# Chapter 1 <br> Introduction to Categories and Categorical Logic 

Samson Abramsky and Nikos Tzevelekos

## Preface

The aim of these notes is to provide a succinct, accessible introduction to some of the basic ideas of category theory and categorical logic. The notes are based on a lecture course given at Oxford over the past few years. They contain numerous exercises, and hopefully will prove useful for self-study by those seeking a first introduction to the subject, with fairly minimal prerequisites. The coverage is by no means comprehensive, but should provide a good basis for further study; a guide to further reading is included.

The main prerequisite is a basic familiarity with the elements of discrete mathematics: sets, relations and functions. An Appendix contains a summary of what we will need, and it may be useful to review this first. In addition, some prior exposure to abstract algebra - vector spaces and linear maps, or groups and group homomorphisms-would be helpful.

## Contents

1 Introduction to Categories and Categorical Logic ..... 1)
Samson Abramsky and Nikos Tzevelekos
1.1 Introduction ..... 7
1.1.1 From Elements To Arrows ..... 8
1.1.2 Categories Defined ..... 9
1.1.3 Diagrams in Categories ..... 10
1.1.4 Examples ..... 12
1.1.5 First Notions ..... 14
1.1.6 Exercises ..... 17
1.2 Some Basic Constructions ..... 17
1.2.1 Initial and Terminal Objects ..... 17
1.2.2 Products and Coproducts ..... 19
1.2.3 Pullbacks and Equalisers ..... 24
1.2.4 Limits and Colimits ..... 27
1.2.5 Exercises ..... 27
1.3 Functors ..... 27
1.3.1 Basics ..... 28
1.3.2 Further Examples ..... 30
1.3.3 Contravariance ..... 32
1.3.4 Properties of Functors ..... 33
1.3.5 Exercises ..... 35
1.4 Natural Transformations ..... 35
1.4.1 Basics ..... 35
1.4.2 Further Examples ..... 38
1.4.3 Functor Categories ..... 40
1.4.4 Exercises ..... 41
1.5 Universality and Adjoints ..... 42
1.5.1 Adjunctions for Posets ..... 43
1.5.2 Universal Arrows and Adjoints ..... 46
1.5.3 Limits and Colimits ..... 52
1.5.4 Exponentials ..... 53
1.5.5 Exercises ..... 55
1.6 The Curry-Howard Correspondence ..... 56
1.6.1 Logic ..... 57
1.6.2 Computation ..... 58
1.6.3 Simply-Typed $\lambda$-calculus ..... 62
1.6.4 Categories ..... 66
1.6.5 Categorical Semantics of Simply-Typed $\lambda$-calculus ..... 67
1.6.6 Completeness? ..... 70
1.6.7 Exercises ..... 73
1.7 Linearity ..... 73
1.7.1 Gentzen Sequent Calculus ..... 74
1.7.2 Linear Logic ..... 75
1.7.3 Linear Logic in Monoidal Categories ..... 78
1.7.4 Beyond the Multiplicatives ..... 82
1.7.5 Exercises ..... 84
1.8 Monads and Comonads ..... 85
1.8.1 Basics ..... 86
1.8.2 (Co)Monads of an Adjunction ..... 88
1.8.3 The Kleisli Construction ..... 89
1.8.4 Modelling of Linear Exponentials ..... 92
1.8.5 Exercises ..... 97
1.9 Review of Sets, Functions and Relations ..... 98
1.10 Guide to Further Reading ..... 100
References ..... 101

## List of Tables

1.1 Examples of Limits and Colimits ..... 27
1.2 The Curry-Howard Correspondence ..... 56
1.3 Natural Deduction System for $\wedge, \supset$. ..... 57
1.4 Examples of $\lambda$-terms. ..... 59
1.5 Simply-Typed $\lambda$-calculus for $\times, \rightarrow$ ..... 63
1.6 Correspondence between Logic and Computation. ..... 65
1.7 Correspondence between Logic and Categories ..... 66
1.8 Structural Rules for Logic. ..... 74
1.9 Gentzen Sequent Calculus for $\wedge, \supset$ ..... 75
1.10 Rules for $\otimes,-$-Linear Logic. ..... 76
1.11 Linear $\lambda$-calculus for $\otimes$, ..... 77
1.12 Categorical Translation of $\otimes,-$-Linear Logic. ..... 82

### 1.1 Introduction

Why study categories - what are they good for? We can offer a range of answers for readers coming from different backgrounds:

- For mathematicians: category theory organises your previous mathematical experience in a new and powerful way, revealing new connections and structure, and allows you to "think bigger thoughts".
- For computer scientists: category theory gives a precise handle on important notions such as compositionality, abstraction, representationindependence, genericity and more. Otherwise put, it provides the fundamental mathematical structures underpinning many key programming concepts.
- For logicians: category theory gives a syntax-independent view of the fundamental structures of logic, and opens up new kinds of models and interpretations.
- For philosophers: category theory opens up a fresh approach to structuralist foundations of mathematics and science; and an alternative to the traditional focus on set theory.
- For physicists: category theory offers new ways of formulating physical theories in a structural form. There have inter alia been some striking recent applications to quantum information and computation.


### 1.1.1 From Elements To Arrows

Category theory can be seen as a "generalised theory of functions", where the focus is shifted from the pointwise, set-theoretic view of functions, to an abstract view of functions as arrows.

Let us briefly recall the arrow notation for functions between sets 1 A function $f$ with domain $X$ and codomain $Y$ is denoted by: $f: X \rightarrow Y$.

$$
\text { Diagrammatic notation: } X \xrightarrow{f} Y \text {. }
$$

The fundamental operation on functions is composition: if $f: X \rightarrow Y$ and $g: Y \rightarrow Z$, then we can define $g \circ f: X \rightarrow Z$ by $g \circ f(x):=g(f(x))^{2}$ Note that, in order for the composition to be defined, the codomain of $f$ must be the same as the domain of $g$.

$$
\text { Diagrammatic notation: } X \xrightarrow{f} Y \xrightarrow{g} Z \text {. }
$$

Moreover, for each set $X$ there is an identity function on $X$, which is denoted by:

$$
\operatorname{id}_{X}: X \longrightarrow X \quad \operatorname{id}_{X}(x):=x
$$

These operations are governed by the associativity law and the unit laws. For $f: X \rightarrow Y, g: Y \rightarrow Z, h: Z \rightarrow W:$

$$
(h \circ g) \circ f=h \circ(g \circ f), \quad f \circ \mathrm{id}_{X}=f=\operatorname{id}_{Y} \circ f .
$$

Notice that these equations are formulated purely in terms of the algebraic operations on functions, without any reference to the elements of the sets $X$, $Y, Z, W$. We will refer to any concept pertaining to functions which can be defined purely in terms of composition and identities as arrow-theoretic. We will now take a first step towards learning to "think with arrows" by seeing how we can replace some familiar definitions couched in terms of elements by arrow-theoretic equivalents; this will lead us towards the notion of category.

We say that a function $f: X \longrightarrow Y$ is:

[^0]\[

$$
\begin{aligned}
& \text { injective if } \forall x, x^{\prime} \in X . f(x)=f\left(x^{\prime}\right) \Longrightarrow x=x^{\prime}, \\
& \text { surjective if } \forall y \in Y . \exists x \in X . f(x)=y, \\
& \text { monic if } \forall g, h . f \circ g=f \circ h \Longrightarrow g=h, \\
& \text { epic } \quad \text { if } \forall g, h . g \circ f=h \circ f \Longrightarrow g=h .
\end{aligned}
$$
\]

Note that injectivity and surjectivity are formulated in terms of elements, while epic and monic are arrow-theoretic.

Proposition 1 Let $f: X \rightarrow Y$. Then,

1. $f$ is injective iff $f$ is monic.
2. $f$ is surjective iff $f$ is epic.

Proof: We show 1. Suppose $f: X \rightarrow Y$ is injective, and that $f \circ g=f \circ h$, where $g, h: Z \rightarrow X$. Then, for all $z \in Z$ :

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

Since $f$ is injective, this implies $g(z)=h(z)$. Hence we have shown that

$$
\forall z \in Z . g(z)=h(z),
$$

and so we can conclude that $g=h$. So $f$ injective implies $f$ monic.
For the converse, fix a one-element set $\mathbf{1}=\{\bullet\}$. Note that elements $x \in X$ are in $1-1$ correspondence with functions $\bar{x}: \mathbf{1} \rightarrow X$, where $\bar{x}(\bullet):=x$. Moreover, if $f(x)=y$ then $\bar{y}=f \circ \bar{x}$. Writing injectivity in these terms, it amounts to the following.

$$
\forall x, x^{\prime} \in X . f \circ \bar{x}=f \circ \bar{x}^{\prime} \Longrightarrow \bar{x}=\bar{x}^{\prime}
$$

Thus we see that being injective is a special case of being monic.
Exercise 2 Show that $f: X \rightarrow Y$ is surjective iff it is epic.

### 1.1.2 Categories Defined

Definition 3 A category $\mathcal{C}$ consists of:

- A collection $\mathrm{Ob}(\mathcal{C})$ of objects. Objects are denoted by $A, B, C$, etc.
- A collection $\operatorname{Ar}(\mathcal{C})$ of arrows (or morphisms). Arrows are denoted by $f, g, h$, etc.
- Mappings dom, cod : $\operatorname{Ar}(\mathcal{C}) \rightarrow \mathrm{Ob}(\mathcal{C})$, which assign to each arrow $f$ its domain dom $(f)$ and its codomain $\operatorname{cod}(f)$. An arrow $f$ with domain $A$ and codomain $B$ is written $f: A \rightarrow B$. For each pair of objects $A, B$, we define the set

$$
\mathcal{C}(A, B):=\{f \in \operatorname{Ar}(\mathcal{C}) \mid f: A \rightarrow B\} .
$$

We refer to $\mathcal{C}(A, B)$ as a hom-set. Note that distinct hom-sets are disjoint.

- For any triple of objects $A, B, C$, a composition map

$$
c_{A, B, C}: \mathcal{C}(A, B) \times \mathcal{C}(B, C) \longrightarrow \mathcal{C}(A, C) .
$$

$c_{A, B, C}(f, g)$ is written $g \circ f$ (or sometimes $\left.f ; g\right)$. Diagrammatically:

$$
A \xrightarrow{f} B \xrightarrow{g} C
$$


The above must satisfy the following axioms.

$$
h \circ(g \circ f)=(h \circ g) \circ f, \quad f \circ \mathrm{id}_{A}=f=\operatorname{id}_{B} \circ f .
$$

whenever the domains and codomains of the arrows match appropriately so that the compositions are well-defined.

### 1.1.3 Diagrams in Categories

Diagrammatic reasoning is an important tool in category theory. The basic cases are commuting triangles and squares. To say that the following triangle commutes

is exactly equivalent to asserting the equation $g \circ f=h$. Similarly, to say that the following square commutes

means exactly that $g \circ f=k \circ h$. For example, the equations

$$
h \circ(g \circ f)=(h \circ g) \circ f, \quad f \circ \mathrm{id}_{A}=f=\operatorname{id}_{B} \circ f,
$$

can be expressed by saying that the following diagrams commute.


As these examples illustrate, most of the diagrams we shall use will be "pasted together" from triangles and squares: the commutation of the diagram as a whole will then reduce to the commutation of the constituent triangles and squares.

We turn to the general case. The formal definition is slightly cumbersome; we give it anyway for reference.

Definition 4 We define a graph to be a collection of vertices and directed edges, where each edge $e: v \rightarrow w$ has a specified source vertex $v$ and target vertex $w$. Thus graphs are like categories without composition and identities $3^{3}$ A diagram in a category $\mathcal{C}$ is a graph whose vertices are labelled with objects of $\mathcal{C}$ and whose edges are labelled with arrows of $\mathcal{C}$, such that, if $e: v \rightarrow w$ is labelled with $f: A \rightarrow B$, then we must have $v$ labelled by $A$ and $w$ labelled by $B$. We say that such a diagram commutes if any two paths in it with common source and target, and at least one of which has length greater than 1 , are equal. That is, given paths

$$
A \xrightarrow{f_{1}} C_{1} \xrightarrow{f_{2}} \cdots C_{n-1} \xrightarrow{f_{n}} B \quad \text { and } \quad A \xrightarrow{g_{1}} D_{1} \xrightarrow{g_{2}} \cdots D_{m-1} \xrightarrow{g_{m}} B,
$$

if $\max (n, m)>1$ then

$$
f_{n} \circ \cdots \circ f_{1}=g_{m} \circ \cdots \circ g_{1}
$$

To illustrate this definition, to say that the following diagram commutes

$$
E \xrightarrow{e} A \xlongequal[g]{f} B
$$

amounts to the assertion that $f \circ e=g \circ e$; it does not imply that $f=g$.

[^1]
### 1.1.4 Examples

Before we proceed to our first examples of categories, we shall present some background material on partial orders, monoids and topologies, which will provide running examples throughout these notes.

## Partial orders

A partial order is a structure $(P, \leq)$ where $P$ is a set and $\leq$ is a binary relation on $P$ satisfying:

- $x \leq x$
(Reflexivity)
- $x \leq y \wedge y \leq x \Rightarrow x=y$
- $x \leq y \wedge y \leq z \Rightarrow x \leq z$

For example, $(\mathbb{R}, \leq)$ and $(\mathcal{P}(X), \subseteq)$ are partial orders, and so are strings with the sub-string relation.

If $P, Q$ are partial orders, a map $h: P \rightarrow Q$ is a partial order homomorphism (or monotone function) if:

$$
\forall x, y \in P . x \leq y \Longrightarrow h(x) \leq h(y)
$$

Note that homomorphisms are closed under composition, and that identity maps are homomorphisms.

Monoids

A monoid is a structure $(M, \cdot, 1)$ where $M$ is a set,

$$
\_\cdot \_: M \times M \longrightarrow M
$$

is a binary operation, and $1 \in M$, satisfying the following axioms.

$$
(x \cdot y) \cdot z=x \cdot(y \cdot z), \quad 1 \cdot x=x=x \cdot 1
$$

For example, $(\mathbb{N},+, 0)$ is a monoid, and so are strings with string-concatenation. Moreover, groups are special kinds of monoids.

If $M, N$ are monoids, a map $h: M \rightarrow N$ is a monoid homomorphism if

$$
\forall m_{1}, m_{2} \in M . h\left(m_{1} \cdot m_{2}\right)=h\left(m_{1}\right) \cdot h\left(m_{2}\right), \quad h(1)=1
$$

Exercise 5 Suppose that $G$ and $H$ are groups (and hence monoids), and that $h: G \rightarrow H$ is a monoid homomorphism. Prove that $h$ is a group homomorphism.

Topological spaces
A topological space is a pair $\left(X, T_{X}\right)$ where $X$ is a set, and $T_{X}$ is a family of subsets of $X$ such that

- $\varnothing, X \in T_{X}$,
- if $U, V \in T_{X}$ then $U \cap V \in T_{X}$,
- if $\left\{U_{i}\right\}_{i \in I}$ is any family in $T_{X}$, then $\bigcup_{i \in I} U_{i} \in T_{X}$.

A continuous map $f:\left(X, T_{X}\right) \rightarrow\left(Y, T_{Y}\right)$ is a function $f: X \rightarrow Y$ such that, for all $U \in T_{Y}, f^{-1}(U) \in T_{X}$.

Let us now see some first examples of categories.

- Any kind of mathematical structure, together with structure preserving functions, forms a category. E.g.
- Set (sets and functions)
- Mon (monoids and monoid homomorphisms)
- Grp (groups and group homomorphisms)
- Vect $_{k}$ (vector spaces over a field $k$, and linear maps)
- Pos (partially ordered sets and monotone functions)
- Top (topological spaces and continuous functions)
- Rel: objects are sets, arrows $R: X \rightarrow Y$ are relations $R \subseteq X \times Y$. Relational composition:

$$
R ; S(x, z) \Longleftrightarrow \exists y \cdot R(x, y) \wedge S(y, z)
$$

- Let $k$ be a field (for example, the real or complex numbers). Consider the following category $\mathrm{Mat}_{k}$. The objects are natural numbers. A morphism $M: \mathbf{n} \rightarrow \mathbf{m}$ is an $\mathbf{n} \times \mathbf{m}$ matrix with entries in $k$. Composition is matrix multiplication, and the identity on $\mathbf{n}$ is the $\mathbf{n} \times \mathbf{n}$ diagonal matrix.
$\diamond$ Monoids are one-object categories. Arrows correspond to the elements of the monoid, with the monoid operation being arrow-composition and the monoid unit being the identity arrow.
$\diamond$ A category in which for each pair of objects $A, B$ there is at most one morphism from $A$ to $B$ is the same thing as a preorder, i.e. a reflexive and transitive relation.

Note that our first class of examples illustrate the idea of categories as mathematical contexts; settings in which various mathematical theories can be developed. Thus for example, Top is the context for general topology, Grp is the context for group theory, etc.

On the other hand, the last two examples illustrate that many important mathematical structures themselves appear as categories of particular kinds. The fact that two such different kinds of structures as monoids and posets should appear as extremal versions of categories is also rather striking.

This ability to capture mathematics both "in the large" and "in the small" is a first indication of the flexibility and power of categories.

Exercise 6 Check that Mon, Vect $_{k}$, Pos and Top are indeed categories.
Exercise 7 Check carefully that monoids correspond exactly to one-object categories. Make sure you understand the difference between such a category and Mon. (For example: how many objects does Mon have?)

Exercise 8 Check carefully that preorders correspond exactly to categories in which each homset has at most one element. Make sure you understand the difference between such a category and Pos. (For example: how big can homsets in Pos be?)

### 1.1.5 First Notions

Many important mathematical notions can be expressed at the general level of categories.

Definition 9 Let $\mathcal{C}$ be a category. A morphism $f: X \rightarrow Y$ in $\mathcal{C}$ is:

- monic (or a monomorphism) if $f \circ g=f \circ h \Longrightarrow g=h$,
- epic (or an epimorphism) if $g \circ f=h \circ f \Longrightarrow g=h$.

An isomorphism in $\mathcal{C}$ is an arrow $i: A \rightarrow B$ such that there exists an arrow $j: B \rightarrow A$ - the inverse of $i$ - satisfying

$$
j \circ i=\mathrm{id}_{A}, \quad i \circ j=\mathrm{id}_{B} .
$$

We denote isomorphisms by $i: A \xrightarrow{\cong} B$, and write $i^{-1}$ for the inverse of $i$. We say that $A$ and $B$ are isomorphic, $A \cong B$, if there exists some $i: A \xlongequal{\cong} B$.

Exercise 10 Show that the inverse, if it exists, is unique.
Exercise 11 Show that $\cong$ is an equivalence relation on the objects of a category.

As we saw previously, in Set monics are injections and epics are surjections. On the other hand, isomorphisms in Set correspond exactly to bijections, in Grp to group isomorphisms, in Top to homeomorphisms, in Pos to order isomorphisms, etc.

Exercise 12 Verify these claims.

Thus we have at one stroke captured the key notion of isomorphism in a form which applies to all mathematical contexts. This is a first taste of the level of generality which category theory naturally affords.

We have already identified monoids as one-object categories. We can now identify groups as exactly those one-object categories in which every arrow is an isomorphism. This also leads to a natural generalisation, of considerable importance in current mathematics: a groupoid is a category in which every morphism is an isomorphism.

Opposite Categories and Duality
The directionality of arrows within a category $\mathcal{C}$ can be reversed without breaking the conditions of being a category; this yields the notion of opposite category.

Definition 13 Given a category $\mathcal{C}$, the opposite category $\mathcal{C}^{\text {op }}$ is given by taking the same objects as $\mathcal{C}$, and

$$
\mathcal{C}^{\mathrm{op}}(A, B):=\mathcal{C}(B, A)
$$

Composition and identities are inherited from $\mathcal{C}$.
Note that if we have

$$
A \xrightarrow{f} B \xrightarrow{g} C
$$

in $\mathcal{C}^{\text {op }}$, this means

$$
A \stackrel{f}{\leftrightarrows} B \stackrel{g}{\leftrightarrows} C
$$

in $\mathcal{C}$, so composition $g \circ f$ in $\mathcal{C}^{\circ \mathrm{p}}$ is defined as $f \circ g$ in $\mathcal{C}$ !
Consideration of opposite categories leads to a principle of duality: a statement $S$ is true about $\mathcal{C}$ if and only if its dual (i.e. the one obtained from $S$ by reversing all the arrows) is true about $\mathcal{C}^{\text {op }}$. For example,

A morphism $f$ is monic in $\mathcal{C}^{\text {op }}$ if and only if it is epic in $\mathcal{C}$.
Indeed, $f$ is monic in $\mathcal{C}^{\text {op }}$ iff for all $g, h: C \rightarrow B$ in $\mathcal{C}^{\text {op }}$,

$$
f \circ g=f \circ h \Longrightarrow g=h,
$$

iff for all $g, h: B \rightarrow C$ in $\mathcal{C}$,

$$
g \circ f=h \circ f \Longrightarrow g=h,
$$

iff $f$ is epic in $\mathcal{C}$. We say that monic and epic are dual notions.
Exercise 14 If $P$ is a preorder, for example $(\mathbb{R}, \leq)$, describe $P^{\circ \mathrm{p}}$ explicitly.

## Subcategories

Another way to obtain new categories from old ones is by restricting their objects or arrows.

Definition 15 Let $\mathcal{C}$ be a category. Suppose that we are given collections

$$
\mathrm{Ob}(\mathcal{D}) \subseteq \mathrm{Ob}(\mathcal{C}), \quad \forall A, B \in \mathrm{Ob}(\mathcal{D}) . \mathcal{D}(A, B) \subseteq \mathcal{C}(A, B)
$$

We say that $\mathcal{D}$ is a subcategory of $\mathcal{C}$ if

$$
A \in \mathrm{Ob}(\mathcal{D}) \Rightarrow \operatorname{id}_{A} \in \mathcal{D}(A, A), \quad f \in \mathcal{D}(A, B), g \in \mathcal{D}(B, C) \Rightarrow g \circ f \in \mathcal{D}(A, C)
$$

and hence $\mathcal{D}$ itself is a category. In particular, $\mathcal{D}$ is:

- A full subcategory of $\mathcal{C}$ if for any $A, B \in \operatorname{Ob}(\mathcal{D}), \mathcal{D}(A, B)=\mathcal{C}(A, B)$.
- A lluf subcategory of $\mathcal{C}$ if $\mathrm{Ob}(\mathcal{D})=\mathrm{Ob}(\mathcal{C})$.

For example, Grp is a full subcategory of Mon (by Exercise 5), and Set is a lluf subcategory of Rel.

Simple cats
We close this section with some very basic examples of categories.

- $\mathbb{1}$ is the category with one object and one arrow, that is,

where the arrow is necessarily id. . Note that, although we say that $\mathbb{1}$ is the one-object/one-arrow category, there is by no means a unique such category. This is explained by the intuitively evident fact that any two such categories are isomorphic. (We will define what it means for categories to be isomorphic later.)
- In two-object categories, there is the one with two arrows, $2:=\bullet \bullet$, and also:

$$
2_{\rightarrow}:=\bullet \longrightarrow \bullet, \quad 2_{\Rightarrow}:=\bullet \longrightarrow \cdot, 2_{\neq}:=\bullet \longrightarrow
$$

Note that we have omitted identity arrows for economy. Categories with only identity arrows, like $\mathbb{1}$ and $\mathfrak{2}$, are called discrete categories.

Exercise 16 How many categories $\mathcal{C}$ with $\operatorname{Ob}(\mathcal{C})=\{\bullet\}$ are there? (Hint: what do such categories correspond to?)

### 1.1.6 Exercises

1. Consider the following properties of an arrow $f$ in a category $\mathcal{C}$.

- $f$ is split monic if for some $g, g \circ f$ is an identity arrow.
- $f$ is split epic if for some $g, f \circ g$ is an identity arrow.
a. Prove that if $f$ and $g$ are arrows such that $g \circ f$ is monic, then $f$ is monic.
b. Prove that, if $f$ is split epic then it is epic.
c. Prove that, if $f$ and $g \circ f$ are iso then $g$ is iso.
d. Prove that, if $f$ is monic and split epic then it is iso.
e. In the category Mon of monoids and monoid homomorphisms, consider the inclusion map

$$
i:(\mathbb{N},+, 0) \longrightarrow(\mathbb{Z},+, 0)
$$

of natural numbers into the integers. Show that this arrow is both monic and epic. Is it an iso?

The Axiom of Choice in Set Theory states that, if $\left\{X_{i}\right\}_{i \in I}$ is a family of non-empty sets, we can form a set $X=\left\{x_{i} \mid i \in I\right\}$ where $x_{i} \in X_{i}$ for all $i \in I$.
f. Show that in Set an arrow which is epic is split epic. Explain why this needs the Axiom of Choice.
g. Is it always the case that an arrow which is epic is split epic? Either prove that it is, or give a counter-example.
2. Give a description of partial orders as categories of a special kind.

### 1.2 Some Basic Constructions

We shall now look at a number of basic constructions which appear throughout mathematics, and which acquire their proper general form in the language of categories.

### 1.2.1 Initial and Terminal Objects

A first such example is that of initial and terminal objects. While apparently trivial, they are actually both important and useful, as we shall see in the sequel.

Definition 17 An object $I$ in a category $\mathcal{C}$ is initial if, for every object $A$, there exists a unique arrow from $I$ to $A$, which we write $\iota_{A}: I \rightarrow A$.

A terminal object in $\mathcal{C}$ is an object $T$ such that, for every object $A$, there exists a unique arrow from $A$ to $T$, which we write $\tau_{A}: A \rightarrow T$.

Note that initial and terminal objects are dual notions: $T$ is terminal in $\mathcal{C}$ iff it is initial in $\mathcal{C}^{\circ p}$. We sometimes write $\mathbf{1}$ for the terminal object and $\mathbf{0}$ for the initial one. Note also the assertions of unique existence in the definitions. This is one of the leitmotifs of category theory; we shall encounter it again in a conceptually deeper form in section 1.5.

Let us examine initial and terminal objects in our standard example categories.

- In $\mathbf{S e t}$, the empty set is an initial object while any one-element set $\{\bullet\}$ is terminal.
- In Pos, the poset $(\varnothing, \varnothing)$ is an initial object while $(\{\bullet\},\{(\bullet, \bullet)\})$ is terminal.
- In Top, the space $(\varnothing,\{\varnothing\})$ is an initial object while $(\{\bullet\},\{\varnothing,\{\bullet\}\})$ is terminal.
- In $\operatorname{Vect}_{k}$, the one-element space $\{0\}$ is both initial and terminal.
- In a poset, seen as a category, an initial object is a least element, while a terminal object is a greatest element.

Exercise 18 Verify these claims. In each case, identify the canonical arrows.
Exercise 19 Identify the initial and terminal objects in Rel.
Exercise 20 Suppose that a monoid, viewed as a category, has either an initial or a terminal object. What must the monoid be?

We shall now establish a fundamental fact: initial and terminal objects are unique up to (unique) isomorphism. As we shall see, this is characteristic of all such "universal" definitions. For example, the apparent arbitrariness in the fact that any singleton set is a terminal object in Set is answered by the fact that what counts is the property of being terminal; and this suffices to ensure that any two concrete objects having this property must be isomorphic to each other.

The proof of the proposition, while elementary, is a first example of distinctively categorical reasoning.

Proposition 21 If $I$ and $I^{\prime}$ are initial objects in the category $\mathcal{C}$ then there exists a unique isomorphism $I \xrightarrow{\cong} I^{\prime}$.

Proof: Since $I$ is initial and $I^{\prime}$ is an object of $\mathcal{C}$, there is a unique arrow $\iota_{I^{\prime}}: I \rightarrow I^{\prime}$. We claim that $\iota_{I^{\prime}}$ is an isomorphism.
Since $I^{\prime}$ is initial and $I$ is an object in $\mathcal{C}$, there is an arrow $\iota_{I}^{\prime}: I^{\prime} \rightarrow I$. Thus we obtain $\iota_{I^{\prime}} ; \iota_{I}^{\prime}: I \rightarrow I$, while we also have the identity morphism $\operatorname{id}_{I}: I \rightarrow I$. But $I$ is initial and therefore there exists a unique arrow from $I$ to $I$, which means that $\iota_{I^{\prime}} ; \iota_{I}^{\prime}=\mathrm{id}_{I}$. Similarly, $\iota_{I}^{\prime} ; \iota_{I^{\prime}}=\mathrm{id}_{I^{\prime}}$, so $\iota_{I^{\prime}}$ is indeed an isomorphism.

Hence, initial objects are "unique up to (unique) isomorphism", and we can (and do) speak of the initial object (if any such exists). Similarly for terminal objects.

Exercise 22 Let $\mathcal{C}$ be a category with an initial object $\mathbf{0}$. For any object $A$, show the following.

- If $A \cong \mathbf{0}$ then $A$ is an initial object.
- If there exists a monomorphism $f: A \rightarrow \mathbf{0}$ then $f$ is an iso, and hence $A$ is initial.


### 1.2.2 Products and Coproducts

### 1.2.2.1 Products

We now consider one of the most common constructions in mathematics: the formation of "direct products". Once again, rather than giving a case-by-case construction of direct products in each mathematical context we encounter, we can express once and for all a general notion of product, meaningful in any category - and such that, if a product exists, it is characterised uniquely up to unique isomorphism, just as for initial and terminal objects. Given a particular mathematical context, i.e. a category, we can then verify whether on not the product exists in that category. The concrete construction appropriate to the context will enter only into the proof of existence; all of the useful properties of the product follow from the general definition. Moreover, the categorical notion of product has a normative force; we can test whether a concrete construction works as intended by verifying that it satisfies the general definition.

In set theory, the cartesian product is defined in terms of the ordered pair:

$$
X \times Y:=\{(x, y) \mid x \in X \wedge y \in Y\} .
$$

It turns out that ordered pairs can be defined in set theory, e.g. as

$$
(x, y):=\{\{x, y\}, y\} .
$$

Note that in no sense is such a definition canonical. The essential properties of ordered pairs are:

1. We can retrieve the first and second components $x, y$ of the ordered pair $(x, y)$, allowing projection functions to be defined:

$$
\pi_{1}:(x, y) \mapsto x, \quad \pi_{2}:(x, y) \mapsto y .
$$

2. The information about first and second components completely determines the ordered pair:

$$
\left(x_{1}, x_{2}\right)=\left(y_{1}, y_{2}\right) \Longleftrightarrow x_{1}=y_{1} \wedge x_{2}=y_{2}
$$

The categorical definition expresses these properties in arrow-theoretic terms, meaningful in any category.

Definition 23 Let $A, B$ be objects in a category $\mathcal{C}$. An $A, B$-pairing is a triple $\left(P, p_{1}, p_{2}\right)$ where $P$ is an object, $p_{1}: P \rightarrow A$ and $p_{2}: P \rightarrow B$. A morphism of $A, B$-pairings

$$
f:\left(P, p_{1}, p_{2}\right) \longrightarrow\left(Q, q_{1}, q_{2}\right)
$$

is a morphism $f: P \rightarrow Q$ in $\mathcal{C}$ such that $q_{1} \circ f=p_{1}$ and $q_{2} \circ f=p_{2}$, i.e. the following diagram commutes.


The $A, B$-pairings form a category $\operatorname{Pair}(A, B)$. We say that $\left(A \times B, \pi_{1}, \pi_{2}\right)$ is a product of $A$ and $B$ if it is terminal in $\operatorname{Pair}(A, B)$.

Exercise 24 Verify that $\operatorname{Pair}(A, B)$ is a category.
Note that products are specified by triples $A \stackrel{\pi_{1}}{\longleftarrow} A \times B \xrightarrow{\pi_{2}} B$, where $\pi_{i}$ 's are called projections. For economy (and if projections are obvious) we may say that $A \times B$ is the product of $A$ and $B$. We say that $\mathcal{C}$ has (binary) products if each pair of objects $A, B$ has a product in $\mathcal{C}$. A direct consequence of the definition, by Proposition 21, is that if products exist, they are unique up to (unique) isomorphism.

Unpacking the uniqueness condition from $\operatorname{Pair}(A, B)$ back to $\mathcal{C}$ we obtain a more concise definition of products which we use in practice.

Definition 25 (Equivalent definition of product) Let $A, B$ be objects in a category $\mathcal{C}$. A product of $A$ and $B$ is an object $A \times B$ together with a pair of arrows $A \stackrel{\pi_{1}}{\longleftarrow} A \times B \xrightarrow{\pi_{2}} B$ such that for every triple $A \stackrel{f}{\longleftrightarrow} C \xrightarrow{g} B$ there exists a unique morphism

$$
\langle f, g\rangle: C \longrightarrow A \times B
$$

such that the following diagram commutes.


$$
\binom{\pi_{1} \circ\langle f, g\rangle=f}{\pi_{2} \circ\langle f, g\rangle=g}
$$

We call $\langle f, g\rangle$ the pairing of $f$ and $g$.
Note that the above diagram features a dashed arrow. Our intention with such diagrams is always to express the following idea: if the undashed part of the diagram commutes, then there exists a unique arrow (the dashed one) such that the whole diagram commutes. In any case, we shall always spell out the intended statement explicitly.

We look at how this definition works in our standard example categories.

- In Set, products are the usual cartesian products.
- In Pos, products are cartesian products with the pointwise order.
- In Top, products are cartesian products with the product topology.
- In Vect $k$, products are direct sums.
- In a poset, seen as a category, products are greatest lower bounds.

Exercise 26 Verify these claims.
The following proposition shows that the uniqueness of the pairing arrow can be specified purely equationally, by the equation:

$$
\forall h: C \rightarrow A \times B . h=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle
$$

Proposition 27 For any triple $A \stackrel{\pi_{1}}{\longleftarrow} A \times B \xrightarrow{\pi_{2}} B$ the following statements are equivalent.
(I) For any triple $A \stackrel{f}{\leftrightarrows} C \xrightarrow{g} B$ there exists a unique morphism $\langle f, g\rangle$ : $C \rightarrow A \times B$ such that $\pi_{1} \circ\langle f, g\rangle=f$ and $\pi_{2} \circ\langle f, g\rangle=g$.
(II) For any triple $A \stackrel{f}{\leftarrow} C \xrightarrow{g} B$ there exists a morphism $\langle f, g\rangle: C \rightarrow$ $A \times B$ such that $\pi_{1} \circ\langle f, g\rangle=f$ and $\pi_{2} \circ\langle f, g\rangle=g$, and moreover, for any $h: C \rightarrow A \times B, h=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle$.

Proof: For $(\mathrm{I}) \Rightarrow(\mathrm{II})$, take any $h: C \rightarrow A \times B$; we need to show $h=$ $\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle$. We have

$$
A \xrightarrow{\pi_{1} \circ h} C \xrightarrow{\pi_{2} \circ h} B
$$

and hence, by (I), there exists unique $k: C \rightarrow A \times B$ such that

$$
\begin{equation*}
\pi_{1} \circ k=\pi_{1} \circ h \quad \wedge \quad \pi_{2} \circ k=\pi_{2} \circ h \tag{*}
\end{equation*}
$$

Note now that $(*)$ holds both for $k:=h$ and $k:=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle$, the latter because of (I). Hence, $h=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle$.

For $(\mathrm{II}) \Rightarrow(\mathrm{I})$, take any triple $A \stackrel{f}{\longleftrightarrow} C \xrightarrow{g} B$. By (II), we have that there exists an arrow $\langle f, g\rangle: C \rightarrow A \times B$ such that $\pi_{1} \circ\langle f, g\rangle=f$ and $\pi_{2} \circ\langle f, g\rangle=g$. We need to show it is the unique such. Let $k: C \rightarrow A \times B$ s.t.

$$
\pi_{1} \circ k=f \quad \wedge \quad \pi_{2} \circ k=g
$$

Then, by (II),

$$
k=\left\langle\pi_{1} \circ k, \pi_{2} \circ k\right\rangle=\langle f, g\rangle
$$

as required.
In the following proposition we give some useful properties of products. First, let us introduce some notation for arrows: given $f_{1}: A_{1} \rightarrow B_{1}, f_{2}: A_{2} \rightarrow B_{2}$, define

$$
f_{1} \times f_{2}:=\left\langle f_{1} \circ \pi_{1}, f_{2} \circ \pi_{2}\right\rangle: A_{1} \times A_{2} \longrightarrow B_{1} \times B_{2} .
$$

Proposition 28 For any $f: A \rightarrow B, g: A \rightarrow C, h: A^{\prime} \rightarrow A$, and any $p: B \rightarrow B^{\prime}, q: C \rightarrow C^{\prime}$,

- $\langle f, g\rangle \circ h=\langle f \circ h, g \circ h\rangle$,
- $(p \times q) \circ\langle f, g\rangle=\langle p \circ f, q \circ g\rangle$.

Proof: For the first claim we have:

$$
\langle f, g\rangle \circ h=\left\langle\pi_{1} \circ(\langle f, g\rangle \circ h), \pi_{2} \circ(\langle f, g\rangle \circ h)\right\rangle=\langle f \circ h, g \circ h\rangle .
$$

And for the second:

$$
\begin{aligned}
(p \times q) \circ\langle f, g\rangle & =\left\langle p \circ \pi_{1}, q \circ \pi_{2}\right\rangle \circ\langle f, g\rangle \\
& =\left\langle p \circ \pi_{1} \circ\langle f, g\rangle, q \circ \pi_{2} \circ\langle f, g\rangle\right\rangle \\
& =\langle p \circ f, q \circ g\rangle .
\end{aligned}
$$

## General Products

The notion of products can be generalised to arbitrary arities as follows. A product for a family of objects $\left\{A_{i}\right\}_{i \in I}$ in a category $\mathcal{C}$ is an object $P$ and morphisms

$$
p_{i}: P \longrightarrow A_{i} \quad(i \in I)
$$

such that, for all objects $B$ and arrows

$$
f_{i}: B \longrightarrow A_{i} \quad(i \in I)
$$

there is a unique arrow

$$
g: B \longrightarrow P
$$

such that, for all $i \in I$, the following diagram commutes.


As before, if such a product exists, it is unique up to (unique) isomorphism. We write $P=\prod_{i \in I} A_{i}$ for the product object, and $g=\left\langle f_{i} \mid i \in I\right\rangle$ for the unique morphism in the definition.

Exercise 29 What is the product of the empty family?
Exercise 30 Show that if a category has binary and nullary products then it has all finite products.

### 1.2.2.2 Coproducts

We now investigate the dual notion to products: namely coproducts. Formally, coproducts in $\mathcal{C}$ are just products in $\mathcal{C}^{\text {op }}$, interpreted back in $\mathcal{C}$. We spell out the definition.

Definition 31 Let $A, B$ be objects in a category $\mathcal{C}$. A coproduct of $A$ and $B$ is an object $A+B$ together with a pair of arrows $A \xrightarrow{\mathrm{in}_{1}} A+B \stackrel{\mathrm{in}_{2}}{\rightleftarrows} B$ such that for every triple $A \xrightarrow{f} C \stackrel{g}{\longleftarrow} B$ there exists a unique morphism

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

such that the following diagram commutes.


We call the $\mathrm{in}_{i}$ 's injections and $[f, g]$ the copairing of $f$ and $g$. As with pairings, uniqueness of copairings can be specified by an equation:

$$
\forall h: A+B \rightarrow C . h=\left[h \circ \mathrm{in}_{1}, h \circ \mathrm{in}_{2}\right]
$$

## Coproducts in Set

This is given by disjoint union of sets, which can be defined concretely e.g. by

$$
X+Y:=\{1\} \times X \cup\{2\} \times Y .
$$

We can define injections

$$
\begin{gathered}
X \xrightarrow{\mathrm{in}_{1}} X+Y \stackrel{\mathrm{in}}{\longleftrightarrow} Y \\
\mathrm{in}_{1}(x):=(1, x), \quad \mathrm{in}_{2}(y):=(2, y) .
\end{gathered}
$$

Also, given functions $f: X \longrightarrow Z$ and $g: Y \longrightarrow Z$, we can define

$$
\begin{gathered}
{[f, g]: X+Y \longrightarrow Z} \\
{[f, g](1, x):=f(x), \quad[f, g](2, y):=g(y) .}
\end{gathered}
$$

Exercise 32 Check that this construction does yield coproducts in Set.
Note that this example suggests that coproducts allow for definition by cases. Let us examine coproducts for some of our other standard examples.

- In Pos, disjoint unions (with the inherited orders) are coproducts.
- In Top, topological disjoint unions are coproducts.
- In $\operatorname{Vect}_{k}$, direct sums are coproducts.
- In a poset, least upper bounds are coproducts.

Exercise 33 Verify these claims.
Exercise 34 Dually to products, express coproducts as initial objects of a category Copair $(A, B)$ of $A, B$-copairings.

### 1.2.3 Pullbacks and Equalisers

We shall consider two further constructions of interest: pullbacks and equalisers.

### 1.2.3.1 Pullbacks

Definition 35 Consider a pair of morphisms $A \xrightarrow{f} C \stackrel{g}{\leftarrow} B$. The pullback of $f$ along $g$ is a pair $A \stackrel{p}{\longleftarrow} D \xrightarrow{q} B$ such that $f \circ p=g \circ q$ and, for any pair $A \stackrel{p^{\prime}}{\stackrel{p^{\prime}}{\prime}} D^{q^{\prime}} B$ such that $f \circ p^{\prime}=g \circ q^{\prime}$, there exists a unique $h: D^{\prime} \rightarrow D$ such that $p^{\prime}=p \circ h$ and $q^{\prime}=q \circ h$. Diagrammatically,


## Example 36

- In Set the pullback of $A \stackrel{f}{\longrightarrow} C \stackrel{g}{\longleftarrow} B$ is defined as a subset of the cartesian product:

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

For example, consider a category $\mathcal{C}$ with

$$
\operatorname{Ar}(\mathcal{C}) \xrightarrow{\text { dom }} \operatorname{Ob}(\mathcal{C}) \stackrel{\text { cod }}{\leftrightarrows} \operatorname{Ar}(\mathcal{C})
$$

Then the pullback of dom along cod is the set of composable morphisms, i.e. pairs of morphisms $(f, g)$ in $\mathcal{C}$ such that $f \circ g$ is well-defined.

- In Set again, subsets (i.e. inclusion maps) pull back to subsets:


Exercise 37 Let $\mathcal{C}$ be a category with a terminal object 1. Show that, for any $A, B \in O b(\mathcal{C})$, the pullback of $A \xrightarrow{\tau_{A}} \mathbf{1} \stackrel{\tau_{B}}{\leftrightarrows} B$ is the product of $A$ and $B$, if it exists.

Just as for products, pullbacks can equivalently be described as terminal objects in suitable categories. Given a pair of morphisms $A \xrightarrow{f} C \stackrel{g}{\longleftrightarrow} B$, we define an $(f, g)$-cone to be a triple $(D, p, q)$ such that the following diagram commutes.


A morphism of $(f, g)$-cones $h:\left(D_{1}, p_{1}, q_{1}\right) \rightarrow\left(D_{2}, p_{2}, q_{2}\right)$ is a morphism $h: D_{1} \rightarrow D_{2}$ such that the following diagram commutes.


We can thus form a category Cone $(f, g)$. A pull-back of $f$ along $g$, if it exists, is exactly a terminal object of Cone $(f, g)$. Once again, this shows the uniqueness of pullbacks up to unique isomorphism.

### 1.2.3.2 Equalisers

Definition 38 Consider a pair of parallel arrows $A \underset{g}{\stackrel{f}{\Longrightarrow}} B$. An equaliser of $(f, g)$ is an arrow $e: E \rightarrow A$ such that $f \circ e=g \circ e$ and, for any arrow $h: D \rightarrow A$ such that $f \circ h=g \circ h$, there is a unique $\hat{h}: D \rightarrow E$ so that $h=e \circ \hat{h}$. Diagrammatically,


As for products, uniqueness of the arrow from $D$ to $E$ can be expressed equationally:

$$
\forall k: D \rightarrow E . \widehat{e \circ k}=k
$$

Exercise 39 Why is $\widehat{e \circ k}$ well-defined for any $k: D \rightarrow E$ ? Prove that the above equation is equivalent to the uniqueness requirement.

Example 40 In Set, the equaliser of $f, g$ is given by the inclusion

$$
\{x \in A \mid f(x)=g(x)\} \hookrightarrow A
$$

This allows equationally defined subsets to be defined as equalisers. For example, consider the pair of maps $\mathbb{R}^{2} \xrightarrow[g]{f} \mathbb{R}$, where

$$
f:(x, y) \mapsto x^{2}+y^{2}, \quad g:(x, y) \mapsto 1
$$

Then, the equaliser is the unit circle as a subset of $\mathbb{R}^{2}$.

### 1.2.4 Limits and Colimits

The notions we have introduced so far are all special cases of a general notion of limits in categories, and the dual notion of colimits.

| Limits | Colimits |
| :--- | :--- |
| Terminal Objects | Initial Objects |
| Products | Coproducts |
| Pullbacks | Pushouts |
| Equalisers | Coequalisers |

Table 1.1: Examples of Limits and Colimits.
An important aspect of studying any kind of mathematical structure is to see what limits and colimits the category of such structures has. We shall return to these ideas shortly.

### 1.2.5 Exercises

1. Give an example of a category where some pair of objects lacks a product or coproduct.
2. (Pullback lemma) Consider the following commutative diagram.


Given that the right hand square $B C E F$ and the outer square $A C D F$ are pullbacks, prove that the left hand square $A B D E$ is a pullback.
3. Consider $A \xrightarrow{f} C \stackrel{g}{\longleftarrow} B$ with pullback $A \stackrel{p}{\longleftarrow} D \xrightarrow{q} B$. For each $A \stackrel{p^{\prime}}{\longleftarrow}$ $D^{\prime} \xrightarrow{q^{\prime}} B^{\prime}$ with $f \circ p^{\prime}=g \circ q^{\prime}$, let $\phi\left(p^{\prime}, q^{\prime}\right): D^{\prime} \rightarrow D$ be the arrow dictated by the pullback condition. Express uniqueness of $\phi\left(p^{\prime}, q^{\prime}\right)$ equationally.

### 1.3 Functors

Part of the "categorical philosophy" is:

We can also apply this to categories!

### 1.3.1 Basics

A "morphism of categories" is a functor.
Definition 41 A functor $F: \mathcal{C} \rightarrow \mathcal{D}$ is given by:

- An object-map, assigning an object $F A$ of $\mathcal{D}$ to every object $A$ of $\mathcal{C}$.
- An arrow-map, assigning an arrow $F f: F A \rightarrow F B$ of $\mathcal{D}$ to every arrow $f$ : $A \rightarrow B$ of $\mathcal{C}$, in such a way that composition and identities are preserved:

$$
F(g \circ f)=F g \circ F f, \quad F \mathrm{id}_{A}=\operatorname{id}_{F A}
$$

Note that we use the same symbol to denote the object- and arrow-maps; in practice, this never causes confusion. Since functors preserve domains and codomains of arrows, for each pair of objects $A, B$ of $\mathcal{C}$, there is a well-defined map

$$
F_{A, B}: \mathcal{C}(A, B) \rightarrow \mathcal{D}(F A, F B)
$$

The conditions expressing preservation of composition and identities are called functoriality.

Example 42 Let $(P, \leq),(Q, \leq)$ be preorders (seen as categories). A functor $F:(P, \leq) \longrightarrow(Q, \leq)$ is specified by an object-map, say $F: P \rightarrow Q$, and an appropriate arrow-map. The arrow-map corresponds to the condition

$$
\forall p_{1}, p_{2} \in P . p_{1} \leq p_{2} \Longrightarrow F\left(p_{1}\right) \leq F\left(p_{2}\right)
$$

i.e. to monotonicity of $F$. Moreover, the functoriality conditions are trivial since in the codomain $(Q, \leq)$ all hom-sets are singletons.
Hence, a functor between preorders is just a monotone map.
Example 43 Let $(M, \cdot, 1),(N, \cdot, 1)$ be monoids. A functor $F:(M, \cdot, 1) \longrightarrow$ $(N, \cdot, 1)$ is specified by a trivial object map (monoids are categories with a single object) and an arrow-map, say $F: M \rightarrow N$. The functoriality conditions correspond to

$$
\forall m_{1}, m_{2} \in M . F\left(m_{1} \cdot m_{2}\right)=F\left(m_{1}\right) \cdot F\left(m_{2}\right), \quad F(1)=1
$$

i.e. to $F$ being a monoid homomorphism.

Hence, a functor between monoids is just a monoid homomorphism.
Other examples are the following.

- Inclusion of a sub-category, $\mathcal{C} \hookrightarrow \mathcal{D}$, is a functor (by taking the identity map for object- and arrow-map).
- The covariant powerset functor $\mathcal{P}$ : Set $\rightarrow$ Set:

$$
X \mapsto \mathcal{P}(X), \quad(f: X \rightarrow Y) \mapsto \mathcal{P}(f):=S \mapsto\{f(x) \mid x \in S\}
$$

- $U:$ Mon $\rightarrow$ Set is the 'forgetful' or 'underlying' functor which sends a monoid to its set of elements, 'forgetting' the algebraic structure, and sends a homomorphism to the corresponding function between sets. There are similar forgetful functors for other categories of structured sets. Why are these trivial-looking functors useful? - We shall see!
- Group theory examples. The assignment of the commutator sub-group of a group extends to a functor from Group to Group; and the assignment of the quotient by this normal subgroup extends to a functor from Group to AbGroup. The assignment of the centraliser of a group does not!
- More sophisticated examples: e.g. homology. The basic idea of algebraic topology is that there are functorial assignments of algebraic objects (e.g. groups) to topological spaces, and variants of this idea ('(co)homology theories') are pervasive throughout modern pure mathematics.

Functors 'of several variables'

We can generalise the notion of a functor to a mapping from several domain categories to a codomain category. For this we need the following definition.

Definition 44 For categories $\mathcal{C}, \mathcal{D}$ define the product category $\mathcal{C} \times \mathcal{D}$ as follows. An object in $\mathcal{C} \times \mathcal{D}$ is a pair of objects from $\mathcal{C}$ and $\mathcal{D}$, and an arrow in $\mathcal{C} \times \mathcal{D}$ is a pair of arrows from $\mathcal{C}$ and $\mathcal{D}$. Identities and arrow composition are defined componentwise:

$$
\operatorname{id}_{(A, B)}:=\left(\mathrm{id}_{A}, \mathrm{id}_{B}\right), \quad(f, g) \circ\left(f^{\prime}, g^{\prime}\right):=\left(f \circ f^{\prime}, g \circ g^{\prime}\right)
$$

A functor 'of two variables', with domains $\mathcal{C}$ and $\mathcal{D}$, to $\mathcal{E}$ is simply a functor:

$$
F: \mathcal{C} \times \mathcal{D} \longrightarrow \mathcal{E}
$$

For example, there are evident projection functors

$$
\mathcal{C} \longleftarrow \mathcal{C} \times \mathcal{D} \longrightarrow \mathcal{D}
$$

### 1.3.2 Further Examples

Set-valued functors
Many important constructions arise as functors $F: \mathcal{C} \rightarrow$ Set. For example:

- If $G$ is a group, a functor $F: G \rightarrow$ Set is an action of $G$ on a set.
- If $P$ is a poset representing time, a functor $F: P \rightarrow$ Set is a notion of set varying through time. This is related to Kripke semantics, and to forcing arguments in set theory.
- Recall that $2_{\Rightarrow}$ is the category $\rightarrow$. Then, functors $F: 2_{\Rightarrow} \rightarrow$ Set correspond to directed graphs understood as in Definition 4. i.e. as structures $(V, E, s, t)$, where $V$ is a set of vertices, $E$ is a set of edges, and $s, t: E \rightarrow V$ specify the source and target vertices for each edge.

Let us examine the first example in more detail. For a group $(G, \cdot, 1)$, a functor $F: G \rightarrow$ Set is specified by a set $X$ (to which the unique object of $G$ is mapped), and by an arrow-map sending each element $m$ of $G$ to an endofunction on $X$, say $m \bullet \_: X \rightarrow X$. Then, functoriality amounts to the conditions

$$
\forall m_{1}, m_{2} \in G . F\left(m_{1} \cdot m_{2}\right)=F\left(m_{1}\right) \circ F\left(m_{2}\right), \quad F(1)=\mathrm{id}_{X}
$$

that is, for all $m_{1}, m_{2} \in G$ and all $x \in X$,

$$
\left(m_{1} \cdot m_{2}\right) \cdot x=m_{1} \cdot m_{2} \cdot x, \quad 1 \cdot x=x
$$

We therefore see that $F$ defines an action of $G$ on $X$.
Exercise 45 Verify that functors $F: \mathcal{1}_{\Rightarrow} \rightarrow$ Set correspond to directed graphs.

## Example: Lists

Data-type constructors are functors. As a basic example, we consider lists. There is a functor

$$
\text { List }: \text { Set } \longrightarrow \text { Set }
$$

which takes a set $X$ to the set of all finite lists (sequences) of elements of $X$. List is functorial: its action on morphisms (i.e. functions, i.e. (functional) programs) is given by maplist:

$$
\begin{gathered}
\frac{f: X \longrightarrow Y}{\operatorname{List}(f): \operatorname{List}(X) \longrightarrow \operatorname{List}(Y)} \\
\operatorname{List}(f)\left[x_{1}, \ldots, x_{n}\right]:=\left[f\left(x_{1}\right), \ldots, f\left(x_{n}\right)\right]
\end{gathered}
$$

We can upgrade List to a functor MList : Set $\rightarrow$ Mon by mapping each set $X$ to the monoid $(\operatorname{List}(X), *, \epsilon)$ and $f: X \rightarrow Y$ to $\operatorname{List}(f)$, as above. The monoid operation $*: \operatorname{List}(X) \times \operatorname{List}(X) \rightarrow \operatorname{List}(X)$ is list concatenation, and $\epsilon$ is the empty list. We call MList $(X)$ the free monoid over $X$. This terminology will be justified in Chapter 5.

Products as functors
If a category $\mathcal{C}$ has binary products, then there is automatically a functor

$$
\__{-}: \mathcal{C} \times \mathcal{C} \longrightarrow \mathcal{C}
$$

which takes each pair $(A, B)$ to the product $A \times B$, and each $(f, g)$ to

$$
f \times g:=\left\langle f \circ \pi_{1}, g \circ \pi_{2}\right\rangle
$$

Functoriality is shown as follows, using proposition 28 and uniqueness of pairings in its equational form.

$$
\begin{aligned}
(f \times g) \circ\left(f^{\prime} \times g^{\prime}\right) & =(f \times g) \circ\left\langle f^{\prime} \circ \pi_{1}, g^{\prime} \circ \pi_{2}\right\rangle=\left\langle f \circ f^{\prime} \circ \pi_{1}, g \circ g^{\prime} \circ \pi_{2}\right\rangle \\
& =\left(f \circ f^{\prime}\right) \times\left(g \circ g^{\prime}\right), \\
\operatorname{id}_{A} \times \operatorname{id}_{B} & =\left\langle\operatorname{id}_{A} \circ \pi_{1}, \operatorname{id}_{B} \circ \pi_{2}\right\rangle=\left\langle\pi_{1} \circ \mathrm{id}_{A \times B}, \pi_{2} \circ \mathrm{id}_{A \times B}\right\rangle=\mathrm{id}_{A \times B}
\end{aligned}
$$

The category of categories
There is a category Cat whose objects are categories, and whose arrows are functors. Identities in Cat are given by identity functors:

$$
\operatorname{Id}_{\mathcal{C}}: \mathcal{C} \longrightarrow \mathcal{C}:=A \mapsto A, f \mapsto f
$$

Composition of functors is defined in the evident fashion. Note that if $F$ : $\mathcal{C} \rightarrow \mathcal{D}$ and $G: \mathcal{D} \rightarrow \mathcal{E}$ then, for $f: A \rightarrow B$ in $\mathcal{C}$,

$$
G \circ F(f):=G(F(f)): G(F(A)) \longrightarrow G(F(B))
$$

so the types work out. A category of categories sounds (and is) circular, but in practice is harmless: one usually makes some size restriction on the categories, and then Cat will be too 'big' to be an object of itself. See Appendix A.

Note that product categories are products in Cat! For any pair of categories $\mathcal{C}, \mathcal{D}$, set

$$
\mathcal{C} \stackrel{\pi_{1}}{\leftrightarrows} \mathcal{C} \times \mathcal{D} \xrightarrow{\pi_{2}} \mathcal{D}
$$

where $\mathcal{C} \times \mathcal{D}$ the product category (defined previously) and $\boldsymbol{\pi}_{i}$ 's the obvious projection functors. For any pair of functors $\mathcal{C} \stackrel{F}{\longleftarrow} \mathcal{E} \xrightarrow{G} \mathcal{D}$, set

$$
\langle F, G\rangle: \mathcal{E} \longrightarrow \mathcal{C} \times \mathcal{D}:=A \mapsto(F A, G A), f \mapsto(F f, G f)
$$

It is easy to see that $\langle F, G\rangle$ is indeed a functor. Moreover, satisfaction of the product diagram and uniqueness are shown exactly as in Set.

### 1.3.3 Contravariance

By definition, the arrow-map of a functor $F$ is covariant: it preserves the direction of arrows, so if $f: A \rightarrow B$ then $F f: F A \rightarrow F B$. A contravariant functor $G$ does exactly the opposite: it reverses arrow-direction, so if $f: A \rightarrow$ $B$ then $G f: G B \rightarrow G A$. A concise way to express contravariance is as follows.

Definition 46 Let $\mathcal{C}, \mathcal{D}$ be categories. A contravariant functor $G$ from $\mathcal{C}$ to $\mathcal{D}$ is a functor $G: \mathcal{C}^{\text {op }} \rightarrow \mathcal{D}$. (Equivalently, a functor $G: \mathcal{C} \rightarrow \mathcal{D}^{\text {op }}$.)

Explicitly, a contravariant functor $G$ is given by an assignment of:

- an object $G A$ in $\mathcal{D}$ to every object $A$ in $\mathcal{C}$,
- an arrow $G f: G B \rightarrow G A$ in $\mathcal{D}$ to every arrow $f: A \rightarrow B$ in $\mathcal{C}$, such that (notice the change of order in composition):

$$
G(g \circ f)=G f \circ G g, \quad G \operatorname{id}_{A}=\operatorname{id}_{G A} .
$$

Note that functors of several variables can be covariant in some variables and contravariant in others, e.g.

$$
F: \mathcal{C}^{\mathrm{op}} \times \mathcal{D} \longrightarrow \mathcal{E}
$$

## Examples of Contravariant Functors

- The contravariant powerset functor, $\mathcal{P}^{\mathrm{op}}: \mathbf{S e t}^{\mathrm{op}} \rightarrow \mathbf{S e t}$, is given by:

$$
\begin{aligned}
\mathcal{P}^{\mathrm{op}}(X) & :=\mathcal{P}(X) . \\
\mathcal{P}^{\mathrm{op}}(f: X \rightarrow Y) & : \mathcal{P}(Y) \longrightarrow \mathcal{P}(X):=T \mapsto\{x \in X \mid f(x) \in T\} .
\end{aligned}
$$

- The dual space functor on vector spaces:

$$
(-)^{*}: \text { Vect }_{k}^{\mathrm{op}} \longrightarrow \operatorname{Vect}_{k}:=V \mapsto V^{*} .
$$

Note that these are both examples of the following idea: send an object $A$ into functions from $A$ into some fixed object. For example, the powerset can be written as $\mathcal{P}(X)=2^{X}$, where we think of a subset in terms of its characteristic function.

Hom-functors

We now consider some fundamental examples of Set-valued functors. Given a category $\mathcal{C}$ and an object $A$ of $\mathcal{C}$, two functors to Set can be defined:

- The covariant Hom-functor at $A$,

$$
\mathcal{C}(A,-): \mathcal{C} \longrightarrow \text { Set }
$$

which is given by (recall that each $\mathcal{C}(A, B)$ is a set):

$$
\mathcal{C}\left(A,_{-}\right)(B):=\mathcal{C}(A, B), \quad \mathcal{C}\left(A,_{-}\right)(f: B \rightarrow C):=g \mapsto f \circ g
$$

We usually write $\mathcal{C}\left(A,_{-}\right)(f)$ as $\mathcal{C}(A, f)$. Functoriality reduces directly to the basic category axioms: associativity of composition and the unit laws for the identity.

- There is also a contravariant Hom-functor,

$$
\mathcal{C}(-, A): \mathcal{C}^{\mathrm{op}} \longrightarrow \text { Set }
$$

given by:

$$
\mathcal{C}(-, A)(B):=\mathcal{C}(B, A), \quad \mathcal{C}(-, A)(h: C \rightarrow B):=g \mapsto g \circ h .
$$

Generalising both of the above, we obtain a bivariant Hom-functor,

$$
\mathcal{C}(-,-): \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \longrightarrow \text { Set }
$$

Exercise 47 Spell out the definition of $\mathcal{C}\left(\__{-},\right): \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \longrightarrow$ Set. Verify carefully that it is a functor.

### 1.3.4 Properties of Functors

Definition 48 A functor $F: \mathcal{C} \rightarrow \mathcal{D}$ is said to be:

- faithful if each map $F_{A, B}: \mathcal{C}(A, B) \rightarrow \mathcal{D}(F A, F B)$ is injective;
- full if each map $F_{A, B}: \mathcal{C}(A, B) \rightarrow \mathcal{D}(F A, F B)$ is surjective;
- an embedding if $F$ is full, faithful, and injective on objects;
- an equivalence if $F$ is full, faithful, and essentially surjective: i.e. for every object $B$ of $\mathcal{D}$ there is an object $A$ of $\mathcal{C}$ such that $F(A) \cong B$;
- an isomorphism if there is a functor $G: \mathcal{D} \rightarrow \mathcal{C}$ such that

$$
G \circ F=\mathrm{Id}_{\mathcal{C}}, \quad F \circ G=\mathrm{Id}_{\mathcal{D}}
$$

We say that categories $\mathcal{C}$ and $\mathcal{D}$ are isomorphic, $\mathcal{C} \cong \mathcal{D}$, if there is an isomorphism between them. Note that this is just the usual notion of isomorphism applied to Cat. Examples:

- The forgetful functor $U:$ Mon $\rightarrow$ Set is faithful, but not full. For the latter, note that not all functions $f: M \rightarrow N$ yield an arrow $f:(M, \cdot, 1) \rightarrow$ $(N, \cdot, 1)$. Similar properties hold for other forgetful functors.
- The free monoid functor MList : Set $\rightarrow$ Mon is faithful, but not full.
- The product functor $\times_{-}: \mathcal{C} \times \mathcal{C} \longrightarrow \mathcal{C}$ is generally neither faithful nor full. For the latter, e.g. in Set, the function $f: \mathbb{N}^{2} \rightarrow \mathbb{N}^{2}:=(m, n) \mapsto(n, n)$ cannot be expressed in the form $f_{1} \times f_{2}$. Faithfulness of the functor is examined in exercise 1.3.5(2).
- There is an equivalence between FDVect $_{k}$ the category of finite dimensional vector spaces over the field $k$, and $\mathrm{Mat}_{k}$, the category of matrices with entries in $k$. Note that these categories are very far from isomorphic! This example is elaborated in exercise 1.3.5(1).


## Preservation and Reflection

Let $P$ be a property of arrows. We say that a functor $F: \mathcal{C} \rightarrow \mathcal{D}$ preserves $P$ if whenever $f$ satisfies $P$, so does $F(f)$. We say that $F$ reflects $P$ if whenever $F(f)$ satisfies $P$, so does $f$. For example:
a. All functors preserve isomorphisms, split monics and split epics.
b. Faithful functors reflect monics and epics.
c. Full and faithful functors reflect isomorphisms.
d. Equivalences preserve monics and epics.

- The forgetful functor $U:$ Mon $\rightarrow$ Set preserves products.

Let us show c ; the rest are given as exercises below. So let $f: A \rightarrow B$ in $\mathcal{C}$ be such that $F f$ is an iso, that is, it has an inverse $g^{\prime}: F B \rightarrow F A$. Then, by fullness, there exists some $g: B \rightarrow A$ so that $g^{\prime}=F g$. Thus,

$$
F(g \circ f)=F g \circ F f=g^{\prime} \circ F f=\mathrm{id}_{F A}=F\left(\mathrm{id}_{A}\right) .
$$

By faithfulness we obtain $g \circ f=\operatorname{id}_{A}$. Similarly, $f \circ g=\operatorname{id}_{B}$ and therefore $f$ is an isomorphism.

Exercise 49 Show items a, b and d above.
Exercise 50 Show the following.

- Functors do not in general reflect monics or epics.
- Faithful functors do not in general reflect isomorphisms.
- Full and faithful functors do not in general preserve monics or epics.


### 1.3.5 Exercises

1. Consider the category FDVect $_{\mathbb{R}}$ of finite dimensional vector spaces over $\mathbb{R}$, and $\mathrm{Mat}_{\mathbb{R}}$ of matrices over $\mathbb{R}$. Concretely, $\mathrm{Mat}_{\mathbb{R}}$ is defined as follows:

$$
\begin{aligned}
O b\left(\mathbf{M a t}_{\mathbb{R}}\right) & :=\mathbb{N}, \\
\operatorname{Mat}_{\mathbb{R}}(n, m) & :=\{M \mid M \text { is an } n \times m \text { matrix with entries in } \mathbb{R}\} .
\end{aligned}
$$

Thus, objects are natural numbers, and arrows $n \rightarrow m$ are $n \times m$ real matrices. Composition is matrix multiplication, and the identity on $n$ is the $n \times n$ identity matrix.
Now let $F:$ Mat $_{\mathbb{R}} \rightarrow \mathbf{F D V e c t}_{\mathbb{R}}$ be the functor taking each $n$ to the vector space $\mathbb{R}^{n}$ and each $M: n \rightarrow m$ to the linear function

$$
F M: \mathbb{R}^{n} \longrightarrow \mathbb{R}^{m}:=\left(x_{1}, \ldots, x_{n}\right) \mapsto\left[x_{1}, \ldots, x_{n}\right] M
$$

with the $1 \times m$ matrix $\left[x_{1}, \ldots, x_{n}\right] M$ considered as a vector in $\mathbb{R}^{m}$. Show that $F$ is full, faithful and essentially surjective, and hence that FDVect $_{\mathbb{R}}$ and $\mathrm{Mat}_{\mathbb{R}}$ are equivalent categories. Are they isomorphic?
2. Let $\mathcal{C}$ be a category with binary products such that, for each pair of objects $A, B$,

$$
\begin{equation*}
\mathcal{C}(A, B) \neq \emptyset . \tag{*}
\end{equation*}
$$

Show that the product functor $F: \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$ is faithful. Would $F$ still be faithful in the absence of condition ( ( ${ }^{*}$ )?

### 1.4 Natural Transformations

"Categories were only introduced to allow functors to be defined; functors were only introduced to allow natural transformations to be defined."

Just as categories have morphisms between them, namely functors, so functors have morphisms between them too - natural transformations.

### 1.4.1 Basics

Definition 51 Let $F, G: \mathcal{C} \rightarrow \mathcal{D}$ be functors. A natural transformation

$$
t: F \longrightarrow G
$$

is a family of morphisms in $\mathcal{D}$ indexed by objects $A$ of $\mathcal{C}$,

$$
\left\{t_{A}: F A \longrightarrow G A\right\}_{A \in O b(\mathcal{C})}
$$

such that, for all $f: A \rightarrow B$, the following diagram commutes.


This condition is known as naturality.
If each $t_{A}$ is an isomorphism, we say that $t$ is a natural isomorphism:

$$
t: F \xrightarrow{\cong} G .
$$

Examples:

- Let Id be the identity functor on Set, and $\times \circ\langle\mathrm{Id}, \mathrm{Id}\rangle$ be the functor taking each set $X$ to $X \times X$ and each function $f$ to $f \times f$. Then, there is a natural transformation $\Delta:$ Id $\longrightarrow \times \circ\langle\mathrm{Id}, \mathrm{Id}\rangle$ given by:

$$
\Delta_{X}: X \longrightarrow X \times X:=x \mapsto(x, x) .
$$

Naturality amounts to asserting that, for any function $f: X \rightarrow Y$, the following diagram commutes.


We call $\Delta$ the diagonal transformation on Set. In fact, it is the only natural transformation between these functors.

- The diagonal transformation can be defined for any category $\mathcal{C}$ with binary products by setting, for each object $A$ in $\mathcal{C}$,

$$
\Delta_{A}: A \longrightarrow A \times A:=\left\langle\mathrm{id}_{A}, \mathrm{id}_{A}\right\rangle
$$

Projections also yield natural transformations. For example the arrows

$$
\pi_{1(A, B)}: A \times B \longrightarrow A
$$

specify a natural transformation $\pi_{1}: \times \rightarrow \boldsymbol{\pi}_{1}$. Note that $\times, \boldsymbol{\pi}_{1}: \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$ are the functors for product and first projection respectively.

- Let $\mathcal{C}$ be a category with terminal object $T$, and let $K_{T}: \mathcal{C} \rightarrow \mathcal{C}$ be the functor mapping all objects to $T$ and all arrows to $\mathrm{id}_{T}$. Then, the canonical
arrows

$$
\tau_{A}: A \longrightarrow T
$$

specify a natural transformation $\tau:$ Id $\rightarrow K_{T}$ (where Id the identity functor on $\mathcal{C}$ ).

- Recall the functor List : Set $\rightarrow$ Set which takes a set $X$ to the set of finite lists with elements in $X$. We can define (amongst others) the following natural transformations,

$$
\text { reverse }: \text { List } \longrightarrow \text { List, unit }: \text { Id } \longrightarrow \text { List , flatten }: \text { List } \circ \text { List } \longrightarrow \text { List },
$$

by setting, for each set $X$,

$$
\begin{aligned}
\text { reverse }_{X} & : \operatorname{List}(X) \longrightarrow \operatorname{List}(X):=\left[x_{1}, \ldots, x_{n}\right] \mapsto\left[x_{n}, \ldots, x_{1}\right] \\
\text { unit }_{X} & : X \longrightarrow \operatorname{List}(X):=x \mapsto[x] \\
\text { flatten }_{X} & : \operatorname{List}(\operatorname{List}(X)) \longrightarrow \operatorname{List}(X) \\
& :=\left[\left[x_{1}^{1}, \ldots, x_{n_{1}}^{1}\right], \ldots,\left[x_{1}^{k}, \ldots, x_{n_{k}}^{k}\right]\right] \mapsto\left[x_{1}^{1}, \ldots \ldots, x_{n_{k}}^{k}\right] .
\end{aligned}
$$

- Consider the functor $P:=\times \circ\langle U, U\rangle$ with $U:$ Mon $\rightarrow$ Set, i.e.

$$
P: \text { Mon } \longrightarrow \text { Set }:=(M, \cdot, 1) \mapsto M \times M, f \mapsto f \times f .
$$

Then, the monoid operation yields a natural transformation $t: P \rightarrow U$ defined by:

$$
t_{(M, \cdot, 1)}: M \times M \longrightarrow M:=\left(m, m^{\prime}\right) \mapsto m \cdot m^{\prime} .
$$

Naturality corresponds to asserting that, for any $f:(M, \cdot, 1) \rightarrow(N, \cdot, 1)$, the following diagram commutes,

that is, for any $m_{1}, m_{2} \in M, f\left(m_{1}\right) \cdot f\left(m_{2}\right)=f\left(m_{1} \cdot m_{2}\right)$.

- If $V$ is a finite dimensional vector space, then $V$ is isomorphic to both its first dual $V^{*}$ and to its second dual $V^{* *}$.
However, while it is naturally isomorphic to its second dual, there is no natural isomorphism to the first dual. This was actually the original example which motivated Eilenberg and Mac Lane to define the concept of natural transformation; here naturality captures basis independence.

Exercise 52 Verify naturality of diagonal transformations, projections and terminals for a category $\mathcal{C}$ with finite products.

Exercise 53 Prove that the diagonal is the only natural transformation Id $\longrightarrow \times \circ\langle\mathrm{Id}, \mathrm{Id}\rangle$ on Set. Similarly, prove that the first projection is the only natural transformation $\times \rightarrow \boldsymbol{\pi}_{1}$ on Set.

### 1.4.2 Further Examples

Natural isomorphisms for products
Let $\mathcal{C}$ be a category with finite products, i.e. binary products and a terminal object 1. Then, we have the following canonical natural isomorphisms.

$$
\begin{aligned}
& a_{A, B, C}: A \times(B \times C) \xrightarrow{\cong}(A \times B) \times C, \\
& s_{A, B}: A \times B \stackrel{\cong}{\cong} B \times A, \\
& l_{A}: \mathbf{1} \times A \xrightarrow{\cong} A, \\
& r_{A}: A \times \mathbf{1} \xrightarrow{\cong} A \text {. }
\end{aligned}
$$

The first two isomorphisms are meant to assert that the product is associative and symmetric, and the last two that $\mathbf{1}$ is its unit. In later sections we will see that these conditions form part of the definition of symmetric monoidal categories.

These natural isomorphisms are defined explicitly by:

$$
\begin{aligned}
a_{A, B, C} & :=\left\langle\left\langle\pi_{1}, \pi_{1} \circ \pi_{2}\right\rangle, \pi_{2} \circ \pi_{2}\right\rangle, \\
s_{A, B} & :=\left\langle\pi_{2}, \pi_{1}\right\rangle \\
l_{A} & :=\pi_{2}, \\
r_{A} & :=\pi_{1} .
\end{aligned}
$$

Since natural isomorphisms are a self-dual notion, similar natural isomorphisms can be defined if $\mathcal{C}$ has binary coproducts and an initial object.

Exercise 54 Verify that these families of arrows are natural isomorphisms.

Natural transformations between Hom-functors

Let $f: A \rightarrow B$ in a category $\mathcal{C}$. Then, this induces a natural transformation

$$
\begin{aligned}
\mathcal{C}(f,-): \mathcal{C}(B,-) & \longrightarrow \mathcal{C}(A,-), \\
\mathcal{C}(f,-)_{C}: \mathcal{C}(B, C) & \longrightarrow \mathcal{C}(A, C):=(g: B \rightarrow C) \mapsto(g \circ f: A \rightarrow C) .
\end{aligned}
$$

Note that $\mathcal{C}(f,)_{C}$ is the same as $\mathcal{C}(f, C)$, the result of applying the contravariant functor $\mathcal{C}(-, C)$ to $f$. Hence, naturality amounts to asserting that, for each $h: C \rightarrow D$, the following diagram commutes.


Starting from a $g: B \rightarrow C$, we compute:

$$
\mathcal{C}(A, h)(\mathcal{C}(f, C)(g))=h \circ(g \circ f)=(h \circ g) \circ f=\mathcal{C}(f, D)(\mathcal{C}(B, h)(g)) .
$$

The natural transformation $\mathcal{C}(-, f): \mathcal{C}\left({ }_{-}, A\right) \rightarrow \mathcal{C}(-, B)$ is defined similarly.
Exercise 55 Define the natural transformation $\mathcal{C}(-, f)$ and verify its naturality.

There is a remarkable result, the Yoneda Lemma, which says that every natural transformation between Hom-functors comes from a (unique) arrow in $\mathcal{C}$ in the fashion described above.

Lemma 1. Let $A, B$ be objects in a category $\mathcal{C}$. For each natural transformation $t: \mathcal{C}\left(A, \__{-}\right) \rightarrow \mathcal{C}\left(B,{ }_{-}\right)$, there is a unique arrow $f: B \rightarrow A$ such that

$$
t=\mathcal{C}(f,-) .
$$

Proof: Take any such $A, B$ and $t$ and let

$$
f: B \longrightarrow A:=t_{A}\left(\mathrm{id}_{A}\right) .
$$

We want to show that $t=\mathcal{C}\left(f, \_\right)$. For any object $C$ and any arrow $g: A \rightarrow C$, naturality of $t$ means that the following commutes.


Starting from id $_{A}$ we have that:

$$
t_{C}\left(\mathcal{C}(A, g)\left(\mathrm{id}_{A}\right)\right)=\mathcal{C}(B, g)\left(t_{A}\left(\mathrm{id}_{A}\right)\right), \text { i.e. } t_{C}(g)=g \circ f
$$

Hence, noting that $\mathcal{C}(f, C)(g)=g \circ f$, we obtain $t=\mathcal{C}\left(f,{ }_{-}\right)$.
For uniqueness we have that, for any $f, f^{\prime}: B \rightarrow A$, if $\mathcal{C}\left(f,{ }_{-}\right)=\mathcal{C}\left(f^{\prime},{ }_{-}\right)$then

$$
f=\operatorname{id}_{A} \circ f=\mathcal{C}(f, A)\left(\mathrm{id}_{A}\right)=\mathcal{C}\left(f^{\prime}, A\right)\left(\mathrm{id}_{A}\right)=\operatorname{id}_{A} \circ f^{\prime}=f^{\prime}
$$

Exercise 56 Prove a similar result for contravariant hom-functors.

Alternative definition of equivalence
Another way of defining equivalence of categories is as follows.
Definition 57 We say that categories $\mathcal{C}$ and $\mathcal{D}$ are equivalent, $\mathcal{C} \simeq \mathcal{D}$, if there are functors $F: \mathcal{C} \rightarrow \mathcal{D}, G: \mathcal{D} \rightarrow \mathcal{C}$ and natural isomorphisms

$$
G \circ F \cong \mathrm{Id}_{\mathcal{C}}, \quad F \circ G \cong \mathrm{Id}_{\mathcal{D}}
$$

### 1.4.3 Functor Categories

Suppose we have functors $F, G, H: \mathcal{C} \rightarrow \mathcal{D}$ and natural transformations

$$
t: F \longrightarrow G, \quad u: G \longrightarrow H
$$

Then, we can compose these natural transformations, yielding $u \circ t: F \rightarrow H$ :

$$
(u \circ t)_{A}:=F A \xrightarrow{t_{A}} G A \xrightarrow{u_{A}} H A .
$$

Composition is associative, and has as identity the natural transformation

$$
I_{F}: F \longrightarrow F:=\left\{\left(I_{F}\right)_{A}:=\operatorname{id}_{A}: F A \longrightarrow F A\right\}_{A}
$$

These observations lead us to the following.
Definition 58 For categories $\mathcal{C}, \mathcal{D}$ define the functor category $\operatorname{Func}(\mathcal{C}, \mathcal{D})$ by taking:

- Objects: functors $F: \mathcal{C} \rightarrow \mathcal{D}$.
- Arrows: natural transformations $t: F \rightarrow G$.

Composition and identities are given as above.

Remark 59 We see that in the category Cat of categories and functors, each hom-set $\operatorname{Cat}(\mathcal{C}, \mathcal{D})$ itself has the structure of a category. In fact, Cat is the basic example of a "2-category", i.e. of a category where hom-sets are themselves categories.

Note that a natural isomorphism is precisely an isomorphism in the functor category. Let us proceed to some examples of functor categories.

- Recall that, for any group $G$, functors from $G$ to Set are $G$-actions on sets. Then, $\operatorname{Func}(G$, Set $)$ is the category of $G$-actions on sets and equivariant functions: $f: X \rightarrow Y$ such that $f(m \bullet x)=m \bullet f(x)$.
- Func $\left(\mathcal{L}_{\Rightarrow}\right.$, Set $)$ : Graphs and graph homomorphisms.
- If $F, G: P \rightarrow Q$ are monotone maps between posets, then $t: F \rightarrow G$ means that

$$
\forall x \in P . F x \leq G x
$$

Note that in this case naturality is trivial (hom-sets are singletons in $Q$ ).
Exercise 60 Verify the above descriptions of $\operatorname{Func}(G$, Set $)$ and $\operatorname{Func}\left(2_{3}\right.$, Set $)$.
Remark 61 The composition of natural transformations defined above is called vertical composition. The reason for this terminology is depicted below.


As expected, there is also a horizontal composition, which is given as follows.


### 1.4.4 Exercises

1. By identifying the relevant functors, express pairing $\left\langle_{-},-\right\rangle$as a natural transformation. What does naturality correspond to explicitly?
2. Show that the two definitions of equivalence of categories, namely
a. $\mathcal{C}$ and $\mathcal{D}$ are equivalent if there is an equivalence $F: \mathcal{C} \rightarrow \mathcal{D}$ (definition (48),
b. $\mathcal{C}$ and $\mathcal{D}$ are equivalent if there are $F: \mathcal{C} \rightarrow \mathcal{D}, G: \mathcal{D} \rightarrow \mathcal{C}$, and isomorphisms $F \circ G \cong \operatorname{ld}_{\mathcal{D}}, G \circ F \cong \mathrm{Id}_{\mathcal{C}}$ (definition 57),
are: equivalent! Note that this will need the Axiom of Choice.
3. Define a relation on objects in a category $\mathcal{C}$ by: $A \cong B$ iff $A$ and $B$ are isomorphic.
a. Show that this relation is an equivalence relation.

Define a skeleton of $\mathcal{C}$ to be the (full) subcategory obtained by choosing one object from each equivalence class of $\cong$ (note that this involves choices, and is not uniquely defined).
b. Show that $\mathcal{C}$ is equivalent to any skeleton.
c. Show that any two skeletons of $\mathcal{C}$ are isomorphic.
d. Give an example of a category whose objects form a proper class, but whose skeleton is finite.
4. Given a category $\mathcal{C}$, we can define a functor

$$
y: \mathcal{C} \longrightarrow \operatorname{Func}\left(\mathcal{C}^{\mathrm{op}}, \text { Set }\right):=A \mapsto \mathcal{C}(-, A), \quad f \mapsto \mathcal{C}(-, f)
$$

Prove carefully that this is indeed a functor. Use exercise 56 to conclude that $y$ is full and faithful. Prove that it is also injective on objects, and hence an embedding. It is known as the Yoneda embedding.
5. Define the horizontal composition $u \bullet t$ of natural transformations explicitly. Prove that it is associative.

### 1.5 Universality and Adjoints

There is a fundamental triad of categorical notions:
Functoriality, Naturality, Universality.

We have studied the first two notions explicitly. We have also seen many examples of universal definitions, notably the various notions of limits and colimits considered in section 1.2, It is now time to consider universality in general; the proper formulation of this fundamental and pervasive notion is one of the major achievements of basic category theory.

Universality arises when we are interested in finding canonical solutions to problems of construction: that is, we are interested not just in the existence of a solution but in its canonicity. This canonicity should guarantee uniqueness, in the sense we have become familiar with: a canonical solution should be unique up to (unique) isomorphism.

The notion of canonicity has a simple interpretation in the case of posets, as an extremal solution: one that is the least or the greatest among all solu-
tions. Such an extremal solution is obviously unique. For example, consider the problem of finding a lower bound of a pair of elements $A, B$ in a poset $P$ : a greatest lower bound of $A$ and $B$ is an extremal solution to this problem. As we have seen, this is the specialisation to posets of the problem of constructing a product:
$\rightsquigarrow$ A product of $A, B$ in a poset is an element $C$ such that $C \leq A$ and $C \leq B$, ( $C$ is a lower bound);
$\rightsquigarrow$ and for any other solution $C^{\prime}$, i.e. $C^{\prime}$ such that $C^{\prime} \leq A$ and $C^{\prime} \leq B$, we have $C^{\prime} \leq C$. ( $C$ is a greatest lower bound.)

Because the ideas of universality and adjunctions have an appealingly simple form in posets, which is, moreover, useful in its own right, we will develop the ideas in that special case first, as a prelude to the general discussion for categories.

### 1.5.1 Adjunctions for Posets

Suppose $g: Q \rightarrow P$ is a monotone map between posets. Given $x \in P$, a $g$ approximation of $x$ (from above) is an element $y \in Q$ such that $x \leq g(y)$. A best $g$-approximation of $x$ is an element $y \in Q$ such that

$$
x \leq g(y) \wedge \forall z \in Q \cdot(x \leq g(z) \Longrightarrow y \leq z)
$$

If a best $g$-approximation exists then it is clearly unique.

## Discussion

It is worth clarifying the notion of best $g$-approximation. If $y$ is a best $g$ approximation to $x$, then in particular, by monotonicity of $g, g(y)$ is the least element of the set of all $g(z)$ where $z \in Q$ and $x \leq g(z)$. However, the property of being a best approximation is much stronger than the mere existence of a least element of this set. We are asking for $y$ itself to be the least, in $Q$, among all elements $z$ such that $x \leq g(z)$. Thus, even if $g$ is surjective, so that for every $x$ there is a $y \in Q$ such that $g(y)=x$, there need not exist a best $g$-approximation to $x$. This is exactly the issue of having a canonical choice of solution.

Exercise 62 Give an example of a surjective monotone map $g: Q \rightarrow P$ and an element $x \in P$ such that there is no best $g$-approximation to $x$ in $Q$.

If such a best $g$-approximation $f(x)$ exists for all $x \in P$ then we have a function $f: P \rightarrow Q$ such that, for all $x \in P, z \in Q$ :

$$
\begin{equation*}
x \leq g(z) \Longleftrightarrow f(x) \leq z \tag{1.1}
\end{equation*}
$$

We say that $f$ is the left adjoint of $g$, and $g$ is the right adjoint of $f$. It is immediate from the definitions that the left adjoint of $g$, if it exists, is uniquely determined by $g$.

Proposition 63 If such a function $f$ exists, then it is monotone. Moreover,

$$
\operatorname{id}_{P} \leq g \circ f, \quad f \circ g \leq \operatorname{id}_{Q}, \quad f \circ g \circ f=f, \quad g \circ f \circ g=g
$$

Proof: If we take $z=f(x)$ in equation (1.1), then since $f(x) \leq f(x), x \leq$ $g \circ f(x)$. Similarly, taking $x=g(z)$ we obtain $f \circ g(z) \leq z$. Now, the ordering on functions $h, k: P \longrightarrow Q$ is the pointwise order:

$$
h \leq k \Longleftrightarrow \forall x \in P . h(x) \leq k(x)
$$

This gives the first two equations.
Now, if $x \leq_{P} x^{\prime}$ then $x \leq x^{\prime} \leq g \circ f\left(x^{\prime}\right)$, so $f\left(x^{\prime}\right)$ is a $g$-approximation of $x$, and hence $f(x) \leq f\left(x^{\prime}\right)$. Thus, $f$ is monotone.

Finally, using the fact that composition is monotone with respect to the pointwise order on functions, and the first two equations:

$$
g=\operatorname{id}_{P} \circ g \leq g \circ f \circ g \leq g \circ \mathrm{id}_{Q}=g
$$

and hence $g=g \circ f \circ g$. The other equation is proved similarly.

Examples:

- Consider the inclusion map

$$
i: \mathbb{Z} \hookrightarrow \mathbb{R}
$$

This has both a left adjoint $f^{L}$ and a right adjoint $f^{R}$, where $f^{L}, f^{R}$ : $\mathbb{R} \rightarrow \mathbb{Z}$. For all $z \in \mathbb{Z}, r \in \mathbb{R}$ :

$$
z \leq f^{R}(r) \Longleftrightarrow i(z) \leq r, \quad f^{L}(r) \leq z \Longleftrightarrow r \leq i(z)
$$

We see from these defining properties that the right adjoint maps a real $r$ to the greatest integer below it (the extremal solution to finding an integer below a given real). This is the standard floor function.
Similarly, the left adjoint maps a real to the least integer above it yielding the ceiling function. Thus:

$$
f^{R}(r)=\lfloor r\rfloor, \quad f^{L}(r)=\lceil r\rceil .
$$

- Consider a relation $R \subseteq X \times Y . R$ induces a function:

$$
f_{R}: \mathcal{P}(X) \longrightarrow \mathcal{P}(Y):=S \mapsto\{y \in Y \mid \exists x \in S . x R y\}
$$

This has a right adjoint $[R]: \mathcal{P}(Y) \longrightarrow \mathcal{P}(X)$ :

$$
S \subseteq[R] T \quad \Longleftrightarrow \quad f_{R}(S) \subseteq T
$$

The definition of $[R]$ which satisfies this condition is:

$$
[R] T:=\{x \in X \mid \forall y \in Y . x R y \Rightarrow y \in T\}
$$

If we consider a set of worlds $W$ with an accessibility relation $R \subseteq W \times W$ as in Kripke semantics for modal logic, we see that $[R]$ gives the usual Kripke semantics for the modal operator $\square$, seen as a propositional operator mapping the set of worlds satisfied by a formula $\phi$ to the set of worlds satisfied by $\square \phi$.
On the other hand, if we think of the relation $R$ as the denotation of a (possibly non-deterministic) program, and $T$ as a predicate on states, then $[R] T$ is exactly the weakest precondition $\mathbf{w p}(R, T)$. In Dynamic Logic, the two settings are combined, and we can write expressions such as $[R] T$ directly, where $T$ will be (the denotation of) some formula, and $R$ the relation corresponding to a program.

- Consider a function $f: X \rightarrow Y$. This induces a function:

$$
f^{-1}: \mathcal{P}(Y) \longrightarrow \mathcal{P}(X):=T \mapsto\{x \in X \mid f(x) \in T\}
$$

This function $f^{-1}$ has both a left adjoint $\exists(f): \mathcal{P}(X) \longrightarrow \mathcal{P}(Y)$, and a right adjoint $\forall(f): \mathcal{P}(X) \longrightarrow \mathcal{P}(Y)$. For all $S \subseteq X, T \subseteq Y$ :

$$
\exists(f)(S) \subseteq T \Longleftrightarrow S \subseteq f^{-1}(T), \quad f^{-1}(T) \subseteq S \Longleftrightarrow T \subseteq \forall(f)(S)
$$

How can we define $\forall(f)$ and $\exists(f)$ explicitly so as to fulfil these defining conditions? - As follows:

$$
\begin{aligned}
& \exists(f)(S):=\{y \in Y \mid \exists x \in X . f(x)=y \wedge x \in S\} \\
& \forall(f)(S):=\{y \in Y \mid \forall x \in X . f(x)=y \Rightarrow x \in S\}
\end{aligned}
$$

If $R \subseteq X \times Y$, which we write in logical notation as $R(x, y)$, and we take the projection function $\pi_{1}: X \times Y \longrightarrow X$, then:

$$
\forall\left(\pi_{1}\right)(R) \equiv \forall y \cdot R(x, y), \quad \exists\left(\pi_{1}\right)(R) \equiv \exists y \cdot R(x, y)
$$

This extends to an algebraic form of the usual Tarski model-theoretic semantics for first-order logic, in which:

> Quantifiers are Adjoints

Couniversality
We can dualise the discussion, so that starting with a monotone map $f$ : $P \rightarrow Q$ and $y \in Q$, we can ask for the best $P$-approximation to $y$ from below: $x \in P$ such that $f(x) \leq y$, and for all $z \in P$ :

$$
f(z) \leq y \Longleftrightarrow z \leq x
$$

If such a best approximation $g(y)$ exists for all $y \in Q$, we obtain a monotone map $g: Q \rightarrow P$ such that $g$ is right adjoint to $f$. From the symmetry of the definition, it is clear that:

$$
f \text { is the left adjoint of } g \Longleftrightarrow g \text { is the right adjoint of } f
$$

and each determines the other uniquely.

### 1.5.2 Universal Arrows and Adjoints

Our discussion of best approximations for posets is lifted to general categories as follows.

Definition 64 Let $G: \mathcal{D} \rightarrow \mathcal{C}$ be a functor, and $C$ an object of $\mathcal{C}$. A universal arrow from $C$ to $G$ is a pair $(D, \eta)$ where $D$ is an object of $\mathcal{D}$ and

$$
\eta: C \longrightarrow G(D),
$$

such that, for any object $D^{\prime}$ of $\mathcal{D}$ and morphism $f: C \rightarrow G\left(D^{\prime}\right)$, there exists a unique morphism $\hat{f}: D \rightarrow D^{\prime}$ in $\mathcal{D}$ such that $f=G(\hat{f}) \circ \eta$. Diagrammatically:


As in previous cases, uniqueness can be given a purely equational specification:

$$
\begin{equation*}
\forall h: D \longrightarrow D^{\prime} . \widehat{G(h) \circ} \eta=h \tag{1.2}
\end{equation*}
$$

Exercise 65 Show that if $(D, \eta)$ and $\left(D^{\prime}, \eta^{\prime}\right)$ are universal arrows from $C$ to $G$ then there is a unique isomorphism $D \cong D^{\prime}$.

Exercise 66 Check that the equational specification of uniqueness (1.2) is valid.

Examples:

- Take $U:$ Mon $\rightarrow$ Set. Given a set $X$, the universal arrow is

$$
\eta_{X}: X \longrightarrow U(\operatorname{MList}(X)):=x \mapsto[x] .
$$

Indeed, for any monoid $(M, \cdot, 1)$ and any function $f: X \rightarrow M$, set

$$
\hat{f}: \operatorname{MList}(X) \longrightarrow(M, \cdot, 1):=\left[x_{1}, \ldots, x_{n}\right] \mapsto f\left(x_{1}\right) \cdots \cdot f\left(x_{n}\right)
$$

It is easy to see that $\hat{f}$ is a monoid homomorphism, and that $U(\hat{f}) \circ \eta_{X}=f$. Moreover, for uniqueness we have that, for any $h: \operatorname{MList}(X) \rightarrow(M, \cdot, 1)$,

$$
\begin{aligned}
U(h) \circ \eta_{X} & =x \widehat{\mapsto h([x])}=\left[x_{1}, \ldots, x_{n}\right] \mapsto h\left(\left[x_{1}\right]\right) \cdots \cdots \cdot h\left(\left[x_{n}\right]\right) \\
& =\left[x_{1}, \ldots, x_{n}\right] \mapsto h\left(\left[x_{1}\right] * \cdots *\left[x_{n}\right]\right) \\
& =\left[x_{1}, \ldots, x_{n}\right] \mapsto h\left(\left[x_{1}, \ldots, x_{n}\right]\right)=h
\end{aligned}
$$

- Let $K: \mathcal{C} \rightarrow \mathbb{1}$ be the unique functor to the one-object/one-arrow category. A universal arrow from the object of $\mathbb{1}$ to $K$ corresponds to an initial object in $\mathcal{C}$.
Indeed, such a universal arrow is given by an object $I$ of $\mathcal{C}$ (and a trivial arrow in $\mathbb{1}$ ), such that for any $A$ in $\mathcal{C}$ (and relevant arrow in $\mathbb{1}$ ) there exists a unique arrow from $I$ to $A$ (such that a trivial condition holds).
- Consider the functor $\left\langle\mathrm{Id}_{\mathcal{C}}, \mathrm{Id}_{\mathcal{C}}\right\rangle: \mathcal{C} \rightarrow \mathcal{C} \times \mathcal{C}$, taking each object $A$ to $(A, A)$ and each arrow $f$ to $(f, f)$. A universal arrow from an object $(A, B)$ of $\mathcal{C} \times \mathcal{C}$ to $\left\langle\operatorname{Id}_{\mathcal{C}}, \operatorname{Id}_{\mathcal{C}}\right\rangle$ corresponds to a coproduct of $A$ and $B$.

Exercise 67 Verify the description of coproducts as universal arrows.
As in the case of posets, a related notion to universal arrows is that of adjunction.

Definition 68 Let $\mathcal{C}, \mathcal{D}$ be categories. An adjunction from $\mathcal{C}$ to $\mathcal{D}$ is a triple $(F, G, \theta)$, where $F$ and $G$ are functors

$$
\mathcal{C} \underset{G}{\stackrel{F}{\rightleftarrows}} \mathcal{D}
$$

and $\theta$ is a family of bijections

$$
\theta_{A, B}: \mathcal{C}(A, G(B)) \xrightarrow{\cong} \mathcal{D}(F(A), B)
$$

for each $A \in O b(\mathcal{C})$ and $B \in O b(\mathcal{D})$, natural in $A$ and $B$. We say that $F$ is left adjoint to $G$, and $G$ is right adjoint to $F$.

Note that $\theta$ should be understood as the "witnessed" form - i.e. arrows instead of mere relations - of the defining condition for adjunctions in the case of posets:

$$
x \leq g(y) \Longleftrightarrow f(x) \leq y
$$

This is often displayed as a two-way 'inference rule':

$$
\frac{A \longrightarrow G B}{F A \longrightarrow B}
$$

Naturality of $\theta$ is expressed as follows: for any $f: A \rightarrow G(B)$ and any $g: A^{\prime} \rightarrow A, h: B \rightarrow B^{\prime}$,

$$
\begin{aligned}
\theta_{A^{\prime}, B}(f \circ g) & =\theta_{A, B}(f) \circ F(g), \\
\theta_{A, B^{\prime}}(G(h) \circ f) & =h \circ \theta_{A, B}(f) .
\end{aligned}
$$

Note that $f, g$ are in $\mathcal{C}$, and $h$ is in $\mathcal{D}$. In one line:

$$
\theta_{A^{\prime}, B^{\prime}}(G(h) \circ f \circ g)=h \circ \theta_{A, B}(f) \circ F(g) .
$$

Diagrammatically:


Thus, $\theta$ is in fact a natural isomorphism

$$
\theta: \mathcal{C}(-, G(-)) \stackrel{ }{\cong} \mathcal{D}\left(F(-), \__{-}\right)
$$

where $\mathcal{C}\left(\__{-}, G(-)\right): \mathcal{C}^{\text {op }} \times \mathcal{D} \rightarrow$ Set is the result of composing the bivariant hom-functor $\mathcal{C}\left(\__{-},\right)$with $\mathrm{Id}_{\mathcal{C}_{\text {op }}} \times G$, and $\mathcal{D}\left(F(-),{ }_{-}\right)$is similar.

In the next propositions we show that universal arrows and adjunctions are equivalent notions.

Proposition 69 (Universals define adjunctions) Let $G: \mathcal{D} \rightarrow \mathcal{C}$. If for every object $C$ of $\mathcal{C}$ there exists a universal arrow $\eta_{C}: C \rightarrow G(F(C))$, then:

1. $F$ uniquely extends to a functor $F: \mathcal{C} \rightarrow \mathcal{D}$ such that $\eta: \operatorname{ld}_{\mathcal{C}} \rightarrow G \circ F$ is a natural transformation.
2. $F$ is uniquely determined by $G$ (up to unique natural isomorphism), and vice versa.
3. For each pair of objects $C$ of $\mathcal{C}$ and $D$ of $\mathcal{D}$, there is a natural bijection:

$$
\theta_{C, D}: \mathcal{C}(C, G(D)) \cong \mathcal{D}(F(C), D)
$$

Proof: For 1, we extend $F$ to a functor as follows. Given $f: C \rightarrow C^{\prime}$ in $\mathcal{C}$, we consider the composition

$$
\eta_{C^{\prime}} \circ f: C \longrightarrow G F C^{\prime}
$$

By the universal property of $\eta_{C}$, there exists a unique arrow $F f: F C \rightarrow F C^{\prime}$ such that the following diagram commutes.


Note that the above is the naturality diagram for $\eta$ on $C$, hence the arrow-map thus defined for $F$ is the unique candidate that makes $\eta$ a natural transformation.
It remains to verify the functoriality of $F$. To show that $F$ preserves composition, consider $g: C^{\prime} \rightarrow C^{\prime \prime}$. We have the following commutative diagram,

from which it follows that

$$
\begin{aligned}
& G(F g \circ F f) \circ \eta_{C}=G F g \circ G F f \circ \eta_{C}=\eta_{C^{\prime \prime}} \circ g \circ f, \\
\therefore & F(g \circ f)=\eta_{C^{\prime \prime} \circ g \circ f=G(F g \circ F f}^{\circ} \circ \eta_{C}=F g \circ F f,
\end{aligned}
$$

where the last equality above holds because of (1.2). The verification that $F$ preserves identities is similar.
For 2, we have that each $F C$ is determined uniquely up to unique isomorphism, by the universal property, and once the object part of $F$ is fixed, the arrow part is uniquely determined.
For 3, we need to define a natural isomorphism $\theta_{C, D}: \mathcal{C}(C, G(D)) \cong$ $\mathcal{D}(F(C), D)$. Given $f: C \rightarrow G D, \theta_{C, D}(f)$ is defined to be the unique arrow $F C \rightarrow D$ such that the following commutes, as dictated by universality.


Suppose that $\theta_{C, D}(f)=\theta_{C, D}(g)$. Then

$$
f=G\left(\theta_{C, D}(f)\right) \circ \eta_{C}=G\left(\theta_{C, D}(g)\right) \circ \eta_{C}=g
$$

Thus $\theta_{C, D}$ is injective. Moreover, given $h: F C \rightarrow D$, by the equational formulation of uniqueness (1.2) we have:

$$
h=\theta_{C, D}\left(G h \circ \eta_{C}\right) .
$$

Thus $\theta_{C, D}$ is surjective. We are left to show naturality, i.e. that the following diagram commutes, for all $h: C^{\prime} \rightarrow C$ and $g: D \rightarrow D^{\prime}$.


We chase around the diagram, starting from $f: C \rightarrow G D$.

$$
\begin{aligned}
\mathcal{D}(F h, g) \circ \theta_{C, D}(f) & =g \circ \theta_{C, D}(f) \circ F h \\
\theta_{C^{\prime}, D^{\prime}} \circ \mathcal{C}(h, G g)(f) & =\theta_{C^{\prime}, D^{\prime}}(G g \circ f \circ h)
\end{aligned}
$$

Now:

$$
\begin{aligned}
g \circ \theta_{C, D}(f) \circ F h & =\theta_{C^{\prime}, D^{\prime}}\left(G\left(g \circ \theta_{C, D}(f) \circ F h\right) \circ \eta_{C^{\prime}}\right) & & \text { by (1.2) } \\
& =\theta_{C^{\prime}, D^{\prime}}\left(G g \circ G\left(\theta_{C, D}(f)\right) \circ G F h \circ \eta_{C^{\prime}}\right) & & \text { functoriality of } G \\
& =\theta_{C^{\prime}, D^{\prime}}\left(G g \circ G\left(\theta_{C, D}(f)\right) \circ \eta_{C} \circ h\right) & & \text { naturality of } \eta \\
& =\theta_{C^{\prime}, D^{\prime}}(G g \circ f \circ h) & & \text { by (1.2). }
\end{aligned}
$$

Proposition 70 (Adjunctions define universals) Let $G: \mathcal{D} \rightarrow \mathcal{C}$ be $a$ functor, $D \in O b(\mathcal{D})$ and $C \in O b(\mathcal{C})$. If, for any $D^{\prime} \in O b(\mathcal{D})$, there is a bijection

$$
\phi_{D^{\prime}}: \mathcal{C}\left(C, G\left(D^{\prime}\right)\right) \cong \mathcal{D}\left(D, D^{\prime}\right)
$$

natural in $D^{\prime}$ then there is a universal arrow $\eta: C \rightarrow G(D)$.
Proof: Take $\eta: C \rightarrow G(D):=\phi_{D}^{-1}\left(\mathrm{id}_{D}\right)$ and, for any $g: C \rightarrow G\left(D^{\prime}\right)$, take $\hat{g}: D \rightarrow D^{\prime}:=\phi_{D^{\prime}}(g)$.
We have that

$$
G(\hat{g}) \circ \eta=G(\hat{g}) \circ \phi_{D}^{-1}\left(\mathrm{id}_{D}\right) \stackrel{\text { nat }}{=} \phi_{D^{\prime}}^{-1}(\hat{g})=g
$$

Moreover, for any $h: D \rightarrow D^{\prime}$,

$$
\phi_{D^{\prime}}(G h \circ \eta)=\phi_{D^{\prime}}\left(G h \circ \phi_{D}^{-1}\left(\operatorname{id}_{D}\right)\right) \stackrel{\text { nat }}{=} \phi_{D^{\prime}}\left(\phi_{D^{\prime}}^{-1}(h)\right)=h,
$$

where equalities labelled with "nat" hold because of naturality of $\phi$.
Corollary $71 \operatorname{Let}(F, G, \theta)$ be an adjunction with $F: \mathcal{C} \rightarrow \mathcal{D}$. Then, for each $C \in O b(\mathcal{C})$ there is a universal arrow $\eta: C \rightarrow G(F(C))$.

## Equivalence of Universals and Adjoints

Thus we see that the following two situations are equivalent, in the sense that each determines the other uniquely.

- We are given a functor $G: \mathcal{D} \rightarrow \mathcal{C}$, and for each object $C$ of $\mathcal{C}$ a universal arrow from $C$ to $G$.
- We are given functors $F: \mathcal{C} \rightarrow \mathcal{D}$ and $G: \mathcal{D} \rightarrow \mathcal{C}$, and a natural bijection

$$
\theta_{C, D}: \mathcal{C}(C, G(D)) \cong \mathcal{D}(F(C), D)
$$

## Couniversal Arrows

Let $F: \mathcal{C} \rightarrow \mathcal{D}$ be a functor, and $D$ an object of $\mathcal{D}$. A couniversal arrow from $F$ to $D$ is an object $C$ of $\mathcal{C}$ and a morphism

$$
\epsilon: F(C) \longrightarrow D
$$

such that, for every object $C^{\prime}$ of $\mathcal{C}$ and morphism $g: F\left(C^{\prime}\right) \rightarrow D$, there exists a unique morphism $\bar{g}: C^{\prime} \rightarrow C$ in $\mathcal{C}$ such that $g=\epsilon \circ F(\bar{g})$.
Diagrammatically:


By exactly similar (but dual) reasoning to the previous propositions, an adjunction implies the existence of couniversal arrows, and the existence of the latter implies the existence of the adjunction. Hence,

$$
\text { Universality } \equiv \text { Adjunctions } \equiv \text { Couniversality }
$$

Some examples of couniversal arrows:

- A terminal object in a category $\mathcal{C}$ is a couniversal arrow from the unique functor $K: \mathcal{C} \rightarrow \mathbb{1}$ to the unique object in $\mathbb{1}$.
- Let $A, B$ be objects of $\mathcal{C}$. A product of $A$ and $B$ is a couniversal arrow from $\left\langle\mathrm{Id}_{\mathcal{C}}, \mathrm{Id}_{\mathcal{C}}\right\rangle: \mathcal{C} \rightarrow \mathcal{C} \times \mathcal{C}$ to $(A, B)$.


### 1.5.3 Limits and Colimits

In the previous paragraph we described products $A \times B$ as couniversal arrows from the diagonal functor $\Delta: \mathcal{C} \rightarrow \mathcal{C} \times \mathcal{C}$ to $(A, B)$. $\Delta$ is the functor assigning $(A, A)$ to each object $A$, and $(f, f)$ to each arrow $f$. Noting that $\mathcal{C} \times \mathcal{C}=\mathcal{C}^{2}$, where $\mathcal{C}^{2}$ is a functor category, this suggests an important generalisation.

Definition 72 Let $\mathcal{C}$ be a category and $\mathcal{I}$ be another category, thought of as an "index category". A diagram of shape $\mathcal{I}$ in $\mathcal{C}$ is just a functor $F: \mathcal{I} \rightarrow \mathcal{C}$. Consider the functor category $\mathcal{C}^{\mathcal{I}}$ with objects the functors from $\mathcal{I}$ to $\mathcal{C}$, and natural transformations as morphisms. There is a diagonal functor

$$
\Delta: \mathcal{C} \longrightarrow \mathcal{C}^{\mathcal{I}}
$$

taking each object $C$ of $\mathcal{C}$ to the constant functor $K_{C}: \mathcal{I} \rightarrow \mathcal{C}$, which maps every object of $\mathcal{I}$ to $C$. A limit for the diagram $F$ is a couniversal arrow from $\Delta$ to $F$.

This concept of limit subsumes products (including infinite products), pullbacks, inverse limits, etc.
For example, take $\mathcal{I}:=2_{\Rightarrow}$ (we have seen this before: $\mathcal{2}_{\Rightarrow}=\bullet$ ). A functor $F$ from $\mathcal{I}$ to $\mathcal{C}$ corresponds to a diagram:


A couniversal arrow from $\Delta$ to $F$ corresponds to the following situation,

i.e. to an equaliser!

By dualising limits we obtain colimits. Some important examples are coproducts, coequalisers, pushouts and $\omega$-colimits.

Exercise 73 Verify that pullbacks are limits by taking:

$$
\mathcal{I}:=\bullet \longrightarrow \bullet \longleftarrow \bullet
$$

Limits as terminal objects
Consider $\Delta: \mathcal{C} \rightarrow \mathcal{C}^{\mathcal{I}}$ and $F: \mathcal{I} \rightarrow \mathcal{C}$. A cone to $F$ is an object $C$ of $\mathcal{C}$ and family of arrows $\gamma$,

$$
\left\{\gamma_{I}: C \longrightarrow F I\right\}_{I \in O b(\mathcal{I})},
$$

such that, for any $f: I \rightarrow J$, the following triangle commutes.


Thus a cone is exactly a natural transformation $\gamma: \Delta C \rightarrow F$. A morphism of cones ('mediating morphism') $(C, \gamma) \longrightarrow(D, \delta)$ is an arrow $g: C \rightarrow D$ such that each of the following triangles commutes.


We obtain a category Cone $(F)$ whose objects are cones to $F$ and whose arrows are mediating morphisms. Then, a limit of $F$ is a terminal object in Cone (F).

### 1.5.4 Exponentials

In Set, given sets $A, B$, we can form the set of functions $B^{A}:=\boldsymbol{\operatorname { S e t }}(A, B)$, which is again a set, i.e. an object of $\boldsymbol{S e t}$. This closure of Set under forming "function spaces" is one of its most important properties.

How can we axiomatise this situation? Once again, rather than asking what the elements of a function space are, we ask instead what we can $d o$ with them operationally. The answer is simple: apply functions to their arguments. That is, there is a map

$$
\mathrm{ev}_{A, B}: B^{A} \times A \longrightarrow B \text { such that } \operatorname{ev}_{A, B}(f, a)=f(a)
$$

We can think of the function as a 'black box': we can feed it inputs and observe the outputs.

Evaluation has the following couniversal property. For any $g: C \times A \rightarrow B$, there is a unique map $\Lambda(g): C \rightarrow B^{A}$ such that the following diagram commutes.


In Set, this is defined by:

$$
\Lambda(g)(c): A \longrightarrow B:=a \mapsto g(c, a) .
$$

This process of transforming a function of two arguments into a functionvalued function of one argument is known as currying, after H. B. Curry. It is an algebraic form of $\lambda$-abstraction.

We are now led to the general definition of exponentials. Note that, for each object $A$ of a category $\mathcal{C}$ with products, we can define a functor

$$
\_\times A: \mathcal{C} \longrightarrow \mathcal{C}
$$

Definition 74 Let $\mathcal{C}$ be a category with binary products. We say that $\mathcal{C}$ has exponentials if for all objects $A$ and $B$ of $\mathcal{C}$ there is a couniversal arrow from $-\times A$ to $B$, i.e. an object $B^{A}$ of $\mathcal{C}$ and a morphism

$$
\mathrm{ev}_{A, B}: B^{A} \times A \longrightarrow B
$$

with the couniversal property: for every $g: C \times A \rightarrow B$, there is a unique morphism $\Lambda(g): C \rightarrow B^{A}$ such that the following diagram commutes.


As before, the couniversal property can be given in purely equational terms, as follows. For every $h: C \rightarrow B^{A}$,

$$
\Lambda\left(\mathrm{ev}_{A, B} \circ h \times \mathrm{id}_{A}\right)=h
$$

Equivalently, $\mathcal{C}$ has exponentials if, for every object $A$, the functor ${ }^{-} \times A$ has a right adjoint, that is, there exists a functor ${ }^{A}{ }^{A}: \mathcal{C} \rightarrow \mathcal{C}$ and a bijection

$$
\Lambda_{B, C}: \mathcal{C}(C \times A, B) \stackrel{\cong}{\cong} \mathcal{C}\left(C, B^{A}\right)
$$

natural in $B, C$. In that case, $\mathrm{ev}_{A, B}:=\Lambda^{-1}\left(\mathrm{id}_{B^{A}}\right)$.

Exercise 75 Derive - ${ }^{A}$ and $\Lambda^{-1}$ of the above description from ev and $\Lambda$ of definition 74 .

Exercise 76 Show that $\mathcal{C}$ has exponentials iff, for every $A, B, C \in \mathrm{Ob}(\mathcal{C})$, there is an object $B^{A}$ and a bijection

$$
\theta_{C}: \mathcal{C}(C \times A, B) \xrightarrow{\cong} \mathcal{C}\left(C, B^{A}\right)
$$

natural in $C$.
Notation 77 The notation $B^{A}$ for exponential objects is standard in the category theory literature. For our purposes, however, it will be more convenient to write $A \Rightarrow B$.

Exponentials bring us to another fundamental notion, this time for understanding functional types, models of $\lambda$-calculus, and the structure of proofs.

Definition 78 A category with a terminal object, products and exponentials is called a Cartesian Closed Category (CCC).

For example, Set is a CCC. Another class of examples are Boolean algebras, seen as categories:

- Products are given by conjunctions $A \wedge B$. We define exponentials as $i m$ plications:

$$
A \Rightarrow B:=\neg A \vee B
$$

- Evaluation is just Modus Ponens,

$$
(A \Rightarrow B) \wedge A \leq B
$$

while couniversality is the Deduction Theorem,

$$
C \wedge A \leq B \Longleftrightarrow C \leq A \Rightarrow B
$$

### 1.5.5 Exercises

1. Suppose that $U: \mathcal{C} \rightarrow \mathcal{D}$ has a left adjoint $F_{1}$, and $V: \mathcal{D} \rightarrow \mathcal{E}$ has a left adjoint $F_{2}$. Show that $V \circ U: \mathcal{C} \rightarrow \mathcal{E}$ has a left adjoint.
2. A sup-lattice is a poset $P$ in which every subset $S \subseteq P$ has a supremum (least upper bound) $\bigvee S$. Let $P, Q$ be sup-lattices, and $f: P \rightarrow Q$ be a monotone map.
a. Show that if $f$ has a right adjoint then $f$ preserves least upper bounds:

$$
f(\bigvee S)=\bigvee\{f(x) \mid x \in S\}
$$

b. Show that if $f$ preserves least upper bounds then it has a right adjoint $g$, given by:

$$
g(y)=\bigvee\{x \in P \mid f(x) \leq y\}
$$

c. Dualise to get a necessary and sufficient condition for the existence of left adjoints.
3. Let $F: \mathcal{C} \rightarrow \mathcal{D}, G: \mathcal{D} \rightarrow \mathcal{C}$ be functors such that $F$ is left adjoint to $G$, with natural bijection $\theta_{C, D}: \mathcal{C}(C, G D) \xrightarrow{\cong} \mathcal{D}(F C, D)$. Show that there is a natural transformation $\varepsilon: F \circ G \rightarrow \mathbf{I d}_{\mathcal{D}}$, the counit of the adjunction. Describe this counit explicitly in the case where the right adjoint is the forgetful functor $U:$ Mon $\rightarrow$ Set.
4. Let $F: \mathcal{C} \rightarrow \mathcal{D}$ and $G: \mathcal{D} \rightarrow \mathcal{C}$ be functors, and assume $F$ is left adjoint to $G$ with natural bijection $\theta$.
a. Show that $F$ preserves epimorphisms.
b. Show that $F$ is faithful if and only if, for every object $A$ of $\mathcal{C}, \eta_{A}: A \rightarrow$ $G F(A)$ is monic.
c. Show that if, for each object $A$ of $\mathcal{C}$, there is a morphism $s_{A}: G F(A) \rightarrow$ $A$ such that $\eta_{A} \circ s_{A}=\operatorname{id}_{G F(A)}$ then $F$ is full.

### 1.6 The Curry-Howard Correspondence

We shall now study a beautiful three-way connection between logic, computation and categories:


Table 1.2: The Curry-Howard Correspondence.
This connection has been known since the 1970's, and is widely used in Computer Science - it is also beginning to be used in Quantum Informatics! It is the upper link (Logic-Computation) that is usually attributed to Haskell B. Curry and William A. Howard, although the idea is related to the operational interpretation of intuitionistic logic given in various formulations by Brouwer, Heyting and Kolmogorov. The link to Categories is mainly due to the pioneering work of Joachim Lambek.

### 1.6.1 Logic

Suppose we ask ourselves the question: What is Logic about? There are two main kinds of answer: one focuses on Truth, and the other on Proof. We focus on the latter, that is, on:

> What follows from what

Traditional introductions to logic focus on Hilbert-style proof systems, that is, on generating the set of theorems of a system from a set of axioms by applying rules of inference (e.g. Modus Ponens).

A key step in logic took place in the 1930's with the advent of Gentzenstyle systems. Instead of focusing on theorems, we look more generally and symmetrically at What follows from what: in these systems the primary focus is on proofs from assumptions. We will examine two such kinds of systems: Natural Deduction systems and Gentzen sequent calculi.

Definition 79 Consider the fragment of propositional logic with logical connectives $\wedge$ and $\supset$. The assertion that a formula $A$ can be proved from assumptions $A_{1}, \ldots, A_{n}$ is expressed by a sequent:

$$
A_{1}, \ldots, A_{n} \vdash A
$$

We use $\Gamma, \Delta$ to range over finite sets of formulas, and write $\Gamma, A$ for $\Gamma \cup\{A\}$. Proofs are built using the proof rules of table 1.3, the resulting proof system is called the Natural Deduction system for $\wedge, \supset$.

| Identity | Conjunction | Implication |
| :---: | :---: | :---: |
| $\frac{\Gamma, A \vdash A}{}$ Id | $\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge$ intro | $\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B} \supset$ intro |
|  | $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge \operatorname{elim}_{1}$ | $\frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B} \supset$ elim |
|  | $\frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge \operatorname{elim}_{2}$ |  |

Table 1.3: Natural Deduction System for $\wedge, \supset$.
For example, we have the following proof of $\supset$-transitivity.
$\frac{\overline{A \supset B, B \supset C, A \vdash B \supset C} \text { Id } \frac{\overline{A \supset B, B \supset C, A \vdash A \supset B}}{} \text { Id } \overline{A \supset B, B \supset C, A \vdash A}}{} \mathrm{Id}$

An important feature of Natural Deduction is the systematic pattern it exhibits in the structure of the inference rules. For each connective $\square$ there are introduction rules, which show how formulas $A \square B$ can be derived, and elimination rules, which show how such formulas can be used to derive other formulas.

Admissibility
We say that a proof rule

$$
\frac{\Gamma_{1} \vdash A_{1} \quad \cdots \quad \Gamma_{n} \vdash A_{n}}{\Delta \vdash B}
$$

is admissible in Natural Deduction if, whenever there are proofs of $\Gamma_{i} \vdash A_{i}$ then there is also a proof of $\Delta \vdash B$. For example, the following Cut rule is admissible.

$$
\frac{\Gamma \vdash A \quad A, \Delta \vdash B}{\Gamma, \Delta \vdash B} \mathrm{Cut}
$$

Exercise 80 Show that the following rules are admissible in Natural Deduction.

1. The Weakening rule:

$$
\frac{\Gamma \vdash B}{\Gamma, A \vdash B}
$$

2. The Cut rule.

Our focus will be on Structural Proof Theory, that is studying the "space of formal proofs" as a mathematical structure in its own right, rather than focussing only on

$$
\text { Provability } \longleftrightarrow \text { Truth }
$$

(i.e. the usual notions of "soundness and completeness"). One motivation for this approach comes from trying to understand and use the computational content of proofs, epitomised in the "Curry-Howard correspondence".

### 1.6.2 Computation

Our starting point in computation is the pure calculus of functions called the $\lambda$-calculus.

Definition 81 Assume a countably infinite set of variables, ranged over by $x, y, z$ and variants. $\lambda$-calculus terms, ranged over by $t, u, v$ etc, are constructed from the following inductive definition.

- Every variable $x$ is a term.
- If $t$ and $u$ are terms, then $t u$ is a term (application).
- If $x$ is a variable and $t$ is a term, then $\lambda x$.t is a term ( $\lambda$-abstraction). $\mathbf{\Delta}$

The above definition can be given in the following compact form, which will be followed in similar definitions in the sequel.

$$
\begin{aligned}
& \text { VA } \ni x, y, z, \ldots \\
& \text { TE } \ni t, u, v \quad::=x|t u| \lambda x . t
\end{aligned}
$$

The computational content of the calculus is exhibited in the following examples. Note that the first example is not part of our formal syntax: it presupposes some encoding of numerals and successors.

$$
\begin{array}{ll}
\lambda x \cdot x+1 & \text { successor function } \\
\lambda x \cdot x & \text { identity function } \\
\lambda f \cdot \lambda x \cdot f x & \text { application } \\
\lambda f \cdot \lambda x \cdot f(f x) & \text { double application } \\
\lambda f \cdot \lambda g \cdot \lambda x \cdot g(f(x)) & \text { composition and application }
\end{array}
$$

Table 1.4: Examples of $\lambda$-terms.
What we also note above is the use of parentheses in order to disambiguate the structure of terms (i.e. the precedence of term constructors). To avoid notational clutter we also use the following conventions.

- Applications associate to the left. For example, $f x y$ stands for $(f x) y$.
- The scope of an abstractions goes as far to the right as possible. For example,

$$
\lambda f .(\lambda x . f(x x)) \lambda x . f(x x) \text { stands for } \lambda f .((\lambda x .(f(x x)))(\lambda x .(f(x x)))) .
$$

The free variables of a term are those that are not bound by any $\lambda$; they can be seen as the assumptions of the term.

Definition 82 The set of free variables of a term $t, \operatorname{fv}(t)$, is given by:

$$
\begin{aligned}
\mathrm{fv}(x) & :=\{x\}, \\
\mathrm{fv}(t u) & :=\mathrm{fv}(t) \cup \mathrm{fv}(u), \\
\mathrm{fv}(\lambda x . t) & :=\mathrm{fv}(t) \backslash\{x\} .
\end{aligned}
$$

The notation $\lambda x . t$ is meant to serve the purpose of expressing formally

$$
\text { the function that returns } t \text { on input } x \text {. }
$$

Thus, $\lambda$ is a binder, that is, it binds the variable $x$ in the 'function' $\lambda x . t$, in the same way that e.g. $\int$ binds $x$ in $\int f(x) d x$. This means that there
should be no difference between $\lambda x . t$ and $\lambda x^{\prime} . t^{\prime}$, where $t^{\prime}$ is obtained from $t$ by swapping $x$ with some fresh variable $x^{\prime}$ (i.e. with some $x^{\prime}$ not appearing free in $t$ ). For example, the terms

$$
\lambda x . x \text { and } \lambda x^{\prime} . x^{\prime}
$$

should be 'equal', as they both stand for the identity function. We formalise this by stipulating that

$$
\text { Terms are identified up to } \alpha \text {-equivalence }
$$

where we say that two terms are $\alpha$-equivalent iff they differ solely in the choice of variables appearing in binding positions. This is formally defined in two steps, as follows.

Definition 83 We define variable-swapping on terms recursively as follows.

$$
\begin{aligned}
(y x) \cdot z & := \begin{cases}y & \text { if } z=x \\
x & \text { if } z=y \\
z & \text { otherwise }\end{cases} \\
(y x) \cdot t u & :=((y x) \cdot t)((y x) \cdot u) \\
(y x) \cdot \lambda z . t & :=\lambda((y x) \cdot z) \cdot((y x) \cdot t)
\end{aligned}
$$

Then, $\alpha$-equivalence, $={ }_{\alpha}$, is the relation on terms defined inductively by $\sqrt{4}$

- $x={ }_{\alpha} x$,
- $t u={ }_{\alpha} t^{\prime} u^{\prime}$ if $t={ }_{\alpha} t^{\prime}$ and $u={ }_{\alpha} u^{\prime}$,
- $\lambda x . t={ }_{\alpha} \lambda x^{\prime} . t^{\prime}$ if, for all $y$ not appearing in $t t^{\prime},(y x) \cdot t={ }_{\alpha}\left(y x^{\prime}\right) \cdot t^{\prime}$.

Equating terms modulo $\alpha$-equivalence means that we work with $\mathrm{TE} /={ }_{\alpha}$ instead of TE. Henceforth, we will refer to elements of TE/ $={ }_{\alpha}$ as terms, and to elements of TE as raw terms. Note that $\alpha$-equivalence is meaningful only on raw terms.

Exercise 84 Prove the following $\alpha$-equivalences.

$$
\lambda x \cdot x={ }_{\alpha} \lambda y \cdot y, \quad \lambda x \cdot \lambda y \cdot x y={ }_{\alpha} \lambda y \cdot \lambda x \cdot y x, \quad x(\lambda x \cdot x)={ }_{\alpha} x(\lambda y \cdot y) .
$$

Exercise 85 Show that, for all raw terms $t, t^{\prime}$ and variables $x, x^{\prime}$, if $t={ }_{\alpha} t^{\prime}$ then $\mathrm{fv}(t)=\mathrm{fv}\left(t^{\prime}\right)$ and $\left(x x^{\prime}\right) \cdot t={ }_{\alpha}\left(x x^{\prime}\right) \cdot t^{\prime}$.
${ }^{4}$ The last clause can be replaced by any of the following:

- ....if, for some $y$ not appearing in $t t^{\prime},(y x) \bullet t=\alpha\left(y x^{\prime}\right) \cdot t^{\prime}$.
- ...if, for all $y$ not appearing free in $t t^{\prime},(y x) \bullet t={ }_{\alpha}\left(y x^{\prime}\right) \bullet t^{\prime}$.
- ...if, for some $y$ not appearing free in $t t^{\prime},(y x) \bullet t={ }_{\alpha}\left(y x^{\prime}\right) \bullet t^{\prime}$.

Moreover, show that, for any $x, x^{\prime} \notin \mathrm{fv}(t), t={ }_{\alpha}\left(x x^{\prime}\right) \cdot t$. Hence infer that, for any $x^{\prime} \notin \mathrm{fv}(t), \quad \lambda x . t={ }_{\alpha} \lambda x^{\prime} .\left(x x^{\prime}\right) \cdot t$.
From the above exercise we obtain that fv and variable-swapping extend to terms (i.e. to $\mathrm{TE} /={ }_{\alpha}$ ) in a straightforward manner. Moreover, we have that, for any term $t$ and any $x^{\prime} \notin \mathrm{fv}(t)$,

$$
\lambda x . t=\lambda x^{\prime} .\left(\begin{array}{ll}
x & x^{\prime}
\end{array}\right) \cdot t .
$$

Since $\lambda$-abstractions stand for functions, an application of a $\lambda$-abstraction on another term should result to a substitution of the latter inside the body of the abstraction.

Definition 86 Define the substitution of a term $t$ for a variable $x$ inside a term inductively by:

$$
\begin{align*}
y[t / x] & := \begin{cases}t & \text { if } y=x \\
y & \text { if } y \neq x\end{cases} \\
(u v)[t / x] & :=(u[t / x])(v[t / x]) \\
(\lambda z . u)[t / x] & :=\lambda z \cdot(u[t / x]) \tag{*}
\end{align*}
$$

where $(*)$ indicates the condition that $z \notin \mathrm{fv}(x t)$.
Note that, due to identification of $\alpha$-equivalent (raw) terms, it is always possible to rename bound variables so that condition $(*)$ be satisfied: for example,

$$
(\lambda z \cdot z x)[z / x]=(\lambda y \cdot y x)[z / x]=\lambda y \cdot y z
$$

Exercise 87 Show that, for all $\lambda$-terms $u, t, t^{\prime}$ and variables $x, x^{\prime}$ such that $x^{\prime} \notin \mathrm{fv}(u) \backslash\{x\}$,

$$
u[t / x]\left[t^{\prime} / x^{\prime}\right]=u\left[\left(t\left[t^{\prime} / x^{\prime}\right]\right) / x\right] .
$$

We proceed to the definition of $\beta$-reduction and $\beta$-conversion. These are relations defined on pairs of terms and express the computational content of the calculus.

Definition 88 We take $\beta$-reduction,$\longrightarrow_{\beta}$, to be the relation defined by:

$$
(\lambda x . t) u \longrightarrow_{\beta} t[u / x]
$$

This extends to arbitrary terms as follows. If $t \longrightarrow_{\beta} t^{\prime}$ then:

$$
t u \longrightarrow_{\beta} t^{\prime} u, \quad u t \longrightarrow_{\beta} u t^{\prime}, \quad \lambda x . t \longrightarrow_{\beta} \lambda x . t^{\prime} .
$$

We take $\beta$-conversion, $={ }_{\beta}$, to be the symmetric reflexive transitive closure of $\beta$-reduction, that is, the equivalence relation induced by:

$$
(\lambda x . t) u={ }_{\beta} t[u / x] .
$$

With $\beta$-reduction we obtain a notion of "computational dynamics". For example:

$$
\begin{aligned}
(\lambda f \cdot f(f y))(\lambda x \cdot x+1) & \longrightarrow_{\beta}(\lambda x \cdot x+1)((\lambda x \cdot x+1) y) \\
& \longrightarrow_{\beta}((\lambda x \cdot x+1) y)+1 \longrightarrow_{\beta}(y+1)+1 \\
(\lambda f \cdot f(f y))(\lambda x \cdot x+1) & \longrightarrow_{\beta}(\lambda x \cdot x+1)((\lambda x \cdot x+1) y) \\
& \longrightarrow_{\beta}(\lambda x \cdot x+1)(y+1) \longrightarrow_{\beta}(y+1)+1
\end{aligned}
$$

Note that in the sequel we will usually write $\beta$-reduction simply by " $\longrightarrow$ ".

### 1.6.3 Simply-Typed $\lambda$-calculus

The 'pure' $\lambda$-calculus we have discussed so far is very unconstrained. For example, it allows self-application, i.e. terms like $x x$ are perfectly legal. On the one hand, this means that the calculus very expressive: for example, we can encode recursion by setting

$$
\mathbf{Y}:=\lambda f \cdot(\lambda x . f(x x)) \lambda x . f(x x)
$$

We have:

$$
\mathbf{Y} t \longrightarrow(\lambda x \cdot t(x x)) \lambda x \cdot t(x x) \longrightarrow t((\lambda x \cdot t(x x)) \lambda x \cdot t(x x)) \longleftarrow t(\mathbf{Y} t)
$$

However, self-application leads also to divergences. The most characteristic example is the following. Setting $\Omega:=(\lambda x . x x) \lambda x . x x$, we have:

$$
\Omega \longrightarrow \Omega \longrightarrow \Omega \longrightarrow \cdots
$$

Historically, Curry extracted Y from an analysis of Russell's Paradox, so it should come as no surprise that it too leads to divergences: setting $t^{\prime}$ to be $\lambda x . t(x x)$,

$$
\mathbf{Y} t \longrightarrow t^{\prime} t^{\prime} \longrightarrow t\left(t^{\prime} t^{\prime}\right) \longrightarrow t\left(t\left(t^{\prime} t^{\prime}\right)\right) \longrightarrow \cdots
$$

The solution is to introduce types. The original idea, due to Church following Russell, was that:

$$
\begin{array}{|l|l|}
\hline \text { Types are there to stop you doing bad things } \\
\hline
\end{array}
$$

However, it has turned out that types constitute one of the most fruitful positive ideas in Computer Science, and provide one of the key disciplines of programming.

Definition 89 Let us assume a set of base types, ranged over by $b$. The simply-typed $\lambda$-calculus is defined as follows.

$$
\begin{array}{ll}
\text { Type } & \mathrm{TY} \ni T, U::=b|T \rightarrow U| T \times U \\
\text { Term } & \mathrm{TE} \ni t, u::=x|t u| \lambda x . t|\langle t, u\rangle| \pi_{1} u \mid \pi_{2} u \\
\text { Typing context } & \Gamma::=\varnothing \mid x: T, \Gamma \quad(x \text { does not appear in } \Gamma)
\end{array}
$$

A typing judgement is a triple of the form

$$
\Gamma \vdash t: T,
$$

which is to be understood as the assertion that term $t$ has the type $T$ under the assumptions that $x_{1}$ has type $T_{1}, \ldots, x_{k}$ has type $T_{k}$, if $\Gamma=x_{1}$ : $T_{1}, \ldots, x_{k}: T_{k}$. A typed term is a term $t$ accompanied with a type $T$ and a context $\Gamma$, such that the judgement $\Gamma \vdash t: T$ is derivable by use of the typing rules of table 1.5

| Variable | Product | Function |
| :---: | :---: | :---: |
| $\Gamma, x: T \vdash x: T$ | $\frac{\Gamma \vdash t: T \quad \Gamma \vdash u: U}{\Gamma \vdash\langle t, u\rangle: T \times U}$ | $\frac{\Gamma, x: U \vdash t: T}{\Gamma \vdash \lambda x \cdot t: U \rightarrow T}$ |
|  | $\frac{\Gamma \vdash v: T \times U}{\Gamma \vdash \pi_{1} v: T}$ | $\frac{\Gamma \vdash t: U \rightarrow T}{} \quad \Gamma \vdash u: U$ |
|  | $\frac{\Gamma \vdash v: T \times U}{\Gamma \vdash \pi_{2} v: U}$ |  |
|  |  |  |

Table 1.5: Simply-Typed $\lambda$-calculus for $\times, \rightarrow$.
Note that contexts are sets, and so $x: T, \Gamma$ stands for $\{x: T\} \cup \Gamma$ with $x$ not appearing in $\Gamma$. As before, terms are identified up to $\alpha$-equivalence.

From the definition of types we see that the simply-typed $\lambda$-calculus is a calculus of functions and products. For example:

$$
\begin{array}{cl}
b \rightarrow b \rightarrow b & \text { first-order function type } \\
(b \rightarrow b) \rightarrow b & \text { second-order function type }
\end{array}
$$

Exercise 90 Can you type the following terms?

$$
\lambda x . x x, \quad \lambda f .(\lambda x . f(x x))(\lambda x . f(x x)) .
$$

Exercise 91 (Weakening \& Cut) Show that Weakening and Cut are admissible in the typing system of the simply-typed $\lambda$-calculus:

$$
\frac{\Gamma \vdash t: T}{\Gamma, x: U \vdash t: T} \text { Weak } \quad \frac{\Gamma \vdash t: T \quad \Gamma, x: T \vdash u: U}{\Gamma \vdash u[t / x]: U} \text { Cut }
$$

We proceed to the rules for reduction and conversion. These are given as in the untyped case, with the addition of $\eta$-rules, which are essentially extensionality principles.

Definition 92 We define $\beta$-reduction, $\longrightarrow_{\beta}$, by the following rules, and let $\beta$-conversion, $={ }_{\beta}$, be its symmetric reflexive transitive closure.

$$
\begin{aligned}
& (\lambda x . t) u \longrightarrow_{\beta} t[u / x] \\
& \pi_{1}\langle t, u\rangle \longrightarrow_{\beta} t \\
& \pi_{2}\langle t, u\rangle \longrightarrow_{\beta} u
\end{aligned}
$$

Moreover, $\eta$-conversion, $=\eta$, is the symmetric reflexive transitive relation obtained by the following rules,

$$
\begin{array}{ll}
t={ }_{\eta} \lambda x \cdot t x & \\
v \notin \mathrm{fv}(t), \text { at function types } \\
v={ }_{\eta}\left\langle\pi_{1} v, \pi_{2} v\right\rangle & \text { at product types }
\end{array}
$$

and $\lambda$-conversion, $={ }_{\lambda}$, is the transitive closure of $={ }_{\beta} \cup=_{\eta}$.
Implicit in the above definition is the fact that $\eta$-rules relate typed terms. For example, $t={ }_{\eta} \lambda x . t x$ has as side condition that $t$ be of function type, i.e. that $t$ be a typed term $\Gamma \vdash t: T \rightarrow U$. Now, following our intuitive interpretation of arrows as functions, we can read this $\eta$-rule as:

$$
t \text { is the function returning } t(x) \text { to every input } x
$$

Note that the above statement is in fact the couniversal property of currying in Set; we will see more on this in the next sections!

Exercise 93 (Subject Reduction) Show that, for any typed term $\Gamma \vdash t$ : $T$, if $t \longrightarrow_{\beta} t^{\prime}$ then $\Gamma \vdash t^{\prime}: T$ is derivable.

## Strong Normalisation

Term reduction results in a normal form: an explicit but much longer expression in which no more reductions are applicable. Formally, a $\lambda$-term is called a redex if it is in one of forms of the left-hand-side of the $\beta$-reduction rules, and therefore $\beta$-reduction can be applied to it. A term is in normal form if it contains no redexes. In the light of the correspondence presented in the next paragraph, a term in normal form corresponds to a proof in which all lemmas have been eliminated.

Fact 94 (SN) For every term $t$, there is no infinite sequence of $\beta$-reductions:

$$
t \longrightarrow t_{0} \longrightarrow t_{1} \longrightarrow t_{2} \longrightarrow \cdots
$$

The above result states that every reduction sequence leads eventually to a term in normal form. Note, though, that reduction to normal form has enormous (non-elementary) complexity.

The correspondence between Logic and Computation
Comparing the following two systems,
Natural Deduction System for $\wedge, \supset$ vs Simply-Typed $\lambda$-calculus for $\times, \rightarrow$
we notice that if we equate

$$
\begin{aligned}
& \wedge \equiv \times \\
& \supset \equiv \rightarrow
\end{aligned}
$$

then they are the same! This is the Logic-Computation part of the CurryHoward correspondence (sometimes: "Curry-Howard isomorphism"). It works on three levels:

| Natural Deduction System | Simply-Typed $\lambda$-calculus |
| :--- | :--- |
| Formulas | Types |
| Proofs | Terms |
| Proof transformations | Term reductions |

Table 1.6: Correspondence between Logic and Computation.
The view of proofs as containing computational content can also be detected in the Brouwer-Heyting-Kolmogorov interpretation of intuitionistic logic:

- A proof of an implication $A \supset B$ is a procedure which transforms any proof of $A$ into a proof of $B$.
- A proof of $A \wedge B$ is a pair consisting of a proof of $A$ and a proof of $B$.

These readings motivate identifying $A \wedge B$ with $A \times B$, and $A \supset B$ with $A \rightarrow B$. Moreover, these ideas have strong connections to computing. The $\lambda$-calculus is a 'pure' version of functional programming languages such as Haskell and SML. So we get a reading of:

> Proofs as Programs

### 1.6.4 Categories

We now have our link between Logic and Computation. We now proceed to complete the triangle of the Curry-Howard correspondence by showing the connection to Categories.

We establish the link from Logic (and Computation) to Categories. Let $\mathcal{C}$ be a cartesian closed category. We shall interpret formulas (or types) as objects of $\mathcal{C}$. A morphism $f: A \rightarrow B$ will then correspond to a proof of $B$ from assumption $A$, i.e. a proof of $A \vdash B$ (a typed term $x: A \vdash t: B$ ). Note that the bare structure of a category only supports proofs from a single assumption. Since $\mathcal{C}$ has finite products, a proof of

$$
A_{1}, \ldots, A_{k} \vdash A
$$

will correspond to a morphism

$$
f: A_{1} \times \cdots \times A_{k} \longrightarrow A
$$

The correspondence is depicted in the next table.

| Axiom | $\overline{\Gamma, A \vdash A} \mathrm{Id}$ | $\overline{\pi_{2}: \Gamma \times A \longrightarrow A}$ |
| :---: | :---: | :---: |
| Conjunction | $\begin{gathered} \frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B} \wedge \mathrm{I} \\ \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A} \wedge \mathrm{E}_{1} \\ \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B} \wedge \mathrm{E}_{2} \end{gathered}$ | $\begin{gathered} f: \Gamma \longrightarrow A \quad g: \Gamma \longrightarrow B \\ \langle f, g\rangle: \Gamma \longrightarrow A \times B \\ \frac{f: \Gamma \longrightarrow A \times B}{\pi_{1} \circ f: \Gamma \longrightarrow A} \\ \frac{f: \Gamma \longrightarrow A \times B}{\pi_{2} \circ f: \Gamma \longrightarrow B} \end{gathered}$ |
| Implication | $\begin{gathered} \frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B} \supset \mathrm{\imath} \\ \frac{\Gamma \vdash A \supset B \quad \Gamma \vdash A}{\Gamma \vdash B} \supset \mathrm{E} \end{gathered}$ | $\begin{gathered} \frac{f: \Gamma \times A \longrightarrow B}{\Lambda(f): \Gamma \longrightarrow(A \Rightarrow B)} \\ \frac{f: \Gamma \longrightarrow(A \Rightarrow B) \quad g: \Gamma \rightarrow A}{\mathrm{ev}_{A, B} \circ\langle f, g\rangle: \Gamma \longrightarrow B} \end{gathered}$ |

Table 1.7: Correspondence between Logic and Categories.
Moreover, the rules for $\beta$ - and $\eta$-conversion are all then derivable from the equations of cartesian closed categories. So cartesian closed categories are models of $\wedge, \supset$-logic at the level of proofs and proof-transformations, and of simply typed $\lambda$-calculus at the level of terms and term-conversions. The connection to computation is examined in more detail below.

Remark 95 In our translation of Logic sequents there is an implicit ordering of assumptions: a set of assumptions is mapped to an assumption product,

$$
\left\{A_{1}, \ldots, A_{n}\right\} \longmapsto A_{1} \times \cdots \times A_{n}
$$

In practice, since for any permutation $A_{1}^{\prime}, \ldots, A_{n}^{\prime}$ of $A_{1}, \ldots, A_{n}$ we have

$$
A_{1} \times \cdots \times A_{n} \cong A_{1}^{\prime} \times \cdots \times A_{n}^{\prime}
$$

such an ordering is harmless.

### 1.6.5 Categorical Semantics of Simply-Typed $\lambda$-calculus

We translate the simply-typed $\lambda$-calculus into a cartesian closed category $\mathcal{C}$, so that to each typed term $x_{1}: T_{1}, \ldots, x_{k}: T_{k} \vdash t: T$ corresponds an arrow

$$
\llbracket t \rrbracket: \llbracket T_{1} \rrbracket \times \cdots \times \llbracket T_{k} \rrbracket \longrightarrow \llbracket T \rrbracket .
$$

The translation if given by the function 【_】defined below ("semantic brackets").

Definition 96 (Semantic translation) Let $\mathcal{C}$ be a CCC and suppose we are given an assignment of an object $\tilde{b}$ to each base type $b$. Then, the translation is defined recursively on types by:

$$
\llbracket b \rrbracket:=\tilde{b}, \quad \llbracket T \times U \rrbracket:=\llbracket T \rrbracket \times \llbracket U \rrbracket, \quad \llbracket T \rightarrow U \rrbracket:=\llbracket T \rrbracket \Rightarrow \llbracket U \rrbracket,
$$

and on typed terms by:

$$
\begin{gathered}
\stackrel{\llbracket \Gamma, x: T \vdash x: T \rrbracket:=\pi_{2}: \llbracket \Gamma \rrbracket \times \llbracket T \rrbracket \longrightarrow \llbracket T \rrbracket}{ } \\
\frac{\llbracket \Gamma \vdash t: T \times U \rrbracket=f: \llbracket \Gamma \rrbracket \longrightarrow \llbracket T \rrbracket \times \llbracket U \rrbracket}{\llbracket \Gamma \vdash \pi_{1} t: T \rrbracket:=\llbracket \Gamma \rrbracket \xrightarrow{f} \llbracket T \rrbracket \times \llbracket U \rrbracket \xrightarrow{\pi_{1}} \llbracket T \rrbracket} \\
\llbracket \Gamma \vdash t: T \rrbracket=f: \llbracket \Gamma \rrbracket \longrightarrow \llbracket T \rrbracket \quad \llbracket \Gamma \vdash u: U \rrbracket=g: \llbracket \Gamma \rrbracket \longrightarrow \llbracket U \rrbracket \\
\llbracket \Gamma \vdash\langle t, u\rangle: T \times U \rrbracket:=\llbracket \Gamma \rrbracket \stackrel{\langle f, g\rangle}{\longrightarrow T \rrbracket \times \llbracket U \rrbracket} \\
\frac{\llbracket \Gamma, x: T \vdash t: U \rrbracket=f: \llbracket \Gamma \rrbracket \times \llbracket T \rrbracket \longrightarrow \llbracket U \rrbracket}{\llbracket \Gamma \vdash \lambda x \cdot t: T \rightarrow U \rrbracket:=\Lambda(f): \llbracket \Gamma \rrbracket \longrightarrow(\llbracket T \rrbracket \Rightarrow \llbracket U \rrbracket)} \\
\frac{\llbracket \Gamma \vdash t: T \rightarrow U \rrbracket=f}{\llbracket \Gamma \vdash t u: U \rrbracket:=\llbracket \Gamma \rrbracket \stackrel{\langle f, g\rangle}{\longrightarrow}(\llbracket T \rrbracket \Rightarrow \llbracket U \rrbracket) \times \llbracket T \rrbracket \xrightarrow{\text { ev }}[\llbracket U \rrbracket}
\end{gathered}
$$

Our aim now is to verify that $\lambda$-conversion (induced by $\beta$ - and $\eta$-rules) is preserved by the translation, i.e. that, for any $t, u$,

$$
t={ }_{\lambda} u \Longrightarrow \llbracket t \rrbracket=\llbracket u \rrbracket .
$$

This would mean that our categorical semantics is sound.
Let us recall some structures from CCC's. Given $f_{1}: D_{1} \rightarrow E_{1}, f_{2}: D_{2} \rightarrow$ $E_{2}$, we defined

$$
f_{1} \times f_{2}=\left\langle f_{1} \circ \pi_{1}, f_{2} \circ \pi_{2}\right\rangle: D_{1} \times D_{2} \longrightarrow E_{1} \times E_{2},
$$

and we showed that $\left(f_{1} \times f_{2}\right) \circ\left\langle h_{1}, h_{2}\right\rangle=\left\langle f_{1} \circ h_{1}, f_{2} \circ h_{2}\right\rangle$. Moreover, exponentials are given by the following natural bijection.

$$
\frac{f: D \times E \longrightarrow F}{\Lambda(f): D \longrightarrow(E \Rightarrow F)}
$$

Equivalently, recall the basic equation:

$$
\mathrm{ev} \circ\left(\Lambda(f) \times \mathrm{id}_{E}\right)=f
$$

where $\Lambda(f)$ is the unique arrow $D \rightarrow(E \Rightarrow F)$ satisfying this equation, with uniqueness being specified by:

$$
\forall h: D \longrightarrow(E \Rightarrow F) . \Lambda\left(\mathrm{ev} \circ\left(h \times \mathrm{id}_{E}\right)\right)=h
$$

Naturality of $\Lambda$ is then proven as follows.
Proposition 97 For any $f: A \times B \rightarrow C$ and $g: A^{\prime} \rightarrow A$,

$$
\Lambda(f) \circ g=\Lambda\left(f \circ\left(g \times \mathrm{id}_{B}\right)\right)
$$

## Proof:

$$
\begin{aligned}
\Lambda(f) \circ g & =\Lambda\left(\mathrm{ev} \circ\left((\Lambda(f) \circ g) \times \operatorname{id}_{B}\right)\right) \\
& \left.=\Lambda\left(\mathrm{ev} \circ\left(\Lambda(f) \times \mathrm{id}_{B}\right) \circ\left(g \times \mathrm{id}_{B}\right)\right)\right)=\Lambda\left(f \circ\left(g \times \operatorname{id}_{B}\right)\right)
\end{aligned}
$$

Substitution Lemma
We consider a simultaneous substitution for all the free variables in a term.

Definition 98 Let $\Gamma=x_{1}: T_{1}, \ldots, x_{k}: T_{k}$. Given typed terms

$$
\Gamma \vdash t: T \quad \text { and } \quad \Gamma \vdash t_{i}: T_{i}, 1 \leq i \leq k
$$

we define $t[\mathbf{t} / \mathbf{x}] \equiv t\left[t_{1} / x_{1}, \ldots, t_{k} / x_{k}\right]$ recursively by:

$$
\begin{aligned}
x_{i}[\mathbf{t} / \mathbf{x}] & :=t_{i} \\
\left(\pi_{i} t\right)[\mathbf{t} / \mathbf{x}] & :=\pi_{i}(t[\mathbf{t} / \mathbf{x}]) \\
\langle t, u\rangle[\mathbf{t} / \mathbf{x}] & :=\langle t[\mathbf{t} / \mathbf{x}], u[\mathbf{t} / \mathbf{x}]\rangle \\
(t u)[\mathbf{t} / \mathbf{x}] & :=(t[\mathbf{t} / \mathbf{x}])(u[\mathbf{t} / \mathbf{x}]) \\
(\lambda x . t)[\mathbf{t} / \mathbf{x}] & :=\lambda x . t[\mathbf{t}, x / \mathbf{x}, x] .
\end{aligned}
$$

Note that, in contrast to ordinary substitution, simultaneous substitution can be defined directly on raw terms, that is, prior to equating them modulo $\alpha$-equivalence. Moreover, we can show that:

$$
t\left[t_{1} / x_{1}, \ldots, t_{k} / x_{k}\right]=t\left[t_{1} / x_{1}\right] \cdots\left[t_{k} / x_{k}\right]
$$

We can now show the following Substitution Lemma.
Proposition 99 For $t, t_{1}, \ldots, t_{k}$ as in the previous definition,

$$
\llbracket t\left[t_{1} / x_{1}, \ldots, t_{k} / x_{k}\right] \rrbracket=\llbracket t \rrbracket \circ\left\langle\llbracket t_{1} \rrbracket, \ldots, \llbracket t_{k} \rrbracket\right\rangle .
$$

Proof: By induction on the structure of $t$.
(1) If $t=x_{i}$ :

$$
\llbracket x_{i}\left[\mathbf{t} / \mathbf{x} \rrbracket \rrbracket=\llbracket t_{i} \rrbracket=\pi_{i} \circ\left\langle\llbracket t_{1} \rrbracket, \ldots, \llbracket t_{k} \rrbracket\right\rangle=\llbracket x_{i} \rrbracket \circ\left\langle\llbracket t_{1} \rrbracket, \ldots, \llbracket t_{k} \rrbracket\right\rangle .\right.
$$

(2) If $t=u v$ then, abbreviating $\left\langle\llbracket t_{1} \rrbracket, \ldots, \llbracket t_{k} \rrbracket\right\rangle$ to $\langle\llbracket \mathbf{t} \rrbracket\rangle$ we have:

$$
\begin{aligned}
\llbracket u v[\mathbf{t} / \mathbf{x}] \rrbracket & =\llbracket(u[\mathbf{t} / \mathbf{x}])(v[\mathbf{t} / \mathbf{x}]) \rrbracket & & \text { Defn of substitution } \\
& =\mathrm{ev} \circ\langle\llbracket u[\mathbf{t} / \mathbf{x}] \rrbracket, \llbracket v\lceil\mathbf{t} / \mathbf{x}] \rrbracket\rangle & & \text { Defn of semantic function } \\
& =\mathrm{ev} \circ\langle\llbracket u \rrbracket \circ\langle\llbracket \mathbf{t} \rrbracket\rangle, \llbracket v \rrbracket \circ\langle\llbracket \mathbf{t} \rrbracket\rangle\rangle & & \text { Induction hyp. } \\
& =\mathrm{ev} \circ\langle\llbracket u \rrbracket, \llbracket v \rrbracket\rangle \circ\langle\llbracket \mathbf{t} \rrbracket\rangle & & \text { Property of products } \\
& =\llbracket u v \rrbracket \circ\langle\llbracket \mathbf{t} \rrbracket\rangle & & \text { Defn of semantic function }
\end{aligned}
$$

(3) If $t=\lambda x . u$ :

$$
\begin{aligned}
\llbracket \lambda x . u[\mathbf{t} / \mathbf{x}] \rrbracket & =\llbracket \lambda x .(u[\mathbf{t}, x / \mathbf{x}, x]) \rrbracket & & \text { Defn. of substitution } \\
& =\Lambda(\llbracket u[\mathbf{t}, x / \mathbf{x}, x] \rrbracket) & & \text { Defn. of semantic function } \\
& =\Lambda(\llbracket u \rrbracket \circ(\langle\llbracket \mathbf{t} \rrbracket\rangle \times \mathrm{id})) & & \text { Induction hyp. } \\
& =\Lambda(\llbracket u \rrbracket) \circ\langle\llbracket \mathbf{t} \rrbracket\rangle & & \text { Prop. } 97 \\
& =\llbracket \lambda x . u \rrbracket \circ\langle\llbracket \mathbf{t} \rrbracket\rangle & & \text { Defn. of semantic function }
\end{aligned}
$$

$(4,5)$ The cases of projections and pairs are left as exercise.
Exercise 100 Complete the proof of the above proposition.

Validating the conversion rules
We can now show that the conversion rules of the $\lambda$-calculus are preserved by the translation, and hence the interpretation is sound. Observe the correspondence between $\eta$-rules and uniqueness (couniversality) principles.

- For $\beta$-conversion: $\quad\left[(\lambda x . t) u=t[u / x], \pi_{1}\langle t, u\rangle=t, \pi_{2}\langle t, u\rangle=u\right]$

$$
\begin{array}{rlr}
\llbracket(\lambda x, t) u \rrbracket & =\mathrm{ev} \circ\langle\Lambda(\llbracket t \rrbracket), \llbracket u \rrbracket\rangle & \text { Defn. of semantics } \\
& =\mathrm{ev} \circ(\Lambda(\llbracket t \rrbracket) \times \mathrm{id}) \circ\left\langle\mathrm{id}_{\llbracket \Gamma \rrbracket}, \llbracket u \rrbracket\right\rangle & \text { Property of } \times \\
& =\llbracket t \rrbracket \circ\left\langle\mathrm{id}_{\llbracket \Gamma \rrbracket}, \llbracket u \rrbracket\right\rangle & \text { Defn. of } \Lambda \\
& =\llbracket t[\mathbf{x}, u / \mathbf{x}, x \rrbracket \rrbracket & \text { Substitution lemma. } \\
\llbracket \pi_{1}\langle t, u\rangle \rrbracket & =\pi_{1} \circ \llbracket\langle t, u\rangle \rrbracket=\pi_{1} \circ\langle\llbracket t \rrbracket, \llbracket u \rrbracket\rangle=\llbracket t \rrbracket . &
\end{array}
$$

- For $\eta$-conversion: $\quad\left[t=\lambda x . t x,\left\langle\pi_{1} t, \pi_{2} t\right\rangle=t\right]$

$$
\begin{array}{rlrl}
\llbracket \lambda x \cdot t x \rrbracket & =\Lambda(\mathrm{ev} \circ(\llbracket t \rrbracket \times \mathrm{id})) & =\llbracket t \rrbracket & \\
\text { Uniqueness equation }(\Rightarrow) \\
\llbracket\left\langle\pi_{1} t, \pi_{2} t\right\rangle \rrbracket & =\left\langle\pi_{1} \circ \llbracket t \rrbracket, \pi_{2} \circ \llbracket t \rrbracket\right\rangle & =\llbracket t \rrbracket & \\
\text { Uniqueness equation }(\times)
\end{array}
$$

### 1.6.6 Completeness?

It is the case that, in a general $\operatorname{CCC} \mathcal{C}$, there may be equalities which are not reflected by the semantic translation, i.e.

$$
\llbracket t \rrbracket=\llbracket u \rrbracket \quad \text { yet } \quad t \neq{ }_{\lambda} u .
$$

In the rest of this section, we show how to construct a $\operatorname{CCC} \mathcal{C}_{\lambda}$ in which equalities between arrows correspond precisely to $\lambda$-conversions between terms. We call $\mathcal{C}_{\lambda}$ a term model, due to its dependence on the syntax.

Definition 101 We define a family of relations on variable-term pairs by setting $(x, t) \sim_{T, U}(y, u)$ if $x: T \vdash t: U$ and $y: T \vdash u: U$ are derivable and

$$
t={ }_{\lambda} u[x / y] .
$$

These are equivalence relations, so we set:

$$
[(x, t)]_{T, U}:=\left\{(y, u) \mid(x, t) \sim_{T, U}(y, u)\right\}
$$

Similarly, $(., t) \sim_{., U}(., u)$ if $\vdash t: U$ and $\vdash u: U$ are derivable and $t={ }_{\lambda} u$. Moreover,

$$
[(\cdot, t)]_{., U}:=\left\{(\cdot, u) \mid(\cdot, t) \sim_{., U}(\cdot, u)\right\}
$$

We denote $[(x, t)]_{T, U}$ simply by $[x, t]$, and $[(\cdot, t)] ., U$ simply by $[., t]$ (these are not to be confused with copairings!). We proceed with $\mathcal{C}_{\lambda}$.

Definition 102 The category $\mathcal{C}_{\lambda}$ is defined as follows. We take as set of objects the set of $\lambda$-types augmented with a terminal object $\mathbf{1}$ :

$$
O b\left(\mathcal{C}_{\lambda}\right):=\{\mathbf{1}\} \cup\{\widetilde{T} \mid T \text { a } \lambda \text {-type }\}
$$

The homsets of $\mathcal{C}_{\lambda}$ contain equivalence relations on typed terms (definition (101), or terminal arrows $\tau$ :

$$
\begin{aligned}
\mathcal{C}_{\lambda}(\widetilde{T}, \widetilde{U}) & :=\{[x, t] \mid x: T \vdash t: U \text { is derivable }\} \\
\mathcal{C}_{\lambda}(\mathbf{1}, \widetilde{U}): & :=\{[\cdot, t] \mid \vdash t: U \text { is derivable }\} \\
\mathcal{C}_{\lambda}(A, \mathbf{1}) & :=\left\{\tau_{A}\right\}
\end{aligned}
$$

The identities are:

$$
\mathrm{id}_{\widetilde{T}}:=[x, x], \quad \mathrm{id}_{\mathbf{1}}:=\tau_{\mathbf{1}},
$$

and arrow composition is defined by:

$$
\begin{aligned}
{[x, t] \circ[y, u] } & :=[y, t[u / x]] \\
{[x, t] \circ[\cdot, u] } & :=[\cdot, t[u / x]] \\
{[\cdot, t] \circ \tau_{A} } & := \begin{cases}{[y, t]} & \text { if } A=\widetilde{U} \\
{[\cdot, t]} & \text { if } A=\mathbf{1}\end{cases} \\
\tau_{B} \circ h & :=\tau_{A} \quad\left(h \in \mathcal{C}_{\lambda}(A, B)\right)
\end{aligned}
$$

Note that, for each variable $x^{\prime}$, any arrow $[x, t]: \widetilde{T} \rightarrow \widetilde{U}$ can be written in the form $\left[x^{\prime}, t^{\prime}\right]$, since $t=\left(t\left[x^{\prime} / x\right]\right)\left[x / x^{\prime}\right]$ and therefore $[x, t]=\left[x^{\prime}, t\left[x^{\prime} / x\right]\right]$.

Proposition $103 \mathcal{C}_{\lambda}$ is a category.
Proof: It is not difficult to see that id's are identities. For associativity, we show the most interesting case (and leave the rest as an exercise):

$$
\begin{aligned}
& {[x, t] \circ([y, u] \circ[z, v])=[x, t] \circ[z, u[v / y]]=[z, t[(u[v / y]) / x]],} \\
& ([x, t] \circ[y, u]) \circ[z, v]=[y, t[u / x]] \circ[z, v]=[z, t[u / x][v / y]] .
\end{aligned}
$$

By exercise 87, the above are equal.
Proposition $104 \mathcal{C}_{\lambda}$ has finite products.
Proof: Clearly, $\mathbf{1}$ is terminal with canonical arrows $\tau_{A}: A \rightarrow \mathbf{1}$. For (binary) products, $\mathbf{1} \times A=A \times \mathbf{1}=A$. Otherwise, define $\widetilde{T} \stackrel{\pi_{1}}{\leftrightarrows} \widetilde{T} \times \widetilde{U} \xrightarrow{\pi_{2}} \widetilde{U}$ by:

$$
\begin{aligned}
\widetilde{T} \times \widetilde{U} & :=\widetilde{T \times U} \\
\pi_{i} & :=\left[x, \pi_{i} x\right] \quad i=1,2 .
\end{aligned}
$$

Given $\widetilde{T} \stackrel{[x, t]}{\rightleftarrows} \widetilde{V} \xrightarrow{[x, u]} \widetilde{U}$, take $\langle[x, t],[x, u]\rangle: \widetilde{V} \rightarrow \widetilde{T} \times \widetilde{U}:=[x,\langle t, u\rangle]$. Then:

$$
\begin{aligned}
\pi_{1} \circ\langle[x, t],[x, u]\rangle & =\left[y, \pi_{1} y\right] \circ[x,\langle t, u\rangle] & & \text { Definitions } \\
& =\left[x, \pi_{1}\langle t, u\rangle\right] & & \text { Defn of composition } \\
& =[x, t] & & \beta \text {-conversion }
\end{aligned}
$$



Proposition $105 \mathcal{C}_{\lambda}$ has exponentials.
Proof: We have that $\mathbf{1} \Rightarrow A=A$ and $A \Rightarrow \mathbf{1}=\mathbf{1}$, with obvious evaluation arrows. Otherwise,

$$
\begin{aligned}
& \widetilde{U} \Rightarrow \widetilde{T}:=\widetilde{U \rightarrow T} \\
& \mathrm{ev}_{\widetilde{U}, \widetilde{T}}:(\widetilde{U} \Rightarrow \widetilde{T}) \times \widetilde{U} \longrightarrow \widetilde{T}:=\left[x,\left(\pi_{1} x\right)\left(\pi_{2} x\right)\right]
\end{aligned}
$$

Given $[x, t]: \widetilde{V} \times \widetilde{U} \rightarrow \widetilde{T}$, take $\quad \Lambda([x, t]):=\left[x_{1}, \lambda x_{2} \cdot t\left[\left\langle x_{1}, x_{2}\right\rangle / x\right]\right]$. Then,

$$
\begin{aligned}
\mathrm{ev} \circ \Lambda([x, t]) \times \mathrm{id} & =\mathrm{ev} \circ\left\langle\Lambda([x, t]) \circ \pi_{1}, \text { id } \circ \pi_{2}\right\rangle \\
& =\mathrm{ev} \circ\left\langle\left[x_{1}, \lambda x_{2} \cdot t\left[\left\langle x_{1}, x_{2}\right\rangle / x\right]\right] \circ\left[y, \pi_{1} y\right],\left[y, \pi_{2} y\right]\right\rangle \\
& =\mathrm{ev} \circ\left\langle\left[y, \lambda x_{2} \cdot t\left[\left\langle\pi_{1} y, x_{2}\right\rangle / x\right]\right],\left[y, \pi_{2} y\right]\right\rangle \\
& =\left[z,\left(\pi_{1} z\right)\left(\pi_{2} z\right)\right] \circ\left[y,\left\langle\lambda x_{2} \cdot t\left[\left\langle\pi_{1} y, x_{2}\right\rangle / x\right], \pi_{2} y\right\rangle\right] \\
& =\left[y,\left(\pi_{1} u\right)\left(\pi_{2} u\right)\right] \stackrel{\beta}{=}\left[y,\left(\lambda x_{2} \cdot t\left[\left\langle\pi_{1} y, x_{2}\right\rangle / x\right]\right)\left(\pi_{2} y\right)\right] \\
& \stackrel{\beta}{=}\left[y, t\left[\left\langle\pi_{1} y, \pi_{2} y\right\rangle / x\right]\right] \stackrel{\eta}{=}[y, t[y / x]]=[x, t] .
\end{aligned}
$$

Uniqueness left as exercise. The case of $[x, t]: \mathbf{1} \times \widetilde{U} \rightarrow \widetilde{T}$ is similar.
Exercise 106 Complete the proof of the previous propositions.
Hence, $\mathcal{C}_{\lambda}$ is a CCC and a sound model of the simply-typed $\lambda$-calculus. Moreover, applying our translation from the $\lambda$-calculus to a CCC (definition 96) we can show that we have

$$
\llbracket \Gamma \vdash t: T \rrbracket=\left[x, t\left[\pi_{i} x / x_{i}\right]_{i=1 . . n}\right]
$$

where $\Gamma=\left\{x_{1}: T_{1}, \ldots, x_{n}: T_{n}\right\}, x \notin \Gamma$ and $x: \prod_{i=1}^{n} T_{i}$. Then,

$$
t={ }_{\lambda} u \Longleftrightarrow \llbracket \Gamma \vdash t: T \rrbracket=\llbracket \Gamma \vdash u: T \rrbracket .
$$

This means that our term model is complete.

### 1.6.7 Exercises

1. Give Natural Deduction proofs of the following sequents.

- $\vdash(A \supset B) \supset((B \supset C) \supset(A \supset C))$
- $\vdash(A \supset(A \supset B)) \supset(A \supset B)$
- $\vdash(C \supset A) \supset((C \supset B) \supset(C \supset(A \wedge B)))$
- $\vdash(A \supset(B \supset C)) \supset((A \supset B) \supset(A \supset C))$

In each case, give the corresponding $\lambda$-term and the corresponding arrow in a $\operatorname{CCC} \mathcal{C}$.
2. For each of the following $\lambda$-terms, find a type for it. Try to find the 'most general' type, built from 'type variables' $\alpha, \beta$ etc. For example, the most general type for the identity $\lambda x . x$ is $\alpha \rightarrow \alpha$. In each case, give the derivation of the type for this term (where you may assume that types can be built up from type variables as well as base types).

- $\lambda f . \lambda x . f x$
- $\lambda x . \lambda y . \lambda z \cdot x(y z)$
- $\lambda x . \lambda y \cdot \lambda z \cdot x z y$
- $\lambda x \cdot \lambda y \cdot x y y$
- $\lambda x . \lambda y \cdot x$
- $\lambda x \cdot \lambda y \cdot \lambda z \cdot x z(y z)$

Reflect a little on the methods you used to do this exercise. Could they be made algorithmic?

### 1.7 Linearity

In the system of Natural Deduction, implicit in our treatment of assumptions in sequents

$$
A_{1}, \ldots, A_{n} \vdash A
$$

is that we can use them as many times as we want (including not at all). In this section we will explore the field that is opened once we apply restrictions to this approach, and thus render our treatment of assumptions more linear (or resource sensitive).

### 1.7.1 Gentzen Sequent Calculus

In order to make the manipulation of assumptions more visible, we now represent the assumptions as a list (possibly with repetitions) rather than a set, and use explicit structural rules to control copying, deletion and interchange of assumptions.

Definition 107 The structural rules for Logic are given in the following table.

| $\frac{\Gamma \vdash A}{} \mathrm{Id}$ | $\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}$ Exch |
| :---: | :---: |
| $\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}$ Contr | $\frac{\Gamma \vdash B}{\Gamma, A \vdash B}$ Weak |

Table 1.8: Structural Rules for Logic.
If we think of using proof rules backwards to reduce the task of proving a given sequent to various sub-tasks, then we see that the Contraction rule lets us duplicate premises, and the Weakening rule lets us discard them, while the Exchange rule merely lets us re-order them. The Identity axiom as given here is equivalent to the one with auxiliary premises given previously in the presence of Weakening.

The structural rules have clear categorical meanings in a category $\mathcal{C}$ with products. Recalling the diagonal transformation $\Delta_{A}:=\left\langle\mathrm{id}_{A}, \mathrm{id}_{A}\right\rangle$ and the symmetry transformation $s_{A, B}:=\left\langle\pi_{2}, \pi_{1}\right\rangle$, the meanings are as follows.

$$
\begin{array}{cc}
\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C} \text { Exch } & f: \Gamma \times A \times B \times \Delta \longrightarrow C \\
\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B} \text { Contr } & \left.\frac{f: \Gamma \times A \times A \longrightarrow B}{} \times s_{A, B} \times \mathrm{id}_{\Delta}\right): \Gamma \times B \times A \times \Delta \longrightarrow C \\
\frac{\Gamma \vdash B}{\Gamma, A \vdash B} \text { Weak } & \left.\frac{f: \Gamma \longrightarrow B}{}+\frac{f \circ \pi_{1}: \Gamma \times A \longrightarrow B}{}\right): \Gamma \times A \longrightarrow B
\end{array}
$$

In order to analyse Natural Deduction, Gentzen introduced sequent calculi based on Left and Right rules, instead of Introduction and Elimination rules. These kind of systems are more adequate for our discussion on linearity.

Definition 108 We define the Gentzen sequent calculus for $\wedge, \supset$ as the proof system obtained by the structural rules (def. 107) and the rules in table 1.9 for connectives.

| Conjunction | Implication | Cut |
| :---: | :---: | :---: |
| $\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B} \wedge \mathrm{R}$ | $\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B} \supset \mathrm{R}$ | $\frac{\Gamma \vdash A B A, \Delta \vdash B}{\Gamma, \Delta \vdash B} \mathrm{Cut}$ |
| $\frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C} \wedge \mathrm{~L}$ | $\frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, A \supset B, \Delta \vdash C} \supset \mathrm{~L}$ |  |

Table 1.9: Gentzen Sequent Calculus for $\wedge, \supset$.
For example, the proof of $\supset$-transitivity is now given as follows.

Exercise 109 Show that the Gentzen-rules are admissible in Natural Deduction. Moreover, show that the Natural Deduction rules are admissible in the Gentzen sequent calculus.

The Cut rule allows the use of lemmas in proofs. It also yields a dynamics of proofs via Cut Elimination, that is, a dynamics of proof transformations towards the goal of eliminating the uses of the Cut rule in a proof, i.e. removing all lemmas and making the proof completely 'explicit', meaning Cut-free. Such transformations are always possible, as is shown in the following seminal result of Gentzen (Hauptsatz).

Fact 110 (Cut Elimination) The Cut rule is admissible in the Gentzen sequent calculus without Cut.

### 1.7.2 Linear Logic

In the presence of the structural rules, the Gentzen sequent calculus is entirely equivalent to the Natural Deduction system we studied earlier. Nevertheless:

[^2] change rule)?

It turns out we can still make good sense of the resulting proofs, terms and categories, but now in the setting of a different, 'resource-sensitive' logic.

Definition 111 Multiplicative Linear logic is a variant of standard logic with linear logical connectives. The multiplicative connectives for conjunction and implication are $\otimes$ and $\multimap$. Proof sequents are of the form $\Gamma \vdash A$, where $\Gamma$ is now a multiset. The proof rules for $\otimes, \multimap-$ Linear Logic, given in table 1.10 , are the multiplicative versions of the Gentzen rules.

| Conjunction | Implication | Cut |
| :---: | :---: | :---: |
| $\frac{\Gamma \vdash A}{\Gamma, \Delta \vdash A \otimes B} \otimes \mathrm{R}$ | $\frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B} \multimap \mathrm{R}$ | $\frac{\Gamma \vdash A}{\Gamma, \Delta \vdash B} A, \Delta \vdash B$ |
| Cut |  |  |
| $\frac{\Gamma, A, B \vdash C}{\Gamma, A \otimes B \vdash C} \otimes \mathrm{~L}$ | $\frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, A \multimap B, \Delta \vdash C} \multimap \mathrm{~L}$ |  |

Table 1.10: Rules for $\otimes,-$-Linear Logic.
Multiplicativity here means the use of disjoint (i.e. non-overlapping) contexts. The use of multisets allows us to omit explicit use of the Exchange rule in our proof system.

Note that the given system satisfies Cut-elimination, and this leans heavily on the $\rightarrow \mathrm{L}$ rule. We could have used instead the following rule,

$$
\frac{\Gamma \vdash A \multimap B \quad \Delta \vdash A}{\Gamma, \Delta \vdash B} \multimap \mathrm{E}
$$

which is more intuitive computationally, but then cut-elimination would fail. Note, though, that:

$$
\multimap \mathrm{L}, \mathrm{Cut}, \mathrm{Id} \equiv \multimap \mathrm{E}, \text { Cut, Id. }
$$

This is shown as follows.

$$
\frac{\Gamma \vdash A \overline{A \multimap B \vdash A \multimap B}}{\frac{\Gamma, A \multimap B \vdash B}{\Gamma} \multimap \mathrm{E} \quad B, \Delta \vdash C} \mathrm{Id}^{\Gamma, A \multimap B, \Delta \vdash C} \operatorname{Cut} \frac{\Gamma \vdash A \multimap B \frac{\Delta \vdash A \quad \overline{B \vdash B}}{A \multimap B, \Delta \vdash B}}{\Gamma, \Delta \vdash B} \text { Id }
$$

The resource-sensitive nature of Linear Logic is reflected in the following exercise.

Exercise 112 Can you construct proofs in Linear Logic of the following sequents? (Hint: Use the Cut Elimination property.)

- $A \vdash A \otimes A$
- $(A \otimes A) \multimap B \vdash A \multimap B$
- $\vdash A \multimap(B \multimap A)$

Related to linear logic is the linear $\lambda$-calculus, which is a linear version of the simply-typed $\lambda$-calculus.

Definition 113 The linear $\lambda$-calculus is defined as follows.
Type $\quad$ TY $\ni T, U::=b|T \multimap U| T \otimes U$
Term $\quad$ TE $\ni t, u::=x|t u| \lambda x . t|t \otimes u|$ let $z$ be $x \otimes y$ in $t$
Typing context $\quad \Gamma::=\varnothing \mid x: T, \Gamma$
Terms are typed by use of the typing rules of table 1.11. Finally, the rules for $\beta$-reduction are:

$$
\begin{aligned}
(\lambda x . t) u & \longrightarrow_{\beta} t[u / x] \\
\text { let } t \otimes u \text { be } x \otimes y \text { in } v & \longrightarrow_{\beta} v[t / x, u / y] .
\end{aligned}
$$

Note here that, again, $x: T, \Gamma$ stands for $\{x: T\} \cup \Gamma$ with $x$ not appearing in $\Gamma$. Note also that Cut-free proofs always yield terms in normal form.

| Variable <br> / Cut | $\frac{\Gamma \vdash T \vdash x: T}{x: T}$ | $\frac{\Gamma \vdash t: T \quad x: T, \Delta \vdash u: U}{\Gamma, \Delta \vdash u[t / x]: U}$ |
| :--- | :---: | :---: |
| Linear <br> Tensor | $\frac{\Gamma \vdash t: T \quad \Delta \vdash u: U}{\Gamma, \Delta \vdash t \otimes u: T \otimes U}$ | $\frac{\Gamma, x: T, y: U \vdash v: V}{\Gamma, z: T \otimes U \vdash \text { let } z \text { be } x \otimes y \text { in } v: V}$ |
| Linear <br> Function | $\frac{\Gamma, x: U \vdash t: T}{\Gamma \vdash \lambda x . t: U \multimap T}$ | $\frac{\Gamma \vdash t: T \quad x: U, \Delta \vdash u: V}{\Gamma, f: T \multimap U, \Delta \vdash u[f t / x]: V}$ |

Table 1.11: Linear $\lambda$-calculus for $\otimes, \multimap$.
Term formation is now highly constrained by the form of the typing judgements. In particular,

$$
x_{1}: A_{1}, \ldots, x_{k}: A_{k} \vdash t: A
$$

now implies that each $x_{i}$ occurs exactly once (free) in $t$.
Moreover, note that, for function application, instead of the rule on the LHS below, we could have used the more intuitive rule on the RHS.

$$
\frac{\Gamma \vdash t: T \quad x: U, \Delta \vdash u: V}{\Gamma, f: T \multimap U, \Delta \vdash u[f t / x]: V} \quad \frac{\Gamma \vdash t: A \multimap B \quad \Delta \vdash u: A}{\Gamma, \Delta \vdash t u: B}
$$

As we did in the logic, we can show that the typing systems with one or the other rule are equivalent.

### 1.7.3 Linear Logic in Monoidal Categories

We proceed to give a categorical counterpart to linearity by providing a categorical interpretation of linear logic. Note that CCC's are no longer adequate for this task as they contain arrows

$$
\Delta_{A}: A \longrightarrow A \times A, \quad \pi_{1}: A \times B \longrightarrow A
$$

which violate linearity. It turns out that the right setting is that of symmetric monoidal closed categories.

Definition 114 A monoidal category is a structure $(\mathcal{C}, \otimes, I, a, l, r)$ where:

- $\mathcal{C}$ is a category,
- $\otimes: \mathcal{C} \times \mathcal{C} \rightarrow \mathcal{C}$ is a functor (tensor),
- $I$ is a distinguished object of $\mathcal{C}$ (unit),
- $a, l, r$ are natural isomorphisms (structural isos) with components:

$$
\begin{aligned}
& a_{A, B, C}: A \otimes(B \otimes C) \xrightarrow{\cong}(A \otimes B) \otimes C \\
& l_{A}: I \otimes A \xrightarrow{\cong} A \quad r_{A}: A \otimes I \xrightarrow{\cong} A
\end{aligned}
$$

such that $l_{I}=r_{I}: I \otimes I \rightarrow I$ and the following diagrams commute.


The monoidal diagrams ensure coherence, described by the slogan:
". . 'all' diagrams involving $a, l$ and $r$ must commute."
Examples:

- Both products and coproducts give rise to monoidal structures - which are the common denominator between them. (But in addition, products
have diagonals and projections, and coproducts have codiagonals and injections.)
- ( $\mathbb{N}, \leq,+, 0)$ is a monoidal category.
- Rel, the category of sets and relations, with cartesian product (which is not the categorical product).
- Vect $_{k}$ with the tensor product.

Let us examine the example of Rel in some detail. We take $\otimes$ to be the cartesian product, which is defined on relations $R: X \rightarrow X^{\prime}$ and $S: Y \rightarrow Y^{\prime}$ as follows.

$$
\forall(x, y) \in X \times Y,\left(x^{\prime}, y^{\prime}\right) \in X^{\prime} \times Y^{\prime} .(x, y) R \otimes S\left(x^{\prime}, y^{\prime}\right) \Longleftrightarrow x R x^{\prime} \wedge y S y^{\prime}
$$

It is not difficult to show that this is indeed a functor. Note that, in the case that $R, S$ are functions, $R \otimes S$ is the same as $R \times S$ in Set. Moreover, we take each $a_{A, B, C}$ to be the associativity function for products (in Set), which is an iso in Set and hence also in Rel. Finally, we take $I$ to be the one-element set, and $l_{A}, r_{A}$ to be the projection functions: their relational converses are their inverses in Rel. The monoidal diagrams commute simply because they commute in Set.

Exercise 115 Verify that $(\mathbb{N}, \leq,+, 0)$ and Vect $_{k}$ are monoidal categories.

Tensors and products
As we mentioned earlier, products are tensors with extra structure: natural diagonals and projections. This fact, which reflects no-cloning and no-deleting of Linear Logic, is shown as follows.

Proposition 116 Let $\mathcal{C}$ be a monoidal category $(\mathcal{C}, \otimes, I, a, l, r)$. $\otimes$ induces a product structure iff there exist natural diagonals and projections, i.e. natural transformations given by arrows

$$
d_{A}: A \longrightarrow A \otimes A, \quad p_{A, B}: A \times B \longrightarrow A, \quad q_{A, B}: A \times B \longrightarrow B
$$

such that the following diagrams commute.


Proof: The "only if" direction is straightforward. For the converse, let $\mathcal{C}$ be monoidal with natural projections and diagonals. Then, we take product
pairs to be pairs of the form

$$
A \stackrel{p_{A, B}}{\rightleftarrows} A \otimes B \xrightarrow{q_{A, B}} B .
$$

Moreover, for any pair of arrows $B \stackrel{f}{\longleftarrow} A \xrightarrow{g} C$, define

$$
\langle f, g\rangle:=A \xrightarrow{d_{A}} A \otimes A \xrightarrow{f \otimes g} B \otimes C .
$$

Then the product diagram commutes. For example:


For uniqueness, if $h: A \rightarrow B \otimes C$ then the following diagram commutes,

(1) naturality of $d$
(2) hypothesis
so $h=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle$.

## SMCC's

Linear Logic is interpreted in monoidal categories with two more pieces of structure: monoidal symmetry and closure. The former allows the Exchange rule to be interpreted, while the latter realises linear implication.

Definition 117 A symmetric monoidal category is a monoidal category $(\mathcal{C}, \otimes, I, a, l, r)$ with an additional natural isomorphism (symmetry),

$$
s_{A, B}: A \otimes B \stackrel{\cong}{\cong} B \otimes A
$$

such that $s_{B, A}=s_{A, B}^{-1}$ and the following diagrams commute.


Definition 118 A symmetric monoidal closed category (SMCC) is a symmetric monoidal category $(\mathcal{C}, \otimes, I, a, l, r, s)$ such that, for each object $A$, there is a couniversal arrow to the functor

$$
-\otimes A: \mathcal{C} \longrightarrow \mathcal{C}
$$

That is, for all pairs $A, B$, there is an object $A \multimap B$ and a morphism

$$
\mathrm{ev}_{A, B}:(A \multimap B) \otimes A \longrightarrow B
$$

such that, for every morphism $f: C \otimes A \rightarrow B$, there is a unique morphism $\Lambda(f): C \rightarrow(A \multimap B)$ such that

$$
\mathrm{ev}_{A, B} \circ\left(\Lambda(f) \otimes \mathrm{id}_{A}\right)=f
$$

Note that, although we use notation borrowed from CCC's (ev, $\Lambda$ ), these are different structures! Examples of symmetric monoidal closed categories are Rel, Vect ${ }_{k}$, and ( a fortiori) cartesian closed categories.

Exercise 119 Show that Rel is a symmetric monoidal closed category.

Linear logic in SMCC's
Just as cartesian closed categories correspond to $\wedge$, - -logic (and simplytyped $\lambda$-calculus), so do symmetric monoidal closed categories correspond to $\otimes, \multimap-l o g i c ~(a n d ~ l i n e a r ~ \lambda-c a l c u l u s) . ~$

So let $\mathcal{C}$ be a symmetric monoidal closed category. The interpretation of a linear sequent

$$
A_{1}, \ldots, A_{k} \vdash A
$$

will be a morphism

$$
f: A_{1} \otimes \cdots \otimes A_{k} \longrightarrow A
$$

To be precise in our interpretation, we will again treat contexts as lists of formulas, and explicitly interpret the Exchange rule by:

$$
\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C} \quad \frac{f: \Gamma \otimes A \otimes B \otimes \Delta \longrightarrow C}{f \circ\left(\operatorname{id}_{\Gamma} \otimes s_{B, A} \otimes \mathrm{id}_{\Delta}\right): \Gamma \otimes B \otimes A \otimes \Delta \longrightarrow C}
$$

The rest of the rules are translated as follows.

| $\overline{A \vdash A}$ | $\overline{\mathrm{id}_{A}: A \longrightarrow A}$ |
| :---: | :---: |
| $\frac{\Gamma \vdash A \quad A, \Delta \vdash B}{\Gamma, \Delta \vdash B}$ | $\frac{f: \Gamma \longrightarrow A}{} \frac{}{g \circ\left(f \otimes \mathrm{id}_{\Delta}\right): \Gamma \otimes \Delta \longrightarrow B}$ |
| $\begin{gathered} \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B} \\ \frac{\Gamma, A, B \vdash C}{\Gamma, A \otimes B \vdash C} \end{gathered}$ | $\begin{gathered} \frac{f: \Gamma \longrightarrow A \quad g: \Delta \longrightarrow B}{f \otimes g: \Gamma \otimes \Delta \longrightarrow A \otimes B} \\ f:(\Gamma \otimes A) \otimes B \longrightarrow C \\ f \circ a_{\Gamma, A, B}: \Gamma \otimes(A \otimes B) \longrightarrow C \end{gathered}$ |
| $\begin{gathered} \frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B} \\ \frac{\Gamma \vdash A \multimap B \quad \Delta \vdash A}{\Gamma, \Delta \vdash B} \end{gathered}$ | $\begin{gathered} \frac{f: \Gamma \otimes A \longrightarrow B}{\Lambda(f): \Gamma \longrightarrow(A \multimap B)} \\ \frac{f: \Gamma \longrightarrow(A \multimap B) \quad g: \Delta \longrightarrow A}{\mathrm{ev}_{A, B} \circ(f \otimes g): \Gamma \otimes \Delta \longrightarrow B} \end{gathered}$ |

Table 1.12: Categorical Translation of $\otimes, \multimap$-Linear Logic.
Note that, because of coherence in monoidal categories, we will not be scholastic with associativity arrows $a$ in our translations and will usually omit them. For the same reason, consecutive applications of tensor will be written without specifying associativity, e.g. $A_{1} \otimes \cdots \otimes A_{n}$.

Exercise 120 Let $\mathcal{C}$ be a symmetric monoidal closed category. Give the interpretation of the - -left rule in $\mathcal{C}$ :

$$
\frac{\Gamma \vdash A \quad B, \Delta \vdash C}{\Gamma, A \multimap B, \Delta \vdash C} \multimap \mathrm{~L}
$$

Exercise 121 Is it possible to translate $\otimes, \multimap-\operatorname{logic}$ into a $\operatorname{CCC} \mathcal{C}$ ? Is this in accordance with linearity of $\otimes,-$-logic?

### 1.7.4 Beyond the Multiplicatives

Linear Logic has three 'levels' of connectives, each describing a different aspect of standard logic:

- The multiplicatives: e.g. $\otimes, \multimap$,
- The additives: additive conjunction \& and disjunction $\oplus$,
- The exponentials, allowing controlled access to copying and discarding.

We focus on additive conjunction and the exponential "!", which will allow us to recover the 'expressive power' of standard $\wedge, \supset$-logic.

Definition 122 The logical connective for additive disjunction is \&, and the related proof rules are the following.

$$
\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \& B} \& \mathrm{R} \quad \frac{\Gamma, A \vdash C}{\Gamma, A \& B \vdash C} \& \mathrm{~L} \quad \frac{\Gamma, B \vdash C}{\Gamma, A \& B \vdash C} \& \mathrm{~L}
$$

So additive conjunction has proof rules that are identical to those of standard conjunction ( $\wedge$ ). Note though that, since by linearity an argument of type $A \& B$ can only be used once, each use of a left rule for $\&$ makes a once-and-for-all choice of a projection. On the other hand, $A \otimes B$ represents a conjunction where both projections must be available.

Additive conjunction can be interpreted in any symmetric monoidal category with products, i.e. a category $\mathcal{C}$ with structure $(\otimes, \times)$ where $\otimes$ is a symmetric monoidal tensor and $\times$ is a product.

$$
\frac{f: \Gamma \longrightarrow A \quad g: \Gamma \longrightarrow B}{\langle f, g\rangle: \Gamma \longrightarrow A \times B} \quad \frac{f: \Gamma \otimes A \longrightarrow C}{f \circ\left(\mathrm{id} \otimes \pi_{1}\right): \Gamma \otimes(A \times B) \longrightarrow C}
$$

Moreover, we can extend the linear $\lambda$-calculus with term constructors for additive conjunction as follows.

$$
\begin{aligned}
\frac{\Gamma \vdash t: A \quad \Gamma \vdash u: B}{\Gamma \vdash\langle t, u\rangle: A \& B} & \frac{\Gamma, x: A \vdash t: C}{\Gamma, z: A \& B \vdash \operatorname{let} z=\langle x,-\rangle \operatorname{in} t: C} \\
& \frac{\Gamma, B \vdash C}{\Gamma, z: A \& B \vdash \operatorname{let} z=\langle-, y\rangle \operatorname{in} t: C}
\end{aligned}
$$

The $\beta$-reduction rules related to these constructs are:

$$
\begin{aligned}
& \text { let }\langle t, u\rangle=\langle x,-\rangle \text { in } v \longrightarrow_{\beta} v[t / x] \\
& \text { let }\langle t, u\rangle=\langle-, y\rangle \text { in } v \longrightarrow_{\beta} v[u / y] .
\end{aligned}
$$

Finally, we can gain back the lost structural rules, in disciplined versions, by introducing an exponential bang operator! which is a kind of modality enabling formulas to participate in structural rules.

Definition 123 The logical connective for bang is !, and the related proof rules are the following.

$$
\frac{\Gamma, A \vdash B}{\Gamma,!A \vdash B}!\mathrm{L} \quad \frac{!\Gamma \vdash A}{!\Gamma \vdash!A}!\mathrm{R} \quad \frac{\Gamma \vdash B}{\Gamma,!A \vdash B} \text { Weak } \quad \frac{\Gamma,!A,!A \vdash B}{\Gamma,!A \vdash B} \text { Contr }
$$

Note that $!\left\{A_{1}, \ldots, A_{n}\right\}:=!A_{1}, \ldots,!A_{n}$.
We can now see the discipline imposed on structural rules: in order for the rules to be applied, the participating formulas need to be tagged with a bang.

Interpreting standard Logic
We are now in position to recover the standard logical connectives $\wedge, \supset$ within Linear Logic. If we interpret

$$
\begin{aligned}
& A \supset B:=!A \multimap B \\
& A \wedge B:=A \& B
\end{aligned}
$$

and each $\wedge$, $\supset$-sequent $\Gamma \vdash A$ as $!\Gamma \vdash A$, then each proof rule of the Gentzen system for $\wedge, \supset$ is admissible in the proof system of Linear Logic for $\otimes,-\infty, \&,!$.

Note in particular that the interpretation

$$
A \supset B:=!A \multimap B
$$

decomposes the fundamental notion of implication into finer notions - like 'splitting the atom of logic'!

### 1.7.5 Exercises

1. Give proofs of the following sequents in Linear Logic.
a. $\vdash A \multimap A$
b. $A \multimap B, B \multimap C \vdash A \multimap C$
c. $\vdash(A \multimap B \multimap C) \multimap(B \multimap A \multimap C)$
d. $A \otimes(B \otimes C) \vdash(A \otimes B) \otimes C$
e. $A \otimes B \vdash B \otimes A$

For each of the proofs constructed give:

- the corresponding linear $\lambda$-term,
- its interpretation in Rel.

2. Consider a symmetric monoidal closed category $\mathcal{C}$.
a. Suppose the sequents $\Gamma_{1} \vdash A, \Gamma_{2} \vdash B$ and $A, B, \Delta \vdash C$ are provable and let their interpretations (i.e. the interpretations of their proofs) in $\mathcal{C}$ be $f_{1}: \Gamma_{1} \rightarrow A, f_{2}: \Gamma_{2} \rightarrow B$ and $g: A \otimes B \otimes \Delta \rightarrow C$ respectively. Find then the interpretations $h_{1}, h_{2}$ of the following proofs.

$$
\frac{\frac{\vdots}{\Gamma_{1} \vdash A} \frac{\vdots}{\Gamma_{2} \vdash B}}{\frac{\vdots}{\Gamma_{1}, \Gamma_{2} \vdash A \otimes B} \otimes \mathrm{R}} \frac{\frac{\vdots}{A, B, \Delta \vdash C}}{\Gamma_{1}, \Gamma_{2}, \Delta \vdash C} \otimes \mathrm{~L} \quad \mathrm{Cut} \quad \frac{\frac{\vdots}{\Gamma_{2} \vdash B} \frac{\vdots}{\Gamma_{1} \vdash A} \frac{\vdots}{\Gamma_{1}, B, \Delta \vdash C}}{\Gamma_{1}, \Gamma_{2}, \Delta \vdash C} \mathrm{Cut}
$$

and show that $h_{1}=h_{2}$.
b. Suppose now $\mathcal{C}$ has also binary products, given by $\times$. Given that the sequents $\Gamma \vdash A, \Gamma \vdash B$ and $A, \Delta \vdash C$ are provable, and that their interpretations in $\mathcal{C}$ are $f_{1}: \Gamma \rightarrow A, f_{2}: \Gamma \rightarrow B$ and $g: A \otimes \Delta \rightarrow C$ respectively, find the interpretations $h_{1}, h_{2}$ of the following proofs.

$$
\frac{\frac{\vdots}{\Gamma \vdash A} \frac{\vdots}{\Gamma \vdash B}}{\Gamma \vdash A \& B} \& \mathrm{R} \frac{\frac{\vdots}{A, \Delta \vdash C}}{\Gamma, \Delta \vdash C} \& \mathrm{~L} \quad \mathrm{Cut} \quad \frac{\frac{\vdots}{\Gamma \vdash A} \frac{\vdots}{A, \Delta \vdash C}}{\frac{C u t}{\Gamma, \Delta \vdash C}} \mathrm{Cut}
$$

and show that $h_{1}=h_{2}$.
3. Show that the condition $l_{I}=r_{I}$ in the definition of monoidal categories is redundant.
Moreover, show that the condition $\mathrm{id}_{A} \otimes l_{B}=a_{A, I, B} \circ r_{A} \otimes \mathrm{id}_{B}$ in the definition of symmetric monoidal categories is redundant.

### 1.8 Monads and Comonads

Recall that an adjunction is given by a triple $\langle F, G, \theta\rangle$, with $F: \mathcal{C} \rightarrow \mathcal{D}$ and $G: \mathcal{D} \rightarrow \mathcal{C}$ being functors, and $\theta$ a natural bijection between homsets. By composing the two functors we obtain endofunctors

$$
G \circ F: \mathcal{C} \longrightarrow \mathcal{C}, \quad F \circ G: \mathcal{D} \longrightarrow \mathcal{D} .
$$

These can be seen as encapsulating the effect of the adjunction inside their domain category. For example, if we consider the functors

$$
\text { MList }: \text { Set } \longrightarrow \text { Mon }, \quad U: \text { Mon } \longrightarrow \text { Set }
$$

then $U \circ$ MList encodes the free monoid construction inside Set.
The study of such endofunctors on their own right gave rise to the notions of monad and comonad, which we examine in this section.

### 1.8.1 Basics

Definition 124 A monad over a category $\mathcal{C}$ is a triple $(T, \eta, \mu)$ where $T$ is an endofunctor on $\mathcal{C}$ and $\eta: \mathbf{I d}_{\mathcal{C}} \rightarrow T, \mu: T^{2} \rightarrow T$ are natural transformations such that the following diagrams commute. (Note that $T^{2}:=T \circ T$, etc.).


We call $\eta$ the unit of the monad, and $\mu$ its multiplication; the whole terminology comes from monoids. Let us now proceed to some examples.

- Let $\mathcal{C}$ be a category with coproducts and let $E$ be an object in $\mathcal{C}$. We can define a monad $(T, \eta, \mu)$ of $E$-coproducts (computationally, $E$-exceptions) by taking $T: \mathcal{C} \rightarrow \mathcal{C}$ to be the functor $\_^{+} E$, and $\eta, \mu$ as follows.

$$
\begin{aligned}
T & :=A \mapsto A+E, f \mapsto f+\mathrm{id}_{E} \\
\eta_{A} & :=A \xrightarrow{\mathrm{in}_{1}} A+E \\
\mu_{A} & :=(A+E)+E \xrightarrow{\left[\mathrm{id}_{A+E}, \mathrm{in}_{2}\right]} A+E
\end{aligned}
$$

As an injection, $\eta$ is a natural transformation. For $\mu$, we can use the properties of the coproduct. For $f: A \rightarrow B$,

$$
\begin{aligned}
T f \circ \mu_{A} & =T f \circ\left[\mathrm{id}_{A+E}, \mathrm{in}_{2}\right]=\left[T f \circ \mathrm{id}_{A+E}, T f \circ \mathrm{in}_{2}\right]=\left[T f, T f \circ \mathrm{in}_{2}\right] \\
& =\left[T f,\left(f+\mathrm{id}_{E}\right) \circ \mathrm{in}_{2}\right]=\left[T f, \mathrm{in}_{2}\right] \\
& =\left[\mathrm{id}_{B+E} \circ T f, \mathrm{in}_{2} \circ \mathrm{id}_{E}\right]=\left[\mathrm{id}_{B+E}, \mathrm{in}_{2}\right] \circ\left(T f+\mathrm{id}_{E}\right) \\
& =\mu_{B} \circ T^{2} f .
\end{aligned}
$$

The monadic diagrams follow in a similar manner. For example,

$$
\begin{aligned}
\mu_{A} \circ \mu_{T A} & =\mu_{A} \circ\left[\mathrm{id}_{T A+E}, \mathrm{in}_{2}\right]=\left[\mu_{A} \circ \mathrm{id}_{T A+E}, \mu_{A} \circ \mathrm{in}_{2}\right]=\left[\mu_{A}, \mu_{A} \circ \mathrm{in}_{2}\right] \\
& =\left[\mu_{A},\left[\mathrm{id}_{A+E}, \mathrm{in}_{2}\right] \circ \mathrm{in}_{2}\right]=\left[\mu_{A}, \mathrm{in}_{2}\right] \\
& =\left[\mathrm{id}_{A+E} \circ \mu_{A}, \mathrm{in}_{2} \circ \mathrm{id}_{E}\right]=\left[\mathrm{id}_{A+E}, \mathrm{in}_{2}\right] \circ\left(\mu_{A}+\mathrm{id}_{E}\right) \\
& =\mu_{A} \circ T \mu_{A} .
\end{aligned}
$$

- Now let $\mathcal{C}$ be a cartesian closed category and let $\xi$ be some object in $\mathcal{C}$. We can define a monad of $\xi$-side-effects by taking $T$ to be the functor $\xi \Rightarrow(-\times \xi)$, and $\eta, \mu$ as follows.

$$
\begin{aligned}
T & :=A \mapsto \xi \Rightarrow(A \times \xi), f \mapsto \xi \Rightarrow\left(f \times \mathrm{id}_{\xi}\right) \\
\eta_{A} & :=\Lambda\left(A \times \xi \xrightarrow{\mathrm{id}_{A \times \xi}} A \times \xi\right) \\
\mu_{A} & :=\Lambda\left(T(T A) \times \xi \xrightarrow{\mathrm{ev}_{\xi, T A \times \xi}} T A \times \xi \xrightarrow{\mathrm{ev}_{\xi, A \times \xi}} A \times \xi\right)
\end{aligned}
$$

Naturality of $\eta, \mu$ follows from naturality of $\Lambda$ : for any $f: A \rightarrow A^{\prime}$,

$$
\begin{aligned}
T f \circ \eta_{A} & =\left(\xi \Rightarrow f \times \mathrm{id}_{\xi}\right) \circ \Lambda\left(\mathrm{id}_{A \times \xi}\right)=\Lambda\left(f \times \mathrm{id}_{\xi} \circ \mathrm{id}_{A \times \xi}\right) \\
& =\Lambda\left(\mathrm{id}_{A^{\prime} \times \xi} \circ f \times \mathrm{id}_{\xi}\right)=\Lambda\left(\mathrm{id}_{A^{\prime} \times \xi}\right) \circ f=\eta_{A^{\prime}} \circ f, \\
\mu_{A^{\prime}} \circ T^{2} f & =\Lambda\left(\mathrm{ev}_{\xi, A^{\prime} \times \xi} \circ \mathrm{ev}_{\xi, T A^{\prime} \times \xi}\right) \circ T^{2} f=\Lambda\left(\mathrm{ev}_{\xi, A^{\prime} \times \xi} \circ \mathrm{ev}_{\xi, T A^{\prime} \times \xi} \circ T^{2} f \times \mathrm{id}_{\xi}\right) \\
& =\Lambda\left(\mathrm{ev}_{\xi, A^{\prime} \times \xi} \circ T f \times \mathrm{id}_{\xi} \circ \mathrm{ev}_{\xi, T A \times \xi}\right)=\Lambda\left(f \times \mathrm{id}_{\xi} \circ \mathrm{ev}_{\xi, A \times \xi} \circ \mathrm{ev}_{\xi, T A \times \xi}\right) \\
& =\left(\xi \Rightarrow f \times \mathrm{id}_{\xi}\right) \circ \Lambda\left(\mathrm{ev}_{\xi, A \times \xi} \circ \mathrm{ev}_{\xi, T A \times \xi}\right)=T f \circ \mu_{A} .
\end{aligned}
$$

The monadic diagrams are shown in a similar manner.

- Our third example employs the functor $U:$ Mon $\rightarrow$ Set. In particular, we take $T:=U \circ \mathrm{MList}$ and $\eta, \mu$ as follows.

$$
\left.\left.\begin{array}{rl}
T:=X \mapsto \bigcup_{n \in \omega}\left\{\left[x_{1}, \ldots, x_{n}\right] \mid x_{1}, \ldots, x_{n} \in X\right\} \\
& f \mapsto\left(\left[x_{1}, \ldots, x_{n}\right] \mapsto\left[f\left(x_{1}\right), \ldots, f\left(x_{n}\right)\right]\right) \\
\eta_{X}:= & x \mapsto[x] \\
\mu_{X}:= & {[ }
\end{array} x_{11}, \ldots, x_{1 n_{1}}\right], \ldots,\left[x_{k 1}, \ldots, x_{k n_{k}}\right]\right] \mapsto\left[x_{11}, \ldots, x_{1 n_{1}}, \ldots, x_{k 1}, \ldots, x_{k n_{k}}\right] .
$$

Naturality of $\eta, \mu$ is obvious - besides, $\eta$ is the unit of the corresponding adjunction. The monadic diagrams are also straightforward: they correspond to the following equalities of mappings (we use $\mathbf{x}$ for $x_{1}, \ldots, x_{n}$ ).


Exercise 125 Show that the $E$-coproduct monad and the $\xi$-side-effect monads are indeed monads.

Our discussion on monads can be dualised, leading us to comonads.

Definition 126 A comonad over a category $\mathcal{C}$ is a triple $(Q, \varepsilon, \delta)$ where $Q$ is an endofunctor on $\mathcal{C}$ and $\varepsilon: Q \rightarrow \mathbf{I d}_{\mathcal{C}}, \delta: Q \rightarrow Q^{2}$ are natural transformations such that the following diagrams commute.

$\varepsilon$ is the counit of the comonad, and $\delta$ its comultiplication. Two of our examples from monads dualise to comonads.

- If $\mathcal{C}$ has finite products then, for any object $S$, we can define the $S$-product comonad with functor $Q:=S \times \ldots$.
- We can form a comonad on Mon with functor $Q:=$ MListo $U$ (and counit that of the corresponding adjunction).

Exercise 127 Give an explicit description of the comonad on Mon with functor $Q:=$ MList $\circ U$ described above. Verify it is a comonad.

### 1.8.2 (Co)Monads of an Adjunction

In the previous section, we saw that an adjunction between Mon and Set yielded a monad on Set (and a comonad on Mon), with its unit being the unit of the adjunction. We now show that this observation generalises to any adjunction. Recall that an adjunction is specified by:

- a pair of functors $\mathcal{C} \underset{G}{\stackrel{F}{\rightleftarrows}} \mathcal{D}$,
- for each $A \in O b(\mathcal{C}), B \in O b(\mathcal{D})$, a bijection $\theta_{A, B}: \mathcal{C}(A, G B) \cong \mathcal{D}(F A, B)$ natural in $A, B$.

For such an adjunction we build a monad on $\mathcal{C}$ : the functor of the monad is simply $T:=G \circ F$, and unit and multiplication are defined by setting

$$
\begin{aligned}
& \eta_{A}: A \longrightarrow G F A:=\theta_{A, F A}^{-1}\left(\mathrm{id}_{F A}\right) \\
& \mu_{A}: G F G F A \longrightarrow G F A:=G\left(\theta_{G F A, F A}\left(\mathrm{id}_{G F A}\right)\right)
\end{aligned}
$$

Observe that $\eta$ is the unit of the adjunction.
Proposition 128 Let $(F, G, \eta)$ be an adjunction. Then the triple ( $T, \eta, \mu$ ) defined above is a monad on $\mathcal{C}$.

Proof: Recall that naturality of $\theta$ means concretely that, for any $f: A \rightarrow$ $G B, g: A^{\prime} \rightarrow A$ and $h: B \rightarrow B^{\prime}$,

$$
\theta_{A^{\prime}, B^{\prime}}(G h \circ f \circ g)=h \circ \theta_{A, B}(f) \circ F g
$$

$\eta$ is the unit of the adjunction and hence natural. We show naturality of $\mu$ :

$$
\begin{aligned}
G F G F f \circ \mu_{B} & =G \theta_{G F B, F B}\left(\operatorname{id}_{G F B}\right) \circ G F G F f=G\left(\theta_{G F B, F B}\left(\mathrm{id}_{G F B}\right) \circ F G F f\right) \\
& \stackrel{\text { nat. } \theta}{=} G \theta_{G F A, F B}\left(\operatorname{id}_{G F B} \circ G F f\right)=G \theta_{G F A, F B}\left(G F f \circ \mathrm{id}_{G F A}\right) \\
& \stackrel{\text { nat. } \theta}{=} G\left(F f \circ \theta_{G F A, F A}\left(\mathrm{id}_{G F A}\right)\right)=G F f \circ \mu_{A} .
\end{aligned}
$$

The monoidal condition for $\mu$ also follows from naturality of $\theta$ :

$$
\begin{aligned}
\mu_{A} \circ \mu_{G F A} & =G\left(\theta\left(\operatorname{id}_{G F A}\right) \circ \theta\left(\operatorname{id}_{G F G F A}\right)\right) \stackrel{\text { nat }}{=} G \theta\left(G \theta\left(\operatorname{id}_{G F A}\right) \circ \operatorname{id}_{G F G F A}\right) \\
& =G \theta\left(\operatorname{id}_{G F A} \circ G \theta\left(\operatorname{id}_{G F A}\right)\right) \stackrel{\text { nat }}{=} G\left(\theta\left(\operatorname{id}_{G F A}\right) \circ F G \theta\left(\operatorname{id}_{G F A}\right)\right) \\
& =\mu_{A} \circ G F \mu_{A}
\end{aligned}
$$

Finally, for the $\eta-\mu$ conditions we also use the universality diagram for $\eta$ and the uniqueness property (in equational form).

$$
\begin{aligned}
\mu_{A} \circ \eta_{G F A} & =G \theta_{G F A, F A}\left(\mathrm{id}_{G F A}\right) \circ \eta_{G F A}=\mathrm{id}_{G F A}, \\
\mu_{A} \circ G F \eta_{G F A} & =G \theta_{G F A, F A}\left(\mathrm{id}_{G F A}\right) \circ G F \eta_{G F A}=G\left(\theta_{G F A, F A}\left(\mathrm{id}_{G F A}\right) \circ F \eta_{G F A}\right) \\
& \stackrel{\text { nat }}{=} G \theta\left(\mathrm{id}_{G F A} \circ \eta_{G F A}\right)=G \theta\left(G \mathrm{id}_{F A} \circ \eta_{G F A}\right)=G \mathrm{id}_{F A}=G F A .
\end{aligned}
$$

Hence, every adjunction gives rise to a monad. It turns out that the converse is also true: every monad is described by means of an adjunction in this way. In particular, there are two canonical constructions of adjunctions from a given monad: the Kleisli construction, and the Eilenberg-Moore construction. These are in a sense minimal and maximal solutions to describing a monad via an adjunction. We describe the former one in the next section.

Finally, note that - because of the symmetric definition of adjunctions the whole discussion can be dualised to comonads. That is, every adjunction gives rise to a comonad with counit that of the adjunction, and also every comonad can be derived from an adjunction in this manner.

### 1.8.3 The Kleisli Construction

The Kleisli construction starts from a monad $(T, \eta, \mu)$ on a category $\mathcal{C}$ and builds a category $\mathcal{C}_{T}$ of $T$-computations, as follows.

Definition 129 Let $(T, \eta, \mu)$ be a monad on a category $\mathcal{C}$. Construct the Kleisli category $\mathcal{C}_{T}$ by taking the same objects as $\mathcal{C}$, and by including an arrow $f_{. T}: A \rightarrow B$ in $\mathcal{C}_{T}$ for each $f: A \rightarrow T B$ in $\mathcal{C}$. That is,

$$
\begin{aligned}
\operatorname{Ob}\left(\mathcal{C}_{T}\right) & :=\operatorname{Ob}(\mathcal{C}) \\
\mathcal{C}_{T}(A, B) & :=\left\{f_{. T} \mid f \in \mathcal{C}(A, T B)\right\}
\end{aligned}
$$

The identity arrow for $A$ in $\mathcal{C}_{T}$ is $\eta_{A . T}$, while the composite of $f_{. T}: A \rightarrow B$ and $g_{. T}: B \rightarrow C$ is $h_{. T}$, where:

$$
h:=A \xrightarrow{f} T B \xrightarrow{T g} T^{2} C \xrightarrow{\mu_{C}} T C .
$$

The conditions for $\mathcal{C}_{T}$ being a category follow from the monadic conditions. For composition with identity, for any $f: A \rightarrow T B$,

$$
\begin{aligned}
& f_{. T} \circ \eta_{A . T}=\left(\mu_{B} \circ T f \circ \eta_{A}\right)_{. T}=\left(\mu_{B} \circ \eta_{B} \circ f\right)_{. T}=f_{. T} \\
& \eta_{B . T} \circ f_{. T}=\left(\mu_{B} \circ T \eta_{B} \circ f\right)_{. T}=f_{. T} .
\end{aligned}
$$

For associativity of composition, for any $f: A \rightarrow T B, g: B \rightarrow T C$ and $h: C \rightarrow T D$,

$$
\begin{aligned}
\left(h_{. T} \circ g_{. T}\right) \circ f_{. T} & =\left(\mu_{D} \circ T h \circ g\right)_{. T} \circ f_{. T}=\left(\mu_{D} \circ T\left(\mu_{D} \circ T h \circ g\right) \circ f\right)_{. T} \\
& =\left(\mu_{D} \circ T \mu_{D} \circ T^{2} h \circ T g \circ f\right)_{. T}=\left(\mu_{D} \circ \mu_{T D} \circ T^{2} h \circ T g \circ f\right)_{. T} \\
& =\left(\mu_{D} \circ T h \circ \mu_{C} \circ T g \circ f\right)_{. T}=h_{. T} \circ\left(g_{. T} \circ f_{. T}\right) .
\end{aligned}
$$

Let us now proceed to build the adjunction between $\mathcal{C}$ and $\mathcal{C}_{T}$ that will eventually give us back the monad $T$. Construct the functors $F: \mathcal{C} \rightarrow \mathcal{C}_{T}$ and $G: \mathcal{C}_{T} \rightarrow \mathcal{C}$ as follows.

$$
\begin{aligned}
& F:=A \mapsto A,(f: A \rightarrow B) \mapsto\left(\left(\eta_{B} \circ f\right)_{\cdot T}: A \rightarrow B\right) \\
& G:=A \mapsto T A,\left(f_{. T}: A \rightarrow B\right) \mapsto\left(\mu_{B} \circ T f: T A \rightarrow T B\right) .
\end{aligned}
$$

Functoriality of $F, G$ follows from the monad laws and the definition of $\mathcal{C}_{T}$. Moreover, for each $A, B \in O b(\mathcal{C})$, construct the following bijection of arrows.

$$
\theta_{A, B}: \mathcal{C}(A, T B) \xrightarrow{\cong} \mathcal{C}_{T}(A, B):=f \mapsto f_{\cdot T}
$$

To establish that $(F, G, \theta)$ is an adjunction we need only show that $\theta$ is natural in $A, B$. So take $f: A \rightarrow T B, g: A^{\prime} \rightarrow A$ and $h_{. T}: B \rightarrow B^{\prime}$. We then have:

$$
\begin{aligned}
\theta_{A^{\prime}, B^{\prime}}\left(G\left(h_{\cdot T}\right) \circ f \circ g\right) & =\theta_{A^{\prime}, B^{\prime}}\left(\mu_{B^{\prime}} \circ T h \circ f \circ g\right)=\left(\mu_{B^{\prime}} \circ T h \circ f \circ g\right)_{. T} \\
& =h_{\cdot T} \circ(f \circ g)_{\cdot T}=h_{\cdot T} \circ\left(\mu_{B} \circ T f \circ \eta_{A} \circ g\right)_{. T} \\
& =h_{\cdot T} \circ f_{\cdot T} \circ\left(\eta_{A} \circ g\right)_{. T}=h_{\cdot T} \circ \theta_{A, B}(f) \circ F g .
\end{aligned}
$$

The final step in this section is to verify that the monad $\left(T^{\prime}, \eta^{\prime}, \mu^{\prime}\right)$ arising from this adjunction is the one we started from. The construction of $T^{\prime}$ follows the recipe given in the previous section, that is:

- $T^{\prime}: \mathcal{C} \rightarrow \mathcal{C}:=G \circ F$. Thus, $T^{\prime}$ maps each object $A$ to $T A$, and each arrow $f: A \rightarrow B$ to $\mu_{B} \circ T \eta_{A} \circ T f=T f$.
- $\eta_{A}^{\prime}: A \rightarrow T A:=\theta_{A, F A}^{-1}\left(\mathrm{id}_{F A}^{\left(\mathcal{C}_{T}\right)}\right)=\theta^{-1}\left(\eta_{A . T}\right)=\eta_{A}$.
- $\mu_{A}^{\prime}: T^{2} A \rightarrow T A:=G \theta_{G F A, F A}\left(\mathrm{id}_{G F A}^{(\mathcal{C})}\right)=G \theta\left(\mathrm{id}_{T A}\right)=\mu_{A} \circ T \mathrm{id}_{T A}=\mu_{A}$.

Thus, we have indeed obtained the initial $(T, \eta, \mu)$.

The Kleisli construction on a comonad
Dually to the Kleisli category of a monad we can construct the Kleisli category of a comonad - and reobtain the comonad through an adjunction between the Kleisli category and the original one. Specifically, given a category $\mathcal{C}$ and a comonad $(Q, \varepsilon, \delta)$ on $\mathcal{C}$, we define the category $\mathcal{C}_{Q}$ as follows.

$$
\begin{aligned}
O b\left(\mathcal{C}_{Q}\right) & :=O b(\mathcal{C}) \\
\mathcal{C}_{Q}(A, B) & :=\left\{f_{. Q} \mid f \in \mathcal{C}(Q A, B)\right\} \\
\operatorname{id}_{A}^{\left(\mathcal{C}_{Q}\right)} & :=\varepsilon_{A . Q} \\
g_{. Q} \circ f_{. Q} & :=\left(g \circ Q f \circ \delta_{A}\right) \cdot . Q
\end{aligned}
$$

The Kleisli category of a comonad will be of use in the next sections, where comonads will be considered for modelling bang of Linear Logic. We end this section by showing a result that will be of use then.

Proposition 130 Let $\mathcal{C}$ be a category and $(Q, \varepsilon, \delta)$ be a comonad on $\mathcal{C}$. If $\mathcal{C}$ has binary products then so does $\mathcal{C}_{Q}$.

Proof: Let $A, B$ be objects in $\mathcal{C}, \mathcal{C}_{Q}$. We claim that their product in $\mathcal{C}_{Q}$ is given by $\left(A \times B, p_{1}, p_{2}\right)$, where

$$
p_{1}:=\left(Q(A \times B){\left.\stackrel{\varepsilon}{\rightarrow} A \times B{\xrightarrow{\pi_{1}}} A\right)_{. Q} \text { }}\right.
$$

and similarly for $p_{2}$. Now, for each $f_{. Q}: C \rightarrow A$ and $g_{. Q}: C \rightarrow B$, setting $\left\langle f_{\cdot Q}, g_{. Q}\right\rangle:=\langle f, g\rangle_{. Q}$ we have:

$$
p_{1} \circ\left\langle f_{\cdot Q}, g_{\cdot Q}\right\rangle=\left(\pi_{1} \circ \varepsilon \circ Q\langle f, g\rangle \circ \delta\right)_{. Q}=\left(\pi_{1} \circ\langle f, g\rangle \circ \varepsilon \circ \delta\right)_{\cdot Q}=f_{\cdot Q}
$$

and similarly $p_{2} \circ\left\langle f_{. Q}, g_{. Q}\right\rangle=g_{. Q}$. Finally, for any $h_{. Q}: C \rightarrow A \times B$,

[^3]\[

$$
\begin{aligned}
\left\langle p_{1} \circ h_{. Q}, p_{2} \circ h_{. Q}\right\rangle & =\left\langle\pi_{1} \circ \varepsilon \circ Q h \circ \delta, \pi_{2} \circ \varepsilon \circ Q h \circ \delta\right\rangle_{. Q}=\left\langle\pi_{1} \circ h, \pi_{2} \circ h\right\rangle_{. Q} \\
& =h_{. Q} .
\end{aligned}
$$
\]

Exercise 131 Show that the Kleisli category $\mathcal{C}_{Q}$ of a comonad $(Q, \varepsilon, \delta)$ has a terminal object when $\mathcal{C}$ does.

### 1.8.4 Modelling of Linear Exponentials

In this section we employ comonads in order to model the exponential bang operator, !, of Linear Logic. Let us start by modelling a weak bang operator, $\hat{!}$, which involves solely the following proof rules.

$$
\frac{\Gamma, A \vdash B}{\Gamma, \hat{\jmath} A \vdash B} \hat{\imath} \mathrm{~L} \quad \frac{\hat{!} B \vdash A}{\hat{\jmath} B \vdash \hat{!} A} \hat{\imath} \mathrm{R}
$$

Observe that, compared to $!, \hat{!}$ is weak in its Right rule, and it also misses Contraction and Weakening.

Let us now assume as given a symmetric monoidal closed category $\mathcal{C}$ along with a comonad $(Q, \varepsilon, \delta)$ on $\mathcal{C}$. As seen previously, $\mathcal{C}$ is a model of $(\otimes-)$ Linear Logic. Moreover, $(\mathcal{C}, Q)$ yields a model of $(\otimes-\hat{!})$-Linear Logic by modelling each formula $\hat{!} A$ by $Q A$ (i.e. $Q$ applied to the translation of $A$ ). The rules for weak bang are then interpreted as follows.

$$
\frac{f: \Gamma \otimes A \longrightarrow B}{f \circ \operatorname{id}_{\Gamma} \otimes \varepsilon_{A}: \Gamma \otimes Q A \longrightarrow B} \quad \frac{f: Q B \longrightarrow A}{Q f \circ \delta_{B}: Q B \longrightarrow Q A}
$$

We know that arrow-equalities in $\mathcal{C}$ correspond to proof-transformations in the proof system. Thus, the comonadic law $\varepsilon_{Q A} \circ \delta_{A}=\operatorname{id}_{Q A}=Q \varepsilon_{A} \circ \delta_{A}$ corresponds to the following transformations.


Exercise 132 Find a proof-transformation corresponding to the comonadic law $\delta_{Q A} \circ \delta_{A}=Q \delta_{A} \circ \delta_{A}$.

In order to extend our translation to the general ! R rule, we need arrows in $\mathcal{C}$ of the form

$$
Q^{2} A_{1} \otimes \cdots \otimes Q^{2} A_{n} \longrightarrow Q\left(Q A_{1} \otimes \cdots \otimes Q A_{n}\right)
$$

Hence, we need to impose (a coherent) distributivity of the tensor - either binary $(\otimes)$ or nullary $(I)$ - over the comonad $Q$. This can be formalised by stipulating that $Q$ be a symmetric monoidal endofunctor.

Definition 133 Let $(\mathcal{C}, \otimes, I, a, l, r, s)$ and $\left(\mathcal{C}^{\prime}, \otimes^{\prime}, I^{\prime}, a^{\prime}, l^{\prime}, r^{\prime}, s^{\prime}\right)$ be symmetric monoidal categories. A functor $F: \mathcal{C} \rightarrow \mathcal{C}^{\prime}$ is called symmetric monoidal if there exist:

- a morphism $m_{0}: I^{\prime} \rightarrow F(I)$,
- a natural transformation $m_{2}: F\left(\_\right) \otimes^{\prime} F\left(\_\right) \rightarrow F\left(\_\otimes_{\_}\right)$,
such that the following diagrams commute.


We may write such an $F$ as $(F, m)$. Moreover, if $(F, m),(G, n): \mathcal{C} \rightarrow \mathcal{C}^{\prime}$ are (symmetric) monoidal functors then a natural transformation $\phi: F \rightarrow G$ is called monoidal whenever the following diagrams commute.


For example, the identity functor is symmetric monoidal. Moreover, if $F$ and $G$ are symmetric monoidal functors then so is $G \circ F$. Other examples are the following.

- The constant endofunctor $K_{I}$, which maps each object to $I$ and each arrow to $\mathrm{id}_{I}$, is symmetric monoidal with structure maps:

$$
m_{0}: I \longrightarrow I:=\mathrm{id}_{I}, \quad m_{2}: I \otimes I \longrightarrow I:=r_{I}
$$

- The endofunctor $\otimes \circ\left\langle\operatorname{ld}_{\mathcal{C}}, \operatorname{ld}_{\mathcal{C}}\right\rangle$, which maps each object $A$ to $A \otimes A$ and each arrow $f$ to $f \otimes f$, is symmetric monoidal with:
$m_{0}: I \longrightarrow I \otimes I:=r_{I}^{-1}, \quad m_{2}:=(A \otimes A) \otimes(B \otimes B) \longrightarrow(A \otimes B) \otimes(A \otimes B)$,
the latter given by use of structural transformations.
Exercise 134 Verify that if $F: \mathcal{C} \rightarrow \mathcal{D}, G: \mathcal{D} \rightarrow \mathcal{E}$ are symmetric monoidal functors then so is $G \circ F$.

Definition 135 A comonad $(Q, \varepsilon, \delta)$ on a SMCC $\mathcal{C}$ is called a monoidal comonad if $Q$ is a symmetric monoidal functor, say $(Q, m)$, and $\varepsilon, \delta$ are monoidal natural transformations. We write $Q$ as $(Q, \varepsilon, \delta, m)$.

Now let us assume $\mathcal{C}$ is a SMCC and $(Q, \varepsilon, \delta, m)$ is a monoidal comonad on $\mathcal{C}$. The coherence of $m_{2}$ with $a$, expressed by the first diagram of symmetric monoidal functors, allows us to generalise $m_{0}$ and $m_{2}$ to arbitrary arities and assume arrows:

$$
m_{n}: Q A_{1} \otimes \cdots \otimes Q A_{n} \longrightarrow Q\left(A_{1} \otimes \cdots \otimes A_{n}\right)
$$

We can give the interpretation of the Right rule for bang as follows.

$$
\frac{f: Q B_{1} \otimes \cdots \otimes Q B_{n} \longrightarrow A}{Q f \circ m_{n} \circ\left(\delta_{B_{1}} \otimes \cdots \otimes \delta_{B_{n}}\right): Q B_{1} \otimes \cdots \otimes Q B_{n} \longrightarrow Q A}
$$

## Contraction and Weakening

Our discussion on the categorical modelling of linear exponentials has only touched the issues of Right and Left rules. However, we also need adequate structure for translating Contraction and Weakening.

$$
\frac{\Gamma,!A,!A \vdash B}{\Gamma,!A \vdash B} \text { Contr } \quad \frac{\Gamma \vdash B}{\Gamma,!A \vdash B} \text { Weak }
$$

For these rules we can use appropriate (monoidal) natural transformations. For Contraction, we stipulate a transformation with components $d_{A}: Q A \rightarrow$ $Q A \otimes Q A$, i.e.

$$
d: Q \longrightarrow \otimes \circ\langle Q, Q\rangle
$$

For Weakening, a transformation with components $e_{A}: Q A \rightarrow I$, i.e.

$$
e: Q \longrightarrow K_{I}
$$

Although the above allow the categorical interpretation of the proof-rules, they do not necessarily preserve the intended proof-transformations. For that, we need to impose some further coherence conditions, which are epitomised in the following notion.

Definition 136 Let $\mathcal{C}$ be a SMCC. A monoidal comonad $(Q, \varepsilon, \delta, m)$ on $\mathcal{C}$ is called a linear exponential comonad if there exist monoidal natural transformations

$$
d: Q \longrightarrow \otimes \circ\langle Q, Q\rangle, \quad e: Q \longrightarrow K_{I},
$$

such that:
(a) for each object $A$, the triple $\left(Q A, d_{A}, e_{A}\right)$ is a commutative comonoid in $\mathcal{C}$, i.e. the following diagrams commute,

(b) for each object $A$, the following diagrams commute.


We write $Q$ as $(Q, \varepsilon, \delta, m, d, e)$.
Exercise 137 Express what it means concretely for $d, e$ to be monoidal natural transformations.

Exercise 138 Give the categorical interpretation of Contraction and Weakening in a SMCC $\mathcal{C}$ with a linear exponential comonad.

### 1.8.4.1 Including Products

We now consider the fragment of Linear Logic which includes all four linear connectives we have seen thus far, i.e. $\otimes \multimap!\&$, and their respective proof rules (see definitions 122,123$)$. The categorical modelling of $(\otimes \multimap!\&)$-Linear Logic requires:

- a symmetric monoidal closed category $\mathcal{C}$,
- a linear exponential comonad $(Q, \varepsilon, \delta, m, d, e)$ on $\mathcal{C}$,
- finite products in $\mathcal{C}$.

The above structure is adequate for modelling the proof rules as we have seen previously. Moreover, it provides rich structure for the Kleisli category $\mathcal{C}_{Q}$. The next result and its proof demonstrate categorically the 'interpretation' of ordinary logic within Linear Logic given by:

$$
A \Rightarrow B \equiv!A \multimap B
$$

Proposition 139 Let $\mathcal{C}$ be a $S M C C$ with finite products and let $(Q, \varepsilon, \delta, m, d, e)$ be a linear exponential comonad on $\mathcal{C}$. Then:
(a) The Kleisli category $\mathcal{C}_{Q}$ has finite products.
(b) There exists an isomorphism $i: Q \mathbf{1} \rightarrow I$ and a natural isomorphism $j$ : $Q\left(\_\times \_\right) \rightarrow Q\left(\_\right) \otimes Q(-)$.
(c) $\mathcal{C}_{Q}$ is cartesian closed, with the exponential of objects $B, C$ being $Q B \multimap C$.

Proof: Part (a) has been shown previously (proposition 130, exercise 131), and part (b) is left as exercise. For (c), we have the following isomorphisms:

$$
\begin{aligned}
\mathcal{C}_{Q}(A \times B, C) & =\mathcal{C}(Q(A \times B), C) & & \text { definition of } \mathcal{C}_{Q} \\
& \cong \mathcal{C}(Q A \otimes Q B, C) & & \text { part }(\mathrm{b}) \\
& \cong \mathcal{C}(Q A, Q B \multimap C) & & \text { monoidal closure of } \mathcal{C} \\
& =\mathcal{C}_{Q}(A, Q B \multimap C) & & \text { defn of } \mathcal{C}_{Q} .
\end{aligned}
$$

Concretely, we obtain $\theta_{A}: \mathcal{C}_{Q}(A \times B, C) \xrightarrow{\cong} \mathcal{C}_{Q}(A, Q B \multimap C)$ by:

$$
\begin{aligned}
\theta_{A} & :=\left(f_{\cdot Q}: A \times B \rightarrow C\right) \longmapsto\left(\Lambda\left(f \circ j_{A, B}^{-1}\right)\right)_{\cdot Q} \\
\theta_{A}^{-1} & :=\left(g_{\cdot Q}: A \rightarrow Q B \multimap C\right) \longmapsto\left(\Lambda^{-1}(g) \circ j_{A, B}\right)_{\cdot Q} .
\end{aligned}
$$

Clearly, $\theta_{A}$ is a bijection. In order to establish couniversality of the exponential, we need also show naturality in $A$ (see exercise 76). So take $f_{. Q}: A \times B \rightarrow C$ and $h_{. Q}: A^{\prime} \rightarrow A$. Note first that the following commutes.


Note also that, for any $h_{i . Q}: A_{i}^{\prime} \rightarrow A_{i}$ in $\mathcal{C}_{Q}, i=1,2$, we have:

$$
h_{1 . Q} \times h_{2 . Q}:=\left(Q\left(A_{1}^{\prime} \times A_{2}^{\prime}\right) \xrightarrow{\left\langle Q \pi_{1}, Q \pi_{2}\right\rangle} Q A_{1}^{\prime} \times Q A_{2}^{\prime} \xrightarrow{h_{1} \times h_{2}} A_{1} \times A_{2}\right)_{. Q}
$$

Thus, noting that $\mathrm{id}_{B}^{\left(\mathcal{C}_{Q}\right)}=\varepsilon_{B . Q}$,

$$
\begin{aligned}
\theta_{A^{\prime}}\left(f_{. Q} \circ h_{\cdot Q} \times \mathrm{id}_{B}^{\left(\mathcal{C}_{Q}\right)}\right) & =\left(\Lambda\left(f \circ Q\left(h \times \varepsilon \circ\left\langle Q \pi_{1}, Q \pi_{2}\right\rangle\right) \circ \delta \circ j^{-1}\right)\right) . Q \\
& =\left(\Lambda\left(f \circ Q(h \times \varepsilon) \circ Q\left\langle Q \pi_{1}, Q \pi_{2}\right\rangle \circ \delta \circ j^{-1}\right)\right) ._{Q} \\
& \stackrel{(x)}{=}\left(\Lambda\left(f \circ Q(h \times \varepsilon) \circ j^{-1} \circ \delta \otimes \delta\right)\right) . Q \\
& =\left(\Lambda\left(f \circ j^{-1} \circ Q h \otimes Q \varepsilon \circ \delta \otimes \delta\right)\right)_{Q} \\
& =\left(\Lambda\left(f \circ j^{-1} \circ(Q h \circ \delta) \otimes \mathrm{id}\right)\right)_{. Q} \\
& =\left(\Lambda\left(f \circ j^{-1}\right) \circ Q h \circ \delta\right)_{. Q}=\theta_{A}(f \cdot Q) \circ h_{\cdot Q}
\end{aligned}
$$

as required.
Exercise 140 Show part (b) of proposition 139] For the defined $j$, show commutativity of (*).

### 1.8.5 Exercises

1. We say that a category $\mathcal{C}$ is well-pointed if it contains a terminal object $\mathbf{1}$ and, for any pair of arrows $f, g: A \rightarrow B$,

$$
f \neq g \Longrightarrow \exists h: 1 \rightarrow A . f \circ h \neq g \circ h .
$$

Let now $\mathcal{C}$ be a well-pointed category with a terminal object $\mathbf{1}$ and binary coproducts, and consider the functor $G: \mathcal{C} \rightarrow \mathcal{C}$ given by:

$$
G:=A \mapsto A+\mathbf{1}, f \mapsto f+\mathrm{id}_{\mathbf{1}} .
$$

If $\mathcal{C}(\mathbf{1}, \mathbf{1}+\mathbf{1})=\left\{\mathrm{in}_{1}, \mathrm{in}_{2}\right\}$ with $\mathrm{in}_{1} \neq \mathrm{in}_{2}$, show that if $(G, \eta, \mu)$ is a monad on $\mathcal{C}$ then, for each object $A$ :

$$
\eta_{A}=A \xrightarrow{\mathrm{in} n_{1}} A+\mathbf{1}, \quad \mu_{A}=(A+\mathbf{1})+\mathbf{1} \xrightarrow{\left[\mathrm{id} \mathrm{~d}_{A+1}, \mathrm{in}_{2}\right]} A+\mathbf{1} .
$$

2. Let $\mathcal{C}$ be a SMCC and let $(Q, \varepsilon, \delta)$ be a comonad on $\mathcal{C}$.
(a) Suppose that the sequents $\hat{!} A \vdash B$ and $\hat{!} B \vdash C$ are provable and let $f: Q A \rightarrow B$ and $g: Q B \rightarrow C$ be their interpretations (i.e. the interpretations of their proofs) in $\mathcal{C}$. Find the interpretations of the sequent $\hat{!} A \vdash \hat{!} C$ which correspond to each of the following proofs and show that the two interpretations are equal.
(b) Find the interpretations in $\mathcal{C}$ of the following proofs; are the interpretations equal?
3. Show that a symmetric monoidal category $\mathcal{C}$ has finite products (given by $\otimes, I$, etc.) iff there are monoidal natural transformations

$$
d: \operatorname{Id}_{\mathcal{C}} \longrightarrow \otimes \circ\left\langle\operatorname{ld}_{\mathcal{C}}, \operatorname{Id}_{\mathcal{C}}\right\rangle, \quad e: \operatorname{Id}_{\mathcal{C}} \longrightarrow K_{I},
$$

such that the following diagram commutes, for any $A \in O b(\mathcal{C})$.


### 1.9 Review of Sets, Functions and Relations

Our aim in this Appendix is to provide a brief review of notions we will assume in the notes. If the first paragraph is not familiar to you, you will need to acquire more background before being ready to read the notes.

Cartesian products, relations and functions
Given sets $X$ and $Y$, their cartesian product is

$$
X \times Y=\{(x, y) \mid x \in X \wedge y \in Y\}
$$

A relation $R$ from $X$ to $Y$, written $R: X \rightarrow Y$, is a subset $R \subseteq X \times Y$. Given such a relation, we write $(x, y) \in R$, or equivalently $R(x, y)$. We compose relations as follows: if $R: X \rightarrow Y$ and $S: Y \rightarrow Z$, then for all $x \in X$ and $z \in Z$ :

$$
R ; S(x, z) \equiv \exists y \in Y . R(x, y) \wedge S(y, z)
$$

A relation $f: X \rightarrow Y$ is a function if it satisfies the following two properties:

- (single-valuedness): if $(x, y) \in f$ and $\left(x, y^{\prime}\right) \in f$, then $y=y^{\prime}$.
- (totality): for all $x \in X$, for some $y \in Y,(x, y) \in f$.

If $f$ is a function, we write $f(x)=y$ or $f: x \mapsto y$ for $(x, y) \in f$. Function composition is written as follows: if $f: X \rightarrow Y$ and $g: Y \rightarrow Z$,

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

It is easily checked that $g \circ f=f ; g$, viewing functions as relations.

## Equality of functions

Two functions $f, g: X \rightarrow Y$ are equal if they are equal as relations, i.e. as sets of ordered pairs. Equivalently, but more conveniently, we can write:

$$
f=g \quad \Longleftrightarrow \forall x \in X . f(x)=g(x)
$$

The right-to-left implication is the standard tool for proving equality of functions on sets. As we shall see, when we enter the world of category theory, which takes a more general view of "arrows" $f: X \rightarrow Y$, for most purposes we have to leave this familiar tool behind!

Making the arrow notation for functions and relations unambiguous
Our definitions of functions and relations, as they stand, have an unfortunate ambiguity. Given a relation $R: X \rightarrow Y$, we cannot uniquely recover its "domain" $X$ and "codomain" $Y$. In the case of a function, we can recover the domain, because of totality, but not the codomain.
Example Consider the set of ordered pairs $\{(n, n) \mid n \in \mathbb{N}\}$, where $\mathbb{N}$ is the set of natural numbers. Is this the identity function $\mathrm{id}_{\mathbb{N}}: \mathbb{N} \longrightarrow \mathbb{N}$, or the inclusion function inc : $\mathbb{N} \hookrightarrow \mathbb{Z}$, where $\mathbb{Z}$ is the set of integers?

We wish to have unambiguous notions of domain and codomain for functions, and more generally relations. Thus we modify our official definition of a relation from $X$ to $Y$ to be an ordered triple $(X, R, Y)$, where $R \subseteq X \times Y$. We then define composition of $(X, R, Y)$ and $(Y, S, Z)$ in the obvious fashion, as $(X, R ; S, Z)$. We treat functions similarly. We shall not belabour this point in the notes, but it is implicit when we set up perhaps the most fundamental example of a category, namely the category of sets.

Size
We shall avoid explicit discussion of set-theoretical foundations in the text, but we include a few remarks for the interested reader. Occasionally, distinctions of set-theoretic size do matter in category theory. One example which does arise in the notes is when we consider Cat, the category of "all" categories. Does this category belong to (is it an object of) itself, at the risk of a Russell-type paradox? The way we avoid this is to impose some set-theoretic limitation of size on the categories gathered into Cat. Cat will then be too big to fit into itself. For example, we can limit Cat to those categories whose collections of objects and arrows form sets in the sense of some standard set theory such as ZFC. Cat will then be a proper class, and will not be an object of itself. One assumption we do make throughout the notes is that the categories we deal with are "locally small", i.e. that all hom-sets are indeed sets. Another place where some technical caveat would be in order is when we form functor categories. In practice, these issues never (well, hardly ever) cause problems, because of the strongly-typed nature of category theory. We leave the interested reader to delve further into these issues by consulting some of the standard texts.

### 1.10 Guide to Further Reading

Of the many texts on category theory, we shall only mention a few, which may be particularly useful to someone who has read these notes and wishes to learn more.

The short text [10] is very nicely written and gently paced; it is probably a little easier going than these notes. A text which is written with a clarity and at a level which makes it ideal as a next step after these notes is [5]. A text particularly useful for its large number of exercises with solutions is [1].

Another very nicely written text, focussing on the connections between categories and logic, and especially topos theory, is 4], recently reissued by Dover Books. A classic text on categorical logic is [6]. A more advanced text on topos theory is 9 .

The text 8 is a classic by one of the founders of category theory. It assumes considerable background knowledge of mathematics to fully appreciate its wide-ranging examples, but it provides invaluable coverage of the key topics.

A stimulating text on the correspondence between computation and logic is [3; ; it is out of print, but available online. A more recent text on this topic is (11.

The 3 -volume handbook [2] provides coverage of a broad range of topics in category theory. The book 7 is somewhat idiosyncratic in style, but offers insights by one of the key contributors to category theory.

## References

1. M. Barr and C. Wells, Category Theory for Computing Science, Third Edition, Publications CRM, 1999.
2. F. Borceux, Handbook of Categorical Algebra Volumes 1-3, Cambridge University Press, 1994.
3. J.-Y. Girard. P. Taylor and Y. Lafont, Proofs and Types, Cambridge University Press, 1989.
4. R. Goldblatt, Topoi, the Categorial Analysis of Logic, North-Holland 1984. Reprinted by Dover Books, 2006.
5. H. Herrlich and G. Strecker, Category Theory, Third Edition, Heldermann Verlag, 2007.
6. J. Lambek and P. J. Scott, Introduction to Higher-Order Categorical Logic, Cambridge University Press, 1986.
7. F. Lawvere and S. Schanuel, Conceptual Mathematics: A First Introduction to Categories, Cambridge University Press, 1997.
8. S. Mac Lane, Categories for the Working Mathematician, Second Edition, Springer, 1998.
9. S. Mac Lane and I. Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory, Springer, 1994.
10. B. Pierce, Basic Category Theory for Computer Scientists, MIT Press, 1991.
11. M. H. Sørensen and P. Urzyczyn, Lectures on the Curry-Howard Isomorphism, Elsevier, 2006.

[^0]:    1 A review of basic ideas about sets, functions and relations, and some of the notation we will be using, is provided in Appendix A.
    ${ }^{2}$ We shall use the notation ":=" for "is defined to be" throughout these notes.

[^1]:    3 This would be a "multigraph" in normal parlance, since multiple edges between a given pair of vertices are allowed.

[^2]:    What happens if we drop the Contraction and Weakening rules (but keep the Ex-

[^3]:    ${ }^{5}$ In some texts, this is called a coKleisli category.

