Lecture Notes on Category Theory

Paul Ossientis

April 11, 2020

Contents

Chapter 1

Category

1.1 Small category

Before we define a category in full generality, we shall focus our attention on the notion of *small category*. This notion is interesting to us because while it essentially describes the notion of *category* itself, it remains simple enough to be compared with various other algebraic structures. For example, consider the case of a monoid: a monoid is essentially a set M together with a binary relation $\circ$ defined on M which is associative, and an element e of M which acts as an identity element for $\circ$. In short a monoid is a tuple $(M, \circ, e)$ containing some data, and which satisfy certain properties. The same is true of a *small category*: it is also a tuple containing some data, and which satisfy certain properties:

Definition 1 *We call* small category *any tuple* $(\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ *with:*

$$
\begin{array}{ll}
(1) & \mathrm{Ob} \textit{ is a set} \\
(2) & \mathrm{Arr} \textit{ is a set} \\
(3) & \mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob} \textit{ is a function} \\
(4) & \mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob} \textit{ is a function} \\
(5) & \mathrm{id} : \mathrm{Ob} \to \mathrm{Arr} \textit{ is a function} \\
(6) & \circ : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr} \textit{ is a partial function} \\
(7) & g \circ f \textit{ is defined} \;\Leftrightarrow\; \mathrm{cod}(f) = \mathrm{dom}(g) \\
(8) & \mathrm{cod}(f) = \mathrm{dom}(g) \;\Rightarrow\; \mathrm{dom}(g \circ f) = \mathrm{dom}(f) \\
(9) & \mathrm{cod}(f) = \mathrm{dom}(g) \;\Rightarrow\; \mathrm{cod}(g \circ f) = \mathrm{cod}(g) \\
(10) & \mathrm{cod}(f) = \mathrm{dom}(g) \wedge \mathrm{cod}(g) = \mathrm{dom}(h) \;\Rightarrow\; (h \circ g) \circ f = h \circ (g \circ f) \\
(11) & \mathrm{dom}\,(\,\mathrm{id}(a)\,) = a = \mathrm{cod}\,(\,\mathrm{id}(a)\,) \\
(12) & \mathrm{dom}(f) = a \;\Rightarrow\; f \circ \mathrm{id}(a) = f \\
(13) & \mathrm{cod}(f) = a \;\Rightarrow\; \mathrm{id}(a) \circ f = f
\end{array}
$$

where (7) – (13) *hold for all* $f, g, h \in \mathrm{Arr}$ *and* $a \in \mathrm{Ob}$*:*

So if $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ is a small category, we have two sets Ob and Arr together with some structure defined on these sets. This feels very much like a monoid, except that we have two sets instead of one and it all looks more complicated. The set Ob is called the *set of objects* of the small category $\mathcal{C}$ and is denoted Ob $\mathcal{C}$, while the set Arr is called the *set of arrows* of the small category $\mathcal{C}$ and is denoted Arr $\mathcal{C}$. An element $x \in \mathrm{Ob}\ \mathcal{C}$ is called an *object* of $\mathcal{C}$, while an element $f \in \mathrm{Arr}\ \mathcal{C}$ is called an *arrow* of $\mathcal{C}$.

As part of the structure defined on the small category $\mathcal{C}$, we have two functions $\mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob}$ and $\mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob}$. Hence, given an arrow f of the small category $\mathcal{C}$, we have two objects $\mathrm{dom}(f)$ and $\mathrm{cod}(f)$ of the small category $\mathcal{C}$. The object $\mathrm{dom}(f)$ is called the *domain* of f. The object $\mathrm{cod}(f)$ is called the *codomain* of f. Note that an arrow f of the small category $\mathcal{C}$ is simply an element of the set Arr $\mathcal{C}$. So it is itself a set but it may not be a function. The words *domain* and *codomain* are therefore overloaded as we are using them in relation to a set f which is possibly not a function. Whenever f is an arrow of the small category $\mathcal{C}$ and a, b are objets, it is common to use the notation $f : a \to b$ as a notational shortcut for the equations $\mathrm{dom}(f) = a$ and $\mathrm{cod}(f) = b$. Once again, it is important to guard against the possible confusion induced by the notation $f : a \to b$ which does not mean that f is function. It simply means that f is an arrow with domain a and codomain b in the small category $\mathcal{C}$.

One of the main ingredients of the structure defining a small category $\mathcal{C}$ is the partial function $\circ : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$, called the *composition operator* in the small category $\mathcal{C}$. The domain of this partial function is made of all ordered pairs (g, f) of arrows in $\mathcal{C}$ for which $\mathrm{cod}(f) = \mathrm{dom}(g)$. As already indicated in definition (1), we use the infix notation $g \circ f$ rather than $\circ(g, f)$ and the arrow $g \circ f$ is called the *composition* of g and f. Once again, we should remember that the notation $g \circ f$ does not mean that g or f are functions. They are simply arrows in the small category $\mathcal{C}$. One key property of the composition operator $\circ$ is the associativity postulated by (10) of definition (1). Note that if $f : a \to b$ and $g : b \to c$, then from properties (8) and (9) of definition (1) we obtain $g \circ f : a \to c$. Furthermore, if $h : c \to d$ we have $h \circ g : b \to d$ and the arrows $(h \circ g) \circ f$ and $h \circ (g \circ f)$ are therefore well-defined arrows with domain a and codomain d. This shows that the expression involved in the associativity condition (10) of definition (1) is always meaningful, involving terms which are well-defined provided $g \circ f$ and $h \circ g$ are themselves well-defined, i.e. provided $\mathrm{cod}(f) = \mathrm{dom}(g)$ and $\mathrm{cod}(g) = \mathrm{dom}(h)$.

Finally, as part of the structure defining the small category $\mathcal{C}$, we have a function $\mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}$ called the *identity operator* on the small category $\mathcal{C}$. Hence, for every object a of $\mathcal{C}$ we have an arrow $\mathrm{id}(a)$ called the *identity at a*. Looking at property (11) of definition (1) we see that $\mathrm{id}(a) : a \to a$. In other words, the arrow $\mathrm{id}(a)$ has domain a and codomain a. Furthermore, looking at properties (12) and (13) of definition (1), for every arrow $f : a \to b$, the composition arrows $\mathrm{id}(b) \circ f$ and $f \circ \mathrm{id}(a)$ are well-defined and both equal to f.

1.2 Category

The notion of *small category* defined in definition (1) is similar to that of any other algebraic structure the reader may be familiar with. It can safely be encoded in set theory as a tuple (which is a set) containing data (which are other sets) which satisfies certain properties. In set theory, everything is a set. A small category $\mathcal{C}$ is a set, its collection of objects $\mathrm{Ob}\ \mathcal{C}$ is a set, its arrows $\mathrm{Arr}\ \mathcal{C}$ form a set, the functions dom, cod, id and the composition operator $\circ$ are all sets (functions are typically encoded as sets of ordered pairs).

Category theory falls outside of set theory. While the definition of a *category* we provide below is formally identical to that of a small category, the object we are defining can no longer be encoded in general as an object of set theory. For example, say we want to speak about the *universe of all sets* or the *universe of all monoids*. These *universes* which are known as *classes* cannot be represented as sets. They are not objects of set theory. Or say we are working within the formal framework of a proof assistant such as Coq, Agda or Lean. These tools are based on type theory and do not fall within the scope of set theory. When defining a *category*, we assume some form of meta-theoretic context, some form of logic, some way of reasoning about objects which may not be sets, where some meaning is attached to the words *tuple*, *collection*, *equality* and *map*. This may sound all very fuzzy, yet we cannot be more formal at this stage.

Definition 2 *We call* category *any tuple* $(\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ *such that:*

$$
\begin{array}{rl}
(1) & \mathrm{Ob}\ \textit{is a collection with equality} \\
(2) & \mathrm{Arr}\ \textit{is a collection with equality} \\
(3) & \mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob}\ \textit{is a map} \\
(4) & \mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob}\ \textit{is a map} \\
(5) & \mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}\ \textit{is a map} \\
(6) & \circ : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}\ \textit{is a partial map} \\
(7) & g \circ f\ \textit{is defined} \ \Leftrightarrow\ \mathrm{cod}(f) = \mathrm{dom}(g) \\
(8) & \mathrm{cod}(f) = \mathrm{dom}(g) \ \Rightarrow\ \mathrm{dom}(g \circ f) = \mathrm{dom}(f) \\
(9) & \mathrm{cod}(f) = \mathrm{dom}(g) \ \Rightarrow\ \mathrm{cod}(g \circ f) = \mathrm{cod}(g) \\
(10) & \mathrm{cod}(f) = \mathrm{dom}(g) \wedge \mathrm{cod}(g) = \mathrm{dom}(h) \ \Rightarrow\ (h \circ g) \circ f = h \circ (g \circ f) \\
(11) & \mathrm{dom}\,(\,\mathrm{id}(a)\,) = a = \mathrm{cod}\,(\,\mathrm{id}(a)\,) \\
(12) & \mathrm{dom}(f) = a \ \Rightarrow\ f \circ \mathrm{id}(a) = f \\
(13) & \mathrm{cod}(f) = a \ \Rightarrow\ \mathrm{id}(a) \circ f = f
\end{array}
$$

where $(7) - (13)$ *hold for all* $f, g, h \in \mathrm{Arr}$ *and* $a \in \mathrm{Ob}$*:*

So let $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ be a category: then $\mathcal{C}$ is a *tuple* but it is no longer a tuple in a set-theoretic sense. We assume given some logical framework where the notion of *tuple* is clear, even if not formally defined. Furthermore, We are no longer imposing that Ob should be a set, but are instead using the phrase *collection with equality*, whatever this may mean in our given logical context. So we shall still make use of the notation Ob $\mathcal{C}$ but this will now refer to the *collection* of all *objects* of the category $\mathcal{C}$.

Notation 1 *Let $\mathcal{C}$ be a category. Its collection of* objects *is denoted* Ob $\mathcal{C}$.

In fact, if a is an object of the category $\mathcal{C}$, we shall abuse notations somewhat by writing '$a \in \mathrm{Ob}\ \mathcal{C}$' or even simply '$a \in \mathcal{C}$' to express the fact that a is an object of $\mathcal{C}$, being understood that this use of the set membership symbol '$\in$' does not mean anything is a set. Since we are stepping out of set theory, the objects of the category $\mathcal{C}$ may not be sets themselves. They are simply members of the *collection* Ob $\mathcal{C}$.

Notation 2 *Let $\mathcal{C}$ be a category. We write $a \in \mathcal{C}$ as a shortcut for $a \in \mathrm{Ob}\ \mathcal{C}$.*

However, properties $(7) - (13)$ of definition (2) are all referring to equalities between objects such that $\mathrm{cod}(f) = \mathrm{dom}(g)$. So it must be the case that the notion of *equality* be meaningful on the collection Ob $\mathcal{C}$. This explains our use of the phrase *collection with equality*: given $a, b \in \mathcal{C}$, the statement $a = b$ while not a set-theoretic equality is nonetheless assumed to be defined.

Similarly, the *collection* of *arrows* of the category $\mathcal{C}$ shall still be denoted Arr $\mathcal{C}$, but is no longer required to be a set. If f is an arrow of the category $\mathcal{C}$ then f itself may not be a set and we may still write '$f \in \mathrm{Arr}\ \mathcal{C}$' simply to indicate that f is a *member* of the *collection* Arr $\mathcal{C}$. Properties (10), (12) and (13) of definition (2) are all referring to equalities between arrows so the *collection* Arr $\mathcal{C}$ must have some notion of *equality* defined on it.

Notation 3 *Let $\mathcal{C}$ be a category. Its collection of* arrows *is denoted* Arr $\mathcal{C}$.

Since Ob and Arr are no longer sets in general, the *maps* $\mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob}$, $\mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob}$, $\mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}$ and the partial map $\circ : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$ cannot possibly be *functions* in the set-theoretic sense. So there must be some meaning to the word *map* (from one *collection* to another) in whatever logical framework we are working in. The *collection* $\mathrm{Arr} \times \mathrm{Arr}$ is not a set, and is simply the *collection* of all 2-dimensional *tuples* made from Arr. Our using the word *map* rather than *function* in definition (2) is simply an attempt at reminding ourselves of the fact these are not set-theoretic functions, eventhough the words *map* and *function* are perfectly interchangeable in standard (set-theoretic) mathematics. Given $f \in \mathrm{Arr}\ \mathcal{C}$, we shall still call the object $\mathrm{dom}(f)$ the *domain* of f and the object $\mathrm{cod}(f)$ the *codomain* of f.

Notation 4 *Let $\mathcal{C}$ be a category. The domain of an arrow $f \in \mathrm{Arr}\ \mathcal{C}$ is denoted* $\mathrm{dom}(f)$*, while its codomain is denoted* $\mathrm{cod}(f)$.

Remark: Notation (4) is potentially ambiguous as a mathematical object f could in principle be an arrow in several categories, and the designations $\mathrm{dom}(f)$ and $\mathrm{cod}(f)$ do not specfify which category is being referred to.

Given $a, b \in \mathcal{C}$, we shall still use the notation $f : a \to b$ as a notational shortcut for $\mathrm{dom}(f) = a$ and $\mathrm{cod}(f) = b$. Hence we state:

Notation 5 *Let $\mathcal{C}$ be a category. We write $f : a \to b$ or $f : a \to b$ @ $\mathcal{C}$ as a shortcut for $f \in \mathrm{Arr}\ \mathcal{C}$ together with $\mathrm{dom}(f) = a$ and $\mathrm{cod}(f) = b$.*

Remark: The qualification @ $\mathcal{C}$ in notation (5) may be useful to disambiguate between several categories in a given context.

The partial map $\circ$ is still the *composition operator* and the arrow $g \circ f$ shall still be called the *composition* of g and f, provided it is defined.

Notation 6 *Let $\mathcal{C}$ be a category. The composition operator on $\mathcal{C}$ is denoted $\circ$, and the composition of two arrows g and f is denoted $g \circ f$ or $g \circ f$ @ $\mathcal{C}$.*

Remark: $\circ$ may also be ambiguous in a context with several categories. It is also the common symbol to refer to standard function composition.

The map $\mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}$ is still the *identity operator* on the category $\mathcal{C}$, and for all $a \in \mathcal{C}$, the arrow $id(a) : a \to a$ is known as the *identity at a*.

Notation 7 *Let $\mathcal{C}$ be a category. We write $\mathrm{id}(a)$ or $\mathrm{id}(a)$ @ $\mathcal{C}$ to denote the* identity at a *in the category $\mathcal{C}$.*

For all arrows $f : a \to b$, it is still the case that the arrows $\mathrm{id}(b) \circ f$ and $f \circ \mathrm{id}(a)$ are well-defined and both equal to f. Just as in the case of a small category, whenever $f : a \to b$, $g : b \to c$ and $h : c \to d$, all the terms involved in the equation $(h \circ g) \circ f = h \circ (g \circ f)$ of definition (2) are well defined.

Proposition 1 *A small category is a category.*

Proof
When considering a small category $\mathcal{C}$, we are implicitely working within a set theoretic framework in which equality between any two sets is always meaningful and elements of sets are themselves sets. So any set can be viewed as a *collection with equality* and hence a small category satisfies definition (2). $\diamond$

1.3 Equality of categories

Whichever logical framework we are working from, we saw that when defining a category $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$, some notion of equality had to be defined on the collections Ob and Arr. Now if $\mathcal{C}' = (\mathrm{Ob}', \mathrm{Arr}', \mathrm{dom}', \mathrm{cod}', \mathrm{id}', \circ')$ is another category, the question may arise as to whether $\mathcal{C} = \mathcal{C}'$. Or indeed, we may simply be asking whether the collections Ob and Ob' are the same, or whether $\mathrm{dom} = \mathrm{dom}'$ etc. It is very difficult for us to carry out any sort of formal reasoning on things without equality. So having equality defined on Ob and Arr is neccessary for definition (2) to even make sense, but it is not enough for us to formally prove anything about categories. Hence we shall assume:

Axiom 1 *A notion of equality exists for* collections.

It is implicit in the statement of axiom (1) that the notion of equality between *collections* should be reflexive, symmetric and transitive. Furthermore:

Axiom 2 *Two collections with identical members are equal.*

In particular, if $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ is a category and we have another partial map $\circ' : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$ such that $g \circ' f$ is defined if and only if $g \circ f$ is defined, then axiom (2) allows us to argue that the domain of $\circ'$ is the same collection as the domain of $\circ$.

Axiom 3 *Let A be a collection and B be a collection with equality. Then two maps $F, G : A \to B$ are equal if and only if $F(x) = G(x)$ for all x in A.*

In particular, if $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ is a category and $\mathrm{dom}' : \mathrm{Arr} \to \mathrm{Ob}$ is another map such that $\mathrm{dom}'(x) = \mathrm{dom}(x)$ for every object $x \in \mathcal{C}$, then $\mathrm{dom}' = \mathrm{dom}$. Or if $\circ' : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$ is another partial map with the same domain as that of $\circ$ and such that $g \circ' f = g \circ f$ when defined, then $\circ' = \circ$.

Axiom 4 *Two tuples with identical entries are equal.*

So if $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ and $\mathcal{C}' = (\mathrm{Ob}', \mathrm{Arr}', \mathrm{dom}', \mathrm{cod}', \mathrm{id}', \circ')$ are two catgeories such that $\mathrm{Ob} = \mathrm{Ob}'$, $\mathrm{Arr} = \mathrm{Arr}'$, $\mathrm{dom} = \mathrm{dom}'$, $\mathrm{cod} = \mathrm{cod}'$, $\mathrm{id} = \mathrm{id}'$ and $\circ = \circ'$ then we have the equality $\mathcal{C} = \mathcal{C}'$.

Notation 8 *If $f : A \to B$ and $g : B \to C$ are maps between collections, we denote $g \circ f$ the map $g \circ f : A \to C$ defined by $(g \circ f)(a) = g(f(a))$ for all $a \in A$.*

1.4 Category of sets

Definition 3 *We call* **Set** *the category* $\mathbf{Set} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ *where*

$$
\begin{array}{ll}
(1) & \mathrm{Ob} = \{\ x \mid x \textit{ is a set}\ \} \\
(2) & \mathrm{Arr} = \{\ (a, b, f) \mid f \textit{ is a function } f : a \to b\ \} \\
(3) & \mathrm{dom}\,(a, b, f) = a \\
(4) & \mathrm{cod}\,(a, b, f) = b \\
(5) & \mathrm{id}(a) = (a, a, i(a)) \\
(6) & (b, c, g) \circ (a, b, f) = (a, c, g \circ f)
\end{array}
$$

where $(3) - (6)$ hold for all sets a, b, c and functions $f : a \to b$, $g : b \to c$, $i(a) : a \to a$ denotes the usual identity function on a, and $g \circ f$ denotes the usual function composition defined by $(g \circ f)(x) = g(f(x))$, for all $x \in a$.

The collection of objects of the category **Set** is defined to be the class of all sets. We are using the set comprehension notation $\{ \ x \mid x \text{ is a set } \}$ to denote this class, but this is an abuse of notation as Ob is not a set but a proper class. One could think of a class as a precicate $P(x)$ of first order logic with one free variable. From this point of view Ob becomes the predicate $\mathrm{Ob}(x) = \top$, i.e. the predicate which returns true for all x. Every set satisfies the predicate Ob, so every set is a member of the class Ob. The class Ob is not a set because the set-theoretic statement $\exists y, \forall z, z \in y \ \Leftrightarrow \ \mathrm{Ob}(z)$ can be proven false. In other words, there exists no set y whose elements z are exactly the sets which satisfy the predicate Ob. There exists no set which contains all sets.

The collection of arrows of the category **Set** is defined to be the class of triples (a, b, f) where a, b are sets and f is a function $f : a \to b$. This last notation is a common set-theoretic shortcut to express the fact that f is a *function* with *domain* a and *range* **which is a subset of** b. A *function* is any set f whose elements are ordered pairs (x, y) and which is functional, i.e. for which the following implication holds for all sets x, y, y':

$$(x, y) \in f \ \wedge \ (x, y') \in f \ \Rightarrow \ y = y'$$

The *domain* of a function f is the set of all sets x for which there exists a set y with $(x, y) \in f$. The *range* of a function f is the set of all sets y for which there exists a set x with $(x, y) \in f$. If x belongs to the domain of a function f, the notation '$f(x)$' commonly refers to the unique set y with $(x, y) \in f$.

Now, as already pointed out the notation $f : a \to b$ only requires that the range of f should be a subset of b. There is no requirement that the range of f should be equal to b. So if $f : a \to b$ and $b \subseteq c$ then $f : a \to c$. This explains why the collection of arrows Arr is defined as a class of triples (a, b, f) rather than a class of functions f. Knowing the function f does not tell you which *codomain* it should have. Any set b which is a superset of its range is a possible codomain. So we keep the set b together with the function f in the triple (a, b, f) so as to remember which codomain is intended for this particular arrow of the category **Set**. Incidentally, we also keep the range a of the function f in the triple (a, b, f) but this is not necessary, as the knowledge of f does allow us to recover its domain a. However, the triple (a, b, f) is convenient, allowing us to treat *domain* and *codomain* uniformly. In fact, it is worth pausing for a second and emphasize the difference between f and (a, b, f):

Definition 4 *Given a function $f : a \to b$ between two sets a and b we say that f is the* untyped function *while the triple (a, b, f) is called the* typed *function.*

Once again, it should be remembered that the collection of arrows Arr is not a set but a proper class, corresponding to the predicate $\mathrm{Arr}(x)$:

$$\mathrm{Arr}(x) = \exists a \, \exists b \, \exists f \, , \ x = (a, b, f) \ \wedge \ f : a \to b$$

Informally, this predicates expresses the fact that x is a typed function. The maps $\mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob}$ and $\mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob}$ for the category **Set** are defined respectively by $\mathrm{dom}(a,b,f)=a$ and $\mathrm{cod}(a,b,f)=b$. This looks simple enough, but for those who worry about foundational issues, we should just note that these are also proper classes which can be encoded as predicates. For example:

$$\mathrm{dom}(x)=\exists u\,\exists v\,,\ x=(u,v)\ \wedge\ \mathrm{Arr}(u)\ \wedge\ (\ \exists a\,\exists b\,\exists f\,,\ u=(a,b,f)\ \wedge\ v=a\ )$$

In other words, any set x satisfies the predicate $\mathrm{dom}(x)$ if and only if it is an ordered pair (u,v) where u satisfies the predicate $\mathrm{Arr}(u)$ and for which there exist sets a,b,f with $u=(a,b,f)$ and $v=a$. In short, (u,v) satisfies the predicate dom if and only if u is an arrow $u=(a,b,f)$ and $v=a$.

We defined the identity operator id by $\mathrm{id}(a)=(a,a,i(a))$ and the composition operator $\circ$ by $(b,c,g)\circ(a,b,f)=(a,c,g\circ f)$ where $g\circ f$ is the usual function composition and $i(a):a\to a$ is the usual identity function. As before, these defined maps are not functional sets of ordered pairs but rather proper classes which we could also encode as precicates of first order logic. One important point to note is the fact that (6) of definition (3) only defines the composition arrow $(b,c,g)\circ(a,b,f)$ where $f:a\to b$ and $g:b\to c$. In other words, the composition $(d,c,g)\circ(a,b,f)$ with $f:a\to b$ and $g:d\to c$ is only defined when $b=d$. Furthermore, the usual function composition $g\circ f$ is a function $g\circ f:a\to c$ which from (2) of definition (3) means that the composed arrow $(b,c,g)\circ(a,b,f)=(a,c,g\circ f)$ is indeed a member of the collection Arr, and the partial map $\circ$ thus defined is indeed a partial map $\circ:\mathrm{Arr}\times\mathrm{Arr}\to\mathrm{Arr}$.

Proposition 2 *The category* **Set** *of definition (3) is a category.*

Proof
Now that we have defined the data $(\mathrm{Ob},\mathrm{Arr},\mathrm{dom},\mathrm{cod},\mathrm{id},\circ)$ of the category **Set**, it is time to check this data actually forms a category. We need to check that conditions (7) – (13) of definition (2) are satisfied.

(7): suppose f^* and g^* are two members of the collection Arr. We need to check that $g^*\circ f^*$ is defined if and only if $\mathrm{cod}(f^*)=\mathrm{dom}(g^*)$. Using (2) of definition (3), f^* can be written $f^*=(a,b,f)$ for some function $f:a\to b$ and g^* can be written $g^*=(d,c,g)$ for some function $g:d\to c$. However, from (6) of definition (3), the arrrow $(d,c,g)\circ(a,b,f)$ is only defined in the case when $b=d$. Furthermore, from (4) of definition (3) we have $\mathrm{cod}(f^*)=b$ and from (3) of definition (3) we have $\mathrm{dom}(g^*)=d$. We conclude that $g^*\circ f^*$ is defined if and only if $\mathrm{cod}(f^*)=\mathrm{dom}(g^*)$ as required.

(8): Let $f^*,g^*\in\mathrm{Arr}$ such that $\mathrm{cod}(f^*)=\mathrm{dom}(g^*)$. We need to show that $\mathrm{dom}(g^*\circ f^*)=\mathrm{dom}(f^*)$. As before, f^* and g^* can be written as $f^*=(a,b,f)$ and $g^*=(b,c,g)$ where $f:a\to b$ and $g:b\to c$. We have $g^*\circ f^*=(a,c,g\circ f)$. Using (3) of definition (3) we obtain $\mathrm{dom}(g^*\circ f^*)=a=\mathrm{dom}(f^*)$.

(9): Let $f^*,g^*\in\mathrm{Arr}$ such that $\mathrm{cod}(f^*)=\mathrm{dom}(g^*)$. We need to show that $\mathrm{cod}(g^*\circ f^*)=\mathrm{cod}(g^*)$. As before, we have $g^*\circ f^*=(a,c,g\circ f)$ and $g^*=(b,c,g)$. Using (4) of definition (3) we obtain $\mathrm{cod}(g^*\circ f^*)=c=\mathrm{cod}(g^*)$.

(10): Let $f^*,g^*,h^*\in\mathrm{Arr}$ with $\mathrm{cod}(f^*)=\mathrm{dom}(g^*)$ and $\mathrm{cod}(g^*)=\mathrm{dom}(h^*)$. We need to show the equality: $(h^*\circ g^*)\circ f^*=h^*\circ(g^*\circ f^*)$. However, f^*,g^*,h^*

can be decomposed as $f^* = (a, b, f)$, $g^* = (b, c, g)$ and $h^* = (c, d, h)$ with $f : a \to b$, $g : b \to c$, and $h : c \to d$. We have:

$$\begin{array}{rcl} (h^* \circ g^*) \circ f^* & = & ((c, d, h) \circ (b, c, g)) \circ (a, b, f) \\ \text{(6) of Def (3)} \to & = & (b, d, h \circ g) \circ (a, b, f) \\ \text{(6) of Def (3)} \to & = & (a, d, (h \circ g) \circ f) \\ \text{assoc of usual composition} \to & = & (a, d, h \circ (g \circ f)) \\ \text{(6) of Def (3)} \to & = & (c, d, h) \circ (a, c, g \circ f) \\ \text{(6) of Def (3)} \to & = & (c, d, h) \circ ((b, c, g) \circ (a, b, f)) \\ & = & h^* \circ (g^* \circ f^*) \end{array}$$

(11): Let a be a set. We need to show that $\mathrm{dom}(\mathrm{id}(a)) = a = \mathrm{cod}(\mathrm{id}(a))$. This follows immediately from $\mathrm{id}(a) = (a, a, i(a))$ which is (5) of definition (3).

(12): Let $f^* = (a, b, f)$ be an arrow with $\mathrm{dom}(f^*) = a$. We need to show that $f^* \circ \mathrm{id}(a) = f^*$, which follows from:

$$\begin{array}{rcl} f^* \circ \mathrm{id}(a) & = & (a, b, f) \circ \mathrm{id}(a) \\ \text{(5) of Def (3)} \to & = & (a, b, f) \circ (a, a, i(a)) \\ \text{(6) of Def (3)} \to & = & (a, b, f \circ i(a)) \\ \text{usual right-identity} \to & = & (a, b, f) \\ & = & f^* \end{array}$$

(13): Let $f^* = (b, a, f)$ be an arrow with $\mathrm{cod}(f^*) = a$. We need to show that $\mathrm{id}(a) \circ f^* = f^*$, which follows from:

$$\begin{array}{rcl} \mathrm{id}(a) \circ f^* & = & \mathrm{id}(a) \circ (b, a, f) \\ \text{(5) of Def (3)} \to & = & (a, a, i(a)) \circ (b, a, f) \\ \text{(6) of Def (3)} \to & = & (b, a, i(a) \circ f) \\ \text{usual left-identity} \to & = & (b, a, f) \\ & = & f^* \end{array}$$

This completes our proof of properties (7) − (13). $\diamond$

Notation 9 *We shall often refer to an arrow (a, b, f) of the category* **Set** *i.e. a typed function, simply as its untyped counterpart f. The context should make it clear that f actually refers to a typed function.*

Remark So on top of its usual set-theoretic meaning for untyped functions, the notation $f : a \to b$ may also have its categorical meaning for typed functions, expressing the fact that f is an arrow of the category **Set** with domain a and codomain b, i.e. that f is really the typed function (a, b, f).

Whenever $f : a \to b$ is an arrow of the category **Set** and $x \in a$, the notation $f(x)$ is not strictly speaking meaningful since f is not a function but a typed function, i.e a tuple (a, b, f). However, it is natural enough to set:

Notation 10 *If* $f : a \to b$ *is an arrow of the category* **Set***, for all* $x \in a$ *we shall write* $f(x)$ *as a shortcut for* $f(x)$ *where* f *is the underlying untyped function.*

Functions in set theory are just sets, and equality between functions is simply the standard equality between sets. As it turns out, if f and g are two functions with the same domain a, then the set equality $f = g$ is equivalent to the *extensional* equality $\forall x \in a \ , \ f(x) = g(x)$:

Proposition 3 *Let* f, g *be two functions with identical domain* a*. We have:*

$$f = g \ \Leftrightarrow \ \forall x \in a \ , \ f(x) = g(x)$$

Proof
($\Rightarrow$): We assume that $f = g$ and $x \in a$. We need to show that $f(x) = g(x)$. However $f(x)$ is defined as the unique set y such that $(x, y) \in f$, while $g(x)$ is the unique set y such that $(x, y) \in g$. Having assumed that $f = g$, both $f(x)$ and $g(x)$ are the unique set y such that $(x, y) \in f$. So we must have $f(x) = g(x)$ by uniqueness.
($\Leftarrow$): We assume that $f(x) = g(x)$ for all $x \in a$. We need to show that $f = g$. Hence we need to show that $f \subseteq g$ and $g \subseteq f$. By symmetry, we can focus on proving $f \subseteq g$ as the same proof will carry over for $g \subseteq f$. So suppose $z \in f$. We need to show that $z \in g$. Having assumed that f is a function, the element z must be an ordered pair $z = (x, y)$. Hence we have $(x, y) \in f$. Having assumed that the domain of f is a, this shows that $x \in a$. Furthermore, from $(x, y) \in f$ we obtain $f(x) = y$. Hence, by assumption we obtain $g(x) = y$. However $g(x)$ is the unique set y' with $(x, y') \in g$. Hence, we have $(x, y) \in g$ and finally $z \in g$. $\diamond$

An important question which will invariably arise is deciding when two arrows of the category **Set** are equal. Although we have defined an arrow to a typed function (a, b, f), as indicated in notation (9) we will often refer to such an arrow simply as f. However, we should not forget that the equality $f = g$ between underlying untyped functions is not enough for two arrows (a, b, f) and (c, d, g) to be equal. The equality between untyped functions will ensure that they have the same domain and the same range, but it does not tell us anything about the intended codomains of their respective typed functions.

Proposition 4 *If two functions* f, g *are equal, they have the same domain.*

Proof
We assume f, g are functions and $f = g$. We need to show f and g have the same domain. By symmetry, it is sufficient to show that the domain of f is a subset of that of g. So let x be an element of the domain of f. There exists some y with $(x, y) \in f$. From $f = g$ we obtain $(x, y) \in g$ and consequently x is also an element of the domain of g. $\diamond$

Proposition 5 *If two functions* f, g *are equal, they have the same range.*

Proof
We assume f, g are functions and $f = g$. We need to show f and g have the same range. By symmetry, it is sufficient to show that the range of f is a subset of that of g. So let y be an element of the range of f. There exists some x with $(x, y) \in f$. From $f = g$ we obtain $(x, y) \in g$ and consequently y is also an element of the range of g. $\diamond$

Proposition 6 *Let $f : a \to b$ and $g : c \to d$ be two arrows of the category* **Set**. *Then $f = g$ if and only if $a = c$, $b = d$ and $f(x) = g(x)$ for all $x \in a$.*

Proof
Let $f^* : a \to b$ and $g^* : c \to d$, that is $f^* = (a, b, f)$ (for some f) and $g^* = (c, d, g)$ (for some g) be two arrows of the category **Set**. We call these arrows f^* and g^* in this proof so as to be very precise on the distinction between an arrow (a typed function) and its underlying function (an untyped function). Looking at definition (3), f and g are functions $f : a \to b$ and $g : c \to d$. So the domain of f is the set a while the domain of g is the set c, and the range of f is a subset of b while the range of g is a subset of d. First we assume that $f^* = g^*$. Then we have the equality between triples $(a, b, f) = (c, d, g)$. Hence we obtain immediately $a = c$ and $b = d$ as requested. However, we also obtain the equality between underlying functions $f = g$. Using proposition (3), we see that $f(x) = g(x)$ for all $x \in a$. By virtue of notation (10), $f^*(x)$ and $g^*(x)$ are notational shortcuts for $f(x)$ and $g(x)$ respectively. Hence we have $f^*(x) = g^*(x)$ for all $x \in a$ as requested. We now assume that $a = c$, $b = d$ and $f^*(x) = g^*(x)$ for all $x \in a$. We need to show that $f^* = g^*$, or in other words $(a, b, f) = (c, d, g)$. Hence it remains to show that $f = g$. This follows from proposition (3) and the fact that $f(x) = g(x)$ for all $x \in a$. $\diamond$

1.5 Opposite category

Definition 5 *Let $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ be a category. We call* opposite category of $\mathcal{C}$*, the category denoted $\mathcal{C}^{op}$ and defined by:*

$$\mathcal{C}^{op} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{cod}, \mathrm{dom}, \mathrm{id}, \circ')$$

where the composition operator $\circ'$ is defined by $f \circ' g = g \circ f$, for all $f, g \in \mathrm{Arr}$.

So if $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ is a category, the opposite category $\mathcal{C}^{op}$ is almost identical, except for the composition operator $\circ'$ which is a flipped version of $\circ$, and for dom and cod which have been swapped with each other. The collection of objects of $\mathcal{C}^{op}$ is the same as that of $\mathcal{C}$, giving us the equality $\mathrm{Ob}\ \mathcal{C}^{op} = \mathrm{Ob}\ \mathcal{C}$. Likewise, the collection of arrows of $\mathcal{C}^{op}$ is the same as that of $\mathcal{C}$, giving us this other equality $\mathrm{Arr}\ \mathcal{C}^{op} = \mathrm{Arr}\ \mathcal{C}$. If we denote $\mathrm{dom}' : \mathrm{Arr} \to \mathrm{Ob}$ and $\mathrm{cod}' : \mathrm{Arr} \to \mathrm{Ob}$ the domain and codomain maps on $\mathcal{C}^{op}$, then $\mathrm{dom}' = \mathrm{cod}$ and $\mathrm{cod}' = \mathrm{dom}$. The identity operator $\mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}$ is the same for both $\mathcal{C}$ and $\mathcal{C}^{op}$, and the composition arrow $f \circ' g$ in $\mathcal{C}^{op}$ is defined whenever the composition arrow $g \circ f$ in $\mathcal{C}$ is defined, and we have $f \circ' g = g \circ f$.

Proposition 7 *Let $\mathcal{C}$ be a category. Then $\mathcal{C}^{op}$ of definition (5) is a category.*

Proof
We need to check that the data $\mathcal{C}^{op} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{cod}, \mathrm{dom}, \mathrm{id}, \circ')$ of definition (5) forms a category, having assumed that the underlying data for $\mathcal{C}$ does. We have indeed two collections Ob and Arr with maps between them $\mathrm{cod} : \mathrm{Arr} \to \mathrm{Ob}$, $\mathrm{dom} : \mathrm{Arr} \to \mathrm{Ob}$, $\mathrm{id} : \mathrm{Ob} \to \mathrm{Arr}$ and partial map $\circ' : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$. So it remains to show that conditions $(7) - (13)$ of definition (2) are satisfied. For the purpose of this proof, we shall denote $\mathrm{dom}' = \mathrm{cod}$ and $\mathrm{cod}' = \mathrm{dom}$.

(7): We need to check that $f \circ' g$ is defined if and only if $\mathrm{cod}'(g) = \mathrm{dom}'(f)$ which is $\mathrm{dom}(g) = \mathrm{cod}(f)$. However by definition, we have set $f \circ' g$ to be defined whenever $g \circ f$ is itself defined, and since $\mathcal{C}$ is a category, this is in turn equivalent to $\mathrm{cod}(f) = \mathrm{dom}(g)$. Hence, we are done.

(8): We need to check that $\mathrm{dom}'(f \circ' g) = \mathrm{dom}'(g)$ which can be written as $\mathrm{cod}(g \circ f) = \mathrm{cod}(g)$ and which is true since $\mathcal{C}$ is a category.

(9): We need to check that $\mathrm{cod}'(f \circ' g) = \mathrm{cod}'(f)$ which can be written as $\mathrm{dom}(g \circ f) = \mathrm{dom}(f)$ and which is true since $\mathcal{C}$ is a category.

(10): Given arrows h, g, f with $\mathrm{cod}'(h) = \mathrm{dom}'(g)$ and $\mathrm{cod}'(g) = \mathrm{dom}'(f)$, we need to check that $(f \circ' g) \circ' h = f \circ' (g \circ' h)$. However, our assumption can be written as $\mathrm{dom}(h) = \mathrm{cod}(g)$ and $\mathrm{dom}(g) = \mathrm{cod}(f)$ and having assumed that $\mathcal{C}$ is a category, by property (10) of definition (2) we have:

$$\begin{array}{rrcl} & (f \circ' g) \circ' h & = & h \circ (f \circ' g) \\ & & = & h \circ (g \circ f) \\ \mathcal{C} \text{ is a category} \rightarrow & & = & (h \circ g) \circ f \\ & & = & f \circ' (h \circ g) \\ & & = & f \circ' (g \circ' h) \end{array}$$

(11): We need to check that $\mathrm{dom}'(\,\mathrm{id}(a)\,) = a = \mathrm{cod}'(\,\mathrm{id}(a)\,)$ for all $a \in \mathcal{C}$, which follows from $\mathrm{dom}' = \mathrm{cod}$, $\mathrm{cod}' = \mathrm{dom}$ and the fact that $\mathcal{C}$ is a category.

(12): We need to check that $f \circ' \mathrm{id}(a) = f$ whenever $\mathrm{dom}'(f) = a$, that is $\mathrm{id}(a) \circ f = f$ whenever $\mathrm{cod}(f) = a$, which follows from $\mathcal{C}$ being a category.

(13): We need to check that $\mathrm{id}(a) \circ' f = f$ whenever $\mathrm{cod}'(f) = a$, that is $f \circ \mathrm{id}(a) = f$ whenever $\mathrm{dom}(f) = a$, which follows from $\mathcal{C}$ being a category.
This completes our proof of properties $(7) - (13)$. $\diamond$

Proposition 8 *Let $\mathcal{C}$ be a category. Then the opposite category of $\mathcal{C}^{op}$ is $\mathcal{C}$, i.e.*

$$(\mathcal{C}^{op})^{op} = \mathcal{C}$$

Proof
Let $\mathcal{C} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ be a category. From definition (5), we have $\mathcal{C}^{op} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{cod}, \mathrm{dom}, \mathrm{id}, \circ')$ and consequently:

$$(\mathcal{C}^{op})^{op} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ'')$$

In order to show that $(\mathcal{C}^{op})^{op} = \mathcal{C}$, by virtue of axiom (4) we simply need to show that the partial maps $\circ, \circ'' : \mathrm{Arr} \times \mathrm{Arr} \to \mathrm{Arr}$ are equal. Given two arrows

f and g, the composition arrow $g \circ'' f$ is defined if and only if $f \circ' g$ is defined, which is itself equivalent to $g \circ f$ being defined. By virtue of axiom (2), both $\circ$ and $\circ''$ are therefore defined on the same collection of arrow tuples (g, f). Furthemore, whenever $g \circ f$ is defined, we have $g \circ'' f = f \circ' g = g \circ f$. Using axiom (3) we conclude that $\circ'' = \circ$ as requested. $\diamond$

1.6 Canonical product of categories

Definition 6 *We call* canonical product *of categories $\mathcal{C}_1$ and $\mathcal{C}_2$ the category denoted $\mathcal{C}_1 \times \mathcal{C}_2$ and defined by $\mathcal{C}_1 \times \mathcal{C}_2 = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ where:*

$$\begin{array}{ll}
(1) & \mathrm{Ob} = \{\ (x_1, x_2) \mid x_1 \in \mathrm{Ob}\ \mathcal{C}_1\ ,\ x_2 \in \mathrm{Ob}\ \mathcal{C}_2\ \} \\
(2) & \mathrm{Arr} = \{\ (f_1, f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}_1\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}_2\ \} \\
(3) & \mathrm{dom}(f_1, f_2) = (\,\mathrm{dom}(f_1)\,,\,\mathrm{dom}(f_2)\,) \\
(4) & \mathrm{cod}(f_1, f_2) = (\,\mathrm{cod}(f_1)\,,\,\mathrm{cod}(f_2)\,) \\
(5) & \mathrm{id}(x_1, x_2) = (\,\mathrm{id}(x_1)\,,\,\mathrm{id}(x_2)\,) \\
(6) & (g_1, g_2) \circ (f_1, f_2) = (\,g_1 \circ f_1\,,\,g_2 \circ f_2\,)
\end{array}$$

where (3) *and* (4) *hold for all $f_1 \in \mathrm{Arr}\ \mathcal{C}_1$ and $f_2 \in \mathrm{Arr}\ \mathcal{C}_2$,* (5) *holds for all $x_1 \in \mathrm{Ob}\ \mathcal{C}_1$ and $x_2 \in \mathrm{Ob}\ \mathcal{C}_2$, and* (6) *holds for all $f_1, g_1 \in \mathrm{Arr}\ \mathcal{C}_1$ and $f_2, g_2 \in \mathrm{Arr}\ \mathcal{C}_2$ for which $g_1 \circ f_1$ and $g_2 \circ f_2$ are defined.*

So if $\mathcal{C}_1$ and $\mathcal{C}_2$ are two categories, the objects of $\mathcal{C}_1 \times \mathcal{C}_2$ are the collection of all tuples (x_1, x_2) where x_1 is an object of $\mathcal{C}_1$ and x_2 is an object of $\mathcal{C}_2$. The set comprehension notation $\{\ (x_1, x_2) \mid x_1 \in \mathrm{Ob}\ \mathcal{C}_1\ ,\ x_2 \in \mathrm{Ob}\ \mathcal{C}_2\ \}$ is of course an abuse of notation as it does not in general represent a set but a collection. We could also have denoted this collection $\mathrm{Ob}\ \mathcal{C}_1 \times \mathrm{Ob}\ \mathcal{C}_2$ using a cartesian product notation, keeping in mind that this is a product of two collections.

Similarly, the arrows of $\mathcal{C}_1 \times \mathcal{C}_2$ are the collection of all tuples (f_1, f_2) where f_1 is an arrow of $\mathcal{C}_1$ and f_2 is an arrow of $\mathcal{C}_2$, a collection which could reasonably be denoted $\mathrm{Arr}\ \mathcal{C}_1 \times \mathrm{Arr}\ \mathcal{C}_2$ instead of the set-comprehension notation.

It should be clear from definition (6) that the notations dom, cod, id and $\circ$ are overloaded, referring either to $\mathcal{C}_1$, $\mathcal{C}_2$ or $\mathcal{C}_1 \times \mathcal{C}_2$. Given our definitions of $\mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2)$ and $\mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2)$, given that we have $\mathrm{dom} : \mathrm{Arr}\ \mathcal{C}_1 \to \mathrm{Ob}\ \mathcal{C}_1$ and $\mathrm{dom} : \mathrm{Arr}\ \mathcal{C}_2 \to \mathrm{Ob}\ \mathcal{C}_2$ it should be clear that (3) of definition (6) defines a map $\mathrm{dom} : \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2)$, and $\mathrm{cod} : \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2)$ follows from (4). Furthermore from $\mathrm{id} : \mathrm{Ob}\ \mathcal{C}_1 \to \mathrm{Arr}\ \mathcal{C}_1$ and $\mathrm{id} : \mathrm{Ob}\ \mathcal{C}_2 \to \mathrm{Arr}\ \mathcal{C}_2$ we obtain $\mathrm{id} : \mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2)$ using (5). Finally using (6), given the partial maps $\circ : \mathrm{Arr}\ \mathcal{C}_1 \times \mathrm{Arr}\ \mathcal{C}_1 \to \mathrm{Arr}\ \mathcal{C}_1$ and $\circ : \mathrm{Arr}\ \mathcal{C}_2 \times \mathrm{Arr}\ \mathcal{C}_2 \to \mathrm{Arr}\ \mathcal{C}_2$ we obtain a partial map $\circ : \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2) \times \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Arr}\ \mathcal{C}_1 \times \mathcal{C}_2$.

Proposition 9 *The canonical product $\mathcal{C}_1 \times \mathcal{C}_2$ of definition (6) is a category.*

Proof
We need to check that the data $\mathcal{C}_1 \times \mathcal{C}_2 = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ of definition (6) forms a category, having assumed $\mathcal{C}_1$ and $\mathcal{C}_2$ are categories. We have

established that Ob and Arr are collections, that dom, cod, id are maps with the appropriate signatures and $\circ$ is a partial map with the appropriate signature. It remains to check properties (7) – (13) of definition (2).

(7): Let $f, g \in \text{Arr } (\mathcal{C}_1 \times \mathcal{C}_2)$. We need to show that $g \circ f$ is defined if and only if $\text{cod}(f) = \text{dom}(g)$. Let $f_1, g_1 \in \text{Arr } \mathcal{C}_1$ and $f_2, g_2 \in \text{Arr } \mathcal{C}_2$ such that $f = (f_1, f_2)$ and $g = (g_1, g_2)$. Then $\text{cod}(f) = (\text{cod}(f_1), \text{cod}(f_2))$ and $\text{dom}(g) = (\text{dom}(g_1), \text{dom}(g_2))$. So we need to show that $g \circ f$ is defined if and only if $\text{cod}(f_1) = \text{dom}(g_1)$ and $\text{cod}(f_2) = \text{dom}(g_2)$. However since $\mathcal{C}_1$ and $\mathcal{C}_2$ are categories, $\text{cod}(f_1) = \text{dom}(g_1)$ is equivalent to $g_1 \circ f_1$ being defined, and $\text{cod}(f_2) = \text{dom}(g_2)$ is equivalent to $g_2 \circ f_2$ being defined. So we need to show that $g \circ f$ is defined if and only if both $g_1 \circ f_1$ and $g_2 \circ f_2$ are defined which follows exactly from definition (6).

(8): Let $f = (f_1, f_2)$ and $g = (g_1, g_2)$ be arrows in $\mathcal{C}_1 \times \mathcal{C}_2$ with the equality $\text{cod}(f) = \text{dom}(g)$, i.e. for which $g_1 \circ f_1$ and $g_2 \circ f_2$ are defined. We need to check the equality $\text{dom}(g \circ f) = \text{dom}(f)$ which goes as follows:

$$\begin{array}{rcl}
\text{dom}(g \circ f) & = & \text{dom}(\,(g_1, g_2) \,\circ\, (f_1, f_2)\,) \\
\text{(6) of def. (6)} \rightarrow & = & \text{dom}(\,(g_1 \circ f_1 \,,\, g_2 \circ f_2)\,) \\
\text{(3) of def. (6)} \rightarrow & = & (\,\text{dom}(g_1 \circ f_1)\,,\, \text{dom}(g_2 \circ f_2)\,) \\
\mathcal{C}_1,\, \mathcal{C}_2 \text{ categories, (8) of def. (2)} \rightarrow & = & (\,\text{dom}(f_1)\,,\, \text{dom}(f_2)\,) \\
\text{(3) of def. (6)} \rightarrow & = & \text{dom}(f_1, f_2) \\
 & = & \text{dom}(f)
\end{array}$$

(9): Let $f = (f_1, f_2)$ and $g = (g_1, g_2)$ be arrows in $\mathcal{C}_1 \times \mathcal{C}_2$ with the equality $\text{cod}(f) = \text{dom}(g)$, i.e. for which $g_1 \circ f_1$ and $g_2 \circ f_2$ are defined. We need to check the equality $\text{cod}(g \circ f) = \text{cod}(g)$ which goes as follows:

$$\begin{array}{rcl}
\text{cod}(g \circ f) & = & \text{cod}(\,(g_1, g_2) \,\circ\, (f_1, f_2)\,) \\
\text{(6) of def. (6)} \rightarrow & = & \text{cod}(\,(g_1 \circ f_1 \,,\, g_2 \circ f_2)\,) \\
\text{(4) of def. (6)} \rightarrow & = & (\,\text{cod}(g_1 \circ f_1)\,,\, \text{cod}(g_2 \circ f_2)\,) \\
\mathcal{C}_1,\, \mathcal{C}_2 \text{ categories, (9) of def. (2)} \rightarrow & = & (\,\text{cod}(g_1)\,,\, \text{cod}(g_2)\,) \\
\text{(4) of def. (6)} \rightarrow & = & \text{cod}(g_1, g_2) \\
 & = & \text{cod}(g)
\end{array}$$

(10): Let $f = (f_1, f_2)$, $g = (g_1, g_2)$ and $h = (h_1, h_2)$ be arrows in $\mathcal{C}_1 \times \mathcal{C}_2$ with the equalities $\text{cod}(f) = \text{dom}(g)$ and $\text{cod}(g) = \text{dom}(h)$, i.e. for which the composition arrows $g_1 \circ f_1$, $g_2 \circ f_2$, $h_1 \circ g_1$ and $h_2 \circ g_2$ are defined. We need to check the equality $(h \circ g) \circ f = h \circ (g \circ f)$ which goes as follows:

$$\begin{array}{rcl}
(h \circ g) \circ f & = & (\,(h_1, h_2) \,\circ\, (g_1, g_2)\,) \,\circ\, (f_1, f_2) \\
\text{(6) of def. (6)} \rightarrow & = & (\,h_1 \circ g_1 \,,\, h_2 \circ g_2\,) \,\circ\, (f_1, f_2) \\
\text{(6) of def. (6)} \rightarrow & = & (\,(h_1 \circ g_1) \circ f_1 \,,\, (h_2 \circ g_2) \circ f_2\,) \\
\mathcal{C}_1,\, \mathcal{C}_2 \text{ categories, (10) of def. (2)} \rightarrow & = & (\,h_1 \circ (g_1 \circ f_1)\,,\, h_2 \circ (g_2 \circ f_2)\,) \\
\text{(6) of def. (6)} \rightarrow & = & (h_1, h_2) \circ (\,(g_1 \circ f_1)\,,\, (g_2 \circ f_2)\,)
\end{array}$$

$$\begin{aligned}
\text{(6) of def. (6)} \rightarrow \quad &= \quad (h_1, h_2) \circ (\,(g_1, g_2) \circ (f_1, f_2)\,) \\
&= \quad h \circ (g \circ f)
\end{aligned}$$

(11): Let $a = (a_1, a_2)$ be an object in $\mathcal{C}_1 \times \mathcal{C}_2$. We need to check that $\mathrm{dom}\,(\,\mathrm{id}(a)\,) = a = \mathrm{cod}\,(\,\mathrm{id}(a)\,)$ which goes as follows:

$$\begin{aligned}
\mathrm{dom}\,(\,\mathrm{id}(a)\,) \quad &= \quad \mathrm{dom}\,(\,\mathrm{id}(a_1, a_2)\,) \\
\text{(5) of def. (6)} \rightarrow \quad &= \quad \mathrm{dom}\,(\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,) \\
\text{(3) of def. (6)} \rightarrow \quad &= \quad (\,\mathrm{dom}\,(\,\mathrm{id}(a_1)\,)\,,\,\mathrm{dom}\,(\,\mathrm{id}(a_2)\,)\,) \\
\mathcal{C}_1,\, \mathcal{C}_2 \text{ categories, (11) of def. (2)} \rightarrow \quad &= \quad (a_1, a_2) \\
&= \quad a
\end{aligned}$$

$$\begin{aligned}
\mathrm{cod}\,(\,\mathrm{id}(a)\,) \quad &= \quad \mathrm{cod}\,(\,\mathrm{id}(a_1, a_2)\,) \\
\text{(5) of def. (6)} \rightarrow \quad &= \quad \mathrm{cod}\,(\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,) \\
\text{(4) of def. (6)} \rightarrow \quad &= \quad (\,\mathrm{cod}\,(\,\mathrm{id}(a_1)\,)\,,\,\mathrm{cod}\,(\,\mathrm{id}(a_2)\,)\,) \\
\mathcal{C}_1,\, \mathcal{C}_2 \text{ categories, (11) of def. (2)} \rightarrow \quad &= \quad (a_1, a_2) \\
&= \quad a
\end{aligned}$$

(12): Let $f = (f_1, f_2)$ be an arrow and $a = (a_1, a_2)$ be an object in $\mathcal{C}_1 \times \mathcal{C}_2$ such that $\mathrm{dom}(f) = a$. We need to show that $f \circ \mathrm{id}(a) = f$ which goes as follows: Using (3) of definition (6) and the condition $\mathrm{dom}(f) = a$ we obtain the equation $(\,\mathrm{dom}(f_1)\,,\,\mathrm{dom}(f_2)\,) = (a_1, a_2)$. Hence, we have:

$$\begin{aligned}
f \circ \mathrm{id}(a) \quad &= \quad (f_1, f_2) \circ \mathrm{id}\,(a_1, a_2) \\
\text{(5) of def. (6)} \rightarrow \quad &= \quad (f_1, f_2) \,\circ\, (\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,) \\
\text{(6) of def. (6)} \rightarrow \quad &= \quad (\,f_1 \circ \mathrm{id}(a_1)\,,\,f_2 \circ \mathrm{id}(a_2)\,) \\
\mathcal{C}_1 \text{ category, } \mathrm{dom}(f_1) = a_1 \rightarrow \quad &= \quad (f_1,\, f_2 \circ \mathrm{id}(a_2)\,) \\
\mathcal{C}_2 \text{ category, } \mathrm{dom}(f_2) = a_2 \rightarrow \quad &= \quad (f_1, f_2) \\
&= \quad f
\end{aligned}$$

(13): Let $f = (f_1, f_2)$ be an arrow and $a = (a_1, a_2)$ be an object in $\mathcal{C}_1 \times \mathcal{C}_2$ such that $\mathrm{cod}(f) = a$. We need to show that $\mathrm{id}(a) \circ f = f$ which goes as follows: Using (4) of definition (6) and the condition $\mathrm{cod}(f) = a$ we obtain the equation $(\,\mathrm{cod}(f_1)\,,\,\mathrm{cod}(f_2)\,) = (a_1, a_2)$. Hence, we have:

$$\begin{aligned}
\mathrm{id}(a) \circ f \quad &= \quad \mathrm{id}\,(a_1, a_2) \,\circ\, (f_1, f_2) \\
\text{(5) of def. (6)} \rightarrow \quad &= \quad (\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,) \,\circ\, (f_1, f_2) \\
\text{(6) of def. (6)} \rightarrow \quad &= \quad (\,\mathrm{id}(a_1) \circ f_1\,,\,\mathrm{id}(a_2) \circ f_2\,) \\
\mathcal{C}_1 \text{ category, } \mathrm{cod}(f_1) = a_1 \rightarrow \quad &= \quad (f_1,\, \mathrm{id}(a_2) \circ f_2\,) \\
\mathcal{C}_2 \text{ category, } \mathrm{cod}(f_2) = a_2 \rightarrow \quad &= \quad (f_1, f_2) \\
&= \quad f
\end{aligned}$$

This completes our proof of properties (7) – (13). $\diamond$

1.7 Hom-sets of a category

Definition 7 *Let* $\mathcal{C}$ *be a category and* $a, b \in \mathcal{C}$*. We call* hom-set of $\mathcal{C}$ associated with the ordered pair (a, b) *the collection denoted* $\mathcal{C}(a, b)$ *and defined as:*

$$\mathcal{C}(a,b) = \{\ f \in \mathrm{Arr}\ \mathcal{C}\ \mid\ f : a \to b\ \}$$

In other words the collection $\mathcal{C}(a, b)$ is the collection of all arrows f in $\mathcal{C}$ such that $\mathrm{dom}(f) = a$ and $\mathrm{cod}(f) = b$. Note that despite being called a 'hom-set', the collection $\mathcal{C}(a, b)$ is generally not a set but an arbitary collection.

Proposition 10 *Let* $\mathcal{C}$ *be a category and* $a, b \in \mathcal{C}$*. Then* $\mathcal{C}^{op}(a, b) = \mathcal{C}(b, a)$*.*

Proof
Denoting $\mathrm{dom}' = \mathrm{cod}$ and $\mathrm{cod}' = \mathrm{dom}$ we have:

$$\begin{array}{rcl}
\mathcal{C}^{op}(a,b) & = & \{\ f \in \mathrm{Arr}\ \mathcal{C}^{op}\ \mid\ f : a \to b\ @\ \mathcal{C}^{op}\ \} \\
\text{def. (5)} \to & = & \{\ f \in \mathrm{Arr}\ \mathcal{C}\ \mid\ f : a \to b\ @\ \mathcal{C}^{op}\ \} \\
\text{def. (5)} \to & = & \{\ f \in \mathrm{Arr}\ \mathcal{C}\ \mid \mathrm{dom}'(f) = a\ ,\ \mathrm{cod}'(f) = b\ \} \\
 & = & \{\ f \in \mathrm{Arr}\ \mathcal{C}\ \mid \mathrm{cod}(f) = a\ ,\ \mathrm{dom}(f) = b\ \} \\
 & = & \{\ f \in \mathrm{Arr}\ \mathcal{C}\ \mid\ f : b \to a\ @\ \mathcal{C}\ \} \\
 & = & \mathcal{C}(b,a)
\end{array}$$

$\diamond$

Proposition 11 *Let* $\mathcal{C}_1$*,* $\mathcal{C}_2$ *be two categories and* $a, b \in \mathcal{C}_1 \times \mathcal{C}_2$*. Then:*

$$\mathcal{C}_1 \times \mathcal{C}_2\,(a,b) = \mathcal{C}_1(a_1,b_1) \times \mathcal{C}_2(a_2,b_2)$$

where it is understood that $a = (a_1, a_2)$ *and* $b = (b_1, b_2)$*.*

Proof
Let $a = (a_1, a_2)$ and $b = (b_1, b_2)$ be objects in the product category $\mathcal{C}_1 \times \mathcal{C}_2$:

$$\begin{array}{rcl}
\mathcal{C}_1 \times \mathcal{C}_2\,(a,b) & = & \mathcal{C}_1 \times \mathcal{C}_2\,[\,(a_1,a_2)\,,\,(b_1,b_2)\,] \\
\text{def. (7)} \to & = & \{\ f \in \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2)\ \mid\ f : (a_1,a_2) \to (b_1,b_2)\ \} \\
\text{def. (6)} \to & = & \{\ (f_1,f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}_1\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}_2 \\
 & & ,\ (f_1,f_2) : (a_1,a_2) \to (b_1,b_2)\ \} \\
 & = & \{\ (f_1,f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}_1\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}_2 \\
 & & ,\ \mathrm{dom}(f_1,f_2) = (a_1,a_2) \\
 & & ,\ \mathrm{cod}(f_1,f_2) = (b_1,b_2)\ \} \\
\text{(3) and (4) of def. (6)} \to & = & \{\ (f_1,f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}_1\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}_2 \\
 & & ,\ (\,\mathrm{dom}(f_1)\,,\,\mathrm{dom}(f_2)\,) = (a_1,a_2) \\
 & & ,\ (\,\mathrm{cod}(f_1)\,,\,\mathrm{cod}(f_2)\,) = (b_1,b_2)\ \} \\
 & = & \{\ (f_1,f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}_1\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}_2
\end{array}$$

$$\begin{array}{rcl} & & ,\ f_1 : a_1 \to b_1\ ,\ f_2 : a_2 \to b_2\ \} \\ \text{product of collections} \to & = & \{\ f_1 \in \text{Arr}\ \mathcal{C}_1 \mid f_1 : a_1 \to b_1\ \} \\ & \times & \{\ f_2 \in \text{Arr}\ \mathcal{C}_2 \mid f_2 : a_2 \to b_2\ \} \\ \text{def. (7)} \to & = & \mathcal{C}_1(a_1, b_1) \times \mathcal{C}_2(a_2, b_2) \end{array}$$

$\diamond$

1.8 Locally small category

Definition 8 *A category $\mathcal{C}$ is said to be* locally small *if and only if the hom-set $\mathcal{C}(a, b)$ associated with every ordered pair of objects (a, b) is actually a set.*

Proposition 12 *A category $\mathcal{C}$ is locally small if and only if $\mathcal{C}^{op}$ is locally small.*

Proof
The category $\mathcal{C}$ being locally small is equivalent to $\mathcal{C}(a, b)$ being a set for all $a, b \in \text{Ob}\ \mathcal{C}$. Since $\text{Ob}\ \mathcal{C} = \text{Ob}\ \mathcal{C}^{op}$ and $\mathcal{C}^{op}(a, b) = \mathcal{C}(b, a)$ from proposition (10), this is in turn equivalent to $\mathcal{C}^{op}(a, b)$ being a set for all $a, b \in \text{Ob}\ \mathcal{C}^{op}$. Hence, it is equivalent to $\mathcal{C}^{op}$ being locally small. $\diamond$

Proposition 13 *The product $\mathcal{C}_1 \times \mathcal{C}_2$ of locally small categories is locally small.*

Proof
Let $\mathcal{C}_1$ and $\mathcal{C}_2$ be two locally small categories. We need to show that the canonical product $\mathcal{C}_1 \times \mathcal{C}_2$ is itself locally small. In other words, given $a, b \in \mathcal{C}_1 \times \mathcal{C}_2$ we need to show that the collection $\mathcal{C}_1 \times \mathcal{C}_2(a, b)$ is actually a set. However from proposition (11) we have $\mathcal{C}_1 \times \mathcal{C}_2\ (a, b) = \mathcal{C}_1(a_1, b_1) \times \mathcal{C}_2(a_2, b_2)$ where $a = (a_1, a_2)$ and $b = (b_1, b_2)$. So the proposition follows from the fact that both $\mathcal{C}_1(a_1, b_1)$ and $\mathcal{C}_2(a_2, b_2)$ are sets, $\mathcal{C}_1$ and $\mathcal{C}_2$ being locally small. $\diamond$

Chapter 2

Functor

2.1 Functor

Definition 9 *We call* functor *from categories* $\mathcal{C}$ *to* $\mathcal{D}$ *any tuple* (F_0, F_1) *with:*

$$\begin{array}{ll}
(1) & F_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{D} \textit{ is a map} \\
(2) & F_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{D} \textit{ is a map} \\
(3) & F_1(f) : F_0(a) \to F_0(b) \\
(4) & F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F_0(a)\,) \\
(5) & F_1(g \circ f) = F_1(g) \circ F_1(f)
\end{array}$$

where $(3) - (5)$ *hold for all* $a, b, c \in \mathcal{C}$*,* $f : a \to b$ *and* $g : b \to c$*.*

Notation 11 *We shall use* $F : \mathcal{C} \to \mathcal{D}$ *as a notational shortcut for the statement that* F is a functor from the category $\mathcal{C}$ to the category $\mathcal{D}$*.*

Notation 12 *If* $F = (F_0, F_1)$ *if a functor from* $\mathcal{C}$ *to* $\mathcal{D}$*, we shall also commonly denote* F_0 *and* F_1 *simply by* F*.*

So if F is a functor $F : \mathcal{C} \to \mathcal{D}$ we effectively have a map $F : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{D}$ acting on objects, and a map $F : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{D}$ acting on arrows. These two maps satisfy the consistency condition (3) of definition (9) i.e. that if f is an arrow $f : a \to b$ in $\mathcal{C}$, then $F(f)$ must be an arrow $F(f) : F(a) \to F(b)$ in $\mathcal{D}$. Furthermore, the functor F must preserve the identity operators on $\mathcal{C}$ and $\mathcal{D}$ which is condition (4) of definition (9): for all objects $a \in \mathcal{C}$, we must have $F(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F(a)\,)$. Note that since $\mathrm{id}(a) : a \to a$, by consistency we have $F(\,\mathrm{id}(a)\,) : F(a) \to F(a)$, and since $\mathrm{id}(\,F(a)\,) : F(a) \to F(a)$ the equality makes sense. Another way to express the preservation of identity operators by F is simply $F \circ \mathrm{id} = \mathrm{id} \circ F$ or $F_1 \circ \mathrm{id} = \mathrm{id} \circ F_0$ to be more explicit. However, we should remember that the notation '$\circ$' in these equality does not refer to the composition operator $\circ$ of either $\mathcal{C}$ or $\mathcal{D}$, nor does it in general refer to the usual function composition since id, F_0 and F_1 are maps between collections and not functions between sets. Now going back to our functor F, it must also preserve the composition operators on $\mathcal{C}$ and $\mathcal{D}$, which is condition (5) of definition (9): For all objects $a, b, c \in \mathcal{C}$ and arrows $f : a \to b$ and $g : b \to c$, we must have $F(g \circ f) = F(g) \circ F(f)$. Note that given these assumptions, the composition arrow $g \circ f$ is well-defined, and by consistency we have $F(f) : F(a) \to F(b)$ and $F(g) : F(b) \to F(c)$, so $F(g) \circ F(f)$ is also well-defined. Furthermore, since $g \circ f : a \to c$ by consistency we have $F(g \circ f) : F(a) \to F(c)$ and since $F(g) \circ F(f) : F(a) \to F(c)$, the equality $F(g \circ f) = F(g) \circ F(f)$ makes sense.

2.2 Identity functor

Definition 10 *Let $\mathcal{C}$ be a category. We call* identity functor on $\mathcal{C}$ *the functor $F : \mathcal{C} \to \mathcal{C}$ defined by $F = (F_0, F_1)$ with:*

$$\begin{array}{ll} (1) & F_0(a) = a \\ (2) & F_1(f) = f \end{array}$$

where (1) *holds for all $a \in \mathrm{Ob}\ \mathcal{C}$ and* (2) *holds for all $f \in \mathrm{Arr}\ \mathcal{C}$.*

Notation 13 *The identity functor on a category $\mathcal{C}$ is denoted $I_{\mathcal{C}}$.*

We shall now check our definition is indeed that of a functor.

Proposition 14 *The identity functor on a category $\mathcal{C}$ is a functor $I_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$.*

Proof
Let $\mathcal{C}$ be a category and $I_{\mathcal{C}} = (F_0, F_1)$ be the identity functor on $\mathcal{C}$. We need to check that properties $(1) - (5)$ of definition (9) are satisfied:

(1): F_0 is indeed a map $F_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{C}$ from (1) of definition (10).

(2): F_1 is indeed a map $F_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{C}$ from (2) of definition (10).

(3): We have $F_1(f) : F_0(a) \to F_0(b)$ whenever $f : a \to b$, since:

$$F_1(f) = f,\ F_0(a) = a,\ F_0(b) = b$$

(4): We have $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F_0(a)\,)$ for all $a \in \mathcal{C}$, since:

$$F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(a),\ F_0(a) = a$$

(5): $F_1(g \circ f) = F_1(g) \circ F_1(f)$ whenever $f : a \to b$ and $g : b \to c$, since:

$$F_1(g \circ f) = g \circ f,\ F_1(g) = g,\ F_1(f) = f$$

$\diamond$

2.3 Constant functor

Definition 11 *Let $\mathcal{C}$ be a category and $c \in \mathcal{C}$. We call* constant functor on $\mathcal{C}$ at c *the functor $F : \mathcal{C} \to \mathcal{C}$ defined by $F = (F_0, F_1)$ with:*

$$\begin{array}{ll} (1) & F_0(a) = c \\ (2) & F_1(f) = \mathrm{id}(c) \end{array}$$

where (1) *holds for all $a \in \mathrm{Ob}\ \mathcal{C}$ and* (2) *holds for all $f \in \mathrm{Arr}\ \mathcal{C}$.*

Notation 14 *The constant functor on a category $\mathcal{C}$ at $c \in \mathcal{C}$ is denoted K_c.*

We shall now check our definition is indeed that of a functor.

Proposition 15 *Given a category $\mathcal{C}$ and $c \in \mathcal{C}$, K_c is a functor $K_c : \mathcal{C} \to \mathcal{C}$.*

Proof
Let $\mathcal{C}$ be a category, $c \in \mathcal{C}$ and $K_c = (F_0, F_1)$ be the constant functor on $\mathcal{C}$ at c. We need to check that properties (1) – (5) of definition (9) are satisfied:
(1): F_0 is indeed a map $F_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{C}$ from (1) of definition (11).
(2): F_1 is indeed a map $F_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{C}$ from (2) of definition (11).
(3): We have $F_1(f) : F_0(a) \to F_0(b)$ whenever $f : a \to b$, since:

$$F_1(f) = \mathrm{id}(c),\ F_0(a) = c,\ F_0(b) = c$$

(4): We have $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F_0(a)\,)$ for all $a \in \mathcal{C}$, since:

$$F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(c),\ F_0(a) = c$$

(5): $F_1(g \circ f) = F_1(g) \circ F_1(f)$ whenever $f : a \to b$ and $g : b \to c$, since:

$$F_1(g \circ f) = \mathrm{id}(c),\ F_1(g) = \mathrm{id}(c),\ F_1(f) = \mathrm{id}(c)$$

and $\mathrm{id}(c) = \mathrm{id}(c) \circ \mathrm{id}(c)$ from (12) (or (13)) of definition (2). $\diamond$

2.4 Functor and opposite category

Proposition 16 *Let $F : \mathcal{C} \to \mathcal{D}$ be a functor where $\mathcal{C}$ and $\mathcal{D}$ are categories. Then F is also a functor $F : \mathcal{C}^{op} \to \mathcal{D}^{op}$.*

Proof
We need to check that properties (1) – (5) of definition (9) are satisfied:
(1): F_0 is indeed a map $F_0 : \mathrm{Ob}\ \mathcal{C}^{op} \to \mathrm{Ob}\ \mathcal{D}^{op}$ since F is a functor and F_0 is therefore a map $F_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{D}$. Furthermore, we have $\mathrm{Ob}\ \mathcal{C}^{op} = \mathrm{Ob}\ \mathcal{C}$ and $\mathrm{Ob}\ \mathcal{D}^{op} = \mathrm{Ob}\ \mathcal{D}$ by virtue of definition (5).
(2): F_1 is indeed a map $F_1 : \mathrm{Arr}\ \mathcal{C}^{op} \to \mathrm{Arr}\ \mathcal{D}^{op}$ since $\mathrm{Arr}\ \mathcal{C}^{op} = \mathrm{Arr}\ \mathcal{C}$ and $\mathrm{Arr}\ \mathcal{D}^{op} = \mathrm{Arr}\ \mathcal{D}$ from definition (5), and F_1 is a map $F_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{D}$.

(3): We have $F_1(f) : F_0(a) \to F_0(b)$ @ $\mathcal{D}^{op}$ whenever $f : a \to b$ @ $\mathcal{C}^{op}$. Indeed, the assumption $f : a \to b$ @ $\mathcal{C}^{op}$ is equivalent to $f : b \to a$ @ $\mathcal{C}$. F being a functor, this implies $F_1(f) : F_0(b) \to F_0(a)$ @ $\mathcal{D}$ which is equivalent to $F_1(f) : F_0(a) \to F_0(b)$ @ $\mathcal{D}^{op}$.

(4): We have $F_1(\,\mathrm{id}(a)$ @ $\mathcal{C}^{op}) = \mathrm{id}(\,F_0(a)\,)$ @ $\mathcal{D}^{op}$ for all $a \in \mathcal{C}^{op}$. This follows from the equality $F_1(\,\mathrm{id}(a)$ @ $\mathcal{C}) = \mathrm{id}(\,F_0(a)\,)$ @ $\mathcal{D}$ and the fact that the notions of objects and identity coincide on a category and its opposite.

(5): We have $F_1(g \circ f$ @ $\mathcal{C}^{op}) = F_1(g) \circ F_1(f)$ @ $\mathcal{D}^{op}$ when $f : a \to b$ @ $\mathcal{C}^{op}$ and $g : b \to c$ @ $\mathcal{C}^{op}$. Indeed, the assumptions $f : a \to b$ @ $\mathcal{C}^{op}$ and $g : b \to c$ @ $\mathcal{C}^{op}$ are equivalent to $f : b \to a$ @ $\mathcal{C}$ and $g : c \to b$ @ $\mathcal{C}$. Hence we have:

$$
\begin{array}{rcl}
F_1(g \circ f \text{ @ } \mathcal{C}^{op}) & = & F_1(f \circ g \text{ @ } \mathcal{C}) \leftarrow \text{ def. (5)} \\
\text{(5) of def. (9)} \rightarrow & = & F_1(f) \circ F_1(g) \text{ @ } \mathcal{D} \\
\text{def. (5)} \rightarrow & = & F_1(g) \circ F_1(f) \text{ @ } \mathcal{D}^{op}
\end{array}
$$

$\diamond$

2.5 Hom-functor of a locally small category

Definition 12 *Let $\mathcal{C}$ be a locally small category. We call* hom-functor associated with $\mathcal{C}$ *the functor $F : \mathcal{C}^{op} \times \mathcal{C} \to \mathbf{Set}$ defined by $F = (F_0, F_1)$ with:*

$$
\begin{array}{ll}
(1) & F_0(a_1, a_2) = \mathcal{C}(a_1, a_2) \\
(2) & F_1(f_1, f_2)(h) = f_2 \circ h \circ f_1
\end{array}
$$

where (2) hold for $a_1, a_2, b_1, b_2 \in \mathcal{C}$, $f_1 : b_1 \to a_1$, $h : a_1 \to a_2$ and $f_2 : a_2 \to b_2$.

Notation 15 *Given a locally small category $\mathcal{C}$, the hom-functor $F = (F_0, F_1)$ associated with $\mathcal{C}$ is denoted $\mathcal{C} = (\mathcal{C}_0, \mathcal{C}_1)$.*

Remark: Using the notation $\mathcal{C}$ to denote both the locally small category $\mathcal{C}$ and its associated hom-functor may appear confusing, but the notation actally makes sense since the equation $F_0(a_1, a_2) = \mathcal{C}(a_1, a_2)$ simply becomes the tautology $\mathcal{C}(a_1, a_2) = \mathcal{C}(a_1, a_2)$. In other words, using $\mathcal{C}$ to denote the hom-functor makes it very easy to remember that when applied to the object (a_1, a_2) of the category $\mathcal{C}^{op} \times \mathcal{C}$, we simply obtain the hom-set $\mathcal{C}(a_1, a_2)$ of the locally small category $\mathcal{C}$.

Given a locally small category $\mathcal{C}$, definition (12) defines a tuple $F = (F_0, F_1)$ where F_0 appears to be a map defined on $\mathrm{Ob}\,\mathcal{C} \times \mathrm{Ob}\,\mathcal{C}$ with values in **Set**, and F_1 appears to be a map defined on $\mathrm{Arr}\,\mathcal{C} \times \mathrm{Arr}\,\mathcal{C}$ with values in some functional space (since it takes an h as argument). Looking at this, it is far from obvious that definition (12) defines a functor $F : \mathcal{C}^{op} \times \mathcal{C} \to \mathbf{Set}$. Hence we state:

Proposition 17 *Let $\mathcal{C}$ be a locally small category. Then the hom-functor F associated with $\mathcal{C}$ is indeed a functor $F : \mathcal{C}^{op} \times \mathcal{C} \to \mathbf{Set}$.*

Proof
We need to check that properties $(1)-(5)$ of definition (9) are satisfied:

(1): We need to show that F_0 is a map $F_0 : \mathrm{Ob}\ (\mathcal{C}^{op} \times \mathcal{C}) \to \mathrm{Ob}\ \mathbf{Set}$. Having defined $F_0(a_1, a_2) = \mathcal{C}(a_1, a_2)$ and the category $\mathcal{C}$ being locally small, we see that $F_0(a_1, a_2)$ is a set for all $a_1, a_2 \in \mathcal{C}$. So F_0 is defined as a map $F_0 : (\mathrm{Ob}\ \mathcal{C}) \times (\mathrm{Ob}\ \mathcal{C}) \to \mathrm{Ob}\ \mathbf{Set}$, and it remains to check that the collections $(\mathrm{Ob}\ \mathcal{C}) \times (\mathrm{Ob}\ \mathcal{C})$ and $\mathrm{Ob}\ (\mathcal{C}^{op} \times \mathcal{C})$ actually coincide, which goes as follows:

$$\begin{aligned}
(\mathrm{Ob}\ \mathcal{C}) \times (\mathrm{Ob}\ \mathcal{C}) &= \{\ (a_1, a_2) \mid a_1 \in \mathrm{Ob}\ \mathcal{C}\ ,\ a_2 \in \mathrm{Ob}\ \mathcal{C}\} \\
\text{def. (5)} \to \quad &= \{\ (a_1, a_2) \mid a_1 \in \mathrm{Ob}\ \mathcal{C}^{op}\ ,\ a_2 \in \mathrm{Ob}\ \mathcal{C}\} \\
\text{(1) of def. (6)} \to \quad &= \mathrm{Ob}\ (\mathcal{C}^{op} \times \mathcal{C})
\end{aligned}$$

(2): We need to show that F_1 is a map $F_1 : \mathrm{Arr}\ (\mathcal{C}^{op} \times \mathcal{C}) \to \mathrm{Arr}\ \mathbf{Set}$. Having defined $F_1(f_1, f_2)$ for any $f_1 : b_1 \to a_1$ and $f_2 : a_2 \to b_2$ where a_1, a_2, b_1, b_2 are arbitrary objects in $\mathcal{C}$, we see that F_1 is a map defined on $(\mathrm{Arr}\ \mathcal{C}) \times (\mathrm{Arr}\ \mathcal{C})$. So we need to check that the collections $(\mathrm{Arr}\ \mathcal{C}) \times (\mathrm{Arr}\ \mathcal{C})$ and $\mathrm{Arr}\ (\mathcal{C}^{op} \times \mathcal{C})$ actually coincide, which goes as follows:

$$\begin{aligned}
(\mathrm{Arr}\ \mathcal{C}) \times (\mathrm{Arr}\ \mathcal{C}) &= \{\ (f_1, f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}\} \\
\text{def. (5)} \to \quad &= \{\ (f_1, f_2) \mid f_1 \in \mathrm{Arr}\ \mathcal{C}^{op}\ ,\ f_2 \in \mathrm{Arr}\ \mathcal{C}\} \\
\text{(2) of def. (6)} \to \quad &= \mathrm{Arr}\ (\mathcal{C}^{op} \times \mathcal{C})
\end{aligned}$$

However, given arrows f_1, f_2 in $\mathcal{C}$, we still need to check that $F_1(f_1, f_2)$ is a member of the collection $\mathrm{Arr}\ \mathbf{Set}$. In other words, we need to check that $F_1(f_1, f_2)$ is a function. Introducing the notations a_1, a_2, b_1, b_2 such that $f_1 : b_1 \to a_1$ and $f_2 : a_2 \to b_2$, our definition states that $F_1(f_1, f_2)(h)$ is defined for any $h : a_1 \to a_2$. In other words, $F_1(f_1, f_2)(h)$ is defined for any h which belongs to the hom-set $\mathcal{C}(a_1, a_2)$. Having assumed that $\mathcal{C}$ is a locally small category, the collection $\mathcal{C}(a_1, a_2)$ is in fact a set, and $F_1(f_1, f_2)$ is therefore a function with domain $\mathcal{C}(a_1, a_2)$, defined by $F_1(f_1, f_2)(h) = f_2 \circ h \circ f_1$. Note that since $f_1 : b_1 \to a_1$ and $f_2 : a_2 \to b_2$, the composition $f_2 \circ h \circ f_1$ is a well-defined arrow in $\mathcal{C}$ whenever $h \in \mathcal{C}(a_1, a_2)$. This arrow has domain b_1 and codomain b_2 and we see that $F_1(f_1, f_2)(h)$ is in fact an element of the hom-set $\mathcal{C}(b_1, b_2)$. So $F_1(f_1, f_2)$ is actually a function $F_1(f_1, f_2) : \mathcal{C}(a_1, a_1) \to \mathcal{C}(b_1, b_2)$. Now looking at definition (3), an arrow of the category $\mathbf{Set}$ is a triple (a, b, f) where a and b are sets and f is a function $f : a \to b$. Hence, strictly speaking our definition of $F_1(f_1, f_2)$ is not a member of $\mathrm{Arr}\ \mathbf{Set}$ but simply a function $F_1(f_1, f_2) : \mathcal{C}(a_1, a_2) \to \mathcal{C}(b_1, b_2)$. However, the triple $(\mathcal{C}(a_1, a_2), \mathcal{C}(b_1, b_2), F_1(f_1, f_2))$ is an arrow of the category $\mathbf{Set}$ and we have agreed in notation (9) to simply refer to this arrow as $F_1(f_1, f_2)$, as the domain $\mathcal{C}(a_1, a_2)$ and intended codomain $\mathcal{C}(b_1, b_2)$ can easily be inferred from the formula $F_1(f_1, f_2)(h) = f_2 \circ h \circ f_1$ for all $h : a_1 \to a_2$. This completes our proof that F_1 is a map $F_1 : \mathrm{Arr}\ (\mathcal{C}^{op} \times \mathcal{C}) \to \mathrm{Arr}\ \mathbf{Set}$.

(3): We need to check that $F_1(f) : F_0(a) \to F_0(b)$ whenever $a, b \in \mathcal{C}^{op} \times \mathcal{C}$ and $f : a \to b$. So let $a = (a_1, a_2) \in \mathcal{C}^{op} \times \mathcal{C}$, $b = (b_1, b_2) \in \mathcal{C}^{op} \times \mathcal{C}$ and let f be an arrow $f = (f_1, f_2) : (a_1, a_2) \to (b_1, b_2)$ @ $\mathcal{C}^{op} \times \mathcal{C}$. From the equalities $F_0(a) = \mathcal{C}(a_1, a_2)$ and $F_0(b) = \mathcal{C}(b_1, b_2)$, it is clear that we simply need to check

$F_1(f_1, f_2) : \mathcal{C}(a_1, a_2) \to \mathcal{C}(b_1, b_2)$. However, we have already established this fact in part (2) of this proof, provided we show that $f_1 : b_1 \to a_1 \ @\ \mathcal{C}$ together with $f_2 : a_2 \to b_2 \ @\ \mathcal{C}$. Hence we need $(f_1, f_2) \in \mathcal{C}(b_1, a_1) \times \mathcal{C}(a_2, b_2)$, knowing that $(f_1, f_2) \in \mathcal{C}^{op} \times \mathcal{C}[\,(a_1, a_2)\,,\,(b_1, b_2)\,]$ by assumption. It is therefore sufficient to prove that the two collections $\mathcal{C}(b_1, a_1) \times \mathcal{C}(a_2, b_2)$ and $\mathcal{C}^{op} \times \mathcal{C}[\,(a_1, a_2)\,,\,(b_1, b_2)\,]$ coincide, which goes as follows:

$$
\begin{array}{rcl}
\mathcal{C}^{op} \times \mathcal{C}[\,(a_1, a_2)\,,\,(b_1, b_2)\,] & = & \mathcal{C}^{op}(a_1, b_1) \times \mathcal{C}(a_2, b_2) \ \leftarrow \ \text{prop. (11)} \\
\text{prop. (10)} \ \rightarrow & = & \mathcal{C}(b_1, a_1) \times \mathcal{C}(a_2, b_2)
\end{array}
$$

(4): We need to check that $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F_0(a)\,)$ whenever $a \in \mathcal{C}^{op} \times \mathcal{C}$. So let $a = (a_1, a_2) \in \mathcal{C}^{op} \times \mathcal{C}$. Since $F_0(a) = \mathcal{C}(a_1, a_2)$, we need to check that $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,\mathcal{C}(a_1, a_2)\,)$. This is an equality between two arrows of the category **Set**, with identical domain and codomain, namely the set $\mathcal{C}(a_1, a_2)$. Given $h \in \mathcal{C}(a_1, a_2)$, from proposition (6) it is sufficient to check that $F_1(\,\mathrm{id}(a)\,)(h) = h$:

$$
\begin{array}{rcl}
F_1(\,\mathrm{id}(a)\,)(h) & = & F_1(\,\mathrm{id}\,(a_1, a_2)\,)(h) \\
\text{(5) of def. (6)} \ \rightarrow & = & F_1(\,\mathrm{id}(a_1) \ @\ \mathcal{C}^{op}\,,\,\mathrm{id}(a_2)\,)(h) \\
\text{def. (5)} \ \rightarrow & = & F_1(\,\mathrm{id}(a_1) \ @\ \mathcal{C}\,,\,\mathrm{id}(a_2)\,)(h) \\
\text{(2) of def. (12)} \ \rightarrow & = & \mathrm{id}(a_2) \circ h \circ \mathrm{id}(a_1) \\
\text{(12) of def. (2)} \ \rightarrow & = & \mathrm{id}(a_2) \circ h \\
\text{(13) of def. (2)} \ \rightarrow & = & h
\end{array}
$$

(5): We need to check that $F_1(g \circ f) = F_1(g) \circ F_1(f)$ whenever $f : a \to b$, $g : b \to c$ and $a, b, c \in \mathcal{C}^{op} \times \mathcal{C}$. So let $a = (a_1, a_2)$, $b = (b_1, b_2)$, $c = (c_1, c_2)$ be objects in $\mathcal{C}^{op} \times \mathcal{C}$, and $f = (f_1, f_1) : a \to b$ and $g = (g_1, g_2) : b \to c$. We have $g \circ f : a \to c$ and consequently $F_1(g \circ f) : F_0(a) \to F_0(c)$. Hence the arrows $F_1(g \circ f)$ and $F_1(g) \circ F_1(f)$ are two arrows in **Set**, with domain $F_0(a) = \mathcal{C}(a_1, a_2)$ and codomain $F_0(c) = \mathcal{C}(c_1, c_2)$. From proposition (6), in order to prove the equality $F_1(g \circ f) = F_1(g) \circ F_1(f)$ it is therefore sufficient to show that the underlying functions coincide for all $h \in \mathcal{C}(a_1, a_2)$ which goes as follows:

$$
\begin{array}{rcl}
F_1(g \circ f)(h) & = & F_1(\,(g_1, g_2) \,\circ\, (f_1, f_2)\,)(h) \\
\text{(6) of def. (6)} \ \rightarrow & = & F_1(\,g_1 \circ f_1 \ @\ \mathcal{C}^{op},\, g_2 \circ f_2\,)\,(h) \\
\text{crucially, def. (5)} \ \rightarrow & = & F_1(\,f_1 \circ g_1\,,\, g_2 \circ f_2\,)\,(h) \\
\text{(2) of def. (12)} \ \rightarrow & = & (g_2 \circ f_2) \,\circ\, h \,\circ (f_1 \circ g_1) \\
\text{associativity of } \circ \text{ in } \mathcal{C} \ \rightarrow & = & g_2 \,\circ\, (f_2 \circ h \circ f_1) \,\circ g_1 \\
\text{(2) of def. (12)} \ \rightarrow & = & g_2 \,\circ\, F_1(f_1, f_2)(h) \,\circ g_1 \\
\text{(2) of def. (12)} \ \rightarrow & = & F_1(g_1, g_2)(\,F_1(f_1, f_2)(h)\,) \\
 & = & F_1(g)(\,F_1(f)(h)\,) \\
 & = & (F_1(g) \circ F_1(f))(h)
\end{array}
$$

$\diamond$

2.6 Composition of functors

Definition 13 *Let $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{E}$ be functors where $\mathcal{C}$,$\mathcal{D}$,$\mathcal{E}$ are categories. We call* composition of G and F *the functor $H : \mathcal{C} \to \mathcal{E}$ defined by:*

$$\begin{array}{ll} (1) & H_0(a) = G_0(F_0(a)) \\ (2) & H_1(f) = G_1(F_1(f)) \end{array}$$

where (1) *holds for all $a \in \mathrm{Ob}\ \mathcal{C}$ and* (2) *holds for all $f \in \mathrm{Arr}\ \mathcal{C}$.*

Notation 16 *The composition of two functors G and F is denoted GF or $G \circ F$.*

Remark: In view of notation (8), (1) and (2) of definition (13) could equally have been written $H_0 = G_0 \circ F_0$ and $H_1 = G_1 \circ F_1$. Note that the overloaded symbol '$\circ$' has many possible meanings: it is the usual symbol for set-theoretic function composition, it also refers to composition of maps between collections as per notation (8), it is the generic symbol for the composition operator in an arbitrary category as per notation (6), and finally it is also used to denote composition of functors as per notation (16).

Proposition 18 *Let $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{E}$ be functors where $\mathcal{C}$,$\mathcal{D}$,$\mathcal{E}$ are categories. Then $G \circ F$ is indeed a functor $G \circ F : \mathcal{C} \to \mathcal{E}$.*

Proof
We need to check that properties $(1) - (5)$ of definition (9) are satisfied:
(1): $(GF)_0 = G_0 \circ F_0$ is indeed a map $(GF)_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{E}$, since:

$$F_0 : \mathrm{Ob}\ \mathcal{C} \to \mathrm{Ob}\ \mathcal{D},\ G_0 : \mathrm{Ob}\ \mathcal{D} \to \mathrm{Ob}\ \mathcal{E}$$

(2): $(GF)_1 = G_1 \circ F_1$ is indeed a map $(GF)_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{E}$, since:

$$F_1 : \mathrm{Arr}\ \mathcal{C} \to \mathrm{Arr}\ \mathcal{D},\ G_1 : \mathrm{Arr}\ \mathcal{D} \to \mathrm{Arr}\ \mathcal{E}$$

(3): We have $(GF)_1(f) : (GF)_0(a) \to (GF)_0(b)$ whenever $f : a \to b$, since:

$$\begin{array}{rcl} f : a \to b & \Rightarrow & F_1(f) : F_0(a) \to F_0(b) \ \leftarrow \text{(3) of def. (9)} \\ \text{(3) of def. (9)} \ \to & \Rightarrow & G_1(F_1(f)) : G_0(F_0(a)) \to G_0(F_0(b)) \\ \text{(1) and (2) of def. (13)} \ \to & \Leftrightarrow & (GF)_1(f) : (GF)_0(a) \to (GF)_0(b) \end{array}$$

(4): We have $(GF)_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,(GF)_0(a)\,)$ for all $a \in \mathcal{C}$, since:

$$\begin{array}{rcl} (GF)_1(\,\mathrm{id}(a)\,) & = & G_1(F_1(\,\mathrm{id}(a)\,)) \ \leftarrow \ \text{(2) of def. (13)} \\ \text{(4) of def. (9)} \ \to & = & G_1(\,\mathrm{id}(\,F_0(a)\,)\,) \\ \text{(4) of def. (9)} \ \to & = & \mathrm{id}(\,G_0(\,F_0(a)\,)\,) \\ \text{(1) of def. (13)} \ \to & = & \mathrm{id}(\,(GF)_0(a)\,) \end{array}$$

(5): $(GF)_1(g \circ f) = (GF)_1(g) \circ (GF)_1(f)$ whenever $f : a \to b$ and $g : b \to c$:

$$\begin{array}{rcll} (GF)_1(g \circ f) & = & G_1(F_1(\,g \circ f\,)) & \leftarrow \text{(2) of def. (13)} \\ \text{(5) of def. (9)} \ \to & = & G_1(\,F_1(g) \circ F_1(f)\,) & \\ \text{(5) of def. (9)} \ \to & = & G_1(F_1(g)) \circ G_1(F_1(f)) & \\ \text{(2) of def. (13)} \ \to & = & (GF)_1(g) \circ (GF)_1(f) & \end{array}$$

◇

2.7 Canonical product of functors

Definition 14 *We call* canonical product *of two functors $F_1 : \mathcal{C}_1 \to \mathcal{D}_1$ and $F_2 : \mathcal{C}_2 \to \mathcal{D}_2$ the functor $G : \mathcal{C}_1 \times \mathcal{C}_2 \to \mathcal{D}_1 \times \mathcal{D}_2$ defined by $G = (G_0, G_1)$ with:*

$$\begin{array}{ll} (1) & G_0(a_1, a_2) = (\,F_1(a_1)\,,\,F_2(a_2)\,) \\ (2) & G_1(f_1, f_2) = (\,F_1(f_1)\,,\,F_2(f_2)\,) \end{array}$$

where $\mathcal{C}_1, \mathcal{C}_2, \mathcal{D}_1, \mathcal{D}_2$ are arbitrary categories, (1) *holds for all $a_1 \in \mathrm{Ob}\ \mathcal{C}_1$ and $a_2 \in \mathrm{Ob}\ \mathcal{C}_2$, and* (2) *holds for all $f_1 \in \mathrm{Arr}\ \mathcal{C}_1$ and $f_2 \in \mathrm{Arr}\ \mathcal{C}_2$.*

Remark: In accordance with notation (12), we are using the same notations F_1 and F_2 in definition (14) to describe the actions of these functors both on objects and on arrows. The alternative would be to use the notations $(F_1)_0$, $(F_1)_1$, $(F_2)_0$ and $(F_2)_1$ which would arguably be harder to read.

Notation 17 *The canonical product of functors F_1 and F_2 is denoted $F_1 \times F_2$.*

Proposition 19 *Let $F_1 : \mathcal{C}_1 \to \mathcal{D}_1$ and $F_2 : \mathcal{C}_2 \to \mathcal{D}_2$ be two functors. Then the product of F_1 and F_2 is indeed a functor $F_1 \times F_2 : \mathcal{C}_1 \times \mathcal{C}_2 \to \mathcal{D}_1 \times \mathcal{D}_2$.*

Proof
Let $G = (G_0, G_1)$ denote the product functor $F_1 \times F_2$. We need to check that properties $(1) - (5)$ of definition (9) are satisfied, which goes as follows:

(1): G_0 is indeed a map $G_0 : \mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Ob}\ (\mathcal{D}_1 \times \mathcal{D}_2)$: firstly, G_0 is defined on the collection of all (a_1, a_2) where $a_1 \in \mathrm{Ob}\ \mathcal{C}_1$ and $a_2 \in \mathrm{Ob}\ \mathcal{C}_2$. According to definition (6), this is precisely the collection $\mathrm{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2)$. Furthermore $G_0(a_1, a_2)$ is defined as $(\,F_1(a_1)\,,\,F_2(a_2)\,)$ and since $F_1 : \mathcal{C}_1 \to \mathcal{D}_1$ while $F_2 : \mathcal{C}_2 \to \mathcal{D}_2$, we have $F_1(a_1) \in \mathrm{Ob}\ \mathcal{D}_1$ together with $F_2(a_2) \in \mathrm{Ob}\ \mathcal{D}_2$. Hence we see that $G_0(a_1, a_2)$ is indeed a member of the collection $\mathrm{Ob}\ (\mathcal{D}_1 \times \mathcal{D}_2)$.

(2): G_1 is indeed a map $G_1 : \mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2) \to \mathrm{Arr}\ (\mathcal{D}_1 \times \mathcal{D}_2)$: G_1 is defined on the collection of all (f_1, f_2) where $f_1 \in \mathrm{Arr}\ \mathcal{C}_1$ and $f_2 \in \mathrm{Arr}\ \mathcal{C}_2$. According to definition (6), this is precisely the collection $\mathrm{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2)$. Furthermore $G_1(f_1, f_2)$ is defined as $(\,F_1(f_1)\,,\,F_2(f_2)\,)$ and since $F_1 : \mathcal{C}_1 \to \mathcal{D}_1$ while we have $F_2 : \mathcal{C}_2 \to \mathcal{D}_2$, we see that $F_1(f_1) \in \mathrm{Arr}\ \mathcal{D}_1$ and $F_2(f_2) \in \mathrm{Arr}\ \mathcal{D}_2$. Hence we conclude that $G_1(f_1, f_2)$ is indeed a member of the collection $\mathrm{Arr}\ (\mathcal{D}_1 \times \mathcal{D}_2)$.

(3): We need to show that $G_1(f) : G_0(a) \to G_0(b)$ whenever $f : a \to b$: let $f \in \text{Arr}\ (\mathcal{C}_1 \times \mathcal{C}_2)$ such that $\text{dom}(f) = a$ and $\text{cod}(f) = b$. Then $f = (f_1, f_2)$ for some $f_1 \in \text{Arr}\ \mathcal{C}_1$ and $f_2 \in \text{Arr}\ \mathcal{C}_2$. Furthermore, since $a, b \in \text{Ob}\ (\mathcal{C}_1 \times \mathcal{C}_2)$, we have $a = (a_1, a_2)$ and $b = (b_1, b_2)$ for some $a_1, b_1 \in \text{Ob}\ \mathcal{C}_1$ and $a_2, b_2 \in \text{Ob}\ \mathcal{C}_2$:

$$\begin{array}{rcl} (a_1, a_2) & = & a \\ & = & \text{dom}\,(f) \\ & = & \text{dom}\,(f_1, f_2) \\ \text{(3) of def. (6)} \ \rightarrow & = & (\,\text{dom}\,(f_1)\,,\,\text{dom}\,(f_2)\,) \end{array}$$

Hence we have $a_1 = \text{dom}(f_1)$ and $a_2 = \text{dom}(f_2)$ and similarly:

$$\begin{array}{rcl} (b_1, b_2) & = & b \\ & = & \text{cod}\,(f) \\ & = & \text{cod}\,(f_1, f_2) \\ \text{(4) of def. (6)} \ \rightarrow & = & (\,\text{cod}\,(f_1)\,,\,\text{cod}\,(f_2)\,) \end{array}$$

from which we conclude that $b_1 = \text{cod}(f_1)$ and $b_2 = \text{cod}(f_2)$. Hence we see that $f_1 : a_1 \to b_1\ @\ \mathcal{C}_1$ and $f_2 : a_2 \to b_2\ @\ \mathcal{C}_2$. Since F_1 and F_2 are functors, using (3) of definition (9) we obtain $F_1(f_1) : F_1(a_1) \to F_1(b_1)\ @\ \mathcal{D}_1$ and likewise $F_2(f_2) : F_2(a_2) \to F_2(b_2)\ @\ \mathcal{D}_2$. In order to show that $G_1(f) : G_0(a) \to G_0(b)$, since we already know that $G_1(f) \in \text{Arr}(\mathcal{D}_1 \times \mathcal{D}_2)$, it remains to show that $\text{dom}\,(G_1(f)) = G_0(a)$ and $\text{cod}\,(G_1(f)) = G_0(b)$, which goes as follows:

$$\begin{array}{rcl} \text{dom}\,(G_1(f)) & = & \text{dom}\,(\,G_1(f_1, f_2)\,) \\ \text{(2) of def. (14)} \ \rightarrow & = & \text{dom}\,(\,F_1(f_1)\,,\,F_2(f_2)) \\ \text{(3) of def. (6)} \ \rightarrow & = & (\,\text{dom}\,(\,F_1(f_1)\,)\,,\,\text{dom}\,(\,F_2(f_2)\,)\,) \\ F_1(f_1) : F_1(a_1) \to F_1(b_1) \ \rightarrow & = & (\,F_1(a_1)\,,\,\text{dom}\,(\,F_2(f_2)\,)\,) \\ F_2(f_2) : F_2(a_2) \to F_2(b_2) \ \rightarrow & = & (\,F_1(a_1)\,,\,F_2(a_2)\,) \\ \text{(1) of def. (14)} \ \rightarrow & = & G_0(a_1, a_2) \\ & = & G_0(a) \end{array}$$

$$\begin{array}{rcl} \text{cod}\,(G_1(f)) & = & \text{cod}\,(\,G_1(f_1, f_2)\,) \\ \text{(2) of def. (14)} \ \rightarrow & = & \text{cod}\,(\,F_1(f_1)\,,\,F_2(f_2)) \\ \text{(4) of def. (6)} \ \rightarrow & = & (\,\text{cod}\,(\,F_1(f_1)\,)\,,\,\text{cod}\,(\,F_2(f_2)\,)\,) \\ F_1(f_1) : F_1(a_1) \to F_1(b_1) \ \rightarrow & = & (\,F_1(b_1)\,,\,\text{cod}\,(\,F_2(f_2)\,)\,) \\ F_2(f_2) : F_2(a_2) \to F_2(b_2) \ \rightarrow & = & (\,F_1(b_1)\,,\,F_2(b_2)\,) \\ \text{(1) of def. (14)} \ \rightarrow & = & G_0(b_1, b_2) \\ & = & G_0(b) \end{array}$$

(4): We have $G_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,G_0(a)\,)$ for all $a = (a_1, a_2) \in \mathcal{C}_1 \times \mathcal{C}_2$:

$$\begin{array}{rcl}
G_1(\,\mathrm{id}(a)\,) & = & G_1(\,\mathrm{id}(\,a_1, a_2\,)\,) \\
\text{(5) of def. (6)} \;\rightarrow & = & G_1(\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,) \\
\text{(2) of def. (14)} \;\rightarrow & = & (\,F_1(\,\mathrm{id}(a_1)\,)\,,\,F_2(\,\mathrm{id}(a_2)\,)\,) \\
\text{(4) of def. (9)} \;\rightarrow & = & (\,\mathrm{id}(\,F_1(a_1)\,)\,,\,\mathrm{id}(\,F_2(a_2)\,)\,) \\
\text{(5) of def. (6)} \;\rightarrow & = & \mathrm{id}\,(\,F_1(a_1)\,,\,F_2(a_2)\,) \\
\text{(1) of def. (14)} \;\rightarrow & = & \mathrm{id}\,(\,G_0(a_1, a_2)\,) \\
& = & \mathrm{id}\,(G_0(a))
\end{array}$$

(5): We need to show that $G_1(g \circ f) = G_1(g) \circ G_1(f)$ for all $f : a \to b \; @ \; \mathcal{C}_1 \times \mathcal{C}_2$ and $g : b \to c \; @ \; \mathcal{C}_1 \times \mathcal{C}_2$. So let $f = (f_1, f_2)$ and $g = (g_1, g_2)$ with $a = (a_1, a_2)$, $b = (b_1, b_2)$ and $c = (c_1, c_2)$. Following the same details as in (3), we have $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$ and similarly $g_1 : b_1 \to c_1$ and $g_2 : b_2 \to c_2$:

$$\begin{array}{rcl}
G_1(g \circ f) & = & G_1(\,(g_1, g_2) \,\circ\, (f_1, f_2)\,) \\
\text{(6) of def. (6)} \;\rightarrow & = & G_1(\,g_1 \circ f_1\,,\,g_2 \circ f_2\,) \\
\text{(2) of def. (14)} \;\rightarrow & = & (\,F_1(\,g_1 \circ f_1\,)\,,\,F_2(\,g_2 \circ f_2\,)\,) \\
\text{(5) of def. (9)} \;\rightarrow & = & (\,F_1(g_1) \,\circ\, F_1(f_1)\,,\,F_2(g_2) \,\circ\, F_2(f_2)\,) \\
\text{(6) of def. (6)} \;\rightarrow & = & (\,F_1(g_1)\,,\,F_2(g_2)\,) \,\circ\, (\,F_1(f_1)\,,\,F_2(f_2)\,) \\
\text{(2) of def. (14)} \;\rightarrow & = & G_1(g_1, g_2) \,\circ\, G_1(f_1, f_2) \\
& = & G_1(g) \,\circ\, G_1(f)
\end{array}$$

$\diamond$

2.8 Cartesian Functor

Definition 15 *The* cartesian functor $F : \mathbf{Set} \times \mathbf{Set} \to \mathbf{Set}$ *is defined by:*

$$\begin{array}{ll}
(1) & F_0(a_1, a_2) = a_1 \times a_2 \\
(2) & F_1(f_1, f_2)(x_1, x_2) = (\,f_1(x_1)\,,\,f_2(x_2)\,)
\end{array}$$

where (1) *holds for all sets* a_1, a_2*, and* (2) *holds for all sets* a_1, a_2, b_1, b_2 *as well as* $f_1 : a_1 \to b_1$*,* $f_2 : a_2 \to b_2$ *together with* $x_1 \in a_1$ *and* $x_2 \in a_2$*.*

Remark: Recall that given two sets a_1 and a_2, the cartesian product $a_1 \times a_2$ of a_1 and a_2 is the set of all ordered pairs (x_1, x_2) where $x_1 \in a_1$ and $x_2 \in a_2$:

$$a_1 \times a_2 = \{(x_1, x_2) \mid x_1 \in a_1,\; x_2 \in a_2\}$$

Notation 18 *The cartesian functor is denoted* $(\times)$ *as an infix operator.*

Remark: So if $F : \mathbf{Set} \times \mathbf{Set} \to \mathbf{Set}$ is the cartesian functor, we shall typicallly write $a_1 \times a_2$ and $f_1 \times f_2$ instead of $F_0(a_1, a_2)$ and $F_1(f_1, f_2)$ respectively.

Remark: If f_1 and f_2 are arrows of the category **Set**, strictly speaking from definition (3) f_1 and f_2 are tuples (a,b,f) where $f : a \to b$. In particular f_1 and f_2 are sets and the cartesian product $f_1 \times f_2$ is meaningful, so we have a notational conflict with $f_1 \times f_2$, the cartesian functor evaluated at (f_1, f_2). Similarly to the composition $\circ$, the symbol $\times$ is highly overloaded and may also be used to denote the canonical product of two categories as in definition (6), or the canonical product of two functors as in notation (17).

Proposition 20 *The cartesian functor* $(\times)$ *is a functor* $(\times) : \mathbf{Set}\times\mathbf{Set} \to \mathbf{Set}$.

Proof
Let $F = (F_0, F_1)$ denote the cartesian functor $(\times)$. We need to check that properties $(1)-(5)$ of definition (9) are satisfied:

(1): F_0 is indeed a map $F_0 : \mathrm{Ob}\ (\mathbf{Set}\times\mathbf{Set}) \to \mathrm{Ob}\ \mathbf{Set}$, since F_0 is defined on the collection of all (a_1, a_2) where a_1 and a_2 are sets and this collection is indeed the collection of all objects of $\mathbf{Set}\times\mathbf{Set}$. Furthemore $F_0(a_1, a_2) = a_1 \times a_2 \in \mathbf{Set}$.

(2): F_1 is indeed a map $F_1 : \mathrm{Arr}\ (\mathbf{Set}\times\mathbf{Set}) \to \mathrm{Arr}\ \mathbf{Set}$: $F_1(f_1, f_2)$ is defined on the collection of all (f_1, f_2) where $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$ for arbitrary a_1, a_2, b_1, b_2. So it is defined on the collection of all (f_1, f_2) where f_1, f_2 are arrows in **Set**, and this collection is indeed $\mathrm{Arr}\ (\mathbf{Set} \times \mathbf{Set})$. Furthemore, if we have $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$, then $F_1(f_1, f_2)(x_1, x_2)$ is defined for all (x_1, x_2) where $x_1 \in a_1$ and $x_2 \in a_2$. So $F_1(f_1, f_2)$ is a function defined on the cartesian product $a_1 \times a_2$. Since $F_1(f_1, f_2)(x_1, x_2)$ is defined as $(\, f_1(x_1)\,,\, f_2(x_2)\,)$ we see that $F_1(f_1, f_2)$ is in fact a function $F_1(f_1, f_2) : a_1 \times a_2 \to b_1 \times b_2$. However, definition (15) does not spell out the fact that $b_1 \times b_2$ is the intended codomain of $F_1(f_1, f_2)$, but this is pretty clear from the context. Furthermore, definition (15) does not explicitely define an arrow $(a_1 \times a_2, b_1 \times b_2, F_1(f_1, f_2))$ of **Set**, but only its underlying function $F_1(f_1, f_2)$. This is also fair enough given the context and in line with notation (9). We conlude that $F_1(f_1, f_2) \in \mathrm{Arr}\ \mathbf{Set}$.

(3): We need to show that $F_1(f) : F_0(a) \to F_0(b)$ whenever $f : a \to b$: let $f \in \mathrm{Arr}\ (\mathbf{Set} \times \mathbf{Set})$ such that $\mathrm{dom}(f) = a$ and $\mathrm{cod}(f) = b$. Then $a = (a_1, a_2)$ for some sets a_1, a_2, $b = (b_1, b_2)$ for some sets b_1, b_2 and $f = (f_1, f_2)$ for some arrows $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$. We just established in (2) the fact that $F_1(f_1, f_2) : a_1 \times a_2 \to b_1 \times b_2$, which is $F_1(f) : F_0(a) \to F_0(b)$ as requested.

(4): We need to show that $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(\,F_0(a)\,)$ for all $a \in \mathbf{Set} \times \mathbf{Set}$. So let $a = (a_1, a_2)$. We need to show that $F_1(\,\mathrm{id}(a)\,) = \mathrm{id}(a_1 \times a_2)$. We already know that $F_1(\,\mathrm{id}(a)\,) : a_1 \times a_2 \to a_1 \times a_2$. So we simply need to check that the underlying function is the usual identity on the set $a_1 \times a_2$. Let $(x_1, x_2) \in a_1 \times a_2$:

$$
\begin{array}{rcl}
F_1(\,\mathrm{id}(a)\,)(x_1,x_2) & = & F_1(\,\mathrm{id}(a_1,a_2)\,)(x_1,x_2) \\
\text{(5) of def. (6)} \ \to & = & F_1(\,\mathrm{id}(a_1)\,,\,\mathrm{id}(a_2)\,)(x_1,x_2) \\
\text{(2) of def. (15)} \ \to & = & (\,\mathrm{id}(a_1)(x_1)\,,\,\mathrm{id}(a_2)(x_2)\,) \\
\text{(5) of def. (3)} \ \to & = & (\,(a_1,a_1,i(a_1))(x_1)\,,\,(a_2,a_2,i(a_2))(x_2)\,) \\
\text{notation (10)} \ \to & = & (\,i(a_1)(x_1)\,,\,i(a_2)(x_2)\,) \\
i(a)(x) = x \ \to & = & (x_1,x_2)
\end{array}
$$

(5): We need to show that $F_1(g \circ f) = F_1(g) \circ F_1(f)$ for $f : a \to b$ @ $\mathbf{Set} \times \mathbf{Set}$ and $g : b \to c$ @ $\mathbf{Set} \times \mathbf{Set}$. So let $f = (f_1, f_2)$ and $g = (g_1, g_2)$ with $a = (a_1, a_2)$, $b = (b_1, b_2)$ and $c = (c_1, c_2)$. Then we have $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$ and similarly $g_1 : b_1 \to c_1$ and $g_2 : b_2 \to c_2$. We know that $F_1(f) : a_1 \times a_2 \to b_1 \times b_2$ and $F_1(g) : b_1 \times b_2 \to c_1 \times c_2$. Hence using proposition (6), in order to show that $F_1(g \circ f) = F_1(g) \circ F_1(f)$ it is sufficient to prove that the underlying functions coincide on the set $a_1 \times a_2$. Given $(x_1, x_2) \in a_1 \times a_2$, we have:

$$\begin{array}{rcl}
F_1(g \circ f)(x_1, x_2) & = & F_1(\,(g_1, g_2) \circ (f_1, f_2)\,)(x_1, x_2) \\
\text{(6) of def. (6)} \ \to & = & F_1(\, g_1 \circ f_1 \,,\, g_2 \circ f_2 \,)(x_1, x_2) \\
\text{(2) of def. (15)} \ \to & = & (\,(g_1 \circ f_1)(x_1)\,,\, (g_2 \circ f_2)(x_2)\,) \\
\circ \text{ in } \mathbf{Set} \ \to & = & (\, g_1(f_1(x_1))\,,\, g_2(f_2(x_2))\,) \\
\text{(2) of def. (15)} \ \to & = & F_1(g_1, g_2)(\, f_1(x_1)\,,\, f_2(x_2)\,) \\
\text{(2) of def. (15)} \ \to & = & F_1(g_1, g_2)(F_1(f_1, f_2)(x_1, x_2)) \\
\circ \text{ in } \mathbf{Set} \ \to & = & (\, F_1(g_1, g_2) \circ F_1(f_1, f_2)\,)(x_1, x_2) \\
 & = & (\, F_1(g) \circ F_1(f)\,)(x_1, x_2)
\end{array}$$

$\diamond$

2.9 Category of small categories

We are now familiar with the notion of *category* as defined in (2) as well as that of functor as defined in (9). Thanks to definition (13), we know how to *compose functors*, and we also have a notion of *identity functor* as defined in (10). So it is very tempting at this stage to wonder whether the *collection of all categories* could be turned into a category itself, in which the objects are categories and the arrows are functors. However, we know from set theory that assuming the existence of the *set of all sets* leads to a contradiction. Those familiar with proof assistants such that Coq, Agda and Lean will also be used to the idea that *the type of all types* does not exist, as we have *universes* and *type levels* instead. So we shall not attempt to define *the category of all catgeories* here. Instead, we shall focus on a collection which is a lot smaller by considering only those categories which are *small*, as per definition (1).

Definition 16 *We call* **cat** *the category* $\mathbf{cat} = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ *where*

$$\begin{array}{ll}
(1) & \mathrm{Ob} = \{\ \mathcal{C} \mid \mathcal{C} \textit{ is a small category}\ \} \\
(2) & \mathrm{Arr} = \{\ (\mathcal{C}, \mathcal{D}, F) \mid \mathcal{C}, \mathcal{D} \in \mathrm{Ob} \textit{ and } F : \mathcal{C} \to \mathcal{D}\ \} \\
(3) & \mathrm{dom}\,(\mathcal{C}, \mathcal{D}, F) = \mathcal{C} \\
(4) & \mathrm{cod}\,(\mathcal{C}, \mathcal{D}, F) = \mathcal{D} \\
(5) & \mathrm{id}(\mathcal{C}) = (\mathcal{C}, \mathcal{C}, I_{\mathcal{C}}) \\
(6) & (\mathcal{D}, \mathcal{E}, G) \circ (\mathcal{C}, \mathcal{D}, F) = (\mathcal{C}, \mathcal{E}, G \circ F)
\end{array}$$

where (3) − (6) *hold for all small categories* $\mathcal{C}, \mathcal{D}, \mathcal{E}$, *and functors* $F : \mathcal{C} \to \mathcal{D}$ *and* $G : \mathcal{D} \to \mathcal{E}$, $I_{\mathcal{C}}$ *is the identity functor of definition (10) and* $G \circ F$ *is the composition of* G *and* F *of definition (13).*

2.10 Category of locally small categories

TODO

Chapter 3

Natural Transformation

3.1 Natural Transformation

Definition 17 *Let $F, G : \mathcal{C} \to \mathcal{D}$ be functors where $\mathcal{C}$ and $\mathcal{D}$ are categories. We call* natural transformation *from F to G any map α defined on* Ob $\mathcal{C}$ *with:*

$$\begin{array}{ll} (1) & \alpha(a) : F(a) \to G(a) \\ (2) & G(f) \circ \alpha(a) = \alpha(b) \circ F(f) \end{array}$$

where (1) *holds for all $a \in$* Ob $\mathcal{C}$ *and* (2) *holds for all $a, b \in$* Ob $\mathcal{C}$ *and $f : a \to b$.*

Notation 19 *We shall use $\alpha : F \Rightarrow G$ as a notational shortcut for the statement that* α is a natural transformation from the functor F to the functor G.

Remark: A mental picture of a natural transformation $\alpha : F \Rightarrow G$ where F and G are two functors between categories $\mathcal{C}$ and $\mathcal{D}$ is as follows:

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, bend right, "G"'] \arrow[r, phantom, "\Downarrow\alpha"] & \mathcal{D} \end{tikzcd}$$

Remark: Given $F, G : \mathcal{C} \to \mathcal{D}$ and $\alpha : F \Rightarrow G$, given $a, b \in$ Ob $\mathcal{C}$ and $f : a \to b$, since F and G are functors we have $F(f) : F(a) \to F(b)$, $G(f) : G(a) \to G(b)$ and from (1) of definition (17), $\alpha(a) : F(a) \to G(a)$ and $\alpha(b) : F(b) \to G(b)$. It follows that both arrows $G(f) \circ \alpha(a)$ and $\alpha(b) \circ F(f)$ are well defined arrows in $\mathcal{D}$ (from $F(a)$ to $G(b)$), and the equality (2) of definition (17) is always meaningful.
Remark: Equality (2) of definition (17) is commonly visualized as:

$$\begin{tikzcd} a \arrow[d, "f"'] & F(a) \arrow[r, "\alpha(a)"] \arrow[d, "F(f)"'] & G(a) \arrow[d, "G(f)"] \\ b & F(b) \arrow[r, "\alpha(b)"'] & G(b) \end{tikzcd}$$

This diagram is called the *naturality square* of the natural transformation α relative to $f : a \to b$. Equality (2) is informally expressed by saying that *the naturality square commutes*, i.e. that both arrows obtained by composition along the two paths from $F(a)$ to $G(b)$ are equal.

Definition 18 *Given $F, G : \mathcal{C} \to \mathcal{D}$ and $\alpha : F \Rightarrow G$, given $a \in \mathcal{C}$ we call* component at a of the natural transformation α*, the arrow $\alpha(a) : F(a) \to G(a)$.*

Remark: The component $\alpha(a)$ of α at $a \in \mathcal{C}$ is an arrow in the category $\mathcal{D}$.

Notation 20 *The component $\alpha(a)$ of α at a is commonly denoted α_a.*

3.2 Identity natural transformation

Definition 19 *Let $F : \mathcal{C} \to \mathcal{D}$ be a functor where $\mathcal{C}, \mathcal{D}$ are categories. We call* identity natural transformation on F*, the natural transformation $\iota_F : F \Rightarrow F$:*

$$(1) \qquad \iota_F(a) = \mathrm{id}(\, F(a)\,)$$

where (1) holds for all $a \in \mathcal{C}$.

Remark: A mental picture of the identity natural transformation $\iota_F : F \Rightarrow F$ where F is a functor between categories $\mathcal{C}$ and $\mathcal{D}$ is as follows:

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, bend right, "F"'] \arrow[r, phantom, "\Downarrow \iota_F"] & \mathcal{D} \end{tikzcd}$$

Proposition 21 *The identity natural transformation $\iota_F : F \Rightarrow F$ on a functor $F : \mathcal{C} \to \mathcal{D}$ where $\mathcal{C}, \mathcal{D}$ are categories is indeed a natural transformation.*

Proof
We need to check that ι_F is a map defined on Ob $\mathcal{C}$ which satisfies (1) and (2) of definition (17). The fact that it is map defined on Ob $\mathcal{C}$ is clear, since $\iota_F(a)$ is defined for all $a \in \mathcal{C}$, that is for all $a \in \mathrm{Ob}\,\mathcal{C}$.

(1): We have $\iota_F(a) : F(a) \to F(a)$ for all $a \in \mathcal{C}$, since $\iota_F(a) = \mathrm{id}(\, F(a)\,)$

(2): We need to check that $F(f) \circ \iota_F(a) = \iota_F(b) \circ F(f)$ for all $f : a \to b$ where $a, b \in \mathcal{C}$. So we need to show that $F(f) \circ \mathrm{id}(\, F(a)\,) = \mathrm{id}(\, F(b)\,) \circ F(f)$, which follows from (12) and (13) of definition (2) and the fact that $F(f) : F(a) \to F(b)$.
$\diamond$

3.3 Composition of natural transformations

Definition 20 *Let $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$ be natural transformations where F, G and H are functors $F, G, H : \mathcal{C} \to \mathcal{D}$ and $\mathcal{C}$, $\mathcal{D}$ are categories. We call* composition of β and α *the natural transformation $\beta \circ \alpha : F \Rightarrow H$ with:*

$$(1) \qquad (\beta \circ \alpha)(a) = \beta(a) \,\circ\, \alpha(a)$$

where (1) holds for all $a \in \mathcal{C}$.

Remark: If $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$ are natural transformations then for all $a \in \mathcal{C}$ we have $\alpha(a) : F(a) \to G(a)$ and $\beta(a) : G(a) \to H(a)$, and $\beta(a) \circ \alpha(a)$ is therefore a well-defined arrow in $\mathcal{D}$.

Proposition 22 *Let $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$ be natural transformations where F, G and H are functors $F, G, H : \mathcal{C} \to \mathcal{D}$ and $\mathcal{C}$, $\mathcal{D}$ are categories. Then $\beta \circ \alpha$ is indeed a natural transformation $\beta \circ \alpha : F \Rightarrow H$.*

Proof
We need to check that $\beta \circ \alpha$ is a map defined on Ob $\mathcal{C}$ which satisfies (1) and (2) of definition (17). As noted above, since $\alpha(a) : F(a) \to G(a)$ and $\beta(a) : G(a) \to H(a)$, the expression $\beta(a) \circ \alpha(a)$ is a well-defined arrow in $\mathcal{D}$ and $(\beta \circ \alpha)(a)$ is thus well-defined for all $a \in \mathcal{C}$. So $\beta \circ \alpha$ is a map defined on Ob $\mathcal{C}$.

(1): We need to check that $(\beta \circ \alpha)(a) : F(a) \to H(a)$ for all $a \in \mathcal{C}$, which is clear since $(\beta \circ \alpha)(a) = \beta(a) \circ \alpha(a)$, $\alpha(a) : F(a) \to G(a)$ and $\beta(a) : G(a) \to H(a)$.

(2): We need $H(f) \circ (\beta \circ \alpha)(a) = (\beta \circ \alpha)(b) \circ F(f)$ for all $a, b \in \mathcal{C}$ and $f : a \to b$, which goes as follows:

$$
\begin{array}{rcll}
H(f) \circ (\beta \circ \alpha)(a) & = & H(f) \circ (\beta(a) \circ \alpha(a)) & \leftarrow \text{ (1) of def. (20)} \\
\circ \text{ associative in } \mathcal{D} \ \rightarrow & = & (H(f) \circ \beta(a)) \circ \alpha(a) & \\
\text{(2) of def. (17)}, \beta : G \Rightarrow H \ \rightarrow & = & (\beta(b) \circ G(f)) \circ \alpha(a) & \\
\circ \text{ associative in } \mathcal{D} \ \rightarrow & = & \beta(b) \circ (G(f) \circ \alpha(a)) & \\
\text{(2) of def. (17)}, \alpha : F \Rightarrow G \ \rightarrow & = & \beta(b) \circ (\alpha(b) \circ F(f)) & \\
\circ \text{ associative in } \mathcal{D} \ \rightarrow & = & (\beta(b) \circ \alpha(b)) \circ F(f) & \\
\text{(1) of def. (20)} \ \rightarrow & = & (\beta \circ \alpha)(b) \circ F(f) &
\end{array}
$$

$\diamond$

Remark: Showing that $\beta \circ \alpha$ is a natural transformation is essentially about proving the equality $H(f) \circ \beta(a) \circ \alpha(a) = \beta(b) \circ \alpha(b) \circ F(f)$, where we no longer care about brackets as composition is associative in $\mathcal{D}$. Informally, this equality amounts to saying that the big rectangle below commutes:

$$
\begin{array}{ccccccc}
a & \quad & F(a) & \xrightarrow{\alpha(a)} & G(a) & \xrightarrow{\beta(a)} & H(a) \\
{\scriptstyle f}\downarrow & & {\scriptstyle F(f)}\downarrow & & \downarrow{\scriptstyle G(f)} & & \downarrow{\scriptstyle H(f)} \\
b & & F(b) & \xrightarrow[\alpha(b)]{} & G(b) & \xrightarrow[\beta(b)]{} & H(b)
\end{array}
$$

We formally proved the commutativity of this rectangle using the commutativity of the individual squares. Forgetting about associativity details:

$$H(f) \circ \beta(a) \circ \alpha(a) = \beta(b) \circ G(f) \circ \alpha(a) = \beta(b) \circ \alpha(b) \circ F(f)$$

We obtain a proof which is a lot simpler.

3.4 Natural transformation paradox

Given two categories $\mathcal{C}$ and $\mathcal{D}$, and a functor $F : \mathcal{C} \to \mathcal{D}$, we saw in proposition (16) that F is also a functor $F : \mathcal{C}^{op} \to \mathcal{D}^{op}$. TODO

3.5 Functor category

In this section, given two categories $\mathcal{C}$ and $\mathcal{D}$, we define a new category denoted $[\mathcal{C}, \mathcal{D}]$ and called the *functor category between* $\mathcal{C}$ *and* $\mathcal{D}$. Heuristically, the functor category between $\mathcal{C}$ and $\mathcal{D}$ is the category in which the objects are the functors between $\mathcal{C}$ and $\mathcal{D}$, and the arrows are the natural transformations between them.

Definition 21 *We call* functor category between the categories $\mathcal{C}$ and $\mathcal{D}$*, the category denoted* $[\mathcal{C}, \mathcal{D}]$ *and defined by* $[\mathcal{C}, \mathcal{D}] = (\mathrm{Ob}, \mathrm{Arr}, \mathrm{dom}, \mathrm{cod}, \mathrm{id}, \circ)$ *where:*

$$
\begin{array}{ll}
(1) & \mathrm{Ob} = \{\ F \mid F : \mathcal{C} \to \mathcal{D}\ \} \\
(2) & \mathrm{Arr} = \{\ (F, G, \alpha) \mid F\ ,\ G : \mathcal{C} \to \mathcal{D} \ \textit{and}\ \alpha : F \Rightarrow G\ \} \\
(3) & \mathrm{dom}\,(F, G, \alpha) = F \\
(4) & \mathrm{cod}\,(F, G, \alpha) = G \\
(5) & \mathrm{id}(F) = (F, F, \iota_F) \\
(6) & (G, H, \beta) \circ (F, G, \alpha) = (F, H, \beta \circ \alpha)
\end{array}
$$

where $(3)-(6)$ *hold for all functors* $F, G, H : \mathcal{C} \to \mathcal{D}$ *and natural transformations* $\alpha : F \Rightarrow G$ *and* $\beta : G \Rightarrow H$*,* $\iota_F : F \Rightarrow F$ *is the identify natural transformation of definition (19) and* $\beta \circ \alpha$ *is the composition of* β *and* α *of definition (20).*

Chapter 4

Adjunction

4.1 Definition

Definition 22 *We call* adjunction *an ordered pair (F, G) where F is a functor $F : \mathcal{C} \to \mathcal{D}$ and G is a functor $G : \mathcal{D} \to \mathcal{C}$ while $\mathcal{C}$ and $\mathcal{D}$ are two locally-small categories for which there exists a natural isomorphism:*

$$\alpha \ : \ \mathcal{D} \circ (F \times I_{\mathcal{D}}) \ \Rightarrow \ \mathcal{C} \circ (I_{\mathcal{C}^{op}} \times G)$$

in the functor category $[\, \mathcal{C}^{op} \times \mathcal{D} \,, \, \mathbf{Set} \,]$, where F also denotes $F : \mathcal{C}^{op} \to \mathcal{D}^{op}$.

Bibliography

[1] Richard Bird, Oege de Moor, (1997). Algebra of Programming