

Chapter 1

Mathematics = Set Theory?

1.1 Where Set Theory Cracks

If it starts to feel like you're working with a system that's *too* powerful, try this: turn the system on itself.

Maybe you'll feed the system to itself, as in the proof of the undecidability of the Halting Problem. Maybe you'll reduce its power to a handful of (say) 3 properties and show that any 2 precludes the possibility of the 3rd:

- the CAP theorems for distributed databases
- Kleinberg's impossibility theorem on clustering algorithms
- Arrow's impossibility theorem on social choice

In this case we simply hold up the mirror to the system. If we can ask for the set of all sets, we can quickly reach a dead end in the form of Russell's paradox.

We have two different trap doors that let us squeeze tighter and keep moving forward.

Zermelo–Fraenkel set theory tightens the standard on set comprehensions. And it makes sense. It's hard to imagine what the first line of 'bad' Python would do:

```
bad_set = {a if p(a)}  
good_set = {b for b in source_set if p(b)}
```

The von Neumann–Bernays–Gödel (NBG) adds to ZF an abstract notion of 'class' allowing us to talk about the 'class of all sets'. Both provide a strong foundation for modern math, but category theory has emerged as a more abstract foundation.

Chapter 2

What Categories Are

2.1 Functions as Sets

We are only going to wallow in the swamp of Set Theory for a brief moment longer. Given a function $f : A \rightarrow B$ we can derive some related sets.

Definition 2.1.1. The *relation*

$$\hat{f} := \{\langle a, f(a) \rangle \mid a \in A\}.$$

Definition 2.1.2. The *image set*

$$f(A) := \{b \in B \mid b = f(a) \text{ for some } a \in A\}.$$

Equivalently

$$f(A) := \{b \mid \langle a, b \rangle \in \hat{f} \text{ for some } a \in A\}.$$

2.2 Composition

The power of composition is that it can't be resisted. Say I hand you an $f : A \rightarrow B$ and a $g : B \rightarrow C$. Then there is a clear procedure for getting from A to C .

- (1) Take your $a \in A$.
- (2) Apply f to a , yielding $f(a)$ which is some $b \in B$.
- (3) Apply g to b , yielding $g(b)$, which is some $c \in C$

This procedure yields the

Definition 2.2.1. *Function composition*

$$g \circ f.$$

Fact 2.2.2. *Functions are associative, or*

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

Exercise 2.2.1. Convince yourself that functions are associative.

2.2.1 Identities

Given any set B there is an important function that comes for free. We call it $1_B : B \rightarrow B$ and it's given by

$$1_B(b) = b$$

for any $b \in B$.

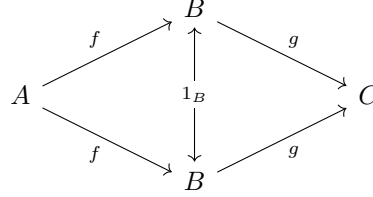
Fact 2.2.3. *Identities are absorbed left and right. In other words, if we have $f : A \rightarrow B$ and a $g : B \rightarrow C$, then*

$$1_B \circ f = f$$

and

$$g \circ 1_B = g.$$

That gives us our first commutative diagram, for which any path is equivalent:



2.3 Category Axioms

We are ready for the abstract view of the above.

Definition 2.3.1. A *category* has

1. a collection of objects:

$$a, b, c.$$

2. a collection of arrows, each with specific domain and codomain:

$$f : a \rightarrow b,$$

$$g : b \rightarrow c.$$

3. an associative composition operation that yields a unique arrow ‘skipping’ aligned domains and codomains:

$$g \circ f : a \rightarrow c.$$

4. an identity arrow for each object:

$$f \circ 1_A = f$$

$$1_B \circ f = f$$

$$g \circ 1_B = g$$

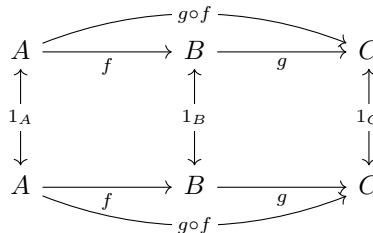
$$1_C \circ g = g$$

For now we will assume that the collection of arrows between any two objects is ‘only’ a set’s worth. In other words we will only deal with *locally small* categories for now. (Riehl Page 7)

Fact 2.3.2. *Diagrams are a convenient way to present categories:*

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

But you must keep in mind there are some implicit things not being shown here. The above is a compact version of the following:



Fact 2.3.3. Saying a diagram ‘commutes’ is a convenient way to present equivalent compositions of arrows. For example, if

$$A \xrightarrow{f} B \xrightarrow{g} C \quad \text{with a curved arrow } h \text{ from } A \text{ to } C$$

commutes then we are saying that

$$g \circ f = h.$$

Definition 2.3.4. Given two objects a, b in a category C , we define the *hom-set*

$$C(a, b) \quad \text{or} \quad \text{hom}(a, b)$$

as the set of all arrows from a to b .

2.4 Comma Categories

Definition 2.4.1. The comma-category $C \downarrow a$ is formed from any category C and any C -object a . Its objects are all the C -arrows with codomain a . (IE, $f_1 : c_1 \rightarrow a$ and $f_2 : c_2 \rightarrow a$). Its arrows are all C -arrows between the objects’ domains, that commute with the ‘object arrows’. (IE, $g : c_1 \rightarrow c_2$ so that $f_1 = f_2 \circ g$). So $C \downarrow a$ looks like this:

$$f_1 \xrightarrow{g} f_2$$

and indicates that this diagram commutes in the original category:

$$\begin{array}{ccc} c_1 & \xrightarrow{g} & c_2 \\ & \searrow f_1 & \swarrow f_2 \\ & a & \end{array}$$

(TODO: Verify category axioms)

Example. Take C to be the preorder on natural numbers, and a to be a given number. For example, let $a = 3$.

$$1 \longrightarrow 2 \longrightarrow \mathbf{3} \longrightarrow 4 \longrightarrow 5 \longrightarrow \dots$$

Objects in $C \downarrow 3$ are statements of ‘ $n \leq 4$ ’ relationships. Arrows in $C \downarrow 3$ from ‘ $m \leq 3$ ’ to ‘ $n \leq 3$ ’ are ‘ $m \leq n$ ’.

$$(1 \leq 3) \xrightarrow{1 \leq 2} (2 \leq 3) \xrightarrow{2 \leq 3} (3 \leq 3) \quad \text{with a curved arrow } 1 \leq 3 \text{ from } (1 \leq 3) \text{ to } (3 \leq 3)$$

Example. Recall $\mathbf{Matr}(\mathbf{k})$ has the natural numbers \mathbb{N} as objects, and $(n \times m)$ matrices as arrows from $m \rightarrow n$.

Then $\mathbf{Matr}(\mathbf{k}) \downarrow 3$ has as objects all $3 \times n$ matrices where $n \in \mathbb{N}$.

Then if object A is a $3 \times m$ matrix, and object B is a $3 \times n$ matrix, then an arrow $C : A \rightarrow B$ is an $n \times m$ matrix such that $A = BC$.

Thus the situation in $\mathbf{Matr}(\mathbf{k}) \downarrow 3$ is

$$A_{3 \times m} \xrightarrow{C_{n \times m}} B_{3 \times n}$$

indicating that this diagram commutes in $\mathbf{Matr}(\mathbf{k})$:

$$\begin{array}{ccc} m & \xrightarrow{C_{n \times m}} & n \\ & \searrow A_{3 \times m} & \swarrow B_{3 \times n} \\ & 3 & \end{array}$$

Chapter 3

Arrows Instead of Epsilon

3.1 Monic arrows

Monic arrows are an abstraction of injective functions.

Definition 3.1.1. An arrow $f : a \rightarrow b$ in a category C is *monic* if for any g_1, g_2 with codomain a , the implication

$$f \circ g_1 = f \circ g_2 \implies g_1 = g_2$$

holds. Or, if the diagram

$$c \begin{array}{c} \xrightarrow{g_1} \\ \xrightarrow{g_2} \end{array} a \xrightarrow{f} b$$

commutes, then $g_1 = g_2$.

Definition 3.1.2. (Alternatively, Riehl pg. 11) An arrow $f : a \rightarrow b$ is *monic* iff for any C -object c , post-composition with f defines an injection $f_* : C(c, a) \rightarrow C(c, b)$. (Here $C(x, y)$ is the set of C -arrows from x to y .)

For both exercises in this section, take the situation to be as follows:

$$s \begin{array}{c} \xrightarrow{h_1} \\ \xrightarrow{h_2} \end{array} a \xrightarrow{f} b \xrightarrow{g} c$$

Where f and g are fixed, and s, h_1, h_2 are ‘any such’ objects/arrows.

Exercise 3.1.1.

Suppose that f and g are both monic, and that $g \circ (f \circ h_1) = g \circ (f \circ h_2)$. Since g is monic, that implies $f \circ h_1 = f \circ h_2$. But since f is monic, that implies $h_1 = h_2$. So using associativity and collapsing the chain of implication gives

$$(g \circ f) \circ h_1 = (g \circ f) \circ h_2 \implies h_1 = h_2.$$

Conclude $g \circ f$ is monic.

Exercise 3.1.2.

Now suppose that $g \circ f$ is monic. If $f \circ h_1 = f \circ h_2$ then clearly $g \circ (f \circ h_1) = g \circ (f \circ h_2)$. Then $(g \circ f) \circ h_1 = (g \circ f) \circ h_2$ and since $g \circ f$ is monic, $h_1 = h_2$. So

$$f \circ h_1 = f \circ h_2 \implies h_1 = h_2,$$

meaning f is monic.

3.2 Epic arrows

Definition 3.2.1. If f is *epic* then commutativity of

$$a \xrightarrow{f} b \begin{array}{c} \xrightarrow{g_1} \\ \xrightarrow{g_2} \end{array} c$$

implies $g_1 = g_2$.

Definition 3.2.2. (Alternatively, Riehl pg. 11) An arrow $f : a \rightarrow b$ is *epic* iff for any C -object c , pre-composition with f defines an injection $f^* : C(b, c) \rightarrow C(a, c)$. (Here $C(x, y)$ is the set of C -arrows from x to y .)

Dually to the exercises proven in the previous section we have

Fact 3.2.3. If $f : a \rightarrow b$ and $g : b \rightarrow c$ are epic, then $g \circ f : a \rightarrow c$ is epic.

Fact 3.2.4. If $g \circ f : a \rightarrow c$ is epic, then $g : b \rightarrow c$ is epic.

3.3 Iso arrows

Definition 3.3.1. An arrow $f : a \rightarrow b$ is *iso* if there exists another arrow $f^{-1} : b \rightarrow a$ such that

$$f \circ f^{-1} = 1_b$$

and

$$f^{-1} \circ f = 1_a.$$

This diagram commutes when the identity loops are included:

$$\begin{array}{ccc} & f & \\ a & \xrightarrow{\quad} & b \\ & f^{-1} & \end{array}$$

Fact 3.3.2. If an arrow is iso then it is epic and monic, but the converse isn't necessarily true. The converse is true in **Set** and any **Topos**.

Exercise 3.3.1. For any object a , the identity morphism 1_a is an inverse to itself and therefore is iso. Simply because

$$1_a \circ 1_a = 1_a.$$

$$\begin{array}{ccc} & 1_a & \\ a & \xrightarrow{\quad} & a \\ & 1_a & \end{array}$$

Exercise 3.3.2. If $f : a \rightarrow b$ is iso then we can retrieve f^{-1} and then plug it right into the definition and find

$$f^{-1} \circ f = 1_a$$

and

$$f \circ f^{-1} = 1_b,$$

indicating that f^{-1} is iso.

$$\begin{array}{ccc} & f^{-1} & \\ a & \xrightarrow{\quad} & b \\ & f & \end{array}$$

Exercise 3.3.3. With $f : a \rightarrow b$ and $g : b \rightarrow c$ both iso, the situation looks like the following:

$$\begin{array}{ccccc} & f & & g & \\ a & \xrightarrow{\quad} & b & \xrightarrow{\quad} & c \\ & f^{-1} & & g^{-1} & \end{array}$$

Now we find that

$$(f^{-1} \circ g^{-1}) \circ (g \circ f) = f^{-1} \circ (g^{-1} \circ g) \circ f = f^{-1} \circ 1_b \circ f = f^{-1} \circ f = 1_a$$

and

$$(g \circ f) \circ (f^{-1} \circ g^{-1}) = g \circ (f \circ f^{-1}) \circ g^{-1} = g \circ 1_b \circ g^{-1} = g \circ g^{-1} = 1_c.$$

Thus $(f^{-1} \circ g^{-1})$ acts as an inverse to $g \circ f$, and $g \circ f$ is iso.

3.4 Isomorphic objects

Definition 3.4.1. Two C -objects a and b are *isomorphic*, or

$$a \cong b$$

if there exists an iso C -arrow

$$f : a \rightarrow b.$$

Definition 3.4.2. A category C is *skeletal* if $a \cong b$ implies $a = b$.

Exercise 3.4.1.

We wish to show that object isomorphism is an equivalence relation, or that it's reflexive, symmetric, and transitive. Fortunately the exercises from section 3 correspond exactly to these properties.

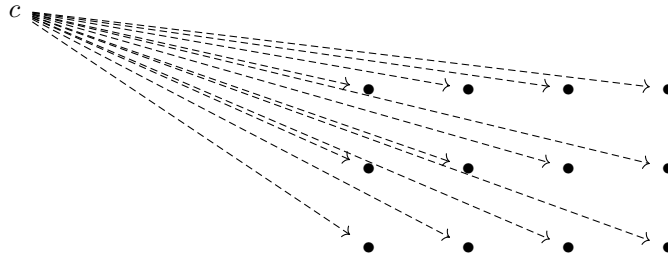
- (i) $a \cong a$ since 1_a is iso.
- (ii) If $a \cong b$ then some $f : a \rightarrow b$ is iso, and therefore $f^{-1} : b \rightarrow a$ is iso and $b \cong a$.
- (iii) If $a \cong b$ and $b \cong c$ then we have iso arrows $f : a \rightarrow b$ and $g : b \rightarrow c$. Then $g \circ f$ is iso, and $a \cong c$.

Exercise 3.4.2.

Suppose a and b are two **Finord**-objects such that $a \cong b$. Then there is some $f : a \rightarrow b$ that is iso. Since **Finord** is a subcategory of **Set**, iso arrows correspond to bijective functions. Then a and b must have the same cardinality, but by the definition of **Finord** distinct objects have distinct cardinalities. So $a = b$ and **Finord** is skeletal.

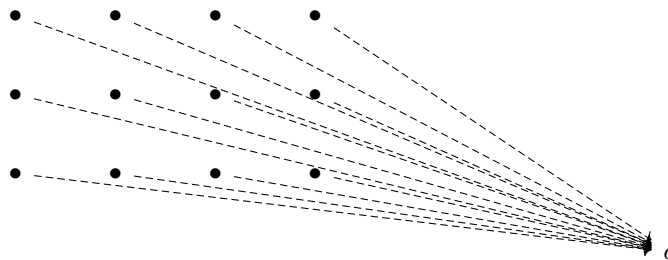
3.5 Initial objects

Definition 3.5.1. An object c is *initial* if for every C -object a , there is exactly one arrow $f : c \rightarrow a$.

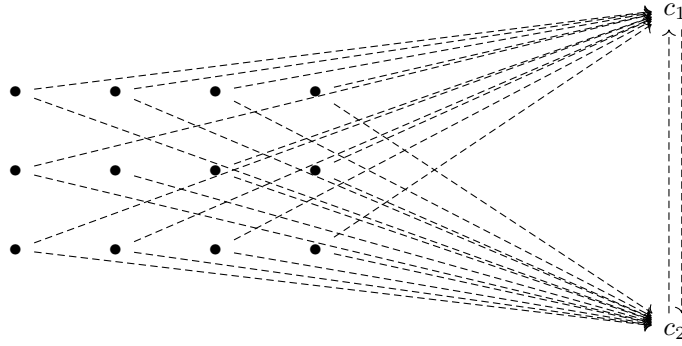


3.6 Terminal objects

Definition 3.6.1. An object c is *terminal* if for every C -object a , there is exactly one arrow $f : a \rightarrow c$.



Exercise 3.6.1. Let c_1 and c_2 be terminal C -objects.



By terminality there is a unique arrow $f_1 : c_1 \rightarrow c_2$ and a unique arrow $f_2 : c_2 \rightarrow c_1$. Then by the category axiom, $f_2 \circ f_1 : c_1 \rightarrow c_1$ and $f_1 \circ f_2 : c_2 \rightarrow c_2$ must exist. But again by terminality, there is a unique arrow $1_{c_1} : c_1 \rightarrow c_1$ and $1_{c_2} : c_2 \rightarrow c_2$, so the composition of f_1 and f_2 must give the identity. Conclude $c_1 \cong c_2$.

Exercise 3.6.2. (i) Terminal objects in \mathbf{Set}^2 are of the form $\langle \{e_1\}, \{e_2\} \rangle$, or pairs of singleton sets.

(ii) Terminal objects in \mathbf{Set}^\rightarrow are arrows with singleton sets as domain and codomain.

(iii) The terminal object in the poset (n, \leq) is the maximal element n , since $m \leq n$ for every m .

Exercise 3.6.3. Suppose $f : 1 \rightarrow a$ has its domain 1 a terminal object, and g_1, g_2 are any two parallel arrows from $c \rightarrow 1$.

$$c \begin{array}{c} \xrightarrow{g_1} \\ \xrightarrow{g_2} \end{array} 1 \xrightarrow{f} a$$

Well, since 1 is terminal the arrow from $c \rightarrow 1$ is unique and we see that $g_1 = g_2$, so regardless of whether $g_1 \circ f = g_2 \circ f$ holds (which it does), we can conclude f is monic.

3.7 Duality

Any category can be turned into its opposite category. So any statement about a category can be dualized with all the arrows reversed.

3.8 Products

Definition 3.8.1. Given C -objects a and b , a *product* is a C -object $a \times b$ and 2 C -arrows pr_a, pr_b .

$$a \xleftarrow{pr_a} a \times b \xrightarrow{pr_b} b$$

For any c, f, g configured as follows

$$\begin{array}{ccccc} & & c & & \\ & f \swarrow & & \searrow g & \\ a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b \end{array}$$

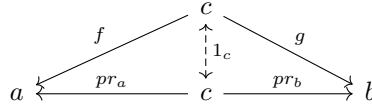
f and g determine a unique $h : c \rightarrow (a \times b)$ so that

$$\begin{array}{ccccc} & & c & & \\ & f \swarrow & \downarrow h & \searrow g & \\ a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b \end{array}$$

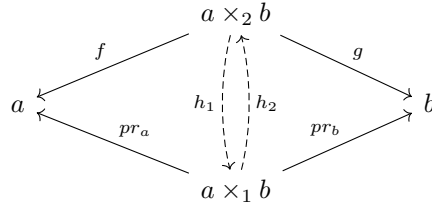
commutes. This is denoted

$$c := \langle f, g \rangle.$$

Fact 3.8.2. If c is a product $a \times b$, any arrow $f : c \rightarrow c$, f must be the identity 1_c . First observe that the identity must exist. Then plug c into definition 3.8.1 to see that f must be the unique arrow with that domain and codomain.

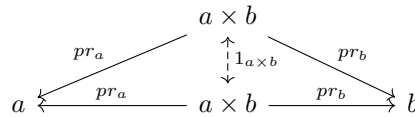


Fact 3.8.3. Any two products of a and b , say $a \times_1 b$ and $a \times_2 b$, are isomorphic to each other. Consider that in the diagram

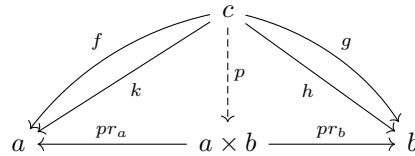


h_1 and h_2 are uniquely determined by symmetric applications of definition 3.8.1. But by fact 3.8.2, composition of h_1 and h_2 must give identities.

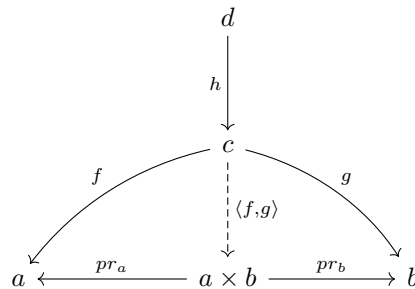
Exercise 3.8.1. The fact that $\langle pr_a, pr_b \rangle = 1_{a \times b}$ follows as a special case of fact 3.8.2, by plugging in the projection functions.



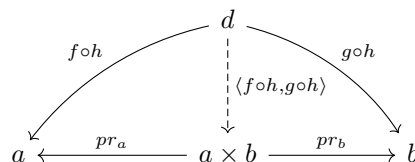
Exercise 3.8.2. Suppose we have parallel $f, k : c \rightrightarrows a$ and $g, h : c \rightrightarrows b$ and $p : c \rightarrow a \times b$ such that $p = \langle f, g \rangle = \langle k, h \rangle$. Then $f = pr_a \circ p$ and $k = pr_a \circ p$. It doesn't take any special cancellation rules to see that identically $f = k$. Similarly $g = h$.



Exercise 3.8.3. Suppose the situation is as follows.



Then by compositionality there must exist $h \circ f : d \rightarrow a$ and $g \circ f : d \rightarrow b$. There also must exist $h \circ \langle f, g \rangle : d \rightarrow a \times b$. By collapsing the diagram to



we see the arrow from $d \rightarrow a \times b$ must be unique, and therefore

$$\langle f \circ h, g \circ h \rangle = h \circ \langle f, g \rangle.$$

Exercise 3.8.4. Suppose a category C has a terminal object t , and products. Let a be a C -object and consider the product $a \times t$.

$$a \longleftarrow a \times t \longrightarrow t$$

Plugging a into the product definition using the arrows given by 1_a and $!$ (the unique arrow to t) yields the unique arrow $h = \langle 1_a, ! \rangle$.

$$\begin{array}{ccccc} & & a & & \\ & \swarrow 1_a & \downarrow h & \searrow ! & \\ a & \xleftarrow{pr_a} & a \times t & \xrightarrow{\quad} & t \end{array}$$

By definition we have that $pr_a \circ h = 1_a$. And since $h \circ pr_a$ maps $a \times t \rightarrow a \times t$ it follows from fact 3.8.2 that $h \circ pr_a = 1_{a \times t}$. Given these two iso arrows we conclude

$$a \cong a \times t.$$

Example. As a specific example of exercise 3.8.4, let's work in **Set** where A is any set, and our terminal set t is any singleton set $\{b\}$.

$$\begin{array}{ccccc} & & A & & \\ & \swarrow 1_a & \downarrow h & \searrow ! & \\ A & \xleftarrow{pr_a} & A \times \{b\} & \xrightarrow{pr_b} & \{b\} \end{array}$$

Now we can explicitly say what all of our functions do:

$$1_a(x) = x$$

$$!(x) = b$$

$$pr_a(\langle a, b \rangle) = a$$

$$pr_b(\langle a, b \rangle) = b$$

$$h(x) = \langle x, b \rangle$$

The fact that $pr_a(h(x)) = x$ and $h(pr_a(\langle a, b \rangle)) = \langle a, b \rangle$ gives the isomorphism

$$A \cong A \times \{b\}.$$

Intuitively, elements in A can be placed in one-to-one correspondence with elements in $A \times \{b\}$ by simply sending x to the tuple $\langle x, b \rangle$.

Definition 3.8.4. Given two products $a \times b$ and $c \times d$, and arrows $f : a \rightarrow c$ and $g : b \rightarrow d$

$$\begin{array}{ccccc} a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b \\ \downarrow f & & & & \downarrow g \\ c & \xleftarrow{pr_c} & c \times d & \xrightarrow{pr_d} & d \end{array}$$

the unique *product arrow* $(f \times g) : a \times b \rightarrow c \times d$ is found as $\langle f \circ pr_a, g \circ pr_b \rangle$.

$$\begin{array}{ccccc}
 a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b \\
 \downarrow f & & \downarrow f \times g & & \downarrow g \\
 c & \xleftarrow{pr_c} & c \times d & \xrightarrow{pr_d} & d
 \end{array}$$

Exercise 3.8.5. In the reflexive case we consider the product arrow $1_a \times 1_b$.

$$\begin{array}{ccccc}
 a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b \\
 \downarrow 1_a & & \downarrow 1_a \times 1_b & & \downarrow 1_b \\
 a & \xleftarrow{pr_a} & a \times b & \xrightarrow{pr_b} & b
 \end{array}$$

By definition we have

$$1_a \times 1_b = \langle 1_a \circ pr_a, 1_b \circ pr_b \rangle = \langle pr_a, pr_b \rangle.$$

Then applying exercise 3.8.1 from here gives the desired result

$$1_a \times 1_b = 1_{a \times b}.$$

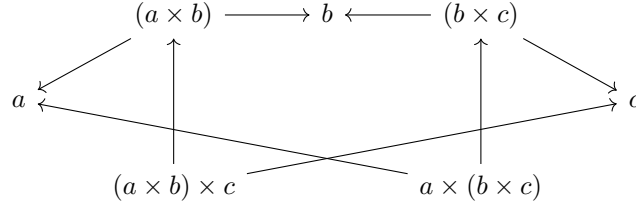
Exercise 3.8.6. The isomorphism $a \times b \cong b \times a$ follows by plugging each object into definition 3.8.1 in relation to the other. In this way the two unique (dashed) arrows are found:

$$\begin{array}{ccccc}
 a & \xleftarrow{\quad} & a \times b & \xrightarrow{\quad} & b \\
 & \searrow & \downarrow & \swarrow & \\
 b & \xleftarrow{\quad} & b \times a & \xrightarrow{\quad} & a
 \end{array}$$

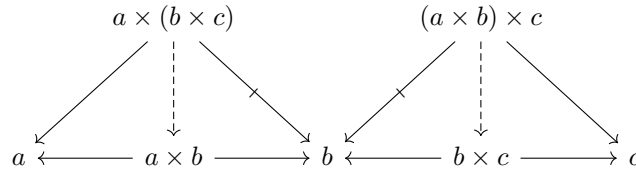
and fact 3.8.2 tells us that they are iso.

Exercise 3.8.7. In this exercise we wish to show that products are associative up to isomorphism. So given C -objects a, b, c ,

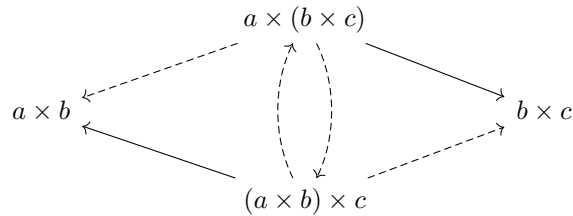
form the products $(a \times b) \times c$ and $a \times (b \times c)$. Here they are with the relevant projection arrows:



We'll use the product definition side-by-side, noting that composition of projections gives us our arrows $a \times (b \times c) \rightarrow b$ and $(a \times b) \times c \rightarrow b$:



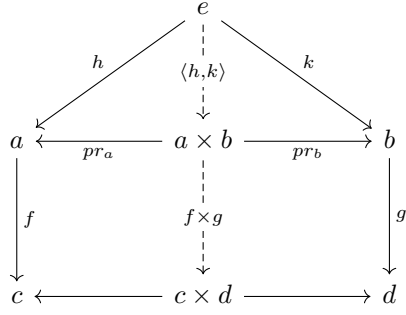
The definition yields unique arrows $a \times (b \times c) \rightarrow b \times c$ and $(a \times b) \times c \rightarrow a \times b$. Using these arrows along-side the 'first-order' projection arrows, we use the product definition again to find the unique arrows



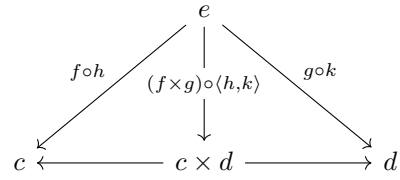
Using fact 3.8.2 gives that $(a \times b) \times c \cong a \times (b \times c)$.

Exercise 3.8.8. Consider the situation of a pair of arrows to the codomain objects of the product arrow's constituent arrows:

(i)

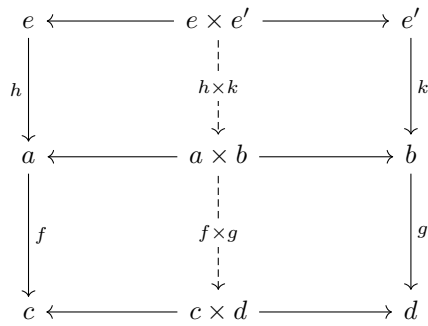


We can use composition to collapse a and b out of the picture:



and thus find that $(f \times g) \circ \langle h, k \rangle$ is the unique arrow determined by $f \circ h$ and $g \circ k$.

(ii) In the situation where we can place two product arrows end-to-end:



we again use composition to collapse the middle level

$$\begin{array}{ccccc}
 e & \xleftarrow{\quad} & e \times e' & \xrightarrow{\quad} & e' \\
 \downarrow f \circ h & & \downarrow (f \times g) \circ (h \times k) & & \downarrow g \circ k \\
 c & \xleftarrow{\quad} & c \times d & \xrightarrow{\quad} & d
 \end{array}$$

Giving (more details needed?)

$$(f \times g) \circ (h \times k) = (f \circ g) \times (h \circ k).$$

3.8.1 Finite Products

Definition 3.8.5. Given a C -object a , the *finite product* (for some integer m) consists of object a^m , and m projection arrows pr_a^m .

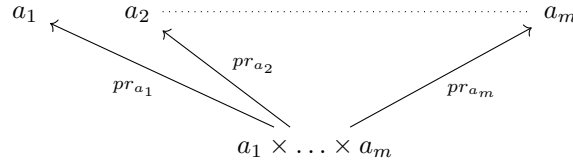
$$\begin{array}{c}
 a \\
 \swarrow \quad \searrow \\
 pr_a^2 \quad \cdots \quad pr_a^m \\
 \swarrow \quad \searrow \\
 a^m
 \end{array}$$

So that for any m parallel arrows $f_i : c \rightarrow a$, there is a unique arrow $\langle f_1, \dots, f_m \rangle : c \rightarrow a^m$ making

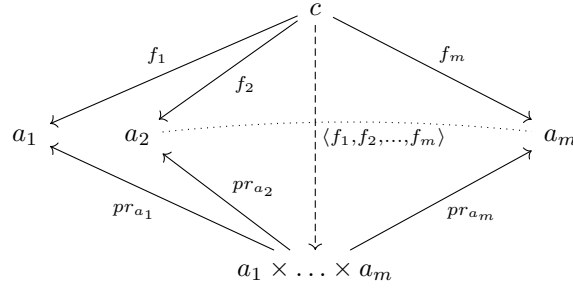
$$\begin{array}{ccc}
 c & & \\
 \downarrow f_1 \quad \downarrow f_2 \quad \cdots \quad \downarrow f_m & & \\
 a & & \\
 \uparrow pr_a^2 \quad \uparrow pr_a^m & & \\
 a^m & & \\
 \swarrow \langle f_1, \dots, f_m \rangle & &
 \end{array}$$

commute.

Definition 3.8.6. A more *general finite product* of m (not necessarily different) C -objects $a_1 \times a_2 \times \dots \times a_m$ consists of a C -object and m projection arrows:

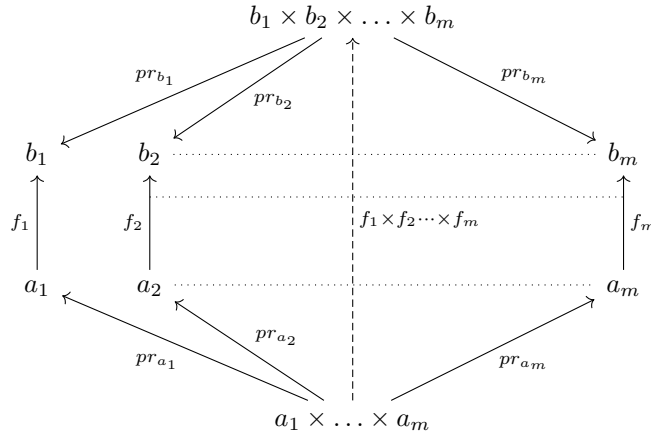


So that for any c -object with m arrows $f_1 : c \rightarrow a_1$, $f_2 : c \rightarrow a_2$, etc, there is a unique arrow $\langle f_1, f_2, \dots, f_m \rangle$ making



commute.

Definition 3.8.7. A *general product arrow* is given by a family of m mappings between the components of two general products.



The product arrow $f_1 \times f_2 \times \dots \times f_m$ is the unique arrow found by using

$$\langle f_1 \circ pr_{a_1}, f_2 \circ pr_{a_2}, \dots, f_m \circ pr_{a_m} \rangle$$

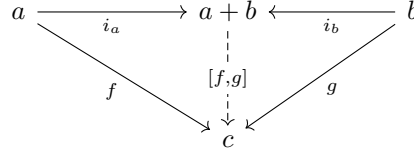
in the general product definition of $b_1 \times b_2 \times \dots \times b_m$.

3.9 Co-products

Definition 3.9.1. A *co-product* of C -objects a and b is given by a C -object denoted $a + b$, and injection functions i_a and i_b .

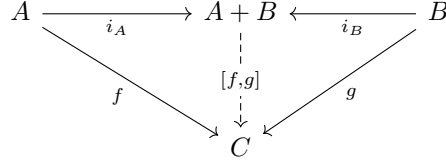
$$a \xrightarrow{i_a} a + b \xleftarrow{i_b} b$$

For any $f : a \rightarrow c$ and $g : b \rightarrow c$, there is a unique arrow $[f, g] : (a + b) \rightarrow c$ so that



commutes.

Exercise 3.9.1. In **Set** we are told that the co-product $A + B$ is the disjoint union, with i_A and i_B being the disjoint identity function (Ie, $i_A(x) = (x, 0)$ for $x \in A$, and $i_B(x) = (x, 1)$ for $x \in B$). Now suppose we have functions $f : A \rightarrow C$ and $g : B \rightarrow C$.

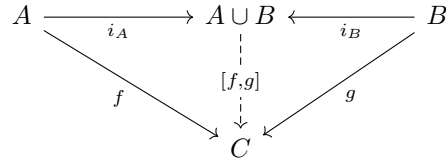


We can find $[f, g] : A + B \rightarrow C$ making this diagram commute by using the rule

$$[f, g](\langle x, y \rangle) = \begin{cases} f(x) & y = 0 \\ g(x) & y = 1. \end{cases}$$

To see that it $[f, g]$ is unique, notice that as given, $[f, g] \circ i_A = f$ iff $[f, g](i_A(x)) = f(x)$ for all $x \in A$. Similarly on the B side - there is no other way to recover the action of f and g out of $A + B$.

Exercise 3.9.2. If $A \cup B = \emptyset$ then we notice that $A \cup B$ satisfies the definition of co-product

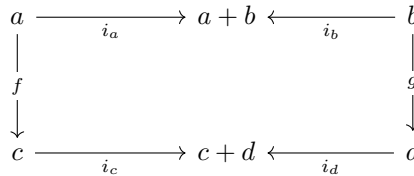


where i_A and i_B are the inclusion functions, and

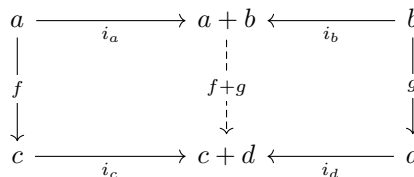
$$[f, g] = \begin{cases} f(x) & x \in A \\ g(x) & x \in B, \end{cases}$$

which is well defined because any x is in either A or B but not both. Then applying the dual of fact 3.8.3, co-products are isomorphic and therefore $A \cup B \cong A + B$.

Definition 3.9.2. Given two co-products $a + b$ and $c + d$, and arrows $f : a \rightarrow c$ and $g : b \rightarrow d$



the co-product arrow $f + g$ is found by using $[i_c \circ f, i_d \circ g]$ in the co-product definition of $c + d$.



3.10 Equalizers

Definition 3.10.1. An arrow i equalizes f and g if they are laid out as follows

$$e \xrightarrow{i} a \rightrightarrows^f_g b$$

where $f \circ i = g \circ i$. Additionally we demand the limiting property: if another e^* and i^* work as above, there is a unique arrow $e^* \rightarrow e$ making

$$\begin{array}{ccc} e & \xrightarrow{i} & a \rightrightarrows^f_g b \\ \uparrow & \nearrow i^* & \\ e^* & & \end{array}$$

commute.

Fact 3.10.2. Every equalizer is monic.

Fact 3.10.3. An epic equalizer is iso.

$$\begin{array}{ccc} e & \xrightarrow{i} & a \rightrightarrows^f_g b \\ \uparrow & \nearrow 1_a & \\ a & & \end{array}$$

Exercise 3.10.1. Working in **Set**, we wish to show that monics are equalizers. Suppose we have some injective function $i : E \rightarrow A$:

$$E \hookrightarrow A \xrightarrow{i}$$

We seek functions which i is an equalizer. Let $f, g : A \rightrightarrows \{0, 1\}$ be given by

$$f(x) = 1$$

$$g(x) = \begin{cases} 1 & x \in i(E) \\ 0 & x \notin i(E). \end{cases}$$

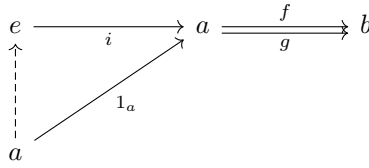
Now clearly $g(i(x)) = f(i(x))$ for all $x \in E$, so $i \circ f = i \circ g$. Supposing that there is another $i^* : C \rightarrow A$ such that $i^* \circ f = i^* \circ g$. We must have ...?

$$\begin{array}{ccc} E & \hookrightarrow & A \rightrightarrows^g_f \{0, 1\} \\ \uparrow & \nearrow i^* & \\ C & & \end{array}$$

Exercise 3.10.2. Working in a poset, suppose that i equalizes f and g . Recall that any 2 parallel arrows are equal. Then in particular $f \circ i = g \circ i$ as follows:

$$e \xrightarrow{i} a \rightrightarrows^f_g b$$

But plugging a and 1_a into the definition for equalizer, since $f \circ 1_a = f = g = g \circ 1_a$ we retrieve the unique arrow $a \rightarrow e$.

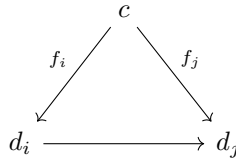


Since we have arrows $a \rightarrow e$ and $e \rightarrow a$ we simply apply the antisymmetric property of posets to determine $e = a$, and the single-arrow property to determine that $i = 1_a$.

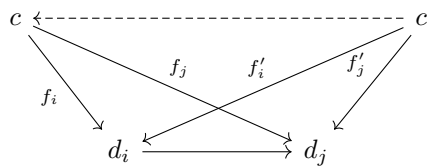
3.11 Limits and co-limits

Definition 3.11.1. A *diagram* is informally a ‘view’ of some objects and arrows within a category. We haven’t looked at functors yet but formally, a diagram in a category C is a functor $F : J \rightarrow C$ where J is an indexing category. (Riehl pg 38)

Definition 3.11.2. Given a diagram D , a D -*cone* consists of a C -object c together with component arrows $f_i : c \rightarrow d_i$ for each $d_i \in d$ that commute with any arrow g in D .



Definition 3.11.3. Given a diagram D , a *limit* for D is a D -cone such that any other D -cone factors through uniquely.



We can apply cones to define some earlier objects in a more succinct way.

Definition 3.11.4. The *product* of a and b is a limit for the arrow-less diagram

$$a \qquad b.$$

Definition 3.11.5. An *equalizer* of f and g is a limit for the diagram

$$a \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} b$$

Definition 3.11.6. A *terminal object* is a limit for the empty diagram.

Exercise 3.11.1. Dualize to reach definitions for *co-cones* and *co-limits*.

3.12 Co-equalizers

Definition 3.12.1. A *co-equalizer* of $f, g : a \rightrightarrows b$ is a co-limit for the diagram

$$a \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} b$$

Let's work backwards from that definition and try to reach something similar to definition 3.10.1 for equalizers! A co-cone over the above diagram consist of an object c , and arrows $f_a : a \rightarrow c$ and $f_b : b \rightarrow c$ such that

$$\begin{array}{ccc} a & \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} & b \\ & \searrow f_a \quad \swarrow f_b & \\ & c & \end{array}$$

commutes. But that means that $f_b \circ f = f_a$ and $f_b \circ g = f_a$. So we can simply drop the f_a and demand an f_b making

$$a \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} b \xrightarrow{f_b} c$$

commute. That gave us our co-cone, but what we want is a co-limit: *the* co-cone that is universal among co-cones. So if we have a c' and f'_b such that $f'_b \circ f = f'_b \circ g$, then c must factor through c' in a unique way:

$$\begin{array}{ccc} a \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} b & \xrightarrow{f_b} & c \\ & \searrow f'_b & \downarrow \text{dashed} \\ & & c' \end{array}$$

3.13 Pullbacks

Definition 3.13.1. Given $f : a \rightarrow c$ and $g : b \rightarrow c$, a *pullback* is a limit for the diagram

$$a \xrightarrow{f} c \xleftarrow{g} b$$

A cone for this diagram takes the form of an object p with arrows $f' : p \rightarrow b$ and $g' : p \rightarrow a$ so that

$$\begin{array}{ccc} p & \xrightarrow{f'} & b \\ \downarrow g' & & \downarrow g \\ a & \xrightarrow{f} & c \end{array}$$

commutes. And the limit means that any cone factors through:

$$\begin{array}{ccccc} & & p' & & \\ & \searrow & & \swarrow & \\ & & p & \xrightarrow{f'} & b \\ & \searrow g'' & \downarrow g' & & \downarrow g \\ & & a & \xrightarrow{f} & c \end{array}$$

Exercise 3.13.1. Suppose

$$\begin{array}{ccc} a & \xrightarrow{g} & b \\ \downarrow & & \downarrow \\ c & \xrightarrow{f} & d \end{array}$$

is a pullback square, and f is monic. Now suppose we have parallel $h_1, h_2 : e \rightrightarrows a$

$$\begin{array}{ccccc} e & \begin{array}{c} \xrightarrow{h_1} \\ \xrightarrow{h_2} \end{array} & a & \xrightarrow{g} & b \\ & & \downarrow f' & & \downarrow g' \\ & & c & \xrightarrow{f} & d \end{array}$$

so that $g \circ h_1 = g \circ h_2$. Then

$$g' \circ g \circ h_1 = g' \circ g \circ h_2.$$

From the original pullback we have $g' \circ g = f \circ f'$, so substituting on both sides gives $f \circ f' \circ h_1 = f \circ f' \circ h_2$. Since f is monic, $f' \circ h_1 = f' \circ h_2$.

But also, substituting on one side gives $g' \circ g \circ h_1 = f \circ f' \circ h_1$. Then

$$\begin{array}{ccc} e & \xrightarrow{g \circ h_1 = g \circ h_2} & b \\ \downarrow f' \circ h_1 = f' \circ h_2 & & \downarrow g' \\ c & \xrightarrow{f} & d \end{array}$$

commutes, meaning it must factor uniquely through our original pullback. Thus the arrow $e \rightarrow a$ is unique, meaning $h_1 = h_2$. Conclude f is monic.

3.14 Pushouts

Definition 3.14.1. Given arrows $f : a \rightarrow b$ and $g : a \rightarrow c$, a *pushout* is a colimit for the diagram

$$b \xleftarrow{f} a \xrightarrow{g} c$$

The co-limit, or universal co-cone for the above diagram consists of an object d and arrows $f' : c \rightarrow d$ and $g' : b \rightarrow d$ so that $g' \circ f = f' \circ g$. Further, if there is a d', f'', g'' behaving the same way, then d will factor uniquely through it.

$$\begin{array}{ccccc} a & \xrightarrow{f} & b & & \\ \downarrow g & & \downarrow g' & \searrow g'' & \\ c & \xrightarrow{f'} & d & & \\ & \searrow f'' & & \nearrow & \\ & & & & d' \end{array}$$

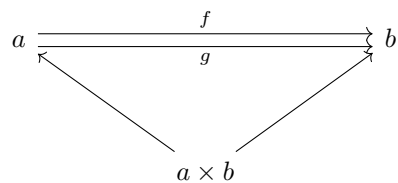
3.15 Completeness

Definition 3.15.1. A category is *complete* if every diagram has a limit.

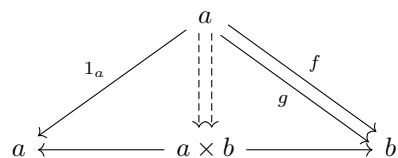
Definition 3.15.2. A category is *finitely complete* if every *finite* diagram has a limit.

Fact 3.15.3. If C has a terminal object and a pullback for each pair of arrows with common codomain, then C is *finitely complete*.

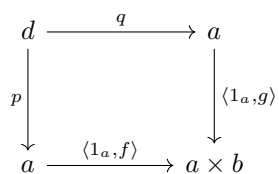
Exercise 3.15.1. Working in a category with pullbacks and products, and given a parallel pair $f, g : a \rightrightarrows b$, first form the product:



Then we can find the two product arrows



$\langle 1_a, f \rangle$ and $\langle 1_a, g \rangle$. Since they share a codomain we can form the pullback:



show $p=q\dots$