λ-calculus & categories!

Paul-André Melliès
mellies@irif.fr

IRIF ← Fundamental Informatics Research

Coq

polarities
linear logic

call-cc

Lambek

λ-calculus
syntax
programming

category
theory
algebra
mathematics

Curry
Howard
Correspondence

Lawvere

homotopy
type
theory

topos
theory

logic
reasoning
proofs

sheaf theory
Grothendieck

reasoning
by
contradiction

$\boxed{\lambda\text{-calculus}}$

simply-typed λ-calculus

$\boxed{\text{simple types}}$

Type of natural numbers

$\boxed{\mathbb{N} \longrightarrow \mathbb{N}}$

$\boxed{\lambda\text{-terms}}$

Church numerals

3, 5, 8

$x \longmapsto x+1$

functions described in the syntax of the λ-calculus.

in connection to algebra.

$\boxed{A \Longrightarrow B}$

the type of $\boxed{\lambda\text{-terms}}$ which take a λ-term of type A as argument and return a λ-term of type B as output.

functions

$\mathbb{N} \Longrightarrow \mathbb{N}$

The approach taken by Lambek

based on the idea of

$\boxed{\boxed{\underline{\underline{\text{free}}}}}$ $\boxed{\text{cartesian closed category.}}$

$\boxed{\text{cartesian category}}$

finite

↓

a word on two letters a, b

$A = \{a, b\}$ <u>alphabet</u> of two letters.

[aabab] [aaabbbaaba]

$\longrightarrow$ $\boxed{\text{automata}}$

Schutzenberger automata $\longleftrightarrow$ $\boxed{\text{monoid}}$

<u>def</u>. A monoid (M, m, e) is a set M

equipped with a binary operation

$$M \times M \xrightarrow{m} M$$

and a constant (= a nullary operation)

singleton set $= \mathbb{1} \xrightarrow{e} M$ $(e \in M)$

multiplication $M^2 \xrightarrow{m} M$

unit $M^0 \xrightarrow{e} M$

$\mathbb{1} \times A \cong A \qquad \mathbb{1} \cong M^0$

unit of the cartesian product of sets

such that :

① multiplication is <u>associative</u>:

$\forall x,y,z \in M \qquad m(x, m(y,z)) = m(m(x,y), z)$

② the unit e is a neutral element:

$\forall x \in M \qquad m(x,e) = x = m(e,x)$

<u>The free monoid.</u>

given a set A (called the alphabet)

we define A^* as the monoid of finite words on the alphabet A equipped with:

— concatenation of words as multiplication

— the empty word ε as neutral element

$$u, v \longmapsto u \cdot v$$

Concatenation of words is associative:

$$([aba] \cdot [ab]) \cdot [aa] = [abaabaa] = [aba] \cdot ([ab] \cdot [aa])$$

<u>Ppty</u>: the free monoid $(A^*, \cdot, \varepsilon)$ is characterized by the property that every function ← set-theoretic

$$f : A \longrightarrow M$$

where $\underset{(M,m,e)}{M}$ is a monoid <u>lifts</u> in a unique way to a homomorphism

$$h : (A^*, \cdot, \varepsilon) \longrightarrow (M, m, e)$$

$\circledast$

<u>def</u>: a homomorphism

$$h : (M_1, m_1, e_1) \longrightarrow (M_2, m_2, e_2)$$

is a function

$$h : M_1 \longrightarrow M_2$$

such that

$$\forall x,y \in M_1 \quad h(m_1(x,y)) = m_2(hx, hy)$$

$$h(e_1) = e_2$$

homomorphism

$$\begin{tikzcd}
A^* \arrow[r, "h"] & M \\
A \arrow[u, "\eta"] \arrow[ur, "f"'] &
\end{tikzcd}$$

$\circledast$

$$\boxed{(M, \bullet_M, e_M)}$$

η is a function

$$\boxed{\begin{array}{ccc} A & \longrightarrow & A^* \\ a & \longmapsto & [a] \\ \text{letter } a & & \text{the word with one letter } a \end{array}}$$

$$h : A^* \longrightarrow M$$

$$[a\,b] \longmapsto f(a) \bullet_M f(b)$$

$$[x_1 \cdots x_n] \longmapsto \underbrace{f(x_1) \cdot_M \cdots \cdot_M f(x_n)}$$

$x_i \in A$

allowed because multiplication in M is associative

$$\varepsilon = [\,] \longmapsto e_M$$

<u>Fact</u>: h is a homomorphism.

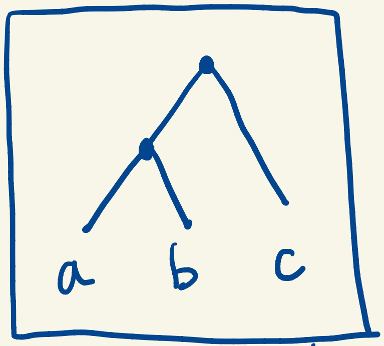
<u>def</u>: a magma (M, m) is a set M equipped with a binary operation:

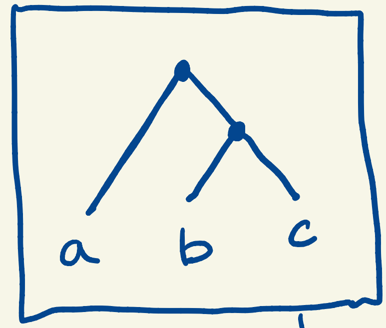
$$M \times M \xrightarrow{m} M$$

<u>Note</u>: m is not necessarily associative!

What is the free magma generated by an alphabet A?

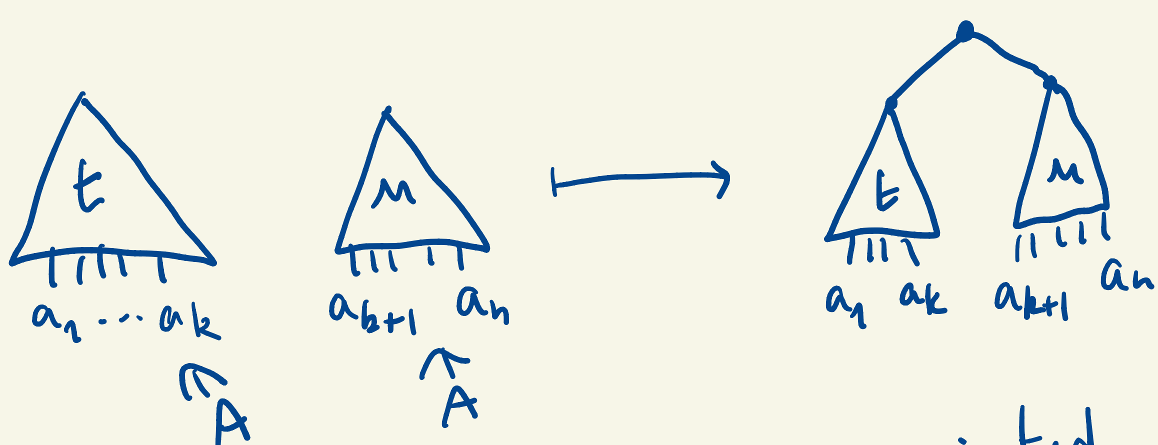
$$f : A \longrightarrow M \text{ magma}$$





$A = \{a, b, c\}$ $\quad m(m(a,b),c)$ $\quad m(a, m(b,c))$

the $\boxed{\text{free magma}}$ generated by A
is the set of binary trees
with leaves in A
equipped with the lifted sum
of binary trees as "multiplication".



a <u>data structure</u> associated to
an algebraic notion
(monad, magma)

construction / recognition

(finite words)

λ-calculus
λ-terms
types

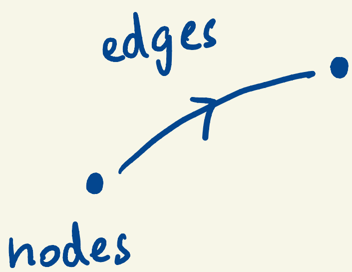
data structure

(monoids)

cartesian closed category

algebraic structure

Categories are graphs!



nodes = objects

edges = maps
arrows
morphisms

What we want:

to think of the big graph of <u>Sets</u>.

the graph $\underline{Set}$ whose nodes are sets
edges are functions

$$A \xrightarrow{f} B$$

f function
A, B sets

[locally small category]

def

A category

• a $\underline{\text{class}}$ of nodes (= objects)
[0]

• between two objects A, B a $\underline{\text{set}}$ of maps
[1]

$$\mathrm{Hom}(A, B)$$

•
$$\begin{tikzcd} & B \arrow[dr, "g"] & \\ A \arrow[ur, "f"] \arrow[rr, "g \circ f"'] & & C \end{tikzcd}$$

for all A, B, C objects

[2] a composition law:

$$\circ_{A,B,C} : \mathrm{Hom}(B,C) \times \mathrm{Hom}(A,B) \longrightarrow \mathrm{Hom}(A,C)$$
$$g \qquad f \longmapsto g \circ f$$

• [2] an identity map

$$A \xrightarrow{id_A} A$$ for all objects A.

$$\begin{tikzcd} & B \arrow[dr, "g"] & \\ A \arrow[ur, "f"] \arrow[rr, "g\circ f"'] & & C \end{tikzcd}$$

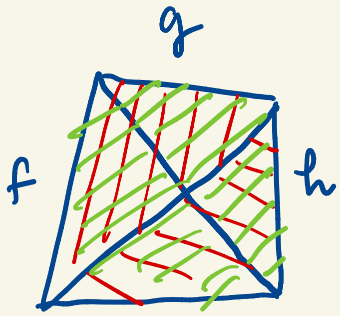
[intuitively
2-dimensional]

[3]

- associativity:

$$\begin{tikzcd} & B \arrow[r, "g"] & C \arrow[dr, "h"] & \\ A \arrow[ur, "f"] \arrow[urr, "g\circ f"] \arrow[rrr, "h\circ(g\circ f)"'] & & & D \end{tikzcd} \qquad \begin{tikzcd} & B \arrow[r, "g"] \arrow[drr, "h\circ g"] & C \arrow[dr, "h"] & \\ A \arrow[ur, "f"] \arrow[rrr, "(h\circ g)\circ f"'] & & & D \end{tikzcd}$$

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



3 dimensional simplex
filling the space
between the two
surfaces.

- $f : A \longrightarrow B$ $\qquad f\circ id_A = f = id_B \circ f$

What is a cartesian product in a category

<u>def</u>: a cartesian product of A and B in a category $\mathcal{C}$ is a triple

$$(A \times B,\ \pi_1,\ \pi_2)$$

consisting of an object $A \times B$ and two maps:

$$A \times B \xrightarrow{\pi_1} A$$

$$A \times B \xrightarrow{\pi_2} B$$

called the projection maps.

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

"universality"

and satisfying the following property:

for every object X and pair of maps

$$X \xrightarrow{f} A \qquad X \xrightarrow{g} B$$

$$\begin{tikzcd}
A & & B \\
& A \times B \arrow[ul, "\pi_1"'] \arrow[ur, "\pi_2"] & \\
& X \arrow[u, "h"] \arrow[uul, bend left, "f"] \arrow[uur, bend right, "g"'] &
\end{tikzcd}$$

in the case of Set

$h : X \longrightarrow A \times B$

$x \longmapsto (fx, gx)$

there exists a $\boxed{\text{unique}}$ map

$$X \xrightarrow{h} A \times B$$

such that the diagram (*) commutes

$$\pi_1 \circ h = f \qquad \pi_2 \circ h = g$$

(end of the definition)

Example: the cartesian product of two sets A and B.

example: the greatest lowerbound of two elements a, b in an ordered set $(A, \leq)$.