Introduction to Category Theory

Alexander Kurz

April 13, 2017

Contents

0 Introduction

Each section is a lecture.

I tried to keep the lectures on a conceptual level. There are almost no proofs. These are delegated to the exercises. If you are new to category theory and you want more than a first overview, do some of the exercises. Best is to have a favourite category at hand to instantiate the definitions and concepts. There are many exercises and one does not necessarily have to do all of them as they usually follow a theme and are sometimes quite similar. Some may be difficult or require extra knowledge, so it may be good to have a tutor at hand.

1 Objects and Arrows

Synopsis. *Theme:* Set theory based on the notion of composition as the primitive notion. *Definitions:* Category, dual category, initial, terminal/final, isomorphism, epi, mono, product, coproduct, equaliser, coequaliser, power. *Concepts:* Set theory without elements, universal constructions, up to canonical isomorphism, instantiating the same definitions in different categories, duality, extending universal constructions from objects to arrows.. *Theorems:* Universal constructions are determined up to canonical isomorphism.

Introduction. Set theory is very useful as a language for mathematics. For example, it allows to make precise the idea of equivalence classes and is therefore the basis of many construction in mathematics and computer science. To give examples of mathematical objects which are equivalence classes, we can start with integers, rationals, and reals. But also in programming, for example in compiler optimisation, we need to know when two programs are equivalent. This leads to an important idea in the semantics of programming languages: The meaning of a program is the equivalence class of all programs that show the same behaviour in all computational situations.

Despite all of this, there are many situations in mathematics and computer science, where it is better to formulate problems in category theory rather than in set-theory. For example, the computational power of computers comes from the fact that finite programs can compute infinite data (eg a program computing all decimals of π can be written in a few lines). Or to say it more mathematically, the computational power of computer comes from the fact that they implement solutions to recursive equations. Now, as we will see later, it turns out that such recursive equations, as soon as we lift them from numbers to types, have solutions not up to equality, but only up to *isomorphism*, which puts us right into categorical territory. Of course, we will encounter many other reasons for why category theory is so important in computer science.

Definition 1. A **category** $\mathcal{A}$ consists of a collection of 'objects' and for all objects $A, B \in \mathcal{A}$ a set $\mathcal{A}(\mathcal{A}, \mathcal{B})$ of 'arrows'. We write $f : A \to B$ for $f \in \mathcal{A}(\mathcal{A}, \mathcal{B})$. For all $A \in \mathcal{A}$ there is an 'identity' arrow $\mathrm{id}_A : A \to A$ and for all A, B, C there is a 'composition' operation $\circ_{ABC} : \mathcal{A}(B, C) \times \mathcal{A}(A, B) \to \mathcal{A}(A, C)$. We write $f \circ g = h$ for $\circ_{ABC}(f, g) = h$. The axioms are for all $h : A \to B$

$$h \circ \mathrm{id}_A = h = \mathrm{id}_B \circ h$$

and for all $f : C \to D, g : B \to C$, $h : A \to B$

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

Remark 1. It is possible to define the notion of a category as a "one-sorted" algebra of arrows only, without referring to objects. Even though that seems to be in the spirit of category theory ("everything is determined by the arrows"), it runs against the practice of category theory in which, most of the time, everything is in fact determined by the objects. We will see many instances of this during the course.

Remark 2. A category with only one object is a monoid. That is, a category can be seen as a many-sorted monoid. Thinking about how important monoids are in computer science and how important the idea is of having many sorts or types, it is not surprising that categories pervade computer science.

Example 3.

1. The category Set has sets as objects and functions as arrows.
2. The category Pos has partially ordered sets as objects and monotone functions as arrows.[1]
3. The category Rel has sets as objects and relations as arrows.
4. The category Monoid of monoids with monoids as objects and monoid morphisms as arrows.
5. If you are familiar with topological or metric spaces, then these structures also form categories. For example, we denote by Top the category of topological spaces and continuous maps.
6. Every monoid is a category with exactly one object.
7. Every poset (partially ordered set) is a category with at most on arrow between any two objects.

Remark 4. The first examples are examples where structures organise themselves in a category. The last examples are examples where a structure appears in the form of a category.

In the following we are going to see that one can perform set-theoretical definitions and constructions without talking about elements, taking instead identies and composition as primitive notions. In particular, we will see category theoretic definitions for

- injective and surjective,
- the empty set and the one-element set,
- disjoint union and cartesian product,
- subsets and quotients,
- powerset,
- natural numbers.

This parallels closely the axiomatisation of Zermelo-Fraenkel set theory.

[1] A partial order on a set X is a relation $\leq \subseteq X \times X$ which is reflexive, symmetric and transitive. A function f is monotne if $x \leq x' \Rightarrow f(x) \leq f(x')$.

Remark 5. It would be a challenging but interesting exercise to detail where in the following properties of categories such as Pos, Rel, Top, Monoid deviate from Set. One would see that some categories, such as Pos are quite similar to Set, whereas others, such as Rel are radically different. Algebraic and topological categories (such as Monoid and Top) are somewhere in between, but the details will depend on exactly which category of algebras or topological spaces one would be looking at.

Definition 2. An arrow m in a category is **mono** if it is left-cancellative, that is, if for all f, g we have that $m \circ f = m \circ g$ implies $f = g$.

We also say that m is a monomorphism, or that m is monic.

Exercise 6. *An arrow in* Set *is mono iff it is injective.*

What can we say about monos in the other examples of categories?

Duality is an important concept in mathematics. In many circumstances, category theory provides exactly what is needed to formalise it.

Definition 3. The **dual** $\mathcal{C}^{\mathrm{op}}$ of the category $\mathcal{C}$ has the same objects as $\mathcal{C}$ and $\mathcal{C}^{\mathrm{op}}(A, B) = \mathcal{C}(B, A)$. Given $f : A \to B$ in $\mathcal{C}$ we write f^{op} to denote $f : B \to A$ in $\mathcal{C}^{\mathrm{op}}$. Identities and composition in $\mathcal{C}^{\mathrm{op}}$ are defined as in $\mathcal{C}$.

Note that duality reverses composition: if $h = f \circ g$ then $h^{\mathrm{op}} = g^{\mathrm{op}} \circ f^{\mathrm{op}}$ (one could argue that it would be more precise to write $h = g \circ^{\mathrm{op}} f$.

Definition 4. An arrow in $\mathcal{C}$ is **epi** iff it is mono in $\mathcal{C}^{\mathrm{op}}$.

Equivalently, an arrow e is epi if for all f, g we have that $f \circ e = g \circ e$ implies $f = g$.

Exercise 7. *An arrow in* Set *is epi iff it is surjective.*

While it is true that being epi captures the set-theoretic property of being surjective in many categories, this is not always the case. For example the inclusion of natural numbers into integers is epi in the category of monoids.

Definition 5. An object A in $\mathcal{C}$ is **initial** if for all $C \in \mathcal{C}$ there is exactly one arrow $A \to C$.

Exercise 8. *An object in* Set *is initial iff it is the empty set.*

Definition 6. An object Z in $\mathcal{C}$ is **final** (or **terminal**) if it is initial in $\mathcal{C}^{\mathrm{op}}$.

Equivalently, Z is final if for all $C \in \mathcal{C}$ there is exactly one arrow $C \to Z$.

Exercise 9. *An object in* Set *is final iff it has exactly one element.*

Definition 7. An arrow $f : X \to Y$ is an **isomorphism**, or **iso**, if there is $g : Y \to X$ such that $f \circ g = \mathrm{id}_Y$ and $g \circ f = \mathrm{id}_X$.

One should check that if f is an iso then its inverse g is uniquely determined.

Exercise 10.

1. *In* Set *evry arrow that is epi and mono is iso.*
2. *Find an example of a category which has an arrow that is both mono and epi but not iso.*
3. *Let X, Y be two final objects. Then they are isomorphic. Moreover, the isomorphism is uniquely determined.*

So far we only have described categorically the empty set and the one-element set. To build bigger sets we can use

Definition 8. Let A, B be two objects in a category $\mathcal{A}$. The **coproduct** of A and B is an object $A+B$ in $\mathcal{A}$ together with two arrows $\mathrm{inl} : A \to A+B$ and $\mathrm{inr} : B \to A+B$ such that for all arrows $f : A \to C, g : A \to C$ there is a unique arrow $h : A+B \to C$ such that $h \circ \mathrm{inl} = f$ and $h \circ \mathrm{inr} = g$.

Exercise 11.

1. *All coproducts of A and B are isomorphic up to 'canonical' isomorphism.* [2]
2. *The coproduct in* Set *is (isomorphic to) the disjoint union.*
3. *Let A^* be the monoid of words over an alphabet A. Show that $A^* + B^* = (A+B)^*$. Conclude that the coproduct in the category of monoids is not disjoint.*

Definition 9. $(A \times B \to A, A \times B \to B)$ is a **product** in $\mathcal{A}$ if it is a coproduct in $\mathcal{A}^{\mathrm{op}}$.

Exercise 12. *Fix a category $\mathcal{A}$.*

1. *Let A, B be two objects in a category $\mathcal{A}$. The product of A and B is an object $A \times B$ in $\mathcal{A}$ together with two arrows $\pi_A : A \times B \to A$ and $\pi_B : A \times B \to B$ such that for all arrows $f : C \to A, g : C \to B$ there is a unique arrow $h : C \to A \times B$ such that $\pi_A \circ h = f$ and $\pi_B \circ h = g$.*
2. *$(A \times B) \times C) \cong A \times (B \times C)$.*
3. *In you favourite axiomatisation of set theory, do you have $(A \times B) \times C) = A \times (B \times C)$?*
4. *Given $f : A \to A'$ and $g : B \to B'$, define $f \times g : A \times B \to A' \times B'$.*

The last item gives a typical example showing how to extend universal constructions from objects to arrows. This gives a first example showing that often, if you know a construction on objects, then you know how to extend it to arrows.

One way to look at what we have done so far is that we gave definitions for $0, 1, +, \times$ for sets instead of numbers. So it will not come as a suprise that next on the list is exponentiation.

[2]'Canonical' here means that even though there may be many isomorphisms, there is only one that commutes with inl and inr.

Definition 10. Let $\mathcal{A}$ be a category with products. The exponentiation of A to the power of B is an object A^B in $\mathcal{A}$ together with an 'evaluation' arrow $\mathrm{eval} : B \times A^B \to A$ such that for all $f : B \times C \to A$ there is a unique $h : C \to A^B$ such that $\mathrm{eval} \circ (\mathrm{id}_B \times h) = f$. A category with products and exponentiation is called **cartesian closed**.

Exercise 13. *Show that exponentiation induces a bijection between arrows $B \times C \to A$ and $C \to A^B$. This is also known as currying.*

Exercise 14. $A^{B \times C} \cong (A^B)^C$. *What about other laws such as $A^{B+C} \cong A^B \times A^C$? For more on this see eg Fiore, Cosmo, and Balat. Remarks on Isomorphisms in Typed Lambda Calculi with Empty and Sum Types http://www.dicosmo.org/Papers/lics02.pdf*

We have built quite an arsenal of universal constructions: initial, terminal, coproduct, product, exponentiation. Moreover, all these universal constructions follow the same pattern: [...] such that for all arrows [...] there is a unique arrow [...] such that the equation [...] holds. Thus, even if this variety of constructions will seem puzzling to the novice, the pattern will become familiar after a while. Moreover, the the pattern is not just pleasing, it has some important consequences. We have seen in particular examples, but that is true for all of them, that

- Objects defined by universal constructions are unique up to canonical isomorphism.
- Universal constructions extend from objects to arrows.[3]

Next on our list are quotients and subsets.

Definition 11. Let $f, g : A \to B$ be two arrows in a category $\mathcal{A}$. The **equaliser** of f, g is an object E together with an arrow $e : E \to A$ such that $f \circ e = g \circ e$ and for all $e' : E' \to A$ with $f \circ e' = g \circ e'$ there is a unique h such that $e \circ h = e'$. (E, e) is a **coequaliser** of (f, g) if (E, e^{op}) is an equaliser of $(f^{\mathrm{op}}, g^{\mathrm{op}})$ in $\mathcal{A}^{\mathrm{op}}$.

Exercise 15.

1. *In* Set *we can take the equaliser to be $E = \{a \in A \mid fa = ga\}$ and e to be the inclusion $E \to A$.*
2. *Let $f, g : B \to A$ be two arrows in a category $\mathcal{A}$. The* **coequaliser** *of f, g is an object E together with an arrow $e : A \to E$ such that $e \circ f = c \circ g$ and for all $e' : A \to E'$ with $e' \circ f = e' \circ g$ there is a unique h such that $h \circ e = e'$.*
3. *In* Set *we can take the coequaliser to be the quotient of A by the equivalence relation generated by $\{(fb, gb) \mid b \in B\}$.*
4. *Equalisers are mono and coequalisers are epi.*
5. *The inclusion of natural numbers with 0 and + into the integers is an epi. But is it a coequaliser?*

[3] In the next section we will learn to say: Universal constructions are functorial.

Nevertheless, we need two more constructions to get an interesting set theory. The first allows us to construct powersets.

Definition 12. An arrow $t : 1 \to \Omega$ form a terminal object to some object Ω is called a **subobject classifier** in the category $\mathcal{A}$, if for all objects A and all monos $m : B \to A$ there is a unique $h : A \to \Omega$ such that $h \circ m = t \circ !$ where $! : B \to 1$ is the unique arrow into the terminal object.

Exercise 16.

1. *In* Set*, we can take* $t : 1 \to \Omega$ *to be the inclusion* $\{1\} \to \{0, 1\}$*. The universal property of the subobject classifier then induces a bijection between subsets* $B \subseteq A$ *and characteristic functions* $A \to \{0, 1\}$*.*
2. *In* Set*, we have that* Ω^A *is the powerset of* A *for all sets* A*.*

Remark 17. A category with terminal object, products, equalisers, exponentiation and subobject classifier is knwon as an **elementary topos**. Elementary toposes can be seen as generalised universes of sets and as models for higher order intuitionistic logic. In particular, a two-element subobject classifier is rather the exception: the typical subobject classifier is not a Boolean algebra but a Heyting algebra.

Finally, we need one more construction, namely an infinite set.

Definition 13. An object N in a category $\mathcal{A}$, together with arrows $z : 1 \to N$ and $s : N \to N$, is a **natural numbers object** if for all N', $z' : 1 \to N'$ and $s' : N' \to N'$ there is a unique $f : N \to N'$ such that $fz = z'$ and $fs = s'f$.

Further Reading To see how the ideas of this section can be used to give a foundation of set-theory see Lawvere, An elementary theory of the category of sets, Reprints in Theory and Applications of Categories, No. 12, 2005, pp. 135. `http://www.tac.mta.ca/tac/reprints/articles/11/tr11.pdf`

Or, the same material developed into a beginner's level text book (and Lawvere is always worth reading): Sets for Mathematics by F. William Lawvere, Robert Rosebrugh.

For a general introduction to category theory as well as for details on how the ideas presented in this section lead to topos theory see Barr and Wells, Toposes, Triples and Theories, Reprints in Theory and Applications of Categories, No. 12 (2005) pp. 1-287. `http://www.tac.mta.ca/tac/reprints/articles/12/tr12.pdf`

A good introduction to topos theory is also MacLane, Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory. The standard reference to the subject is Johnstone's Sketches of an Elephant: A Topos Theory Compendium.

2 Functors and Natural Transformations

Synopsis. *Definitions:* Functor, natural transformations, functor categories, presheaves. *Concepts:* Objects as types, functors as type constructors or parametric types, natural transformations as parametric functions; Yoneda lemma to characterise natural transformations *Theorems:* Yoneda lemma.

Introduction. In the previous section, we have seen how to use definitions by universal properties in order to build set theory based on the notion of function instead of elementship. [4] In this section, we will enter territory that cannot, or only with considerable notational difficulty, treated in the traditional set theoretic way. In particular, we encounter *natural transformations*, the notion for which category theory was invented. It plays an important role in many areas of mathematics and theoretical computer science.

Notice: From now on examples are exercises. I am aware that these notes are a bit rough, for the missing details consult the excellent `http://www.staff.science.uu.nl/~ooste110/syllabi/catsmoeder.pdf`.

Definition 14. Let $\mathcal{A}, \mathcal{B}$ be two categories. A **functor** $F : \mathcal{A} \to \mathcal{B}$ is a function from the objects of $\mathcal{A}$ to the objects of $\mathcal{B}$ and, for each pair A, A' of objects of $\mathcal{A}$ a function $F_{AA'} : \mathcal{A}(A, A') \to \mathcal{B}(FA, FA')$ that preserves identities and composition.

Example 18. Universal properties give rise to functors. (Assume in the following that the relevant categories admit the respective universal constructions.) Fix an object A in the category $\mathcal{A}$.

1. The identity $\mathrm{Id}_{\mathcal{A}} : \mathcal{A} \to \mathcal{A}$ is a functor.
2. $FX = A \times X$ defines a functor $\mathcal{A} \to \mathcal{A}$.
3. $F(X, Y) = X \times Y$ defines a functor $\mathcal{A} \times \mathcal{A} \to \mathcal{A}$. (That requires the definition of $\mathcal{A} \times \mathcal{A}$ first.)
4. $FX = X^A$ defines a functor $\mathcal{A} \to \mathcal{A}$.
5. $FX = A^X$ defines a functor $\mathcal{A}^{\mathrm{op}} \to \mathcal{A}$.
6. ...
7. ... add your own ...
8. ...

Remark 19. 1. Above, it is enough to define functors on objects only.

[4] Actually, nothing in the axioms of a category captures the idea of a function. Indeed, relations form a category as well. Nevertheless, the definitions in the last section were found by thinking of arrows as functions. So which of the axioms of the last section capture something specific about sets and functions?

2.

An example where the definition of the action on maps does not follow (in an obvious way) from a universal property.

Example 20. Define $\mathcal{P} : \mathsf{Set} \to \mathsf{Set}$ as $\mathcal{P}X$ the powerset of X and on maps as direct image.

We can think of functors as data type constructors: We have seen the type of pairs of the type of sets. Here is another important example.

Example 21. Define $\mathsf{List} : \mathsf{Set} \to \mathsf{Set}$ as $\mathsf{List}X$ the set of lists over X. On functions, List works 'pointwise'.

Another useful way to think about a functor F is as an object FX parameterised by X.

But, then, what are the arrows between these parameterised objects?

Definition 15. Let $F, G : \mathcal{A} \to \mathcal{B}$. A **natural transformation** $\tau : F \to G$ is a parameterised collection $\tau_A : FA \to GA$ satisfying $Gf \circ \tau_A = \tau_{A'} \circ Ff$ for all $f : A \to A'$.

How do we know that this definition is the 'right' definition? We will collect some evidence in the form of examples and theorems.

Example 22.

1. There is only one natural transformation $\mathrm{Id}_{\mathsf{Set}} \to \mathrm{Id}_{\mathsf{Set}}$.
2. Let $F, G : \mathsf{Set} \to \mathsf{Set}$ be given by $FX = A \times X$ and $GX = B \times X$. The natural transformations $F \to G$ are in bijection with functions $A \to B$.
3. Abstracting from the order or multiplicity of the elements of a list gives a natural transformation $\mathsf{List} \to \mathcal{P}$.

Taking these examples further, what are the natural transformations

$$\mathsf{List} \to \mathsf{List}$$

A precise answer involves some combinatorial details, but here is it familiar terms: *The natural transformations* $\mathsf{List} \to \mathsf{List}$ *are all functions* $\mathsf{List}(A) \to \mathsf{List}(A)$ *that can be programmed parametric in* A.

Proposition 23. *Let $\mathcal{A}, \mathcal{B}$ be two categories. Then taking functors $\mathcal{A} \to \mathcal{B}$ as objects and natural transformations as arrow, we obtain a category of functors $[\mathcal{A}, \mathcal{B}]$.*

Remark 24. In our definition of a category, the collection of objects formed a class, such as the universe of sets itself, and the homsets where sets. Then $[\mathcal{A}, \mathcal{B}]$ need not be a category, because there are too many both functors and natural transformations. There are two common solutions. One is to restrict $\mathcal{A}$ to a so-called small category, that is, a category with a set of objects, not a proper class of objects. The other is to stipulate that every collection of objects is a set in some universe, see the section on Foundations in MacLane, Categories for the working mathematician.

The details can be safely ignored most of the time, but what is important is the size distinction we have in the definition of categories: Where as the collection of objects is allowed to be large, the homsets need to be small. We will be able to explain this later ...

We say that a category is **small** if the collection of objects is a set. One can think of the definition of natural transformation as exactly what is needed to make the following work:

Theorem 25. *The category of small categories is cartesian closed, that is, it has products and exponentiation.*

Remark 26. In the last section section, we have seen that set theory can be developed in terms of category theory. But one can push this even further and the idea of the category of categories as a foundation of mathematics has been influential.

We started with the idea of sets and functions forming a category, with sets as objects and functions as arrows. Then we added the idea of functors as arrows between categories and saw that categories and functors form again a category. We also said that functors are parameterised objects. What then do we get in the special case where functors are parameterised sets?

These parameterised sets are known as presheaves and play a central role in all of category theory. They also have many applications in computer science.

Definition 16. Let $\mathcal{C}$ be a small category. The category $[\mathcal{C}^{\mathrm{op}}, \mathsf{Set}]$ is called the **category of presheaves** on $\mathcal{C}$.

Example 27 (Presheaves as transition systems)**.** Let ω be the category of ordinals with an arrow $n \to m$ iff $n \le m$. The a presheaf $P : \omega^{\mathrm{op}} \to \mathsf{Set}$ is the set of trees with $P(n \to n+1)$ mapping nodes at depth $n+1$ to their parent at depth n. How would you describe natural transformations?

Example 28 (Presheaves for variable binding)**.** Let Inj be the category of finite sets with injective maps. Then the terms of the untyped lambda calculus can be seen as a presheaf $\Lambda : \mathsf{Inj} \to \mathsf{Set}$ where $\Lambda(X)$ is the set of lambda-terms with free variable in X. For a variable $x \notin X$, varibale binding $\lambda x.$ can be described as an operation

$$\lambda x. : \Lambda(X \cup \{x\}) \longrightarrow \Lambda(X)$$

In the previous examples, we have seen that as soon as we work with paremeterised sets (ie presheaves) mundane arrows such as simulations or algebraic operations become natural transformations. So we need a tool to work with natural transformations.

Theorem 29 (Yoneda Lemma)**.** *Let F be a presheaf on $\mathcal{C}$. There is a bijection between natural transformations $\mathcal{C}(,C) \to F$ and FC.*

Definition 17. . A functor $F : \mathcal{A} \to \mathcal{B}$ is

- full if $F_{AA} : \mathcal{A}(A, A) \to \mathcal{B}(FA, FA)$ is surjective
- faithful $F_{AA} : \mathcal{A}(A, A) \to \mathcal{B}(FA, FA)$ is injective
- fully faithful $F_{AA} : \mathcal{A}(A, A) \to \mathcal{B}(FA, FA)$ is an isomorphism

for all $A, A \in \mathcal{A}$.

Corollary 30. *The Yoneda embedding $\mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathcal{C}]$ is fully faithful.*

Exercise 31. *Use the Yoneda lemma to characterise the natural transformations between Set-functors such as $A \times X \to B \times X$ or $X^A \to X^B$, etc*

3 Limits and Adjoints

Synopsis. *Theme:* From posets to categoris to 2-categories, or from 2 to Set to Cat. *Definitions:* Limits, colimits, adjoints. *Concepts:* Category theory is generalised lattice theory. *Theorems:* Limits in functor categories are computed pointwise; a category with large limits and colimits is a preorder; ...
Synopsis.

Definition 18. Let $\mathcal{C}$ be a poset and $\mathcal{A} \subseteq \mathcal{C}$. Then $\bigwedge \mathcal{A}$, the **meet** or **greatest lower bound** of $\mathcal{A}$, is definec by the property that for all $c \in \mathcal{C}$ such that $c \leq \mathcal{A}$ we have $c \leq \bigwedge \mathcal{A}$.

To generalise this from posets to categories we need to make explicit the indexing of the elements of $\mathcal{C}$, which will be done by a *functor* $D : \mathcal{A} \to \mathcal{C}$ also called a **diagram**. As for posets, we will require the indices to form a set, or, rather, a $\mathcal{A}$ to be a *small category.* A short and elegant way to say what a cone $C \in \mathcal{C}$ over D is, is to consider both D and C as functors, and to say that a **cone** over a diagram D is an object $C \in \mathcal{C}$ and a natural transformation $\gamma : D \to C$.

Definition 19. Let $\mathcal{C}$ be a category, let $\mathcal{A}$ be a small categeory and $D : \mathcal{A} \to \mathcal{C}$. Then $\gamma : D \to \lim D$, the **limit** of D, is defined by the universal property that for all $C \in \mathcal{C}$ and all natural transformations $\alpha : C \to D$ there is a unique $h : C \to \lim D$ such that $\gamma \circ h = \alpha$.

Definition 20. Let $\mathcal{C}$ be a poset and $\mathcal{A} \subseteq \mathcal{C}$. Then $\bigvee \mathcal{A}$, the **join** or **least upper bound** of $\mathcal{A}$, is definec by the property that for all $c \in \mathcal{C}$ such that $\mathcal{A} \leq c$ we have $\bigvee \mathcal{A} \leq c$.

Definition 21. Let $\mathcal{C}$ be a category, and $D : \mathcal{A} \to \mathcal{C}$ a diagram. Then $\gamma : \mathrm{colim} D \to D$ is the **colimit** of D if for all $C \in \mathcal{C}$ and all natural transformations $\alpha : D \to C$ there is a unique $h : \mathrm{colim} D \to C$ such that $h \circ \gamma = \alpha$.

If $\mathcal{A}, \mathcal{C}$ are categories and $D : \mathcal{A} \to \mathcal{C}$ is a functor, then the **dual functor** $D^{\mathrm{op}} : \mathcal{A}^{\mathrm{op}} \to \mathcal{C}^{\mathrm{op}}$ acts the same as D on objects and arrows of $\mathcal{A}$. If $C, D : \mathcal{A} \to \mathcal{C}$ are functors and $\tau : C \to D$ is a natural transformation, then the **dual natural transformation** $\tau^{\mathrm{op}} : D^{\mathrm{op}} \to C^{\mathrm{op}}$ is the natural transformation given by $(\tau^{\mathrm{op}})_A = (\tau_A)^{\mathrm{op}}$.

Note that arrows and natural transformation change direction under duality, but functors do not.

Important: $\gamma : D \to C$ is a colimit in $\mathcal{C}$ iff $\gamma^{\mathrm{op}} : C \to D^{\mathrm{op}}$ is a limit in $\mathcal{C}^{\mathrm{op}}$.

Example 32. See Section 1: terminal, product, equaliser are limits, initial, coproduct, coequaliser are colimits.

Whereas the definition of coproduct is abstract, in concrete examples one can give explicit descriptions.

Exercise 33. *Let $D : \mathcal{I} \to \mathsf{Set}$ be a diagram. Show that* $\mathrm{colim} D$ *is the disjoint union of all Di, $i \in \mathcal{I}$, modulo the equivalence relation generated by $(i, x) \equiv (j, Df(x))$ where $i \in \mathcal{I}$, $x \in Di$, $f : i \to j$ is an arrow in $\mathcal{I}$.*

A category is **discrete** if the only arrows it has are identities. A category has **products** if it has limits of all diagrams $D : \mathcal{A} \to \mathcal{C}$ where $\mathcal{A}$ is discrete. A category is called **complete** if it has limits (ie limits for all diagrams) and **cocomplete** if it has all colimits.

Theorem 34. *A category is complete if it has products and equalisers.*

Theorem 35. *Limits and colimits in functor categories are computed pointwise.*

Theorem 36. *The Yoneda embedding $\mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathsf{Set}]$ preserves limits.*

Definition 22. Let $\mathcal{A}$ and $\mathcal{B}$ be two preorders and $L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two monotone functions. Then we say that L is the **left-adjoint** of R and R is the **right-adjoint** of L and write $L \dashv R$ iff

$$LA \le B \;\Leftrightarrow\; A \le RB$$

for all $A \in \mathcal{A}$ and all $B \in \mathcal{B}$.

Example 37. Let $\models$ be a relation on $Models \times Formulas$. Define $M : \mathcal{P}(Formulas) \to \mathcal{P}(Models)$ as

$$M(\mathcal{T}) = \{m \in Models \mid m \models \varphi \; \forall \varphi \in \mathcal{T}\}$$

and $T : \mathcal{P}(Models) \to \mathcal{P}(Formulas)$

$$T(\mathcal{M}) = \{\varphi \in Formulas \mid m \models \varphi \; \forall m \in \mathcal{M}\}$$

Then $MTM = M$ and $TMT = T$.

The importance of the next proposition is that it gives you different ways of showing adjointness.

Proposition 38. *$L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two monotone functions*

1. *$L \dashv R$ iff $A \le RLA$ and ($A \le RB \Rightarrow LA \le B$) for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
2. *$L \dashv R$ iff $LRB \le B$ and ($LA \le B \Rightarrow A \le RB$) for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
3. *$L \dashv R$ iff $A \le RLA$ and $LRB \le B$ for all $A \in \mathcal{A}$,$B \in \mathcal{B}$.*
4. *$L \dashv R$ iff $LRL = L$ and $RLR = R$.*
5. *$L \dashv R$ iff $LA = \bigwedge\{B \mid A \le RB\}$ for all $A \in \mathcal{A}$.*
6. *$L \dashv R$ iff $RB = \bigvee\{A \mid LA \le B\}$ for all $B \in \mathcal{B}$.*
7. *$L \dashv R$ only if L preserves all joins and R preserves all meets.*
8. *L has a right-adjoint iff $\mathcal{A}$ has and L preserves all joins of the form $\bigvee\{A \mid LA \le B\}$.*
9. *R has a left-adjoint iff $\mathcal{B}$ has and R preserves all meets of the form $\bigwedge\{B \mid A \le RB\}$.*

In particular, if $\mathcal{A}$ and $\mathcal{B}$ are complete, that is, have all joins and meets, then preserving all joins is equivalent to having a right adjoint and preserving all meets is equivalent to having a left-adjoint.

Corollary 39. *If L is a left-adjoint of $R : \mathcal{B} \to \mathcal{A}$, then L is uniquely determined up to iso. If $\mathcal{B}$ is a poset, then L is uniquely determined.*

Definition 23. Let $\mathcal{A}$ and $\mathcal{B}$ be two categories and $L : \mathcal{A} \to \mathcal{B}$ and $R : \mathcal{B} \to \mathcal{A}$ be two functors. Then we say that L is the left-adjoint of R and R is the right-adjoint of L and write $L \dashv R$ if there is an isomorphism

$$\mathcal{B}(LA, B) \ \cong\ \mathcal{A}(A, RB)$$

natural in $A \in \mathcal{A}$ and all $B \in \mathcal{B}$.

Theorem 40. *The data of an adjunction can be given equivalently in any of the following ways.*

1. *For all $A \in \mathcal{A}$ an arrow $\eta_A : A \to RLA$, called the unit, such that for all $f : A \to RB$ there is a unique $f^\sharp : LA \to B$ such that $Rf^\sharp \circ \eta = f$.*
2. *For all $B \in \mathcal{B}$ an arrow $\epsilon_B : LRB \to B$, called the counit, such that for all $g : LA \to B$ there is a unique $g^\flat : A \to RB$ such that $\epsilon_B \circ Lg^\flat = g$.*
3. *Two natural transformations $\eta : \mathrm{Id} \to RL$ (the unit) and $\epsilon : LR \to \mathrm{Id}$ such that $\epsilon L \circ L\eta = \mathrm{Id}$ and $R\epsilon \circ \eta R = \mathrm{Id}$.*

Theorem 41. *Adjoints are determined up to unique isomorphism.*

Theorem 42. *Adjoints compose.*

Theorem 43. *Left adjoints preserve colimits (and, by duality, right adjoints preserve limits).*

Theorem 44. *A category that is both complete and cocomplete is a preorder.*

Theorem 45. *If R preserves limits and is determined by a small subcategory, then R is a right adjoint.*

Theorem 46. *Let $\mathcal{A} \to \mathcal{B}$ be fully faithful and have a left-adjoint. Then $\mathcal{A}$ is complete/cocomplete if $\mathcal{B}$ is.*

Example 47. Let $\mathcal{A}$ be a category of algebras given by a signature Σ and equations E such as monoids, or lattices, or Boolean algebras, etc. For the purpose of this example, consider a signature Σ for which all operations have finite arity (that is, all operations take a finite number of arguments). One can formalise this by saying that Σ is a function $\mathbb{N} \to \mathsf{Set}$ giving for each arity $n \in \mathbb{N}$ the set of opertion symbols of this arity.

Whatever the signature and equations, there always is a functor $U : \mathcal{A} \to \mathsf{Set}$ mapping an algebra to its underlying set. And there always is a left-adjoint $F : \mathsf{Set} \to \mathcal{A}$ mapping a set X to the free algebra FX over X. Explicitely, FX is constructed by taking the set of all terms with variables from X and operation symbols from Σ and quotienting by the equations E.

What does the data of an adjunction mean in this example? The unit

$$\eta_X : X \to UFX$$

says that every variable is a term. The counit

$$\epsilon_A : FUA \to A$$

not only says that every algebra is the quotient of a free algebra (Exercise: show that ϵ_A is onto). It, moreover, describes the operations on A to which each term gives rise: If t is a term in variables X, that is $t \in UFX$, the interpretation t^A of t in A should be a function $UA^X \to UA$. So given a 'valuation of variables' $v : X \to UA$, we can indeed form

$$UFX \xrightarrow{UFv} UFUA \xrightarrow{U\epsilon} UA$$

evaluating a term $t \in UFX$ to an element $a \in A$.

4 Algebra, Coalgebras, Monads, ...

Synopsis. *Theme:* Representation results (all set-functors can be presented by equation on flat terms, all set-monads can be presented by operations and equations) *Definitions:* Algebra, coalgebra, monad, algebras for a monad, Kleisli-category, ... *Concepts:* Free algebras, induction, initial and final semantics, Kleisli categories as categories of relations, ... *Theorems:* representation results plus various theorems on monads

4.1 Algebras for a functor

In universal algebra, to specify a class of algebras one starts with a signature $\Sigma : \mathbb{N} \to \mathsf{Set}$, or, equivalently, with a polynomial functor $F_\Sigma(X) = \coprod_{n\in\mathbb{N}} \Sigma(n) \bullet X^n$, where $\Sigma(n) \bullet X^n$ denotes the $\Sigma(n)$-fold coproduct of the set X^n. To regard a signature Σ as a functor $F_\Sigma : \mathsf{Set} \to \mathsf{Set}$ allows us to say that an algebra is simply an arrow

$$F_\Sigma(X) \to X$$

in the category Set and that an algebra homomorphism $f : X \to X'$ is a commuting square

$$\begin{tikzcd} F_\Sigma(X) \arrow[r] \arrow[d, "F_\Sigma(f)"'] & X \arrow[d, "f"] \\ F_\Sigma(X') \arrow[r] & X' \end{tikzcd} \tag{1}$$

Example 48. Suppose we specify a signature consisting of two binary operations $*$ and $+$ and one nullary operation e. Thus, the corresponding $\Sigma : \mathbb{N} \to \mathsf{Set}$ is defined by putting $\Sigma(2) = \{*, +\}$, $\Sigma(0) = \{e\}$ and $\Sigma(n) = \emptyset$ otherwise. The appropriate polynomial endofunctor $F_\Sigma : \mathsf{Set} \to \mathsf{Set}$ then collapses to

$$F_\Sigma(X) = \{e\} \bullet X^0 + \{*, +\} \bullet X^2$$

since the signature Σ is empty for $n \notin \{0, 2\}$. A typical element of $F_\Sigma(X)$ therefore can be conceived as having one of the following three forms:

$$e, x * y, x + y$$

where we denoted by 0 the unique element of the set X^0 and (x, y) denotes an arbitrary element of X^2.

Thus, elements of $F_\Sigma(X)$ are precisely the *flat terms* in variables X for the signature Σ. A mapping $a : F_\Sigma(X) \to X$ that makes X into an algebra for F_Σ is then simply the *interpretation* of flat terms in X. Thus, the mapping a sends the above three typical elements to their "meanings" in X.

It is now straightforward to verify that the commutative square (1) encodes precisely the fact that the mapping $f : X \to X'$ respects the operations e, $*$ and $+$.

In category theory, the notion of algebra for a signature is generalised to the notion of an algebra for a functor. Looking at (1) above, we see that it makes sense to speak of algebras $FX \to X$ and their homomorphisms whenever we have a functor $\mathcal{C} \to \mathcal{C}$ on an arbitrary category $\mathcal{C}$.

Definition 24. Let $T : \mathcal{C} \to \mathcal{C}$ be a functor. The category $\mathsf{Alg}(T)$ has as objects T-algebras $TA \to A$ and arrows $(TA \xrightarrow{\alpha} A) \longrightarrow (TA' \xrightarrow{\alpha'} A')$ are arrows $f : A \to A'$ in $\mathcal{C}$ such that $f \circ \alpha = \alpha' \circ Tf$.

What is gained by this generalisation?

Answer 1. Maybe not too much, as long as one stays in sets, that is, as long as one takes $\mathcal{C} = \mathsf{Set}$. Let us call a functor $\mathsf{Set} \to \mathsf{Set}$ *finitary* if it is fully determined by its action on finite sets. Without going into the category theoretic definition of finitary, it suffices to say here that an arbitrary functor $F : \mathsf{Set} \to \mathsf{Set}$ is finitary iff there is a signature Σ such that F is a quotient of some F_Σ,

$$F_\Sigma \twoheadrightarrow F\,.$$

It follows that for any finitary $F : \mathsf{Set} \to \mathsf{Set}$, an F-algebra $FX \to X$ is nothing but an algebra

$$F_\Sigma X \twoheadrightarrow FX \longrightarrow X$$

for the signature Σ (and the *equations* defining the quotient $F_\Sigma \twoheadrightarrow F$). To summarize, the study of algebras for (finitary) functors $\mathsf{Set} \to \mathsf{Set}$ does not lead beyond the study of varieties in universal algebra, that is, algebras defined by operations and equations. In fact, algebras for a functor are algebras for operations and equations only involving flat terms. For a detailed account on algebras for a functor see the monograph by Adamek and Trnkova.

Answer 2. Quite a lot is gained when moving to other categories $\mathcal{C}$ than Set. Ever since the work of Scott and others on domain theory and program semantics, type constructors T have been viewed as functors and semantic domains as (particular) algebras $TX \to X$, see e.g. the handbook article by Abramsky and Jung on Domain Theory. Typically, the category $\mathcal{C}$ is a category of partial orders or metric spaces, possibly with some completeness requirements.

Another interesting choice for $\mathcal{C}$ is the category which is *dual* to Set. One then obtains the notion of a *coalgebra* for a functor $T : \mathsf{Set} \to \mathsf{Set}$ as a function

$$X \to TX.$$

As opposed to what we have seen in Answer 1 above, the fact that a finitary T is a quotient $F_\Sigma \twoheadrightarrow T$ of a polynomial functor F_Σ does not allow us to reduce the notion of a T-coalgebra to the notion of a coalgebra $X \to F_\Sigma X$ for a signature Σ. Going beyond polynomial functors will lead to new and interesting examples, as we are going to see next.

On the other hand, we still have an induction principle for all T-algebras: Let $\iota : TI \to I$ be an initial object in $\mathsf{Alg}(T)$ (assuming that such an object exists). Then initiality means that for all $\alpha : TA \to A$ there is a unique $f : I \to A$ such that

$$\begin{tikzcd} TI \arrow[r, "\iota"] \arrow[d, "Tf"'] & I \arrow[d, "f"] \\ TI \arrow[r, "\alpha"] & A \end{tikzcd} \tag{2}$$

This is **induction**: To define an arrow on all 'terms' from the recursive data-type I to the data-type A, it is enough to specify one-step (or flat) T-operations α on A.

Exercise 49. *Show how the diagram* (2) *specialises to account for induction on the natural numbers.*

4.2 Coalgebras for a functor

Examples of coalgebras below show that coalgebras for polynomial functors F_Σ are of interest, but also that new phenomena such as *bisimulation* come into focus when going beyond polynomial functors.

Example 50. *Coalgebras for polynomial functors describe infinite trees.* For example, an element x in a coalgebra $\xi : X \to X + X$ can be seen as an infinite stream of left/right decisions: in state x, taking a transition by applying ξ yields a successor state $\xi(x)$ in either the left or the right component of $TX = X + X$.

Similarly, a state in a coalgebra $X \to A + B \times X \times X$ represents a possibly infinite tree with leaves labelled by elements of A and non-leaf nodes labelled by elements of B. Here, the polynomial functor is $TX = A + B \times X \times X$.

Example 51. *Coalgebras for the powerset functor are transition systems.* Here the functor T assigns the powerset PX to every set X. Thus a coalgebra $\xi : X \to TX$ can be seen as describing the behaviour of a nondeterministic transition system: the "next state" $\xi(x)$ of a state x is, in fact, the subset $\xi(x) \subseteq X$ of all possible states into which x can evolve.

Example 52. *Coalgebras for the distribution functor are probabilistic transition systems.* Denote by DX the set of all functions $p : X \to [0;1]$ that have a finite support (i.e., such that $p(x) = 0$ for all but finitely many $x \in X$) and that satisfy $\sum_{x\in X} p(x) = 1$. Then a coalgebra $\xi : X \to DX$ describes a transition system with $\xi(x) : X \to [0;1]$ giving the probability $\xi(x)(x')$ that x evolves to x'.

In universal coalgebra, a notion coined by Rutten in the eponymous article, therefore, it is important to develop the theory of T-coalgebras parametric in a functor T, much in the same way as universal algebra is done parametrically in a signature Σ. Some questions that arise in that context are:

- For which functors $T : \mathsf{Set} \to \mathsf{Set}$ is there a final coalgebra?
- Can the behavioural equivalence given by the final coalgebra be characterised in terms of bisimulations?
- In universal algebra every signature Σ gives rise to an equational logic. Can we associate a coalgebraic logic to every functor $T : \mathsf{Set} \to \mathsf{Set}$?
- How much of this can be done axiomatically, replacing Set by general categories $\mathcal{C}$?

Of course, there are many further topics in coalgebra, for example, the use of coalgebra to solve recursive equations or to describe and derive congruence formats of process algebras or to extend and apply coalgebraic logic to description logics and knowledge representation.

4.3 Algebras for a monad

We need notation for 'whiskering' a natural transformation with a functor. Let $\tau : F \to G : \mathcal{A} \to \mathcal{B}$ and let $L : \mathcal{A}' \to \mathcal{A}$ and $R : \mathcal{B} \to \mathcal{B}'$. Then we have natural transformations $(\tau L)_{A'} = \tau_{LA'}$ and $(R\tau)_A = R(\tau_A)$. [5] Of course, we now can also write $R\tau L$. This notation also suggests to write τA instead of τ_A.

Definition 25. A **monad** (M, η, μ) on a category $\mathcal{C}$ is a functor $M : \mathcal{C} \to \mathcal{C}$ and two natural transformations $\eta : \mathrm{Id} \to M$ and $\mu : MM \to M$ such that $\mu \circ M\eta = \mu \circ \eta M = \mathrm{Id}$ and $\mu \circ \mu M = \mu \circ M\mu$.

Definition 26. An **algebra for a monad** (M, η, μ) is an algebra $\alpha : MA \to A$ for the functor M satisfying $\alpha \circ \eta A = \mathrm{id}_A$ and $\alpha \circ \mu A = \alpha \circ M\alpha$. We denote the category of algebras for a monad, also known as the category of **Eilenberg-Moore algebras**, by $\mathsf{MAlg}(M, \eta, \mu)$ or simply by $\mathsf{MAlg}(M)$.

The next two examples show that monads are ubiquitous.

Example 53. Given a signature Σ and equations E, the category of algebras for $\langle \Sigma, E \rangle$ is a category of algebras for a monad.

Example 54. Given and adjunction $F \dashv U : \mathcal{A} \to \mathcal{C}$, then UF is a monad.

In the following we are going to look at two representation theorem, that tell us that all monads arise in this way.

Theorem 55. *Let $U : \mathsf{MAlg} \to \mathcal{C}$ be the forgetful functor. Then U has a left adjoint and $UF = M$.*

Is this the only adjunction that generates M? Not at all, but it is the terminal one, in a sense. The initial one is the following.

Definition 27. Given a category $\mathcal{C}$, let M be a map (which is not required to be functorial at this stage) from objects of $\mathcal{C}$ to objects of $\mathcal{C}$ (think of $M = \mathcal{P}$ and $\mathcal{C} = \mathsf{Set}$). Let $\eta_X : X \to MX$ be a collection of arrows in $\mathcal{C}$ and let $(-)^\sharp_{Y,Z}$ be a collection of functions $\mathcal{C}(Y, MZ) \to \mathcal{C}(MY, MZ)$. Then $(M, \eta, (-)^\sharp)$ is called a **Kleisli triple** if

$$\eta^\sharp = \mathrm{id}$$
$$f^\sharp \circ \eta = f$$
$$(g^\sharp \circ f)^\sharp = g^\sharp \circ f^\sharp$$

[5] Exercise: Show that τL and $R\tau$ are natural transformations, given that τ is natural.

Example 56. The assignment $X \mapsto \mathcal{P}X$, together with the collection $\eta_X(x) = \{x\}$ of singleton maps and with

$$g^\sharp(b) = \bigcup\{g(y) \mid y \in b\}$$

is easily seen to form a Kleisli triple.

In fact, the above axiomatisation allows us to extend the assignment $X \mapsto MX$ uniquely to a functor $M : \mathcal{C} \to \mathcal{C}$ such that the collection η_X becomes a natural transformation from Id to M, and, when one defines $\mu_X : MMX \to MX$ by putting $\mu_X = (\mathrm{id}_{MX})^\sharp$, then μ is a natural transformation from MM to M. The above axioms guarantee that (M, η, μ) is a monad. Conversely, every monad (M, η, μ) yields a Kleisli triple by defining $f^\sharp = \mu_Z \circ Mf : MY \to MZ$ for $f : Y \to MZ$. (All details about this are eg discribed in the monograph by Manes on algebraic theories but also many other places such as in MacLane.)

Kleisli triples gives rise to categories, the Kleisli categories, which tend to resemble categories of relations.

Definition 28. Given a Kleisli triple $(M, \eta, (-)^\sharp)$ on a category $\mathcal{C}$, the **Kleisli category** $Kl(M)$ has the same objects as $\mathcal{C}$ and arrows $X \rightharpoonup Y$ in $Kl(M)$ are arrows $X \to MY$ in $\mathcal{C}$. The identity on X in $Kl(M)$ is given by η_X and the composition $g \cdot f$ in $Kl(M)$ is given by the composition $g^\sharp \circ f$ in $\mathcal{C}$.

As to be expected for a 'category of relations' there is an identity-on-objects functor

$$(-)_\star : \mathcal{C} \to Kl(M)$$

taking an arrow $f : X \to Y$ to a 'map' $\eta_Y \circ f : X \rightharpoonup Y$. On the other hand, there need to be no analogue of the converse of a relation nor of the order between relations.

Theorem 57. *Let $F \dashv U : \mathcal{A} \to \mathcal{C}$. Then there are 'comparison functors' $Kl(UF) \xrightarrow{K} \mathcal{A} \xrightarrow{L} \mathsf{MAlg}(UF)$ commuting with the respective forgetful functors.*

Theorem 58. *For any monad M on set, one can find a class of operations Σ and a class of equations E, such that $\mathsf{MAlg}(M) \cong \mathsf{Alg}(\Sigma, E)$.*

5 Two remarks

We will do some 'mental gymnastics' motivated by Pawel's lectures and discuss the small vs large issue, introducing Kan extensions.

5.1 Monoids

We started by saying that category theory is a theory of composition. Let us go back to basics. The basic theory of composition is given by the equational theory of monoids

$(A, e, \cdot)$ where e is the identity and $\cdot$ is associative. The paragdigmatic monoid, the free monoid over a set A, is the monoid A^* of finite words with the empty word and concatenation as operations. But monoids alone cannot serve as a theory of composition, because the set A of things we want to compose may have some more structure than just being mere letters or words. We may want to compose functions, or relations, or processes, or computer programs, or some kind of systems, etc

So going back to the definition of a monoid as consisting of

$$1 \to A$$
$$A \times A \to A$$

we see that its very definition presupposes some structure such as 1 and $\times$ and, depending on the things we want to compose, this structure will take different forms.

So really we should consider monoids in a category. This gives us some more generality, so for example, we can now say that a topological monoid is a monoid in the category of topological spaces.

But this is not general enough. It is not always the case that composition

$$A \times A \to A$$

should be defined on a cartesian or categorical product $A \times A$. In fact, what is required to make the idea of a monoid work, is only that of a 'monoidal product'

$$A \otimes A \to A.$$

This can be formalised as a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ satisfying itself the properties of a monoid and giving us the notion of a (strict) monoidal category.

Example 59. The category of endo-functors $\mathcal{C} \to \mathcal{C}$ with composition is a monoidal category.

We motivated monoidal categories by saying that they provide the minimal environment in which one can talk meaningfully about monoids. So what a monoid in the category of monoids?

Example 60. A monoid in the category of endofunctors is a monad.

We started with monoids as a theory of composition, introduced categories and monoidal categories to say what a monoid is and then recovered monads as particular monoids. Now let us go one step further and recover categories as monads.

Before we can do this, we need to think of monads in a category (or rather 2-category): (M, η, μ) is given by an arrow $M : C \to C$ and 2-cells $\mathrm{Id} \to M$ and $\mu : MM \to M$.

Example 61. Let Span be the category which has sets as objects and spans $X \leftarrow S \to Y$ as arrows $X \to Y$. The 2-cells $S \to S'$ are 1-cells that commute with the legs of the spans. What is a monad in Span?

As we have learned from Pawel this is more than only 'mental gymnastics' but this is genuinly useful ... all what Pawel showed us is ultimately based on this.

5.2 Small vs large, generation and Kan extensions

We have seen the small vs large issues in many places.

- In the definition of a category the homs are small but the collection of objects may be large (unless we have a small category).
- In universal properties we quantify over a large set of objects. And this is where the power of universal properties resides.
- Completeness and cocompleteness of categories was defined wrt to small limits and colimits ... since a category with all large limits and colimits is a preorder.
- A functor that is determined on a small subcategory and preserves limits is a right adjoint.
- If functor $T : \mathsf{Set} \to \mathsf{Set}$ is determined on a small subcategory then
  - T is a quotient of a signature functor
  - initial and free T-algebras exist
  - final and cofree T-algebras exists
- For all functors $T : \mathsf{Set} \to \mathsf{Set}$ there is a final T-coalgebra, but its carrier is not a (small) set, so does not exists in Set, but it is a proper class (large set). Thus the final coalgebra is then really the final coalgebra not for T but for an extension of T to an enlargement of Set.

One way of tackling the problem is to think about generators in a category. For example, the category Set is generated by 1: Every set is the quotient of a coproduct of 1.

Example 62. 1. In the universal definition of a product one does not need to quantify over all objects in Set, in fact, 1 is enough.

2. Every colimit preserving functor $\mathsf{Set} \to \mathsf{Set}$ is a left-adjoint.

3. Every left-adjoint $\mathsf{Set} \to \mathsf{Set}$ is of the form $X \mapsto A \times X$ form some $A \in \mathsf{Set}$.

4. The inclusion $\{1\} \to \mathsf{Set}$ is dense, that is, there is a bijection between colimit preserving functor $\mathsf{Set} \to \mathcal{C}$ and functors $\{1\} \to \mathsf{Set}$.

Definition 29. Given $F : \mathcal{A} \to \mathcal{B}$ and $K : \mathcal{A} \to \mathcal{C}$ the left Kan-extension of F along K is a functor $\mathrm{Lan}_K F : \mathcal{C} \to \mathcal{B}$ together with a natural transformation $\gamma : F \to (\mathrm{Lan}_K F)K$ such that for all $H : \mathcal{C} \to \mathcal{B}$ and all $\alpha : F \to HK$ there is a unique $\beta : \mathrm{Lan}_K F \to H$ such that $\alpha = \beta \circ \gamma$.

This is often remembered as

$$\frac{F \to HK}{\overline{\mathrm{Lan}_K F \to H}}$$

Example 63.

1. What is the left Kan-extension of $F : \{*\} \to \mathsf{Set}$, $F(*) = A$ along $\{*\} \to \mathsf{Set}$?

2. Let Fin be the category of finite sets. What is the left Kan extension of $X \mapsto A \times X$, $X \mapsto \mathsf{List}(X)$, ... along $\mathsf{Fin} \to \mathsf{Set}$?

3. Let Fin be the category of finite sets. What is the left Kan extension of $X \mapsto \mathcal{P}(X)$, ... along $\mathsf{Fin} \to \mathsf{Set}$?

Definition 30. A functor $F : \mathsf{Set} \to \mathsf{Set}$ is called **finitary** iff it is the left Kan extension of its restriction along $\mathsf{Fin} \to \mathsf{Set}$.

Example 64.

1. The forgetful functor from algebras to sets is finitary if all operations take only a finite number of arguments.

2. If $T : \mathsf{Set} \to \mathsf{Set}$ is a finitary functor, then the initial T-algebra can by built as the colimit
$$0 \to T0 \to T(T0) \to \ldots \to T^n(0) \to \ldots \mathrm{colim}_{n \in \mathbb{N}} T^n(0)$$

3. If $T : \mathsf{Set} \to \mathsf{Set}$ is a finitary functor, then the final T-coalgebra can by built as the limit
$$1 \leftarrow \ldots T^n(1) \ldots \leftarrow \lim_{n \in \mathbb{N}} T^n(1) \leftarrow \ldots T^m(\lim_{n \in \mathbb{N}} T^n(1)) \ldots \leftarrow \lim_{m \in \mathbb{N}} T^m(\lim_{n \in \mathbb{N}} T^n(1))$$