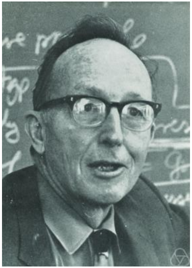
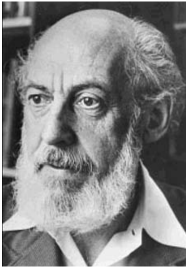
Categories for the Lazy Functional Programmer

Thorsten Altenkirch

School of Computer Science
University of Nottingham

April 15, 2011

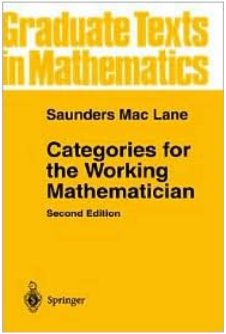
Intro

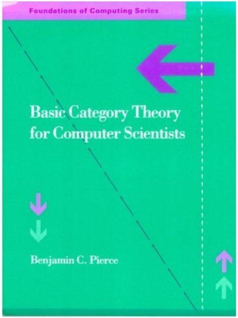
Saunders MacLane (1909 - 2005) Samuel Eilenberg (1913 - 1998)

- Originally: tool for algebraic topology.
- Relevance for Computer Science (Lambek's obs)
  E.g. *Cartesian Closed Cats* $\approx$ *Simply Typed* λ*-calculus*
- Categorical concepts in Haskell: `Functor`, `Monad`, ...
- Is Category Theory *Abstract Nonsense* ?
- Is Category Theory an alternative to Set Theory?

Books



MacLane



Pierce

Awodey

Overview

The category **Set**

Objects: Sets

$$|\mathbf{Set}| = \mathrm{Set}$$

Morphisms : Functions, given $A, B \in |\mathbf{Set}|$

$$\mathbf{Set}(A, B) = A \to B$$

Identity: Given $A \in \mathrm{Set}$

$$\begin{aligned} \mathrm{id}_A &\in \mathbf{Set}(A, A) \\ \mathrm{id}_A &= \lambda a.a \end{aligned}$$

Composition: Given $f \in \mathbf{Set}(B, C), g \in \mathbf{Set}(A, B)$:

$$\begin{aligned} f \circ g &\in \mathbf{Set}(A, C) \\ f \circ g &= \lambda a.f\,(g\,a) \end{aligned}$$

Laws:

$$\begin{aligned} f \circ \mathrm{id} &= f \\ \mathrm{id} \circ f &= f \\ (f \circ g) \circ h &= f \circ (g \circ h) \end{aligned}$$

Exercise 1

Derive the laws for **Set** using only the equations of the simply typed λ-calculus, i.e.

$$\beta \quad (\lambda x.t)u = t[x := u]$$

$$\eta \quad \lambda x.t\,x = t \text{ if } x \notin \mathrm{FV}\ t$$

$$\xi \quad \frac{t = u}{\lambda x.t = \lambda x.u}$$

Definition: **C** is a category

A (large) set of objects:

$$|\mathbf{C}| \in \mathrm{Set}_1$$

Morphisms: For every $A, B \in |\mathbf{C}|$ a *homset*

$$\mathbf{C}(A, B) \in \mathrm{Set}$$

Identity: For any $A \in |\mathbf{C}|$:

$$\mathrm{id}_A \in \mathbf{C}(A, A)$$

Composition: For $f \in \mathbf{C}(B, C), g \in \mathbf{C}(A, B)$:

$$f \circ g \in \mathbf{C}(A, C)$$

Laws:

$$\begin{aligned} f \circ \mathrm{id} &= f \\ \mathrm{id} \circ f &= f \\ (f \circ g) \circ h &= f \circ (g \circ h) \end{aligned}$$

Size matters

- I assume as given a predicative hierarchy of set-theoretic universes:

  $$\mathrm{Set} = \mathrm{Set}_0 \in \mathrm{Set}_1 \in \mathrm{Set}_2 \in \ldots$$

  which is cummulative

  $$\mathrm{Set}_0 \subseteq \mathrm{Set}_1 \subseteq \mathrm{Set}_2 \subseteq \ldots$$

- To accomodate categories like **Set** we allow that the objects are a large set ($|\mathbf{C}| \in \mathrm{Set}_1$) but require the hom**sets** to be proper sets $\mathbf{C}(A, B) \in \mathrm{Set} = \mathrm{Set}_0$.
- A category is *small*, if the objects are a set $|\mathbf{C}| \in \mathrm{Set}$
- We can repeat this definition at higher levels, a category at level n has as objects $|\mathbf{C}| \in \mathrm{Set}_{n+1}$ and homsets $\mathbf{C}(A, B) \in \mathrm{Set}_n$

Dual category

Given a category $\mathbf{C}$ there is a dual category $\mathbf{C}^{\mathrm{op}}$ with

Objects $|\mathbf{C}^{\mathrm{op}}| = |\mathbf{C}|$

Homsets $\mathbf{C}^{\mathrm{op}}(A, B) = \mathbf{C}(B, A)$

and composition defined backwards.

Notation

For $n \in \mathbb{N}$ we define

$$\bar{n} = \{i < n\}$$

Question

How many elements are in $\mathbf{Set}(\bar{2}, \bar{3})$ and in $\mathbf{SET}^{\mathrm{op}}(\bar{2}, \bar{3})$?

Isomorphism

An isomorphism between $A, B \in |\mathbf{C}|$ is given by two morphisms $f \in \mathbf{C}(A, B)$ and $f^{-1} \in \mathbf{C}(B, A)$ such that $f \circ f^{-1} = \mathrm{id}$, $f^{-1} \circ f = \mathrm{id}$:

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}"] \arrow[r, bend left, "f"] & B \arrow[l, bend left, "f^{-1}"] \arrow[loop right, "\mathrm{id}"] \end{tikzcd}$$

We say that A and B are isomorphic $A \simeq B$.

- Isomorphic sets are the same upto a *renaming* of elements.
- Concepts in category theory are usually defined *up to isomorphism*.

Exercise 2

Which of the following isomorphisms hold in **Set**:

$$
\begin{aligned}
\bar{2}+\bar{2} &\simeq \bar{4}\\
\bar{2}\times\bar{2} &\simeq \bar{4}\\
\bar{2}\to\bar{2} &\simeq \bar{4}\\
\mathbb{N}+\mathbb{N} &\simeq \mathbb{N}\\
\mathbb{N}\times\mathbb{N} &\simeq \mathbb{N}\\
\mathbb{N}\to\mathbb{N} &\simeq \mathbb{N}
\end{aligned}
$$

$A \times B$ is cartesian product

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$

$A + B$ is disjoint union

$$A + B = \{\mathrm{inl}\, a \mid a \in A\} \cup \{\mathrm{inr}\, b \mid b \in B\}$$

Monomorphism

$f \in \mathbf{C}(B, C)$ is a monomorphism (short *mono*), if for all $g, h \in \mathbf{C}(A, B)$

$$\frac{f \circ g = f \circ h}{g = h}$$

- In **Set** monos are precisely the injective functions.
- We draw monos as $A \rightarrowtail B$

Epimorphism

$f \in \mathbf{C}(A, B)$ is a epimorphism (short *epi*), if for all $g, h \in \mathbf{C}(B, C)$

$$\frac{g \circ f = h \circ f}{g = h}$$

- In **Set** epis are precisely the surjective functions.
- We draw epis as $A \twoheadrightarrow B$

Exercise 3

Show that every iso is both mono and epi.

Exercise 4

Show that the bijections (i.e. functions that are both mono and epi) in **Set** are precisely the isos.

Exercise 5

Show that in **Set** every morphism $f \in A \to B$ can be written as a composition of an epi and a mono:

$$\begin{tikzcd}
B & \\
 & C \arrow[ul, tail, "m"'] \\
A \arrow[uu, "f"] \arrow[ur, two heads, "e"] &
\end{tikzcd}$$

Monoids

Definition: Monoid

A monoid $(M, e, *)$ is given by $M \in \mathrm{Set}$, $e \in M$ and $(*) \in M \to M \to M$ such that:

$$\begin{aligned} x * e &= x \\ e * x &= x \\ (x * y) * z &= x * (y * z) \end{aligned}$$

Example

$(\mathbb{N}, 0, +)$ is a (commutative) monoid.

Question

Give an example of a non-commutative monoid.

- Monoids correspond to categories with one object.

Monoid as a category

Every monoid $(M, e, *)$ gives rise to a category $\mathbf{M}$

Objects: $|\mathbf{M}| = \{()\}$

Morphisms $\mathbf{M}((), ()) = M$

e is the identity, $*$ is composition.

Preorder

$(A, \sqsubseteq)$ with $A \in \mathrm{Set}$ and $(\sqsubseteq) \in A \to A \to \mathrm{Prop}$ is a preorder if R is

- reflexive $\forall a \in A. a \sqsubseteq a$
- transitive $\forall a, b, c \in A. a \sqsubseteq b \to b \sqsubseteq c \to a \sqsubseteq c$

Example

$(\mathbb{N}, \leq)$ is a preorder.

- $(\mathbb{N}, \leq)$ is a partial order, because it also satisfies

$$\frac{m \leq n \qquad n \leq m}{m = n}$$

Question

Give an example of a preorder, which is not a partial order.

- Preorders correspond to categories where the homsets have at most one element.

A preorder as a category

A preorder $(A, \sqsubseteq)$ can be viewed as a category $\mathbf{A}$:

Objects $|\mathbf{A}| = A$

Homsets $\mathbf{A}(a, b) = \begin{cases} \{()\} & \text{if } a \sqsubseteq b \\ \{\} & \text{otherwise} \end{cases}$

- Monoids and preorders are degenerate categories.

Categories of sets with structure

The category of Monoids: **Mon**

Objects: Monoids $(M, e, *)$

Morphisms $\mathbf{Mon}((M, e, *), (M', e', *'))$ is given by $f \in M \to M'$ such that $f\,e = e'$ and $f\,(x * y) = (f\,x) *' (f\,y)$.

Example

The embedding $i \in \mathbf{Mon}((\mathbb{N}, 0, +), (\mathbb{Z}, 0, +))$ with $i\,n = n$

Exercise 6

Show that i is a mono and an epi but not an iso in **Mon**.

Exercise 7

Define the category **Pre** of preorders and monotone functions.

Finite Sets

FinSet

Objects: Finite Sets

Morphisms: Functions

- **FinSet** is a full subcategory of **Set**.

FinSetSkel

Objects: $\mathbb{N}$

Morphisms: $\mathbf{FinSetSkel}(m, n) = \bar{m} \to \bar{n}$

- **FinSetSkel** is skeletal, any isomorphic objects are equal.
- **FinSet** and **FinSetSkel** are equivalent (in the appropriate sense).

Computational Effects

Error

Given a set of Errors $E \in \mathrm{Set}$

Objects: Sets

Morphisms: $\mathbf{Error}(A, B) = A \to B + E$

State

Given a set of states: $S \in \mathrm{Set}$

Objects: Sets

Morphisms: $\mathbf{State}(A, B) = A \times S \to B \times S$

Exercise 8

Define identity and composition for both categories.

λ-terms

Lam

Objects: Finite sets of variables

Morphisms: **Lam**$(X, Y) = Y \to \mathrm{Lam}\, X$ where $\mathrm{Lam}\, X$ is the set of λ-terms whose free variables are in X.

Exercise 9

1. Define identity and composition.
2. Extend the definition to typed λ-calculus.

Product categories

Given categories $\mathbf{C}, \mathbf{D}$ we define $\mathbf{C} \times \mathbf{D}$:

Objects: $\mathbf{C} \times \mathbf{D}$

Morphisms: $\mathbf{C} \times \mathbf{D}((A, B), (C, D)) = \mathbf{C}(A, C) \times \mathbf{D}(B, D)$

We abbreviate $\mathbf{C}^2 = \mathbf{C} \times \mathbf{C}$

Slice categories

Given a category $\mathbf{C}$ and an object $A \in |\mathbf{C}|$ we define $\mathbf{C}/A$ as:

Objects: $|\mathbf{C}/\mathbf{A}| = \Sigma B \in |\mathbf{C}|.\mathbf{C}(B, A)$

Morphisms: $\mathbf{C}/\mathbf{A}((B, f), (C, g))$:

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

Computable sets

ω-Set

Objects: A Set A and a relation $\Vdash_A \subseteq \mathbb{N} \times A$ such that $\forall a \in A. \exists i \in \mathbb{N}. i \Vdash_A a$.

Morphisms:

$$\begin{aligned} &\omega-\mathbf{Set}((A, \Vdash_A), (B, \Vdash_B)) \\ &= \{f \in A \to B \mid \exists i \in \mathbb{N}. \forall j, a. j \Vdash_A a \\ &\quad \to \exists k. \{i\} j \downarrow k \wedge k \Vdash_B f\, a\} \end{aligned}$$

where $\{i\} j \downarrow k$ means the ith Turing machine applied to input j terminates and returns k.

Partial computations

ω-CPO

Objects: $(A, \sqsubseteq_A, \bigsqcup_A)$ such that $(A, \sqsubseteq_A)$ is a partial order, and

$$\bigsqcup_A \in \{f \in \mathbb{N} \to A \mid \forall i.fi \sqsubseteq_A f(i+1)\} \to A$$

is the least upper bound of a chain, i.e. $\forall i.f\,i \sqsubseteq \bigsqcup_A f$ and $(\forall i.f\,i \sqsubseteq a) \to \bigsqcup_A f \sqsubseteq a$.

Morphisms: $\omega-\mathbf{CPO}((A, \sqsubseteq_A, \bigsqcup_A), (B, \sqsubseteq_B, \bigsqcup_B))$ is given by functions $f \in A \to B$ which are:

monotone $\dfrac{a \sqsubseteq_A b}{f\,a \sqsubseteq f\,b}$

continuous $f(\bigsqcup_A h) = \bigsqcup_B (f \circ h)$

Definition: Functor

Given categories $\mathbf{C}, \mathbf{D}$ a functor $F \in \mathbf{C} \to \mathbf{D}$ is given by

a map on objects $F \in |\mathbf{C}| \to |\mathbf{D}|$

maps on morphisms Given $f \in \mathbf{C}(A, B)$, $F\,f \in \mathbf{D}(F\,A, F\,B)$

such that

$$\begin{aligned} F\,\mathrm{id}_A &= \mathrm{id}_{F\,A} \\ F\,(f \circ g) &= (F\,f) \circ (F\,g) \end{aligned}$$

- A functor $F \in \mathbf{C} \to \mathbf{C}$ is called an *endofunctor*.

Example

$\mathrm{List} : \mathbf{Set} \to \mathbf{Set}$, the list functor on morphisms is given by map

$$\begin{aligned} \mathrm{map}\,f\,[] &= [] \\ \mathrm{map}\,f\,(a : as) &= f\,a : \mathrm{map}\,f\,as \end{aligned}$$

We just write $\mathrm{List}\,f = \mathrm{map}\,f$.

Exercise 10

Show that List satisfies the functor laws.

Question

We consider endofunctors on **Set**, given maps on objects:

1. Is $F_1\,X = X \to \mathbb{N}$ a functor?
2. Is $F_2\,X = X \to X$ a functor?
3. Is $F_3\,X = (X \to \mathbb{N}) \to \mathbb{N}$ a functor?

- All type expressions with only positive occurences of a set variable give rise to (covariant) functors in **Set** $\to$ **Set**.
- All type expressions with only negative occurences of a set variable give rise to (contravariant) functors in **Set**$^{\mathrm{op}}$ $\to$ **Set**.

Exercise 11

Is there a type-expression which is not positive but still gives rise to a covariant endofunctor on **Set**?

Definition: natural transformation

Given functors $F, G \in \mathbf{C} \to \mathbf{D}$ a natural transformation $\alpha : F \to G$ is given by a family of maps

$$\alpha \in \Pi_{A \in |\mathbf{C}|} \mathbf{D}(F\,A, G\,A)$$

such that for any $f \in \mathbf{C}(A, B)$

$$\begin{tikzcd} F\,A \arrow[r, "\alpha_A"] \arrow[d, "F\,f"'] & G\,A \arrow[d, "G\,f"'] \\ F\,B \arrow[r, "\alpha_B"'] & G\,B \end{tikzcd}$$

Exercise 12

1. Show that $\mathrm{reverse} \in \Pi X \in \mathrm{Set}.\mathrm{List}\,X \to \mathrm{List}\,X$ is a natural transformation.
2. Give a family of maps with the same type, which is not natural.

Functor categories

Given categories $\mathbf{C}, \mathbf{D}$ the functor category $\mathbf{C} \to \mathbf{D}$ is given by:

Objects: Functors $F \in \mathbf{C} \to \mathbf{D}$

Morphisms Given $F, G \in \mathbf{C} \to \mathbf{D}$, a morphism is a natural transformation $\alpha \in F \to G$

- If $\mathbf{C}$ is small, the functor category

$$\mathbf{PSh}\,\mathbf{C} = \mathbf{C}^{\mathrm{op}} \to \mathbf{Set}$$

is called *the category of presheaves over* $\mathbf{C}$.

Exercise 13

Spell out the details of the objects and morphisms of $\mathbf{PSh}\,(\mathbb{N}, \leq)$.

We define a functor Y, the Yoneda embedding:

$$Y \in \mathbf{C} \to \mathbf{PSh}\,\mathbf{C}$$
$$Y\,A = \lambda X.\mathbf{C}(X, A)$$

Exercise 14

Show that Y is a functor.

The Yoneda Lemma

Given $F \in \mathbf{PSh}\,\mathbf{C}$ the following are naturally isomorphic in $A \in |\mathbf{C}|$

$$\mathbf{PSh}\,\mathbf{C}(Y\,A, F) \simeq F\,A$$

Exercise 15

Prove the Yoneda Lemma.

The category of categories

CAT

The category of categories is given by:

Objects: Categories

Morphisms: Functors

- This is a category on level 1, $|\textbf{CAT}| \in \mathrm{Set}_2$.
- **CAT** is a 2-category because its homsets are categories themselves and there is a horizontal composition of natural transformations.

Horizontal composition of natural transformations

If $\alpha \in F \to F', \beta \in G \to G'$ then

$$\alpha \cdot \beta \in F \circ G \to F' \circ G'$$
$$(\alpha \cdot \beta)_A = \beta_{G\,A} \circ F\,(\alpha_A)$$

Question

What is the difference between $rev \circ rev$ and $rev \cdot rev$?

Question

We could have defined $\alpha \cdot \beta$ as

$$(\alpha \cdot \beta)_A = G'(\alpha_A) \circ \beta_{FA}$$

Why is this definition equivalent?

Free Monoids

- The forgetful functor:

$$\begin{aligned} U &\in \mathbf{Mon} \to \mathbf{Set} \\ U\,(M, e, *) &= M \end{aligned}$$

- Can we go the other way?
- The free functor:

$$\begin{aligned} F &\in \mathbf{Set} \to \mathbf{Mon} \\ F\,A &= (\mathrm{List}\,A, [], (++)) \end{aligned}$$

- How to specify that F is *free*?

We construct two natural families of maps:

$$\mathbf{Mon}(F\,A, (M, e, *)) \underset{\phi^{-1}}{\overset{\phi}{\rightleftarrows}} \mathbf{Set}(A, U\,(M, e, *))$$

$$\begin{aligned}
&\phi \in (\mathrm{List}\,A \to M) \to A \to M \\
&\phi\,f\,a = f\,[a] \\
&\phi^{-1} \in (A \to M) \to (\mathrm{List}\,A \to M) \\
&\phi^{-1}\,g\,[] = e \\
&\phi^{-1}\,g\,(a :: as) = (g\,a) * (\phi^{-1}\,g\,as)
\end{aligned}$$

Exercise 16

Show:

1. $\phi \circ \phi^{-1} = \mathrm{id}$
2. $\phi^{-1} \circ \phi = \mathrm{id}$

Definition: Adjunction

Given functors:

$$\begin{tikzcd} \mathbf{C} \arrow[r, bend left, "U"] & \mathbf{D} \arrow[l, bend left, "F"] \end{tikzcd}$$

we say that F is left adjoint to U ($F \dashv U$)
or U is right adjoint to F
if there is a natural isomorphism (in $A \in |\mathbf{D}|, B \in |\mathbf{C}|$)

$$\begin{tikzcd} \mathbf{C}(F\,A, B) \arrow[r, bend left, "\phi"] & \mathbf{D}(A, U\,B) \arrow[l, bend left, "\phi^{-1}"] \end{tikzcd}$$

A semilattice (with zero) is a monoid $(M, e, *)$ such that:

commutative , if for all $x, y \in M$:

$$x * y = y * x$$

idempotent , if for all $x \in M$:

$$x * x = x$$

- We define **SLat** as the category of semilattices with zero.
- Morphisms and forgetful functors are defined as for **Mon**

Exercise 17

Construct the free functor $F \in \mathbf{Set} \to \mathbf{SLat}$ and show that F is left adjoint to $U \in \mathbf{SLat} \to \mathbf{Set}$.

Products in **Set**

$$\begin{tikzcd}
 & C \arrow[dl, "f"'] \arrow[d, dashed, "<f,g>"] \arrow[dr, "g"] & \\
A & A \times B \arrow[l, "\pi_0"] \arrow[r, "\pi_1"'] & B
\end{tikzcd}$$

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$
$$\pi_0\,(a, b) = a$$
$$\pi_1\,(a, b) = b$$
$$< f, g > \; c = (f\,c, f\,c)$$

Laws:

$$\pi_0 \circ < f, g >= f$$
$$\pi_1 \circ < f, g >= g$$
$$\frac{\pi_0 \circ h = f \quad \pi_1 \circ h = g}{h =< f, g >}$$

Products

Given objects $A, B \in |\mathbf{C}|$ we say that $A \times B$ is their product if the morphisms π_0, π_1 exists and for every f, g there is a morphism $< f, g >$ so that the following diagram commutes:

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[d, dashed, "<f,g>"] \arrow[dr, "g"] & \\ A & A \times B \arrow[l, "\pi_0"] \arrow[r, "\pi_1"'] & B \end{tikzcd}$$

Moreover, the morphism $< f, g >$ is the unique morphism which makes this diagram commute, i.e.

$$\frac{\pi_0 \circ h = f \quad \pi_1 \circ h = g}{h =< f, g >}$$

Exercise 18

Show that products in **C** give rise to a functor $(\times) \in \mathbf{C}^2 \to \mathbf{C}$.

Exercise 19

Show that the following equation holds

$$< f, g > \circ h =< f \circ h, g \circ h >$$

Exercise 20

Show that the following isomorphism exist in all categories with products:

$$A \times B \simeq B \times A$$

and that the assignment is natural in A, B.

Coproducts in **Set**

$$
\begin{tikzcd}
 & C & \\
A \arrow[ur, "f"] \arrow[r, "\mathrm{inl}"'] & A+B \arrow[u, dashed, "{[f,g]}"] & B \arrow[ul, "g"'] \arrow[l, "\mathrm{inr}"]
\end{tikzcd}
$$

$$
\begin{aligned}
A + B &= \{\mathrm{inl}\, a \mid a \in A\} \cup \{\mathrm{inr}\, b \mid b \in B\} \\
[f, g]\,(\mathrm{inl}\, a) &= f\, a \\
[f, g]\,(\mathrm{inr}\, b) &= g\, b
\end{aligned}
$$

Laws:

$$
\begin{aligned}
[f, g] \circ \mathrm{inl} &= f \\
[f, g] \circ \mathrm{inr} &= g
\end{aligned}
$$

$$
\frac{h \circ \mathrm{inl} = f \quad h \circ \mathrm{inr} = g}{h = [f, g]}
$$

Coproducts

Given objects $A, B \in |\mathbf{C}|$ we say that $A + B$ is their coproduct if the morphisms $\mathrm{inl}, \mathrm{inr}$ exists and for every f, g there is a morphism $[f, g]$ so that the following diagram commutes:

$$\begin{tikzcd} & C & \\ A \arrow[ur, "f"] \arrow[r, "\mathrm{inl}"'] & A + B \arrow[u, dashed, "{[f,g]}"] & B \arrow[ul, "g"'] \arrow[l, "\mathrm{inr}"] \end{tikzcd}$$

Moreover, the morphism $[f, g]$ is the unique morphism which makes this diagram commute, i.e.

$$\frac{h \circ \mathrm{inl} = f \quad h \circ \mathrm{inr} = g}{h = [f, g]}$$

- Products and coproducts are dual concepts:
  Products in $|\mathbf{C}|$ are coproducts in $|\mathbf{C}^{\mathrm{op}}|$ and vice versa.
- Products and coproducts are left and right adjoints of the diagonal functor:

$$\Delta \in \mathbf{C} \to \mathbf{C}^2$$
$$\Delta\, A = (A, A)$$

$$
\begin{tikzcd}[column sep=huge]
\mathbf{C} \arrow[r, "\Delta" description] & \mathbf{C}^2 \arrow[l, bend right=60, "(+)"'] \arrow[l, bend left=60, "(\times)"] \arrow[l, phantom, shift right=12pt, "\perp"] \arrow[l, phantom, shift left=12pt, "\perp"]
\end{tikzcd}
$$

Terminal objects

$1 \in |\mathbf{C}|$ is a terminal object, if for any object $A \in \mathbf{C}$ there is exactly one arrow $!_A$:

$$A \xrightarrow[!_A]{} 1$$

Initial objects

$0 \in |\mathbf{C}|$ is an initial object, if for any object $A \in \mathbf{C}$ there is exactly one arrow $?_A$:

$$0 \xrightarrow[?_A]{} A$$

Question

What are initial and terminal objects in **Set**?

Exercise 21

Show that any two terminal objects are isomorphic.

Global elements

- In **Set** we have that

$$\mathbf{Set}(1, A) \simeq A$$

- Hence the elements of $\mathbf{C}(1, A)$ are called the **global elements** of A.
- A category $\mathbf{C}$ is *well pointed*, if for $f, g \in \mathbf{C}(A, B)$ we have

$$\frac{\forall a \in \mathbf{C}(1, A). f \circ a = g \circ a}{f = g}$$

- **Set** is well pointed.

Exercise 22

Consider **PSh** $(\mathbb{N}, \leq)$ again. What is the terminal object and what are global elements? Show that **PSh** $(\mathbb{N}, \leq)$ is not well pointed.

Exercise 23

Construct the following isomorphism in **Set**:

$$A \times (B + C) \simeq A \times B + A \times C$$

Exercise 24

Show that **CMon** (the category of commutative monoids) has products and coproducts.

Exercise 25

Give a counterexample for the isomorphism:

$$A \times (B + C) \simeq A \times B + A \times C$$

in **CMon**.

Exponentials in **Set**

- In **Set** we have the curry/uncurry isomorphism:

$$A \times B \to C \simeq A \to (B \to C)$$

- Indeed this is an adjunction $F \dashv G$ for

$$F, G \in \mathbf{Set} \to \mathbf{Set}$$
$$F\,X = X \times B$$
$$G\,X = B \to X$$

$$\mathbf{Set}(F\,A, C) \simeq \mathbf{Set}(A, G\,C)$$

Exponentials

Given a category **C** with products. We say that the object $B \in |\mathbf{C}|$ is exponentiable, if the functor $F\,X = X \times B$ has a right adjoint $F \dashv G$, which we write as $G\,X = B \to X$.
A category with products where all objects are exponentiable is called **cartesian closed**.

- $B \to C$ is often written as C^B.

Question

What are the exponentials in **FinSetSkel**?

Exercise 26

Show that the category of typed λ-terms is cartesian closed.

- Indeed, this is the initial cartesian closed category (or the classifying category).

Exercise 27

Show that in a cartesian closed category with coproducts we have that

$$A \times (B + C) \simeq (A \times B) + (A \times C)$$

Corollary

CMon *is not cartesian closed.*

Exercise 28

Show that the presheaf categories ($\mathbf{PSh}\,\mathbf{C}$) are cartesian closed.

Exercise 29

Is there a cartesian closed category whose dual is also cartesian closed?

Pullbacks

Given arrows $f \in \mathbf{C}(A, C)$ and $g \in \mathbf{C}(B, C)$, $(f \times_C g, \pi_0, \pi_1)$ is their pullback, if the diagram below commutes and for every (D, p_0, p_1) there is a unique arrow $< p_0, p_1 >$ such that the diagram commutes:

$$\begin{tikzcd}
D \arrow[dr, dashed, "<p_0,p_1>"] \arrow[drr, bend left, "p_0"] \arrow[ddr, bend right, "p_1"'] & & \\
& f \times_C g \arrow[r, "\pi_0"'] \arrow[d, "\pi_1"'] & A \arrow[d, "f"] \\
& B \arrow[r, "g"'] & C
\end{tikzcd}$$

- Pullbacks in **Set**:

$$f \times_C g = \{(a, b) \in A \times B \mid f\,a = g\,b\}$$

Pushouts

Given arrows $f \in \mathbf{C}(A, B)$ and $g \in \mathbf{C}(A, C)$, $(f +^A g, \mathrm{inl}, \mathrm{inr})$ is their pushout, if the diagram below commutes and for every (D, i_0, i_1) there is a unique arrow $[p_0, p_1]$ such that the diagram commutes:

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, "g"'] & B \arrow[d, "\mathrm{inl}"] \arrow[ddr, bend left, "i_1"] & \\
C \arrow[r, "\mathrm{inr}"'] \arrow[drr, bend right, "i_0"'] & f +^A g \arrow[dr, dashed, "{[i_0, i_1]}"] & \\
 & & D
\end{tikzcd}$$

Exercise 30

What are pushouts in **Set**?

Limits and colimits

Given a small category of diagrams **D**, a **D**-diagram in **C** is given by a functor $F \in \mathbf{D} \to \mathbf{C}$. A cone of a diagram is given by an object $D \in \mathbf{C}$ and a natural transformation $\alpha \in \mathrm{K}_D \to F$ where $\mathrm{K}_D\, X = D$ is a constant functor.
Morphisms between cones (D, α) and (E, β) are given by $f \in D \to E$ such that $\alpha \circ f = \beta$.
The limit of F is the terminal object in the category of cones.
Dually, a cocone is given by a natural transformation $\alpha \in F \to \mathrm{K}_D$, and a morphism of cocones (D, α) and (E, β) are given by $f \in D \to E$ such that $f \circ \alpha = \beta$.
The colimit of F is the initial object in the category of cocones.

Examples

- Products are given by limits of

$$\bullet \qquad \bullet$$

  Note that we are leaving out identity arrows.
- Dually, coproducts are given by colimits of the same diagram.
- Pullbacks are limits of

$$\begin{tikzcd} & \bullet \arrow[d] \\ \bullet \arrow[r] & \bullet \end{tikzcd}$$

- Pushouts are colimits of the dual diagram:

$$\begin{tikzcd} \bullet \arrow[r] \arrow[d] & \bullet \\ \bullet & \end{tikzcd}$$

- Equalizers are limits of

$$\begin{tikzcd} \bullet \arrow[r, shift left] \arrow[r, shift right] & \bullet \end{tikzcd}$$

- Dually, coequalizers are colimits of the same diagram.

Exercise 31

What are equalizers and coequalizers in **Set**?

Exercise 32

Show that pullbacks can be constructed from equalizers and products.

- Actually, all finite limits can be constructed from equalizers and finite products (i.e. binary products and terminal objects).

- Diagrams of $(\mathbb{N}, \leq)$ are called ω-chains:

$$A\,0 \xrightarrow[a\,0]{} A\,1 \xrightarrow[a\,1]{} A\,2 \xrightarrow[a\,2]{} \dots$$

  Note that we are leaving out the composites of arrows.

- An ω-chain in **Set** is given by

$$\begin{aligned} &A \in \mathbb{N} \to \mathrm{Set} \\ &a \in \Pi n \in \mathbb{N}.A\,n \to A\,(n+1) \end{aligned}$$

- We write $\mathrm{colim}\,(A, a)$ for the colimit of an ω-chain.

Exercise 33

What is the colimit of the following chain?

$$\begin{aligned} A\,n &= \bar{n} \\ a\,n\,i &= i \end{aligned}$$

- Dually, Diagrams of $(\mathbb{N}, \geq)$ are called ω-cochains:

$$A\,0 \xleftarrow{a\,0} A\,1 \xleftarrow{a\,1} A\,2 \xleftarrow{a\,2} \dots$$

- An ω-cochain in **Set** is given by

$$\begin{aligned} A &\in \mathbb{N} \to \mathrm{Set} \\ a &\in \Pi n \in \mathbb{N}.A\,(n+1) \to A\,n \end{aligned}$$

- We write $\lim\,(A, a)$ for the limit of an ω-cochain.

Exercise 34

Given a set $X \in \mathrm{Set}$. What is the limit of the following chain?

$$\begin{aligned} A\,n &= \bar{n} \to X \\ a\,n\,f &= \lambda i.f\,i \end{aligned}$$

- Natural numbers $\mathbb{N} \in \mathrm{Set}$ are given by:

$$\begin{array}{rcl} 0 & \in & \mathbb{N} \\ & \simeq & 1 \to \mathbb{N} \\ \mathrm{S} & \in & \mathbb{N} \to \mathbb{N} \end{array}$$

- We can combine the two constructors in one morphism:

$$[0, \mathrm{S}] \in 1 + \mathbb{N} \to \mathbb{N}$$

- The functor $T\,X = 1 + X$ is called the signature functor.
- A pair $(A \in \mathrm{Set}, f \in 1 + A \to A)$ is a $1+$-algebra.

- For any $1+$-algebra (A, f) there is a unique morphism $\mathrm{fold}\,(A, f)$ such that the following diagram commutes:

$$\begin{tikzcd}
1+\mathbb{N} \arrow[r, "{[0,\mathrm{S}]}"] \arrow[d, "1+(\mathrm{fold}\,(A,f))"'] & \mathbb{N} \arrow[d, "\mathrm{fold}\,(A,f)"] \\
1+A \arrow[r, "f"'] & A
\end{tikzcd}$$

with

$$\begin{aligned}
\mathrm{fold}\,(A, f)\,0 &= f\,(\mathrm{inl}\,()) \\
\mathrm{fold}\,(A, f)\,(\mathrm{S}\,n) &= f\,(\mathrm{inr}\,(\mathrm{fold}\,(A, f)\,n))
\end{aligned}$$

Exercise 35

Define addition $(+) \in \mathbb{N} \to \mathbb{N} \to \mathbb{N}$ using fold.

T-algebras

Given an endofunctor $T \in \mathbf{C} \to \mathbf{C}$ the category of T-algebras is given by

Objects T-algebras (A, f) with

$$T\,A \xrightarrow[f]{} A$$

Morphisms Given T-algebras (A, f),(B, g) a T-algebra morphism is a morphism $h \in \mathbf{C}(A, B)$ such that

$$\begin{tikzcd} T\,A \arrow[r, "f"'] \arrow[d, "T\,h"'] & A \arrow[d, "h"'] \\ T\,B \arrow[r, "g"'] & B \end{tikzcd}$$

commutes.

Initial T-algebras

The initial object (if it exists) in the category of T-algebras is denoted as $(\mu T, \mathrm{in}_T)$. For every T-algebra (A, f) there is a unique morphism $\mathrm{fold}_T\,(A, f)$ such that

$$\begin{tikzcd} T\,(\mu\,T) \arrow[r, "\mathrm{in}_T"] \arrow[d, "T\,(\mathrm{fold}\,(A{,}f))"'] & \mathbb{N} \arrow[d, "\mathrm{fold}\,(A{,}f)"] \\ T\,A \arrow[r, "f"'] & A \end{tikzcd}$$

commutes.

- Given $A \in \mathrm{Set}$ the set of streams over A: A^ω comes with two destructors

$$\begin{aligned} \mathrm{hd} &\in A^\omega \to A \\ \mathrm{tl} &\in A^\omega \to A^\omega \end{aligned}$$

- We can combine the two destructors in one morphism:

$$< \mathrm{hd}, \mathrm{tl} > \in A^\omega \to A \times A^\omega$$

- A pair $(X \in \mathrm{Set}, f \in X \to A \times X)$ is a $A\times$-coalgebra.

- For any $A\times$-algebra (X, f) there is a unique morphism $\mathrm{unfold}\,(X, f)$ such that the following diagram commutes:

$$\begin{tikzcd} X \arrow[r, "f"'] \arrow[d, "\mathrm{unfold}\,(X,f)"'] & A \times X \arrow[d, "A\times\mathrm{unfold}\,(X,f)"] \\ A^\omega \arrow[r, "<\mathrm{hd},\mathrm{tl}>"'] & A \times A^\omega \end{tikzcd}$$

with

$$\begin{aligned} \mathrm{hd}(\mathrm{unfold}\,(X, f)\,x) &= \pi_0(f\,x) \\ \mathrm{tl}(\mathrm{unfold}\,(X, f)\,x) &= \mathrm{unfold}\,(X, f)\,(\pi_1\,(f\,x)) \end{aligned}$$

Exercise 36

Define the function $\mathrm{from} \in \mathbb{N} \to \mathbb{N}^\omega$, which produces the stream of natural numbers starting with a given number, using unfold.

T-coalgebras

Dually, given an endofunctor $T \in \mathbf{C} \to \mathbf{C}$ the category of T-coalgebras is given by

Objects T-coalgebras (A, f) with

$$A \xrightarrow[f]{} T\,A$$

Morphisms Given T-coalgebras (A, f),(B, g) a T-coalgebra morphism is a morphism $h \in \mathbf{C}(A, B)$ such that

$$\begin{array}{ccc} A & \xrightarrow{f} & T\,A \\ {\scriptstyle h}\downarrow & & \downarrow{\scriptstyle T\,h} \\ B & \xrightarrow[g]{} & T\,B \end{array}$$

commutes.

Terminal T-coalgebras

The terminal object (if it exists) in the category of T-coalgebras is denoted as $(\nu T, \mathrm{out}_T)$. For every T-coalgebra (A, f) there is a unique morphism $\mathrm{unfold}_T\,(A, f)$ such that

$$
\begin{array}{ccc}
A & \xrightarrow{f} & T\,A \\
{\scriptstyle \mathrm{unfold}\,(A,f)}\downarrow & & \downarrow{\scriptstyle T\,(\mathrm{unfold}\,(X,f))} \\
\nu\,T & \xrightarrow[\mathrm{out}_T]{} & T\,(\nu\,T)
\end{array}
$$

Lambek's lemma

- Initial algebras and terminal coalgebras are always isomorphisms.
- We construct the inverse of $\mathrm{in}_T \in \mathbf{C}(T\,(\mu\,T), \mu\,T)$ as

$$\begin{aligned}\mathrm{in}_T^{-1} &\in \mathbf{C}(\mu\,T, T\,(\mu\,T))\\ \mathrm{in}_T^{-1} &= \mathrm{fold}_T\,(T\,(\mu\,T), T\,\mathrm{in}_T)\end{aligned}$$

- Dually, we construct an inverse to out_T.

Exercise 37

Construct explicitely the inverses to $[0, S]$ (for natural numbers) and $< \mathrm{hd}, \mathrm{tl} >$ (for streams).

Exercise 38

Prove Lambek's lemma, i.e. show that in_T^{-1} is inverse to in_T.

- A functor T is called ω-cocontinous if it preserves colimits of ω-chains, that is

$$T\,(\mathrm{colim}\,(A, a)) \simeq \mathrm{colim}\,(\lambda n. T\,(A\,n), \lambda n. T\,(a\,n))$$

- We can construct the initial T-algebra of an ω-cocontinous functor T by constructing the colimit of the following chain:

$$0 \xrightarrow[?]{} T\,0 \xrightarrow[T\,?]{} T^2\,0 \xrightarrow[T^2\,?]{} \dots$$

Exercise 39

Complete the construction, and show that the colimit is indeed an initial T-algebra.

Exercise 40

Dualize the previous slide. What is an ω-continous functor? How can we construct its terminal coalgebra?

Exercise 41

Which of the following endofunctors on Set are ω-cocontinous, and which are ω-continous:

$$\begin{array}{rcl} T_1\,X & = & X \times X \\ T_2\,X & = & \mathbb{N} \to X \\ T_3\,X & = & (X \to \mathbb{N}) \to \mathbb{N} \end{array}$$

- We define the functor of binary trees with labelled leafs:

$$BT \in \mathbf{Set} \to \mathbf{Set}$$
$$BT\,X = \mu Y.X + Y \times Y$$

We write $\mathrm{L} = \mathrm{in} \circ \mathrm{inl}$ and $\mathrm{N} = \mathrm{in} \circ \mathrm{inr}$ for the constructors.

- The natural transformation η constructs a leaf:

$$\eta_A \in A \to BT\,A$$
$$\eta_A = \lambda a.\mathrm{L}\,a$$

- We define a natural transformation bind, which replaces each leaf by a tree.

$$\mathrm{bind}_{A,B} \in (A \to BT\,B) \to BT\,A \to BT\,B$$
$$\mathrm{bind}_{A,B}\,f\,(\mathrm{L}\,a) = f\,a$$
$$\mathrm{bind}_{A,B}\,f\,(\mathrm{N}\,(l, r)) = \mathrm{N}\,(\mathrm{bind}_{A,B}\,f\,l, \mathrm{bind}_{A,B}\,f\,r)$$

- Haskell's $(>>=)$ can be defined as $a >>= f = \mathrm{bind}\,f\,a$.

Monads (Kleisli triple)

A monad on **C** is a triple (T, η, bind) with

$$\begin{aligned} T &\in \mathbf{C} \to \mathbf{C} \\ \eta &\in \mathbf{C}(A, T\,A) \\ \mathrm{bind} &\in \mathbf{C}(A, T\,B) \to \mathbf{C}(T\,A, T\,B) \end{aligned}$$

such that

$$\begin{aligned} (\mathrm{bind}\,\eta) &= \mathrm{id} \\ \mathrm{bind}\,(f) \circ \eta &= f \\ (\mathrm{bind}\,f) \circ (\mathrm{bind}\,g) &= \mathrm{bind}\,((\mathrm{bind}\,f) \circ g) \end{aligned}$$

Exercise 42

Show that the operations on binary trees satisfy the laws of a monad.

Exercise 43

Show that the following functors over **Set** give rise to monads (assuming $E, S \in \mathrm{Set}$):

$$
\begin{aligned}
T_{\mathrm{Error}}\,X &= E + X \\
T_{\mathrm{State}}\,X &= S \to (X \times S)
\end{aligned}
$$

Monad

A monad on **C** is a triple (T, η, μ) with

$$\begin{array}{rcl} T & \in & \mathbf{C} \to \mathbf{C} \\ \eta & \in & I \to T \\ \mu & \in & T^2 \to T \end{array}$$

(where $T^2 = T \circ T$) such that the following diagrams commute.

$$\begin{tikzcd} T \arrow[r, "\eta T"] \arrow[d, "T\eta"'] \arrow[dr, equal] & T^2 \arrow[d, "\mu"] \\ T^2 \arrow[r, "\mu"'] & T \end{tikzcd} \qquad \begin{tikzcd} T^3 \arrow[r, "T\mu"] \arrow[d, "\mu T"'] & T^2 \arrow[d, "\mu"] \\ T^2 \arrow[r, "\mu"'] & T \end{tikzcd}$$

Exercise 44

Show that the two definitions are equivalent.

- We define infinite, labelled binary trees:

$$BT^{\infty} \in \mathbf{Set} \to \mathbf{Set}$$
$$BT^{\infty}\, X = \nu Y.X \times (Y \times Y)$$

- The operation ϵ extracts the top label:

$$\epsilon \in BT^{\infty}\, A \to A$$
$$\epsilon\,(a,(l,r)) = a$$

- cobind relabels a tree recursively:

$$\mathrm{cobind} \in (BT^{\infty}\, A \to B) \to (BT^{\infty}\, A \to BT^{\infty}\, B)$$
$$\mathrm{cobind}\, f\, t = (f\, t, \mathrm{cobind}\, f\,(\pi_2 t), \mathrm{cobind}\, f\,(\pi_3 t))$$

Exercise 45

Show that $(BT^{\infty}, \epsilon, \mathrm{cobind})$ is a comonad, i.e. a monad in $\mathbf{Set}^{\mathrm{op}}$.

Kleisli category

Given a monad (T, η, bind) on $\mathbf{C}$ we define the Kleisli category $\mathbf{C}_T$ as:

Objects: $|\mathbf{C}|$

Morphisms: $\mathbf{C}_T\,A\,B = \mathbf{C}(A, T\,B)$

Identity: $\eta \in \mathbf{C}_T\,A\,A$

Composition: Given $f \in \mathbf{C}_T\,B\,C$, $g \in \mathbf{C}_T\,A\,B$ we define

$$f \circ_T g = (\mathrm{bind}\,f) \circ g$$

Exercise 46

Verify that that $\mathbf{C}_T$ is indeed a category.

Exercise 47

Explicitely construct the Kleisli-categories of T_{Error} and T_{State}

Given an adjunction $F \dashv U$

$$\begin{tikzcd} \mathbf{D}(F\,A, B) \arrow[r, bend left, "\phi"] & \mathbf{C}(A, U\,B) \arrow[l, bend left, "\phi^{-1}"] \end{tikzcd}$$

we define:

$$\begin{array}{rcl} \eta & \in & \mathbf{C}(A, U\,(F\,A)) \\ \eta & = & \phi\,(\mathrm{id}_{F\,A}) \\ \epsilon & \in & \mathbf{D}(F, U\,B)B \\ \epsilon & = & \phi^{-1}\,(\mathrm{id}_{U\,B}) \end{array}$$

this gives rise to a monad (T, ϵ, μ) on **C**

$$\begin{array}{rcl} T & = & UF \\ \mu & = & U\epsilon F \end{array}$$

Exercise 48

Spell out the constructed monad in the case where $F \in \mathbf{Set} \to \mathbf{Mon}$ is the free monad functor and $U \in \mathbf{Mon} \to \mathbf{Set}$ the forgetful functor

Exercise 49

Verify the monad laws of the construction of a monad from an adjunction.

- Using $\mathbf{C}_T$ we can also go the other way: $\mathbf{C}_T$ gives rise to an adjunction $F_T \dashv U_T$ such that $T = U_T \circ F_T$:

$$
\begin{aligned}
F_T &\in \mathbf{C} \to \mathbf{C}_T \\
F_T A &= A \\
F_T\, f &= \eta \circ f \\
U_T &\in \mathbf{C}_T \to \mathbf{C} \\
U_T\, A &= T\, A \\
U_T\, f &= \mu \circ T\, f
\end{aligned}
$$

Exercise 50

Verify that $F_T \dashv U_T$.

- This is not the only way to factor a monad into an adjunction. Another construction is the Eilenberg-Moore category $\mathbf{C}^T$, indeed the two are initial and terminal objects in the category of factorisations.