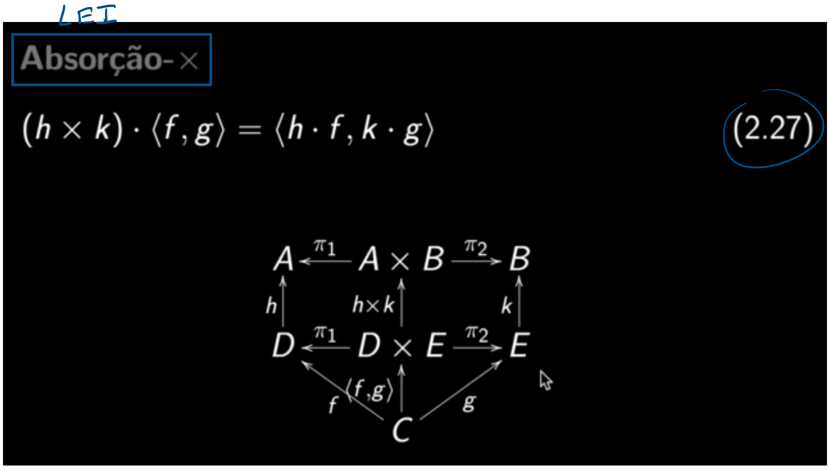
$\Longrightarrow$





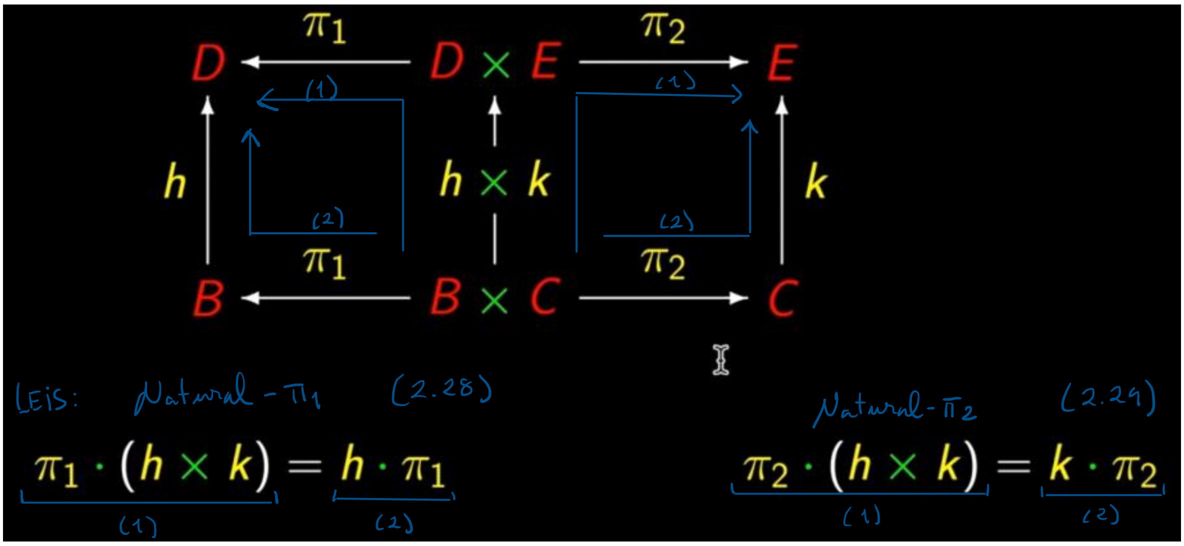


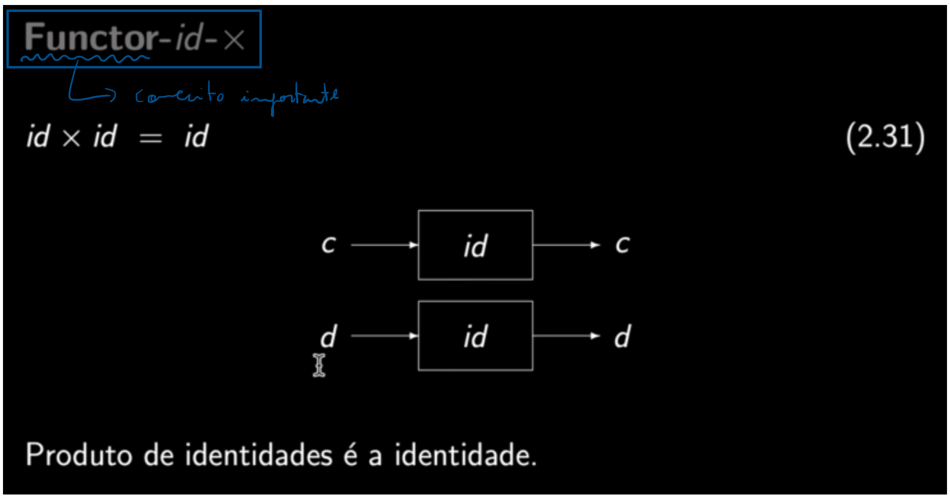
LEI



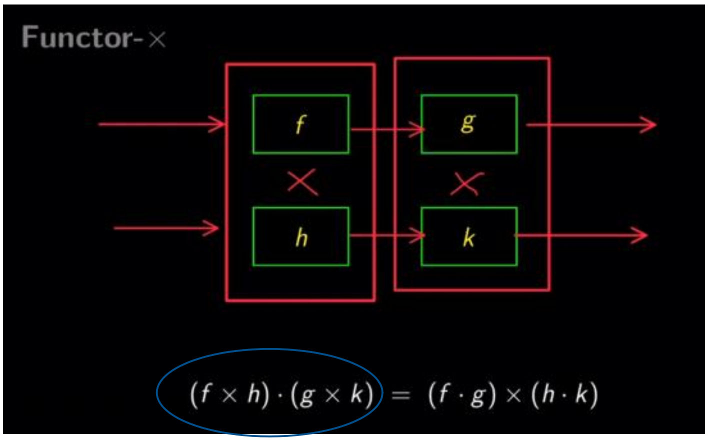
HASKELL one writes (A, B) to denote $A \times B$, for A and B two predefined datatypes, fst to denote π_1 and snd to denote π_2. In the C programming language this datatype is called the "struct datatype",

```
struct {
  A first;
  B second;
};
```

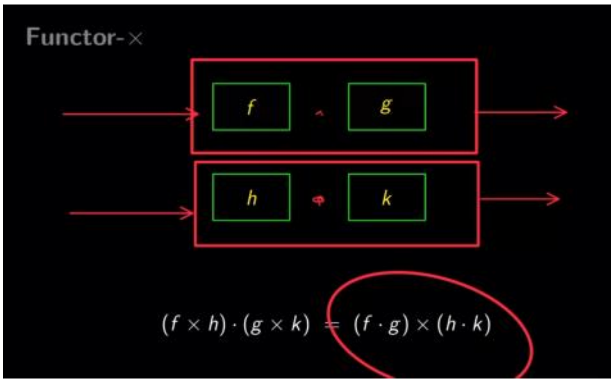




$$(f \times h) \cdot (g \times k) = (f \cdot h) \times (h \cdot k) \qquad (2.30)$$



=



Reflexão-× $\quad \langle \pi_1, \pi_2 \rangle = id \qquad (2.32)$

Eq-× $\quad \langle i, j \rangle = \langle f, g \rangle \iff \begin{cases} i = f \\ j = g \end{cases} \qquad (2.64)$

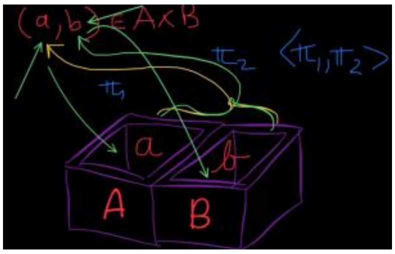
$$\begin{tikzcd}
A & A \times B \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & B \\
& A \times B \arrow[ul] \arrow[u, "\langle \pi_1, \pi_2 \rangle"] \arrow[ur] &
\end{tikzcd}$$

$\iff id$

Split gera pares

Par

$(a, b) \in A \times B$



$$f, g : B \leftarrow A$$

$$\underbrace{f = g}_{\text{Sem variáveis}} \quad \text{iff} \quad \underbrace{\langle \forall a : \overset{\text{Entrada}}{a \in A} : f\,a \overset{\text{Saída}}{\underset{\text{em } B}{=_B}} g\,a \rangle}_{\text{Com variáveis}}$$

Sem variáveis

$\Rightarrow$ Point-free

↳ Com variáveis

Expressões lógicas

Propriedade Universal

Universal-×

Existência

$$k = \langle f, g \rangle \Rightarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases}$$

"Existe uma solução — $k = \langle f, g \rangle$ — para as equações da direita"

VS

Universal-×

Unicidade

$$k = \langle f, g \rangle \Leftarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases}$$

"As equações da direita só têm uma solução: $k = \langle f, g \rangle$"

Existência de uma solução VS Unicidade de uma solução

$$id = \langle f, g \rangle \Leftrightarrow \begin{cases} \pi_1 = f \\ \pi_2 = g \end{cases}$$

Dedução da propriedade Reflexão-X a partir da propriedade Universal deste combinador.

$\langle h, k \rangle = \langle f, g \rangle$

(1 eq., 4 incógnitas)

$$\begin{aligned} & \langle h, k \rangle = \langle f, g \rangle \\ \Leftrightarrow\ & \{ \text{universal-}\times \} \\ & \begin{cases} \pi_1 \cdot \langle h, k \rangle = f \\ \pi_2 \cdot \langle h, k \rangle = g \end{cases} \\ \Leftrightarrow\ & \{ \text{cancelamento-}\times \} \\ & \begin{cases} h = f \\ k = g \end{cases} \end{aligned}$$

Dedução do Eq-X

$$\begin{tikzcd} B & B \times A \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & A \\ & A \times B \arrow[u, "\langle \pi_2, \pi_1 \rangle"] \arrow[ul, "\pi_2"] \arrow[ur, "\pi_1"'] & \end{tikzcd}$$

Equação

$\langle \pi_2, \pi_1 \rangle \circ k = id$

$$\begin{tikzcd} A \times B \arrow[r, bend left, "\langle \pi_2, \pi_1 \rangle"] & B \times A \arrow[l, bend left, "k = \langle \pi_2, \pi_1 \rangle = SWAP"] \end{tikzcd}$$

Universal-X

$\left\{ k = \langle f, g \rangle \Leftrightarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases} \right.$

$SWAP \circ SWAP = id$

$\langle \pi_2, \pi_1 \rangle \cdot k = id$

$\Leftrightarrow \langle \pi_2 \cdot k, \pi_1 \cdot k \rangle = id$

$\Leftrightarrow \begin{cases} \pi_2 \cdot k = \pi_1 \\ \pi_1 \cdot k = \pi_2 \end{cases} \Leftrightarrow \begin{cases} \pi_1 \cdot k = \pi_2 \\ \pi_2 \cdot k = \pi_1 \end{cases}$ Universal-X

$\left\{\pi_1 \cdot k = \pi_2 \quad \middle| \quad \pi_2 \cdot k = \pi_1\right\}$ Universal-$\times$

$\Leftrightarrow \ k = \langle \pi_2, \pi_1 \rangle$ YEP

$f \cdot g$ Composição sequencial

Composição paralela $\langle f, g \rangle$

Associatividade

$(f \cdot g) \cdot h = f \cdot (g \cdot h)$

Associatividade ?

$\langle \langle f, g \rangle, h \rangle = \langle f, \langle g, h \rangle \rangle$? $\longrightarrow$ NÃO

Mas existe uma relação entre as duas expressões funcionais.

$$\begin{tikzcd} A & A \times B \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & B \\ & D \arrow[ul, "f"] \arrow[u, "\langle f, g \rangle"] \arrow[ur, "g"'] & \end{tikzcd}$$

$$\begin{tikzcd} B & B \times C \arrow[l, "\pi_2"'] \arrow[r, "\pi_1"] & C \\ & D \arrow[ul, "g"] \arrow[u, "\langle g, h \rangle"] \arrow[ur, "h"'] & \end{tikzcd}$$

$$\begin{tikzcd} A \times B & (A \times B) \times C \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & C \\ & D \arrow[ul, "\langle f, g \rangle"] \arrow[u, "\langle \langle f, g \rangle, h \rangle"] \arrow[ur, "h"'] & \end{tikzcd}$$

$\neq$

Diferentes,
mas relacionadas.

$$\begin{tikzcd} A & A \times (B \times C) \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & B \times C \\ & D \arrow[ul, "f"] \arrow[u, "\langle f, \langle g, h \rangle \rangle"] \arrow[ur, "\langle g, h \rangle"'] & \end{tikzcd}$$

$$\begin{tikzcd} (A \times B) \times C \arrow[rr, bend left, "k"] & & A \times (B \times C) \\ & D \arrow[ul, "\langle \langle f, g \rangle, h \rangle"] \arrow[ur, "\langle f, \langle g, h \rangle \rangle"'] & \end{tikzcd}$$

Portanto, $k \cdot \underbrace{\langle \langle f, g \rangle, h \rangle}_{id\ ?} = \langle f, \langle g, h \rangle \rangle$

$\langle \langle f, g \rangle, h \rangle = id$

Universal-$\times$ $\Leftrightarrow$ $\begin{cases} \pi_1 = \langle f, g \rangle \\ \pi_2 = h \end{cases}$

Universal-$\times$ $k = \langle f, g \rangle \Leftrightarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases}$

Universal-$\times$ $\Leftrightarrow$ $\boxed{\begin{cases} \pi_1 \cdot \pi_1 = f \\ \pi_2 \cdot \pi_1 = g \\ \pi_2 = h \end{cases}}$

$k = \langle \pi_1 \cdot \pi_1, \langle \pi_2 \cdot \pi_1, \pi_2 \rangle \rangle$

$\Leftrightarrow \ \{ \pi_2 = id \cdot \pi_2 \}$

$$\begin{tikzcd} B & B \times C \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & C \\ & \arrow[ul, "f"] \arrow[u, "\langle f, g \rangle"] \arrow[ur, "g"'] & \end{tikzcd}$$

$k = \langle \pi_1 \cdot \pi_1, \langle \pi_2 \cdot \pi_1, id \cdot \pi_2 \rangle \rangle$

$\{ f \times g = \langle f \cdot \pi_1, g \cdot \pi_2 \rangle \}$

(tirar o da esquerda
e aplicar f, tirar o da
direita e aplicar g)

$$\begin{tikzcd} C & C \times D \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & D \\ A \arrow[u, "f"] & A \times B \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] \arrow[u, "f \times g"] & B \arrow[u] \end{tikzcd}$$

$$\begin{tikzcd} C & C \times D \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] & D \\ A \arrow[u, "f"] & A \times B \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] \arrow[u, "f \times g"] & B \arrow[u, "g"'] \end{tikzcd}$$

$\Rightarrow \quad f \times g = \langle f \cdot \pi_1, g \cdot \pi_2 \rangle$

$\{ f \times g = \langle f \cdot \pi_1 , g \cdot \pi_2 \rangle \}$ (tirar o da esquerda e aplicar f, tirar o da direita e aplicar g) é equivalente a f x g (a, b) = (f a, g b)

$\Leftrightarrow \quad k = \langle \pi_1 \cdot \pi_1 , \pi_2 \times id \rangle$

$$\begin{tikzcd} (A \times B) \times C \arrow[rr, bend left, "\langle \pi_1 \cdot \pi_1 , \pi_2 \times id \rangle"] & & A \times (B \times C) \arrow[ll, bend left, "j"] \\ & D \arrow[ul, "\langle\langle f, g\rangle, h\rangle"] \arrow[ur, "\langle f, \langle g, h\rangle\rangle"'] & \end{tikzcd}$$

Exercício

$j \cdot \underbrace{\langle f, \langle g, h \rangle \rangle}_{id\,?} = \langle \langle f, g \rangle , h \rangle \quad \Rightarrow \quad j \cdot id = j$

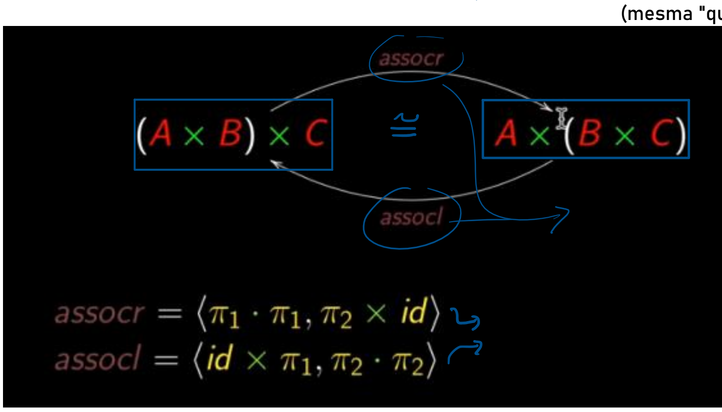
$\langle f, \langle g, h \rangle \rangle = id$

$\Leftrightarrow \begin{cases} \pi_1 = f \\ \pi_2 = \langle g, h \rangle \end{cases}$

$\Leftrightarrow \begin{cases} \pi_1 = f \\ \pi_1 \cdot \pi_2 = g \\ \pi_2 \cdot \pi_2 = h \end{cases}$

$$\begin{aligned} j &= \langle \langle \pi_1 , \pi_1 \cdot \pi_2 \rangle , \pi_2 \cdot \pi_2 \rangle \\ &= \langle \langle \underline{id} \times \pi_1 , \underline{\pi_1} \cdot \pi_2 \rangle , \pi_2 \cdot \pi_2 \rangle \\ &= \langle id \times \pi_1 , \pi_2 \cdot \pi_2 \rangle \end{aligned}$$

(mesma "quantidade de
" em ambos os



s
do

a

Inversas uma da outra

$assocr \cdot assocl = id$ (esquerda)
$assocl \cdot assocr = id$ (direita)

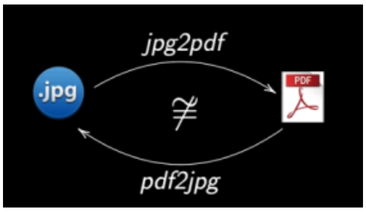
$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, phantom, "\cong"] & B \arrow[l, bend left, "g"] \end{tikzcd}$$

$f \cdot g = id$
$g \cdot f = id$

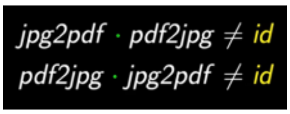
$\Rightarrow$ Não há perda de informação. Isomorfismo. (Exemplo: SWAP)

$\underbrace{\text{ISO}}_{\text{a mesma}} + \underbrace{\text{MORFISMO}}_{\text{forma}}$

"FORMA SEMELHANTE"



Neste caso, há perda de informação, logo não há isomorfismo.



Apesar de poder haver isomorfismo a nível visual (avaliação humana).

Conversão de formatos

$$\begin{tikzcd} A \arrow[r, "exporta"] \arrow[d, dash, "f"'] \arrow[dr, phantom, "(\cong)"] & C \arrow[d, dash, "r"] \\ {} & {} \end{tikzcd}$$

Para valer a pena, tanto a exportação como a importação têm de custar pouco em termos de recursos.

GESTÃO E REUTILIZAÇÃO

tem de custar pouco em termos de recursos.

Necessidade $f \downarrow$ $(\cong)$ $\downarrow r$ Reutilizável

$B \xleftarrow{\text{importa}} D$

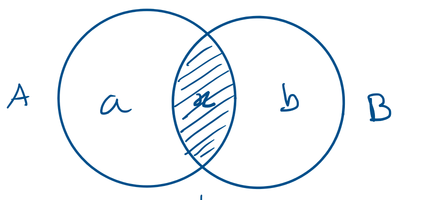
$f = \text{importa} \cdot r \cdot \text{exporta}$

GESTÃO E REUTILIZAÇÃO DE CÓDIGO

EM HASKELL:

```
Prelude> split f g a = (f a , g a)
Prelude> f>< g = split (f.p1)(g.p2)
Prelude>
Prelude> swap = split p2 p1
Prelude> assocl = split (id >< p1) (p2.p2)
Prelude> assocr = split (p1.p1) (p2 >< id)
Prelude>
Prelude> assocl ("Texto",(10, False))
(("Texto",10),False)
Prelude> (assocr.assocl) ("Texto",(10, False))
("Texto",(10,False))
```

$A \cup B = \{a | a \in A\} \cup \{b | b \in B\}$



A ou B?

Reuniões disjuntas

$\{(1, a) | a \in A\} \cup \{(2, b) | b \in B\} = A + B$

"Etiqueta 1 e etiqueta 2"

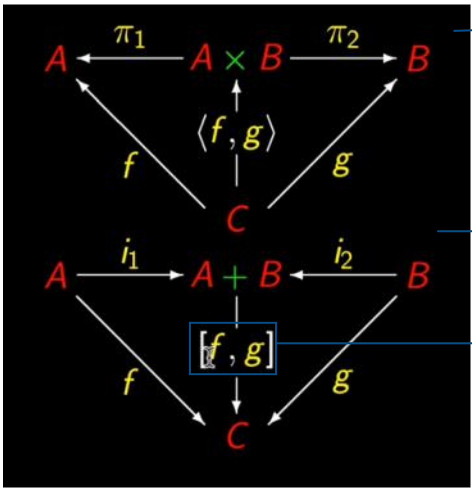
$A + B = \{i_1\, a \,|\, a \in A\} \cup \{i_2\, b \,|\, b \in B\}$
$i_1 a = (1, a) \wedge i_2 b = (2, b)$

$m : A + B \to C$
$i_1 : A \to A + B$
$i_2 : B \to A + B$

$$\begin{tikzcd} A \arrow[r, "i_1"] \arrow[dr, "f"'] & A + B \arrow[d, "m"] & B \arrow[l, "i_2"'] \arrow[dl, "g"] \\ & C & \end{tikzcd}$$

$f = m \cdot i_1$
$g = m \cdot i_2$

O "m" quanto tem entradas do tipo A vai "ativar o f",
e quando tem entradas do tipo B vai "ativar o g".



Funções correm em paralelo.
Projeções p1 e p2.
Produto de A e B.

PRODUTO

Lê-se "ou f, ou g"

Funções correm em alternativa.
Injeções i1 e i2.
Soma de A e B.

CO-PRODUTO

UNIVERSAL-+

$k = [f, g] \Leftrightarrow \begin{cases} k \cdot i_1 = f \\ k \cdot i_2 = g \end{cases}$

Comparar com

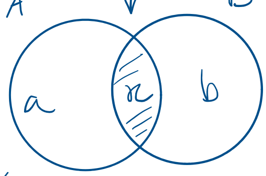
$k = \langle f, g \rangle \Leftrightarrow \begin{cases} \pi_1 \cdot k = f \\ \pi_2 \cdot k = g \end{cases}$

Como identificá-los como pertencentes a A ou a B?

$m : A \cup B \to C$

$A \cup B = \{a \,|\, a \in A\} \cup \{b \,|\, b \in B\}$

Podem haver a's em A que pertencem a B, e vice-versa



Não sei

Podem haver a's em A que pertencem a B, e vice-versa

↪ Não são disjuntos

$\hookrightarrow \{(\underline{1},a) \mid a \in A\} \cup \{(\underline{2},b) \mid b \in B\}$

$\hookrightarrow$ União Disjunta : $\underline{A+B} \neq A \cup B$

Abstração : $A+B = \{i_1\, a \mid a \in A\} \cup \{i_2\, b \mid b \in B\}$

$i_1\ a = (1, a)$ $\qquad i_1 : A \to A+B$

$i_2\ b = (2, b)$ $\qquad i_2 : B \to A+B$

$m : A+B \to C$

COPRODUTO

$$\begin{tikzcd} A \arrow[r, "i_1"] \arrow[dr, "f"'] & A+B \arrow[d, "m"] & B \arrow[l, "i_2"'] \arrow[dl, "g"] \\ & C & \end{tikzcd}$$

Um tipo habitado ou por A ou por B

O f é aquilo que o m faz quando tem a certeza que a entrada veio do lado do A. E o g é aquilo que o m faz quando tem a certeza que a sua entrada é um B do lado direito.

Se tivermos duas funções, uma de A para C e outra de B para C, i.e. que partilham o tipo de saída, é possível construir uma função m que decide que função usar com base no tipo de entrada.

Em suma, g e f não operam em paralelo, operam em alternância. Mas quem é que decide? A etiqueta do valor que está na união disjunta.

$m = [f, g]$

"ou f ou g"

"either f or g"

$\rightsquigarrow$ Expressões condicionais?

$$K = [f, g] \iff \begin{cases} K \cdot i_1 = f \\ K \cdot i_2 = g \end{cases}$$

$A+B$ na linguagem de programação C

```
struct {
    int tag; // 1,2
    union {
        A if A
        B if B
    } data;
}
```

or in PASCAL,

```
record
  case tag : integer
  of x =
    1 : (P : A);
    2 : (S : B)
end;
```

$[f, g] : A+B \to C$

$$[f, g]\, x = \begin{cases} x = i_1\ a \Rightarrow f\ a \\ x = i_2\ b \Rightarrow g\ a \end{cases}$$

Condição

Se $x = i_1\ a$ então $f\ a$

$$\begin{tikzcd} D \arrow[r, "i_1"] \arrow[d, "f"'] \arrow[dr] & D+E \arrow[d, "f+g"] & E \arrow[l, "i_2"'] \arrow[d, "g"] \arrow[dl] \\ {} \arrow[r, "i_1"] & {} & {} \arrow[l, "i_2"'] \end{tikzcd}$$

$\longrightarrow f + g = [i_1 \cdot f, i_2 \cdot g]$

f + g vai ativar o f ou o g conforme a sua entrada, mantendo as etiquetas

$$\begin{tikzcd}
{} \arrow[d, "f"] & {} \arrow[d, "f+g"] & {} \arrow[d, "g"] \\
B \arrow[r, "i_1"] \arrow[dr, "h"'] & B+C \arrow[d, "{[h,k]}"] & C \arrow[l, "i_2"'] \arrow[dl, "k"] \\
& F &
\end{tikzcd}$$

$$\longrightarrow \quad f + g = [i_1 \cdot f, i_2 \cdot g]$$

f + g vai ativar o f ou o g conforme a sua entrada, mantendo as etiquetas na saída (o que permite propagar o efeito de alternância).
Se for do tipo D, ativa F, e depois coloca a mesma etiqueta que recebeu.
Se for do tipo E, ativa G, e depois coloca a mesma etiqueta que recebeu.

$$[h,k] \cdot (f+g) = [h \cdot f, k \cdot g] \qquad (2.43)$$

A soma de funções f + g recebe um que pode ser D, se for ativa f, ou que pode ser E, se for ativa g, devolvendo novamente uma união disjunta, ou seja, etiquetando o resultado de acordo com o tipo que recebeu.
A alternativa [f, g] recebe esse tipo que é uma união disjunta, caso seja do tipo B, ativa h, e caso seja do tipo C, ativa K.
Portanto, o a alternativa [h, k] após a soma f + g pode ser descrita como a alternativa de h após f, e de k após g - recebe tipo D+E, com base no tipo ativa h . f ou k . g, e devolve um tipo F.

Fusão-+ $\quad f \cdot [h,k] = [f \cdot h, f \cdot k]$

Reflexão-+ $\quad [i_1, i_2] = id$

Universal-+	$k = [f\,,g] \;\Leftrightarrow\; \begin{cases} k \cdot i_1 = f \\ k \cdot i_2 = g \end{cases}$
Cancelamento-+	$\begin{cases} [f\,,g] \cdot i_1 = f \\ [f\,,g] \cdot i_2 = g \end{cases}$
Reflexão-+	$[i_1\,,i_2] \;=\; id_{A+B}$
Fusão-+	$f \cdot [g\,,h] = [f \cdot g\,, f \cdot h]$
Def-+	$f + g = [i_1 \cdot f\,, i_2 \cdot g]$
Absorção-+	$[g\,,h] \cdot (i + j) = [g \cdot i\,, h \cdot j]$
Natural-i_1	$(i + j) \cdot i_1 \;=\; i_1 \cdot i$
Natural-i_2	$(i + j) \cdot i_2 \;=\; i_2 \cdot j$
Functor-+	$(g \cdot h) + (i \cdot j) = (g + i) \cdot (h + j)$
Functor-id-+	$id_A + id_B = id_{A+B}$
Eq-+	$[f\,,g] = [h\,,k] \;\Leftrightarrow\; \begin{cases} f = h \\ g = k \end{cases}$

$$f = g \iff f\,x = g\,x$$