A natural introduction to basic category theory

Semitransparent Observer
2023 at Edinburgh

Contents

Chapter 1

A natural introduction to category theory

1.1 Terminology explaination

i.e.: that is

e.g.: for example

etc.: and so on

UoE: University of Edinburgh

$\exists!$: exists a unique

1.2 Introduction

This part gives a brief review of basic set theory([1],[2]), then goes to needed category theory(strict symmetric monoidal category, monoidal functor, and colimit) naturally([3],[4]).

"The art of doing mathematics consists in finding that special case which contains all the germs of generality."

——David Hilbert

I will try to introduce new things inherently by generalizing the concepts familiar to readers.

The whole content of Category theory is far beyond what I will talk about. I strongly recommend readers read some formal books about it, this note is an "introduction of introduction" to category theory.

READ WITH TWO SCREENS!!!

1.3 Basic Set Theory

Let's consider UoE and a family with 3 members Alex, Bob, and Carol. Alex and Bob are former students of UoE, they got married and their son Carol is now a student of UoE.

1.3.1 Object

Def. an *object* is anything you wish to consider as an object.

e.g. former students of UoE, UoE, this sentence is an object, •, you, and the word "you" are objects.

1.3.2 Set, intersection, and union

Def. a *set* S is a collection of **distinct** objects called elements, e.g.:

- U = {UoE}, UoE itself;
- fs = {Alex, Bob}, the couple/former students of UoE; Us = {UoE, Carol}, UoE and a current student; f ={Alex, Bob, Carol}, the family;
- Uf = {UoE, Alex, Bob, Carol}, UoE, its former students, and a current student.

Def. *cardinality* of a set S is the number of elements in S, written as $\mathrm{num}(S)$;

The order we write the elements in a set makes no difference.

i.e. f = {Alex, Bob, Carol} = {Alex, Carol, Bob} etc.;

Def. a *pair* is a set with 2 elements. e.g. fs = {Alex, Bob}, Us = {UoE, Carol}.

Def. set A is a *subset* of set S if $\forall a \in A, a \in S$, written as $A \subseteq S$.

e.g. a set Family = {Alex, Bob, Carol}$\subseteq Uf$.

Def. *empty set* $\varnothing = \{\}$ is the unique set containing no object.

- for any set S, $\varnothing \subseteq S$.

1.3.3 Union and intersection

(Suppose we want to get a set containing all the elements of two sets, we put all the elements of them in one set:)

Def. the *union* of two sets A and B is $\{\forall x | x \in A \text{ or } x \in B\}$, written as $A \bigcup B$.

e.g. take a photo of UoE and Carol, with his family standing by, then we have figures of UoE, Alex, Bob, and Carol:

$\{\mathrm{UoE}, \mathrm{Carol}\} \cup \{\mathrm{Alex}, \mathrm{Bob}, \mathrm{Carol}\} = \{\mathrm{UoE}, \mathrm{Alex}, \mathrm{Bob}, \mathrm{Carol}\}$.

(Suppose we want to get a union of two sets containing all their information, we should first record the same element in different sets differently to make a difference, then we union them together.)

For a set S, S' is a set $\{(s,')|\forall s \in S\}$.

Def. the *disjoint union* of two sets A and B is a set $A' \bigcup B''$, written as $A \bigsqcup B$.

e.g. take a photo of UoE and Carol then another photo of his family, then we have figures of UoE, Carol, Alex, Bob, and another Carol:

$\{\mathrm{UoE}, \mathrm{Carol}\} \bigsqcup \{\mathrm{Alex}, \mathrm{Bob}, \mathrm{Carol}\} = \{\mathrm{UoE}', \mathrm{Carol}', \mathrm{Alex}'', \mathrm{Bob}'', \mathrm{Carol}''\}$.

(Suppose we want to get a set containing all the common elements of two sets, we fetch every element and check if it appears in both of them:)

Def. the *intersection* of two sets A and B is $\{x|x \in A \text{ and } x \in B\}$,

written as $A \bigcap B$.

e.g. take a photo of UoE and Carol then another photo of his family, the figure of Carol is in both of them:

$\{\mathrm{UoE}, \mathrm{Carol}\} \cap \{\mathrm{Alex}, \mathrm{Bob}, \mathrm{Carol}\} = \{\mathrm{Carol}\}$.

1.3.4 Ordered pair and n-tuple

(If Alex, Bob, and Carol are singing "Ode to Joy", the tone is like 3345 5432. The notes are sung in order and the same note can appear repeatedly. We may construct a set called "n-tuple" to describe it, which is defined recursively from ordered pair(2-tuple).)

For two objects a_1 and a_2,

Def. a *ordered pair* (a_1, a_2) is a set $\{\{a_1\}, \{a_1, a_2\}\}$.

- $(a_2, a_1) = \{\{a_2\}, \{a_2, a_1\}\} \neq \{\{a_1\}, \{a_1, a_2\}\} = (a_1, a_2)$, hence the order do matters;
- $(\bullet, \bullet) = \{\{\bullet\}, \{\bullet\}\} = \{\{\bullet\}\}$;

For a sequence of n objects $a_1, a_2, \cdots, a_n$,

Def. the *n-tuples* $A = (a_1, a_2, \cdots, a_n)$ is an ordered pair $((a_1, a_2, \cdots, a_{n-1}), a_n)$[2].

- for $a_1, a_2, \cdots, a_n$, the n-tuples is:
$$(a_1, a_2, \cdots, a_n) = \{(a_1, a_2, \cdots, a_{n-1}), \{(a_1, a_2, \cdots, a_{n-1}), a_n\}\} = \cdots;$$
- the index of element of A is $I = (1, 2, \cdots, n)$,
$\forall i \in I$, $A_i = a_i$ is the ith element of A;

e.g. the tone of "Ode to Joy" is $(3, 3, 4, 5, 5, 4, 3, 2)$;

e.g. $(\bullet, \bullet, \bullet) = ((\bullet, \bullet), \bullet) = (\{\{\bullet\}\}, \bullet) = \{\{\{\{\bullet\}\}\}, \{\{\{\bullet\}\}, \bullet\}\}$.

(There can be some "relation" between elements of sets:

e.g. "$\in$", Alice $\in f$, Bob $\in f$, Bob $\in fs$;

e.g. "$\subseteq$", $fs \subseteq f$, $s \subseteq f$, $f \subseteq f$;

e.g. "$=$" on $\mathbb{R}$, $1 = 1$, $2.0 = 2$, $0.999\cdots = 1$ etc;

e.g. "$\neq$" on $\mathbb{R}$, whenever "$a = b$" doesn't hold;

e.g. "$\leq_{\mathbb{N}}$" on $\mathbb{N}$: $0 \leq_{\mathbb{N}} 2$, "$\leq_{\mathbb{R}^+}$" on $\mathbb{R}^+$: $1 \leq_{\mathbb{R}^+} e^2$.

e.g. in the set Family, there are many *relations* between the family members: Alice *is the mother of* Carol, Bob *is the husband of* Alice, Alice *married* Bob, Bob *married* Alice, Carol *is not married to* Bob, Carol *is* Carol, they *love* each other, and there's no *hate* between them.)

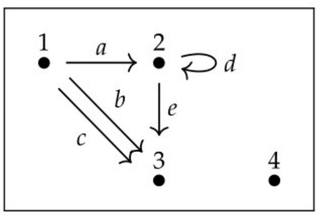
Before introducing the rigid definition of relation, we may introduce

- "graph", a way to represent relations intuitively;
- "cartesian product", the "universe" relations live in.

1.3.5 Graph and path

Def. a *directed graph* $G = (V, A, s, t)$ consists of:

- a set V of vertices;
- a set A of arrows.
  - an arrow $\mathbf{a} \in A$ is a 3-tuple (a, s, t), $s, t \in V$. s/t is the source/target of a. $\mathbf{a} = (a, s, t)$ may drawn as
    $s \vdash a \rightarrow t$, $s \xmapsto{a} t$, or $s \xrightarrow{a} t$.

$G =$ 

arrow a	**source** $s(a) \in V$	**target** $t(a) \in V$
a	1	2
b	1	3
c	1	3
d	2	2
e	2	3

Figure 1.1: graph G

Def. a *path* p in a graph G is a n-tuple of arrows such that the target of one arrow is the source of the arrow on its left.

- n is called the length of the path.

e.g. $1 \xrightarrow{a} 2 \xrightarrow{d} 2 \xrightarrow{e} 3$ is a length 3 path.

1.3.6 Cartesian product

We may construct a new set from two old ones.

For a 2-tuple of sets (A, B),
Def. their *cartesian product* $A \times B$ is a set consists of 2-tuples:

$$A \times B = \{\forall (a, b) | a \in A \text{ and } b \in B\}.$$

e.g. 2-dimensional Cartesian coordinate system $(x, y) \in \mathbb{R} \times \mathbb{R}$.

1.3.7 Relation, domain, range, and codomain

$\mathrm{dom}(R) \times \mathrm{cod}(R)$ is the "universe" a relation R lives in:

Def. a *relation* R is a set of 2-tuples $R \subseteq \mathrm{dom}(R) \times \mathrm{cod}(R)$, $(a, b) \in R$ is written as aRb.
e.g.$\leq_\mathbb{N} \subseteq \mathbb{N} \times \mathbb{N}$, $(1, 2) \in \leq_\mathbb{N}$, written as $1 \leq_\mathbb{N} 2$;
e.g. for any set A, $=_A$ is a relation $\{\forall (a, a) | a \in A\}$, written as $a =_A a$.

Def. the *domain* of R is a set $\mathrm{dom}(R) = \{\forall a | \exists b, (a, b) \in R\}$.

Def. the *range* of R is a set $\mathrm{ran}(R) = \{\forall b | \exists a, (a, b) \in R\}$.

Def. a *codomain* of R is a set $\mathrm{cod}(R) \supseteq \mathrm{ran}(R)$.
Following notations are used later.

- an element $r = (a, b) \in R$ is drawn as $a \vdash_R \rightarrow b$, $a \xmapsto{(a,b)} b$, or $a \xmapsto{r} b$;
- aRb, bRc is written as $aRbRc$. Let $s = (b, c)$, drawn as:
  $a \xmapsto{r} b \xmapsto{s} c$.

e.g.:

- $\leq_\mathbb{N}$ is a relation, $f = (0, 2) \in \leq_\mathbb{N}$, $0 \leq_\mathbb{N} 2$,
  drawn as $0 \vdash_{\leq_\mathbb{N}} \rightarrow 2$, $0 \xmapsto{(0,2)} 2$, or $0 \xmapsto{f} 2$.
- $2 \leq_\mathbb{N} 4$, $4 \leq_\mathbb{N} 5$ is written as $2 \leq_\mathbb{N} 4 \leq_\mathbb{N} 5$. Let $g = (4, 5)$, drawn as: $2 \xmapsto{f} 4 \xmapsto{g} 5$;
- $\mathrm{dom}(\leq_\mathbb{N}) = \mathrm{ran}(\leq_\mathbb{N}) = \mathbb{N}$, $\mathrm{cod}(\leq_\mathbb{N})$ can be any set containing $\mathbb{N}$, such as Z, $\mathbb{R}^+$, and $\mathbb{C}$;
- $\leq_\mathbb{N} \subset \leq_{\mathbb{R}^+}$.

Every relation R lives in a set constructed from its domain $\mathrm{dom}(R)$ and a codomain $\mathrm{cod}(R)$.

For two relations R_1 and R_2,

Def. the *product* $R_1 \times R_2$ is a relation

$(R_1, R_2) \in (\mathrm{dom}(R_1) \times \mathrm{dom}(R_2)) \times (\mathrm{cod}(R_1) \times \mathrm{cod}(R_2))$, such that:

$$(x, y)(R_1, R_2)(x', y') \text{ if and only if } xR_1x' \text{ and } yR_2y'.$$

For a relation $R \subseteq A \times B$,

Def. the *inverse relation* $R^{-1} \in B \times A$ is a relation $\{\forall (b, a) | (a, b) \in B\}$.

e.g. inverse of $\leq_{\mathbb{N}}$ is $\geq_{\mathbb{N}}$. $\because 1 \leq_{\mathbb{N}} 2, \therefore 2 \geq_{\mathbb{N}} 1$.

1.3.8 Function and composition

There's a special type of relation, namely "function". A function f assigns every element d in its domain $\mathrm{dom}(f)$ only one element c in its codomain $\mathrm{cod}(f)$.

e.g. in the alphabet, lower-case "a" has its upper-case "A", written as (a, A);

e.g. *all the cities* is a set, and the cities' location is described by latitude and longitude. Location of Edinburgh is $(55°57'\mathrm{N}, 3°11'\mathrm{W})$, written as

$(\mathrm{Edinburgh}, (55°57'\mathrm{N}, 3°11'\mathrm{W}))$.

For two sets S and T,

Def. a *function* $f : S \to T$ is a subset $f \subseteq S \times T$, such that

$\forall x \in S$, $\exists! y \in T$, such that $(x, y) \in f$:

- $(x, y) \in f$ is written as $f(x) = y$, y is called the *value* of x;
- function f is written as:

$$\begin{aligned} f : S &\longrightarrow T \\ x &\longmapsto f(x) \end{aligned}$$

  drawn as an arrow (f, S, T): $S \xrightarrow{f} T$

  and $(f, x, f(x))$: $x \xmapsto{f} f(x)$ for an element $x \in S$.

The range of f is written as $f(S)$.

e.g. $\mathrm{dom}(f) = \mathbb{R}$, $\mathrm{cod}(f) = \mathbb{R}$, and $\mathrm{ran}(f) = [-1, 1]$.

$$\begin{aligned} f : \mathbb{R} &\longrightarrow \mathbb{R} \\ x &\longmapsto \sin(x). \end{aligned}$$

For functions $f : A \to B'$ and $g : B \to C$, if $\mathrm{ran}(f) \subseteq \mathrm{dom}(g) = B$(i.e., $\mathrm{dom}(g)$ is a codomain of f) we may construct a new function $gf : A \to C$ in a natural way:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[rr, bend left, "gf"] & B \arrow[r, "g"] & C \end{tikzcd} \qquad \begin{tikzcd} a \arrow[r, mapsto, "f"] \arrow[rr, mapsto, bend left, "gf"] & f(a) \arrow[r, mapsto, "g"] & g(f(a)) \end{tikzcd} .$$

Def. *compsite* is a function $\circ$:

$$\begin{aligned} \circ : \{\forall (f,g) | \mathrm{ran}(f) \subseteq \mathrm{dom}(g)\} &\longrightarrow \{\forall h : \mathrm{dom}(f) \to \mathrm{cod}(g)\} \\ (f,g) &\longmapsto gf (\text{or } g \circ f) \end{aligned}$$

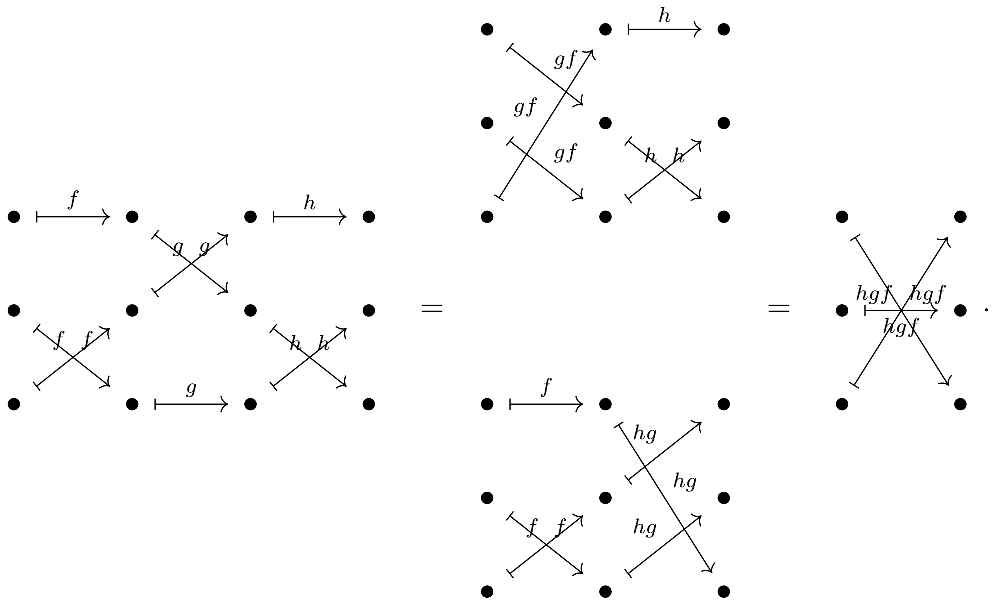
- the *composition* of (f,g) is gf;

Def. a n-tuple of functions $fun = (f,g,h,\cdots)$ is *composable* if $\forall i, i+1 \in I, \mathrm{ran}(fun_i) \subseteq \mathrm{dom}(fun_{i+1})$: e.g for functions $f : A \to B, g : B \to C$, and $h : C \to D$, 3-tuple (f,g,h) is composable:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[rr, bend left, "gf"] \arrow[rrr, bend left=50, "h(gf)"] \arrow[rrr, bend right=50, "(hg)f"'] & B \arrow[r, "g"] \arrow[rr, bend right, "hg"'] & C \arrow[r, "h"] & D \end{tikzcd}$$

the function $h(gf) = (hg)f = hgf : A \to D$ is the composition of (f,g,h).

for functions $f, g, h : (\bullet,\bullet,\bullet) \to (\bullet,\bullet,\bullet)$:



Def. a graph of sets and functions is *commute* if compositions of functions along different paths are the same.

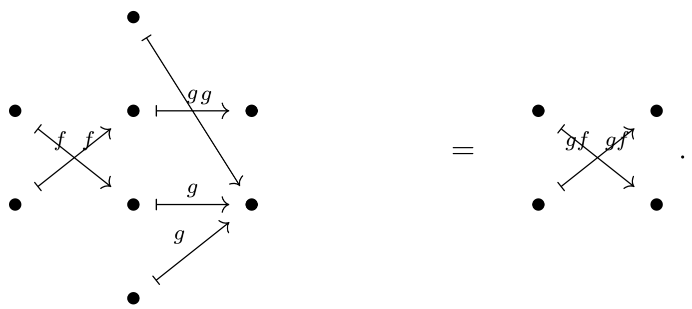
We may classify functions by their properties:

For two sets S and T,
Def. a function $f_i : S \to T$ is *injective* if $\forall y \in \mathrm{ran}(f_i) \subseteq T$, $\exists! x \in S$, such that $(x, y) \in f_i$.

Def. a function $f_s : S \to T$ is *surjective* if $\forall y \in T, \exists x \in S$, such that $(x, y) \in f_s$ (i.e. the range $f_s(S)$ and the codomain $\mathrm{cod}(f_s)$ of f_s are the same).

Def. a function f_b is *bijective* if it's injective and surjective (e.g. f, g, h). i.e. $\forall y \in T, \exists! x \in S$ such that $(x, y) \in f_b$.

e.g. injection $f : (\bullet, \bullet) \to (\bullet, \bullet, \bullet, \bullet)$, surjection $g : (\bullet, \bullet, \bullet, \bullet) \to (\bullet, \bullet)$, and bijection $gf : (\bullet, \bullet) \to (\bullet, \bullet)$:



injection: surjection: bijection:

e.g. $$f : [0, \frac{\pi}{2}] \longrightarrow \mathbb{R} \qquad x \longmapsto \sin(x);$$
$$f : [0, \pi] \longrightarrow [0, 1] \qquad x \longmapsto \sin(x);$$
$$f : [0, \frac{\pi}{2}] \longrightarrow [0, 1] \qquad x \longmapsto \sin(x);$$

e.g. injections eb_A, eb_B(embedding), surjections pr_A, pr_B(projection):

$$eb_A : A \longrightarrow A \bigsqcup B \qquad a \longmapsto a',$$
$$eb_B : B \longrightarrow A \bigsqcup B \qquad b \longmapsto b'';$$

$$\begin{tikzcd}
A \arrow[r, "eb_A"] & A \bigsqcup B \\
A \times B \arrow[u, "pr_A"] \arrow[r, mapsto, "pr_B"] & B \arrow[u, "eb_B"']
\end{tikzcd}
\qquad
\begin{tikzcd}
a \arrow[r, "eb_A"] & a'/b'' \\
(a, b) \arrow[u, "pr_A"] \arrow[r, mapsto, "pr_B"] & b \arrow[u, "eb_B"']
\end{tikzcd}$$

$$pr_1 : A \times B \longrightarrow A \qquad (a, b) \longmapsto a,$$
$$pr_2 : A \times B \longrightarrow B \qquad (a, b) \longmapsto b.$$

For a set S,

Def. a *identity function* is a bijection:

$$\begin{aligned} \mathrm{id}_S : S &\longrightarrow S \\ x &\longmapsto x. \end{aligned}$$

For a bijection $f \subseteq S \times T$,

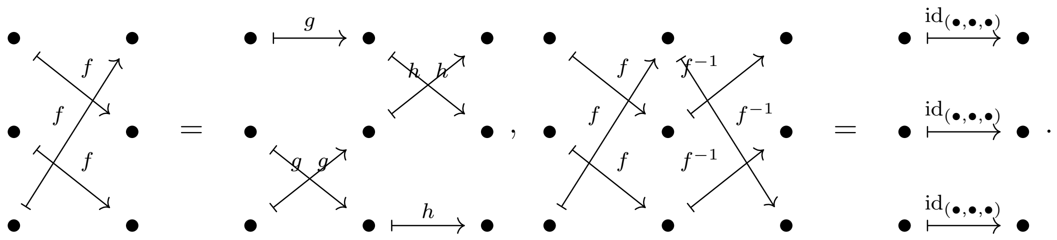
Def. the *inverse function* $f^{-1} \in T \times S$ is the inverse relation of f, i.e. $\{\forall (y, x) | (x, y) \in f\}$. f^{-1} is a bijection:

- $\because f$ is a bijection, $\therefore \forall y \in T, \exists! x \in S$ such that $(x, y) \in f$, $\therefore \forall y \in T, \exists! x \in S$ such that $(y, x) \in f^{-1}$, $\therefore f^{-1} : T \to S$ is a function;
- $\because f$ is a function, $\therefore \forall x \in S, \exists! y \in T$ such that $(x, y) \in f$, $\therefore \forall x \in S, \exists! y \in T$ such that $(y, x) \in f^{-1}$, $\therefore f^{-1}$ is also a bijection.

e.g. for $f : S \to T$ and $f^{-1} : T \to S$,

- $\forall x \in S$, $(f^{-1} f)(x) = f^{-1}(f(x)) = x$, $f^{-1} f = \mathrm{id}_S$;
- $\forall y \in T$, $(f f^{-1})(y) = f(f^{-1}(y)) = y$, $f f^{-1} = \mathrm{id}_T$.

e.g. bijections $f, g, h : (\bullet, \bullet, \bullet) \to (\bullet, \bullet, \bullet)$, and $\mathrm{id}_{(\bullet,\bullet,\bullet)}$, $f = gh$, $ff^{-1} = \mathrm{id}_{(\bullet,\bullet,\bullet)}$:



Notation: sometimes $f\left((x_1, x_2)\right)$ is written as $x_1 f x_2$.

e.g. $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, $+(1, 3) = 4$ is written as $1 + 3 = 4$.

Similar to product of relations, for two functions

$$\begin{aligned} f : A &\longrightarrow B \\ x &\longmapsto f(x) \end{aligned} \qquad\qquad \begin{aligned} g : C &\longrightarrow D \\ y &\longmapsto g(y), \end{aligned}$$

we may construct a product function (f, g), such that:

$$\begin{aligned} (f,g) : A \times C &\longrightarrow B \times D \\ (x,y) &\longmapsto (f(x), g(y)) \end{aligned}$$

e.g. for any function f, we may define a functions dom/cod to get f's domain/codomain:

$$\begin{aligned} \text{dom/cod} : \text{function} &\longrightarrow \text{domain/codomain} \\ f &\longmapsto \text{dom}(f)/\text{cod}(f); \end{aligned}$$

1.4 Category theory in need

The main goal of this part is to introduce the concept of symmetric monoidal category and monoidal functor in a natural way. The idea is compromised in the following graph:

$$\begin{tikzcd}
(P, \leq) \arrow[d, "+\otimes"'] \arrow[rr, "\leq \Rightarrow f : a \to b"] & & (M, \otimes, I) \arrow[d, "\text{is a}"] \arrow[ll, "+\leq"] \\
(P, \leq, \otimes, I) \arrow[dr, "\leq \Rightarrow f : a \to b"'] & & \mathbf{C} \arrow[dl, "+\otimes"] \\
& (\mathbf{C}, \otimes, I) &
\end{tikzcd} .$$

We begin with some trivial things.

1.4.1 "Less than equal to" and "addition" on natural numbers

Consider two relations $\leq_{\mathbb{N}}$ and addition $+$ on $\mathbb{N}$:

"+" is a function

$$\begin{aligned} + : \mathbb{N} \times \mathbb{N} &\longrightarrow \mathbb{N} \\ (a,b) &\longmapsto a + b, \end{aligned}$$

such that:

- *associativity*: $\forall a, b, c \in \mathbb{N}$, $(a + b) + c = a + (b + c)$;
  - e.g. $(2 + 4) + 5 = 2 + (4 + 5) = 11$.
- *commutativity*: $\forall p, q \in \mathbb{N}$, $p + q = q + p$;
- *unit object*: an element $0 \in \mathbb{N}$, such that:

- *left unity*: $\forall x \in \mathbb{N}$, $0 + x = x$;
- *right unity*: $\forall x \in \mathbb{N}$, $x + 0 = x$.
  * e.g. $0 + 6 = 6 + 0 = 6, 3 + 5 = 5 + 3 = 8$.

These are the data of a monoid $(\mathbb{N}, +, 0)$.

$\leq_{\mathbb{N}} \subseteq \mathbb{N} \times \mathbb{N}$ is a relation such that:

- *transitivity*: $\forall a, b, c \in \mathbb{N}$, if $a \leq_{\mathbb{N}} b \leq_{\mathbb{N}} c$, then $a \leq_{\mathbb{N}} c$:
  - e.g. $2 \leq_{\mathbb{N}} 4 \leq_{\mathbb{N}} 5$, then $2 \leq_{\mathbb{N}} 5$;
- *reflexivity*: $\forall x \in \mathbb{N}$, $x \leq_{\mathbb{N}} x$:
  - e.g. $3 \leq_{\mathbb{N}} 3$.

These are the data of a preorder $(\mathbb{N}, \leq_{\mathbb{N}})$.

"+" preserves the preorder in the following way:

- *monotonicity*: $\forall a_1, b_1, a_2, b_2 \in M$, if $a_1 \leq_{\mathbb{N}} b_1$ and $a_2 \leq_{\mathbb{N}} b_2$, then $a_1 \cdot b_1 \leq_{\mathbb{N}} a_2 \cdot b_2$. e.g.:

$$\begin{tikzcd}
(1,5) \arrow[r, mapsto, "{(\leq_{\mathbb{N}}, \leq_{\mathbb{N}})}"] \arrow[d, mapsto, "+"] & (3,7) \arrow[d, mapsto, "+"] & \in & (\mathbb{N}, \mathbb{N}) \arrow[d] \\
6 \arrow[r, mapsto, "\leq_{\mathbb{N}}"] & 10 & \in & \mathbb{N}.
\end{tikzcd}$$

The combination of all the data above is the data of the symmetric monoidal preorder $(\mathbb{N}, \leq_{\mathbb{N}}, +, 0)$:

1.4.2 "Less than equal to" and "multiplication" on positive real numbers

Let's consider relation $\leq_{\mathbb{R}^+}$ and multiplication $\cdot$ in $\mathbb{R}$:

$\cdot$ is a function

$$\begin{aligned} \cdot : \mathbb{R}^+ \times \mathbb{R}^+ &\longrightarrow \mathbb{R}^+ \\ (a, b) &\longmapsto a \cdot b, \end{aligned}$$

such that:

- *associativity*: $\forall a, b, c \in \mathbb{R}^+$, $(a \cdot b) \cdot c = a \cdot (b \cdot c)$;
  - $(e^2 \cdot e^4) \cdot e^5 = e^2 \cdot (e^4 \cdot e^5) = e^1 1$.
- *commutativity*: $\forall p, q \in \mathbb{R}^+$, $p \cdot q = q \cdot p$;
  - e.g. $1 \cdot e^6 = e^6 \cdot 1 = e^6, e^3 \cdot e^5 = e^5 \cdot e^3 = e^8$.
- *unit object*: an element $1 \in \mathbb{R}^+$, such that:
  - *left unity*: $\forall x \in \mathbb{R}^+$, $1 \cdot x = x$;
  - *right unity*: $\forall x \in \mathbb{R}^+$, $x \cdot 1 = x$.

These are the data of a monoid $(\mathbb{R}^+, \cdot, 1)$.

$\leq_{\mathbb{R}^+} \subseteq \mathbb{R}^+ \times \mathbb{R}^+$ is a relation such that:

- *transitivity*: $\forall a, b, c \in \mathbb{R}^+$, if $a \leq_{\mathbb{R}^+} b \leq_{\mathbb{R}^+} c$, then $a \leq_{\mathbb{R}^+} c$:
  - e.g. $e^2 \leq_{\mathbb{N}} e^4 \leq_{\mathbb{N}} e^5$, then $e^2 \leq_{\mathbb{N}} e^5$.
- *reflexivity*: $\forall x \in \mathbb{R}^+$, $x \leq_{\mathbb{R}^+} x$:
  - e.g. $e^3 \leq_{\mathbb{R}^+} e^3$.

These are the data of a preorder $(\mathbb{R}^+, \leq_{\mathbb{R}^+})$.

"$\cdot$" preserves the preorder in the following way:

- *monotonicity*: $\forall a_1, b_1, a_2, b_2 \in \mathbb{R}^+$, if $a_1 \leq_{\mathbb{R}} b_1$ and $a_2 \leq_{\mathbb{R}} b_2$, then $a_1 \cdot b_1 \leq_{\mathbb{R}} a_2 \cdot b_2$. e.g.:

$$\begin{tikzcd}
(e, e^5) \arrow[r, mapsto, "(\leq_{\mathbb{R}}, \leq_{\mathbb{R}})"] \arrow[d, mapsto, "\cdot"] & (e^3, e^7) \arrow[d, mapsto, "\cdot"] & \in & (\mathbb{R}^+, \mathbb{R}^+) \arrow[d] \\
e^6 \arrow[r, mapsto, "\leq_{\mathbb{R}}"] & e^{10} & \in & \mathbb{R}^+.
\end{tikzcd}$$

The combination of all the data above is the data of the symmetric monoidal preorder $(\mathbb{R}^+, \leq_{\mathbb{R}}, \cdot, 0)$.

1.4.3 Monoid and generator

Def. a *commutative monoid* $\mathbf{M} = (M, \otimes, I)$ consists of:

- an underlying set M;
- a function $\otimes : M \times M \to M$(in convention, $\forall a, b \in M, \otimes(a, b)$ is written as $a \otimes b$),such that:

- *associativity*: $\forall a, b, c \in M$, $(a \otimes b) \otimes c = a \otimes (b \otimes c)$;
- *commutativity*: $\forall p, q \in M$, $p \otimes q = q \otimes p$.

- a unique element $I \in M$, such that:
  - *left unity*: $\forall x \in M$, $I \otimes x = x$;
  - *right unity*: $\forall x \in M$, $x \otimes I = x \otimes I = x$.

e.g. $(\mathbb{N}, +, 0), (\mathbb{R}^+, \times, 1)$.

Similar to subsets of a set, a monoid have submonoids.

For a monoid $\mathbf{M} = (M, \otimes)$

Def. a *submonoid* of $\mathbf{M}$ is a monoid $\mathbf{M'} = (M', \otimes)$ such that $M' \subseteq M$, written as $\mathbf{M'} \subseteq \mathbf{M}$.

e.g. $(\{3n | \forall n \in \mathbb{N}\}, +, 0) \subseteq (\mathbb{N}, +, 0) \subseteq (\mathrm{Z}, +, 0) \subseteq (\mathbb{R}, +, 0)$.

Def. a *generating set* of $\mathbf{M}$ is a subset $S \subseteq M$ such that $\forall x \in M$ is a product of some elements from S.

e.g. for $(\{3n | \forall n \in \mathbb{N}\}, +, 0)$, $\{0, 3\}$; for $(\mathbb{N}, +, 0)$, $\{0, 1\}$; for $(\mathrm{Z}, +, 0)$, $\{1, -1\}$.

1.4.4 Preorder and category

Def. a *preorder* $\mathbf{P} = (P, \leq)$ is a 2-tuple consists of:

- a *underlying set* P;
- a*preorder relation* $\leq \subseteq P \times P$, $(a, b) \in \leq$ is written as $a \leq b$ such that:
  - *transitivity*: for $\forall a, b, c \in P$, if $a \leq b \leq c$, then $a \leq c$:

$$\begin{tikzcd} a \arrow[r, mapsto, "\leq" description] \arrow[rr, mapsto, bend left, "\leq"] & b \arrow[r, mapsto, "\leq" description] & c, \end{tikzcd} \quad \begin{tikzcd} a \arrow[r, mapsto, "r"] \arrow[rr, mapsto, bend left, "sr"] & b \arrow[r, mapsto, "s"] & c \end{tikzcd};$$

here $r = (a, b)$, $s = (b, c)$, and $sr = (a, c)$.

* when $a \leq b$ and $b \leq a$, a and b can be distinct objects.
  e.g. for "married" on (family, family), Alice married Bob, Bob married Alice, but Alice is not Bob;
* if $a \leq b \leq c \leq d$, we may get $a \leq d$ in 2 ways:

$$\begin{tikzcd} a \arrow[r, mapsto, "\leq" description] \arrow[rr, mapsto, bend left, "\leq"] \arrow[rrr, mapsto, bend left=50, "\leq"] \arrow[rrr, mapsto, bend right=50, "\leq"'] & b \arrow[r, mapsto, "\leq" description] \arrow[rr, mapsto, bend right, "\leq"'] & c \arrow[r, mapsto, "\leq" description] & d, \end{tikzcd} \quad \begin{tikzcd} a \arrow[r, mapsto, "r"] \arrow[rr, mapsto, bend left, "sr"] \arrow[rrr, mapsto, bend left=50, "t(sr)"] \arrow[rrr, mapsto, bend right=50, "(ts)r"'] & b \arrow[r, mapsto, "s"] \arrow[rr, mapsto, bend right, "ts"'] & c \arrow[r, "t"] & d. \end{tikzcd}$$

upper: $\because a \le b \le c; \therefore a \le c$, $\because a \le c \le d; \therefore a \le d$;
lower: $\because b \le c \le d, \therefore b \le d$, $\because a \le b \le d, \therefore a \le d$;
The right graph:let $r = (a,b), s = (b,c), t = (c,d)$,
then $sr = (b,d), ts = (a,c)$, finally $t(sr) = (ts)r = (a,d) \in \le$.

- *reflexivity*: $\forall x \in P$, $x \le x$(i.e. $=_P \subseteq \le 0$. This leads to the following 2 properties: $\forall a, b \in P$, if $a \le b$(i.e. $r = (a,b) \in \le$), then:
  * *left unity*: $a \le a \le b$, $a \le b$: $\begin{tikzcd} a \arrow[r, mapsto, "\le"] \arrow[rr, bend left, "\le"] & a \arrow[r, mapsto, "\le"] & b \end{tikzcd}$,

    let $1_a = (a,a)$, then $1_a r = r$: $\begin{tikzcd} a \arrow[r, mapsto, "1_a"] \arrow[rr, bend left, "r"] & a \arrow[r, mapsto, "r"] & b \end{tikzcd}$;
  * *right unity*: $a \le b \le b$, $a \le b$: $\begin{tikzcd} a \arrow[r, mapsto, "\le"] \arrow[rr, bend left, "\le"] & b \arrow[r, mapsto, "\le"] & b \end{tikzcd}$,

    let $1_b = (b,b)$, then $r1_b = r$: $\begin{tikzcd} a \arrow[r, mapsto, "r"] \arrow[rr, bend left, "r"] & b \arrow[r, mapsto, "1_b"] & b \end{tikzcd}$.

If $a \le b$, we say a is smaller than b, b is larger than a.
e.g. for any set A, $(A, =_A)$; $(\mathbb{N} \times \mathbb{N}, \le_{\mathbb{N}\times\mathbb{N}})$; (all the sets, $\subseteq$).

Similar to subset, for a preorder $\mathbf{P} = (P, \le_P)$,
Def. preorder $\mathbf{A} = (A, \le_A)$ is a *subpreorder* of $\mathbf{P}$ if $A \subseteq P$ and $\le_A \subseteq \le_P$.
e.g. $(\mathbb{N}, \le_{\mathbb{N}}) \subseteq (\mathbb{R}^+, \le_{\mathbb{R}^+})$.

For $\le$ in $(P, \le)$, the inverse relation is $\ge$, $(P, \ge)$ is also a preorder.
Def. a *category* $\mathbf{C} = (\mathrm{Obj}(\mathbf{C}), \mathrm{Mor}(\mathbf{C}))$ is a 2-tuple consists of:

- *objects*: a collection of objects $\mathrm{Obj}(\mathbf{C})$;
- *objects*: a collection of objects $\mathrm{Obj}(\mathbf{C})$;
  - similar to underlying set P.
- *morphisms*: a collection of morphisms $\mathrm{Mor}(\mathbf{C})$(. A morphism f is drawn as.

$$f : \mathrm{Obj}(\mathbf{C}) \xrightarrow{f} \mathrm{Obj}(\mathbf{C})$$
$$\mathrm{dom}(f) \xrightarrow{f} \mathrm{cod}(f)$$
$$a \xrightarrow{f} f(a)$$

  $\mathrm{dom}(f)/\mathrm{cod}(f)$ is called the domain/codomain of a morphism f.
  - for a morphism $f : a \to b$, $a, b \in \mathrm{Obj}(\mathbf{C})$:
    * f is called a morphism from a to b;

    * similar to $a \leq b$.

  - a category C is *small* if $\mathrm{Obj}(\mathbf{C})$ and $\mathrm{Mor}(\mathbf{C})$ are sets.

Similar to composition of functions,

Def. a n-tuple of morphisms $mor = (f, g, h, \cdots)$ is *composable* if $\forall i, i+1 \in I, \mathrm{cod}(mor_i) \subseteq \mathrm{dom}(mor_{i+1}))$, such that:

  - *composite rule*: $\forall f, g \in \mathrm{Mor}(\mathbf{C})$, if $\mathrm{cod}(f) = \mathrm{dom}(g)$, then there's a specify morphism gf:

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[rr, bend left, "gf"] & b \arrow[r, "g"] & c \end{tikzcd};$$

    * similar to transitivity and composition of functions.

  - *associativity*: $\forall f, g, h \in \mathrm{Mor}(\mathbf{C})$, if (f, g, h) is composable, then $h(gf) = (hg)f = hgf$:

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[rr, bend left, "gf"] \arrow[rrr, bend left=50, "h(gf)"] \arrow[rrr, bend right=50, "(hg)f"'] & b \arrow[r, "g"] \arrow[rr, bend right, "hg"'] & c \arrow[r, "h"] & d \end{tikzcd}$$

    * similar to $a \leq b \leq c \leq d$.

- *identity*: $\forall x \in \mathrm{Obj}(\mathbf{C})$, there's an unique morphism $1_x : x \to x$ called identity morphism,

  - similar to reflexivity, $=_A$ and id_S,
    $1_{\mathbf{C}}$ is the collection of all identity morphisms of $\mathbf{C}$.

such that $\forall f : a \to b \in \mathrm{Mor}(\mathbf{C})$,

  - *left unity*: $1_a f = f$: $\begin{tikzcd} a \arrow[r, "1_a"] \arrow[rr, bend left, "1_a f = f"] & a \arrow[r, "f"] & b \end{tikzcd}$, similar to $1_a r = r$;

  - *right unity*: $f 1_b = f$: $\begin{tikzcd} a \arrow[r, "f"] \arrow[rr, bend left, "f 1_b = f"] & b \arrow[r, "1_b"] & b \end{tikzcd}$, similar to $r 1_b = r$.

e.g. a preorder $(P, \leq)$ is a special category, there is at most 1 morphism between 2 objects. Conversely, for a category $\mathbf{C}$ and $\forall a, b \in \mathrm{Obj}(\mathbf{C})$, if we substitute all the morphisms from a to b by (a, b), then category $\mathbf{C}$ becomes a preorder $\mathbf{C}$.

e.g $\mathbf{Set} = (\mathrm{sets}, \mathrm{functions})$ is a category, consists of:

- *objects*: $\mathrm{Obj}(\mathbf{Set})$ consists of all the sets;
- *morphisms*: $\mathrm{Mor}(\mathbf{Set})$ consists of all the functions between sets;
- *identity morphisms*: identity functions.

e.g a monoid $\mathbf{M} = (M, \otimes, I)$ is a category, consists of:

- *object*: an object m;
- *morphisms*: $\forall x \in M$ is a morphism $x : m \to m$. composition of $(a, b) \in M \times M$ is $a \otimes b$;
- *identity morphisms*: I.

Similar to subpmoid and subpreorder, for a category $\mathbf{C}$,

Def. category $\mathbf{A}$ is a *subcategory* of $\mathbf{C}$ if $\mathrm{Obj}(A) \subseteq \mathrm{Obj}(C)$ and $\mathrm{Mor}(A) \subseteq \mathrm{Mor}(C)$.

Similar to generating set of monoid, for a category $\mathbf{C}$,

Def. a *generating set* of $\mathbf{C}$ is a subset $S \subseteq \mathrm{Mor}(\mathbf{C})$ such that $\forall \in \mathrm{Mor}(\mathbf{C})$ is a composition of some morphisms from S.

Similar to commute graph of functions,

Def. a graph of a category is *commute* if compositions of morphisms along different paths are the same.

Similar to the inverse of preorder relation, if we inverse all the morphisms in a category $\mathbf{C}$, we get its opposite category $\mathbf{C}^{op}$.

Def. the *opposite* category $\mathbf{C}^{op}$ of a category $\mathbf{C}$ is a category consists of:

- *objects*: $\mathrm{Obj}(\mathbf{C^{op}}) = \mathrm{Obj}(\mathbf{C})$;
- *morphisms*: $\forall f : a \to b \in \mathrm{Mor}(\mathbf{C})$, a morphism $f^{op} : b \to a \in \mathrm{Mor}(\mathbf{C}^{op})$.

Similar to a bijection between sets, isomorphism is a "bijective" morphism .

For a category $\mathbf{C}$, $a, b \in \mathrm{Obj}(\mathbf{C})$,

Def. a morphism $f : a \to b$ is an *isomorphism* if $\exists f^{-1} : b \to a$ such that $f^{-1}f = 1_a$ and $ff^{-1} = 1_b$.

$$\begin{tikzcd}
a \arrow[loop left, "ff^{-1}=1_a"] \arrow[r, bend left, "f"] & b \arrow[l, bend left, "f^{-1}"] \arrow[loop right, "f^{-1}f=1_b"]
\end{tikzcd}$$

If isomorphism $f : a \to b$ exists, we say that a and b are equal up to isomorphism(s), written as $a \cong b$.

Similar to the cartesian product of two sets, we may define the product of two categories.

For categories $\mathbf{C}$ and $\mathbf{D}$,

Def. the *product* $\mathbf{C} \times \mathbf{D}$ of $(\mathbf{C}, \mathbf{D})$ consists of:

- *objects*: $\mathrm{Obj}(\mathbf{C} \times \mathbf{D}) = \mathrm{Obj}(\mathbf{C}) \times \mathrm{Obj}(\mathbf{D})$,
  e.g. $(c, d) \in (\mathrm{Obj}(\mathbf{C}), \mathrm{Obj}(\mathbf{D}))$.
- *morphisms*: $\mathrm{Mor}(\mathbf{C} \times \mathbf{D}) = \mathrm{Mor}(\mathbf{C}) \times \mathrm{Mor}(\mathbf{D})$,
  e.g. $(f, g) \in (\mathrm{Mor}(\mathbf{C}), \mathrm{Mor}(\mathbf{D}))$.

As a generalization, morphisms are far more complicated than "$\leq$" or functions, it not only contains the information about its domain and codomain. We will witness this.

Def. the

1.4.5 Monotone function and functor

Let's consider a function f from preorder $(\mathbb{N}, \leq_{\mathbb{N}})$ to preorder $(\mathbb{R}^+, \leq_{\mathbb{R}^+})$:

$$\begin{aligned} f : \mathbb{N} &\longrightarrow \mathbb{R}^+ \\ x &\longmapsto e^x \end{aligned}$$

- *preservation of preorder*: If $a \leq_{\mathbb{N}} b$ in $\mathbb{N}$, then $e^a \leq_{\mathbb{R}^+} e^b$ in $\mathbb{R}^+$.
  e.g. $2 \leq_{\mathbb{N}} 3$, $e^2 \leq_{\mathbb{R}^+} e^3$:

$$\begin{tikzcd}
2 \arrow[r, mapsto, "{(a,b)\in\leq_P}"] \arrow[d, mapsto, "f"] \arrow[dr, phantom, "\downarrow (f,f)"] & 3 \arrow[d, mapsto, "f"] & (\mathbb{N}, \leq_{\mathbb{N}}) \arrow[d, "f"] \\
e^2 \arrow[r, mapsto, "{(e^2,e^3)\in\leq_Q}"'] & e^3 & (\mathbb{R}^+, \leq_{\mathbb{R}^+}) .
\end{tikzcd}$$

f also "takes" $r \in\leq_{\mathbb{N}}$ to $s \in\leq_{\mathbb{N}}$ e.g. $(f, f)(2, 3) = (e^2, e^3) \in\leq_{\mathbb{R}^+}$, f is called a monotone function:

For two preorders $(P, \leq_P)$ and $(Q, \leq_Q)$,

Def. a function $f : P \to Q$ is *monotone* if

- *monotonicity*$a \leq_P b$ then $f(a) \leq_Q f(b)$ for all elements $a, b \in P$, or equivalently
- $\forall (a, b) \in\leq_P$, $(f, f)(a, b) = (f(a), f(b)) \in\leq_Q$:

$$\begin{tikzcd}
a \arrow[r, mapsto, "{(a,b)\in\leq_P}"] \arrow[d, mapsto, "f"] \arrow[dr, phantom, "\downarrow (f,f)"] & b \arrow[d, mapsto, "f"] & (P, \leq_P) \arrow[d, "f"] \\
f(a) \arrow[r, mapsto, "{(f(a),f(b))\in\leq_Q}"'] & f(b) & (Q, \leq_Q) .
\end{tikzcd}$$

A monotone function f naturally preserves transitivity and reflexivity:

- *transitivity*: if $a \leq_P b \leq_P c$, then $f(a) \leq_Q f(b) \leq_Q f(c)$:

$$\begin{tikzcd} a \arrow[r, mapsto, "\leq_P"] \arrow[d, mapsto, "f"] & b \arrow[r, mapsto, "\leq_P"] \arrow[d, mapsto, "f"] & c \arrow[d, mapsto, "f"] & (P, \leq_P) \arrow[d, "f"] \\ f(a) \arrow[r, mapsto, "\leq_Q"] & f(b) \arrow[r, mapsto, "\leq_Q"] & f(c) & (Q, \leq_Q)\,; \end{tikzcd}$$

- *reflexivity*: $x \leq_P x$ and $f(x) \leq_Q f(x)$ always hold:

$$\begin{tikzcd} x \arrow[loop left, "\leq_P"] \arrow[d, mapsto, "f"] & \in & P \arrow[d, "f"] \\ f(x) \arrow[loop left, "\leq_Q"] & \in & Q. \end{tikzcd}$$

If we generalize this to morphisms in category, we can define a "function" from one category to another called functor.

a subcategory of $\mathbf{D}$. Def. a *functor* $F : \mathbf{C} \to \mathbf{D}$ consists of:

- a function

$$\begin{aligned} F_{\text{Obj}} : \text{Obj}(\mathbf{C}) &\longrightarrow \text{Obj}(\mathbf{D}) \\ a &\longmapsto F(a) \end{aligned}$$

  - similar to monotone function $f : P \to Q$;

- a function

$$\begin{aligned} F_{\text{Mor}} : \text{Mor}(\mathbf{C}) &\longrightarrow \text{Mor}(\mathbf{D}) \\ f &\longmapsto F(f) \end{aligned}$$

  - similar to $(f, f) : \leq_P \to \leq_Q$.

  such that $F(\mathbf{I})$ is a subcategory of D

- *compatibility*: $\forall f \in \text{Mor(C)}$ and $F(f) \in \text{Mor(D)}$,
  $\text{dom}(F(f)) = F(\text{dom}(f)) \in \text{Obj(D)}, \text{cod}(F(f)) = F(\text{cod}(f)) \in \text{Obj(D)}$; e.g.

$$\begin{tikzcd} a \arrow[rr, "f"] \arrow[d, "F"] & \arrow[d, "F"] & b \arrow[d, "F"] & \mathbf{C} \arrow[d, "F"] \\ F(a) \arrow[rr, "F(f)"'] & {} & F(b) & \mathbf{D} \end{tikzcd}$$

  - similar to monotonicity.

- *functoriality*:
  - if (f, g) is composable in $\mathbf{C}$, then $(F(f), F(g))$ is composable in $\mathbf{D}$, such that $F(gf) = F(g)F(f)$:

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[d, "F"] \arrow[rr, bend left, "gf"] & b \arrow[r, "g"] \arrow[d, "F"] & c \arrow[d, "F"] & \mathbf{C} \arrow[d] \\ F(a) \arrow[r, "F(f)"] \arrow[rr, bend right, "F(gf)=F(g)F(f)"'] & F(b) \arrow[r, "F(g)"] & F(c) & \mathbf{D}; \end{tikzcd}$$

    - $*$ similar to the preservation of transitivity.
  - for each object $x \in \mathrm{C}$, $F(1_x) = 1_{F(x)}$:

$$\begin{tikzcd} a \arrow[loop left, "1_x"] \arrow[d, "F"] & \mathbf{C} \arrow[d] \\ F(x) \arrow[loop left, "1_{F(x)}"] & \mathbf{D}. \end{tikzcd}$$

    - $*$ similar to the preservation of reflexivity.

Similar to **Set**, if we view functors as a generalization of functions, we may define the category of categories:

e.g **Cat** is a category, Obj(**Cat**) is the collection of all categories, Mor(**Cat**) is the collection of all the functors, identity morphisms are identity functors.

With the product of categories and functors, we may generalize symmetric monoidal preorder into strict symmetric monoidal category, the core concept in 2TQFT.

1.4.6 Monoidal preorder and category

A function that maps elements of monoid A to elements of another monoid B preserving the multiplicative structure is called a homomorphism between monoids.

If we combine the data of "$\leq$" and "$\otimes$" together, we get a new structure $(PM, \leq, I, \otimes)$, where commutativity is replaced by symmetry.

e.g. $(\mathbb{N}, \leq_{\mathbb{N}}, 0, +)$, $(\mathbb{N}, \leq_{\mathbb{R}^+}, 0, \times)$.

Def. a *symmetric monoidal preorder* $\mathbf{PM} = (PM, \leq, \otimes, I)$ consists of:

- a preorder $(PM, \leq)$;
- a monotone function $\otimes : PM \times PM \to PM$, such that:

- *associativity*: $\forall a, b, c \in PM$, $(a \otimes b) \otimes c = a \otimes (b \otimes c)$;
- *symmetry*: $\forall p, q \in M$, $p \otimes q = q \otimes p$.

- a unit element $I \in PM$, such that $\forall x \in PM$,
  - *left unity*: $I \otimes x = x$;
  - *right unity*: $x \otimes I = x$.

These data agree with the ones in $(\mathbb{N}, \leq, +, 0)$.

In order to generalize the underlying set PM into a category, we need to generalize the function $\otimes : PM \times PM \to PM$ to a special functor $\otimes : \mathrm{C} \times \mathrm{C} \to \mathrm{C}$. Before that, we need to define what is the product of categories and what is a functor between categories.

For a category $\mathbf{C}$,

Def. a *strict symmetric monoidal category* $(\mathbf{C}, \otimes, I, \sigma)$ consists of:

- a category $\mathbf{C}$;
- *monoidal product*: a functor $\otimes : \mathbf{C} \times \mathbf{C} \to \mathbf{C}$,

  functoriality: for two morphisms $f_1 : a_1 \to b_1$ and $f_2 : a_2 \to b_2$, then there exist a morphism $f_1 \otimes f_2 : a_1 \otimes b_1 \to a_2 \otimes b_2$:

$$
\begin{tikzcd}
(a_1, a_2) \arrow[rr, "{(f_1, f_2)}"] \arrow[d, "\otimes"] & {} \arrow[d, "\otimes"] & (b_1, b_2) \arrow[d, "\otimes"] & (\mathbf{C}, \mathbf{C}) \arrow[d, "\otimes"] \\
a_1 \otimes b_2 \arrow[rr, "f_1 \otimes f_2"'] & {} & b_1 \otimes b_2 & \mathbf{C};
\end{tikzcd}
$$

  such that:

  - *associativity*: $\forall a, b, c \in \mathrm{Obj}(\mathbf{C})$, $(a \otimes b) \otimes c = a \otimes (b \otimes c)$;
  - *braid*: $\forall a, b \in \mathrm{Obj}(\mathbf{C})$, $\exists \sigma_{a,b} : a \otimes b \approxeq b \otimes a \in \mathrm{Mor}(\mathbf{C})$, such that $\sigma_{a,b}\sigma_{b,a} = 1_{a \otimes b}$.

- *unit object*: an object $I \in \mathrm{Obj}(\mathbf{C})$, such that $\forall x \in \mathrm{Obj}(\mathbf{C})$,
  - *left unity*: $I \otimes x = x$;
  - *right unity*: $x \otimes I = x$.

e.g.$(\mathbf{Set}, \cup, \varnothing)$, $(\mathbf{Vect}_{\Bbbk}, \otimes, \Bbbk)$.

1.4.7 Monoidal monotone and monoidal functor

f also takes $+$ to $\cdot$, we may write it as $f(+) = \cdot$:

- *preservation of multiplication*: $\forall a, b \in \mathbb{N}, f(a + b) = f(a) \cdot f(b)$. e.g.:

$$
\begin{tikzcd}
(1,3) \arrow[rr, mapsto, "+"] \arrow[d, mapsto, "{(f,f)}"] & \arrow[d, phantom, "\downarrow f"] & 4 \arrow[d, mapsto, "f"] \\
(e, e^3) \arrow[rr, mapsto, "\cdot"'] & & e^4;
\end{tikzcd}
$$

These are data of monoidal monotone.

For two symmetric monoidal preorders $\mathbf{P} = (P, \leq_P, \otimes_P, I_P)$ and $\mathbf{Q} = (Q, \leq_Q, \otimes_Q, I_Q)$,
Def. a *monoidal monotone* $f : \mathbf{P} \to \mathbf{Q}$ is a monotone function $f : P \to Q$, such that:

- *preservation of multiplication*: $\forall a, b \in P, f(a) \otimes_Q f(b) \leq_Q f(a \otimes_P b)$:

$$
\begin{tikzcd}
(a,b) \arrow[rr, mapsto, "\otimes_P"] \arrow[d, mapsto, "{(f,f)}"] & \arrow[d, phantom, "\downarrow f"] & a \otimes_P b \arrow[d, mapsto, "f"] \\
(f(a), f(b)) \arrow[r, mapsto, "\otimes_Q"'] & f(a) \otimes_Q f(b) \leq_Q f(a \otimes_P b); &
\end{tikzcd}
$$

- *preservation of unit element*: $f(I_P) = I_Q$;

Then we substitute preorders by categories.

For two strict symmetric monoidal categories $(\mathbf{C}, \otimes_{\mathbf{C}}, I_{\mathbf{C}})$ and $(\mathbf{D}, \otimes_{\mathbf{D}}, I_{\mathbf{D}})$,
Def. a *symmetric monoidal functor* $F : (\mathbf{C}, \otimes_{\mathbf{C}}, I_{\mathbf{C}}) \to (\mathbf{D}, \otimes_{\mathbf{D}}, I_{\mathbf{D}})$ is a functor such that:

- *preservation of multiplication*: $\forall a, b \in \mathrm{Obj}(\mathbf{C})$, a morphism
  $f : F(a) \otimes_{\mathbf{D}} F(b) \to F(a \otimes_{\mathbf{C}} b)$:

$$
\begin{tikzcd}
(a,b) \arrow[rr, "\otimes_{\mathbf{C}}"] \arrow[d, "{(F,F)}"] & \arrow[d, phantom, "\downarrow F"] & a \otimes_{\mathbf{C}} b \arrow[d, "F"] \\
(F(a), F(b)) \arrow[r, "\otimes_{\mathbf{D}}"'] & F(a) \otimes_{\mathbf{D}} F(b) \to F(a \otimes_{\mathbf{C}} b); &
\end{tikzcd}
$$

- *preservation of unit element*: $F(I_{\mathbf{C}}) = I_{\mathbf{D}}$;

1.4.8 Largest and smallest, terminal and initial

A subset of $\mathbb{R}$ has infimum(the largest element of lower bounds) and supremum(the smallest element of upper bounds).

For $A \subseteq \mathbb{R}$,
Def. the *lower class* is a set $L_b \subseteq \mathbb{R}$ such that

$\forall a \in A$ and $x \in L_b$, $a \leq_{\mathbb{R}^+} x$.

Def. the *infimum* is an element $\inf(A) \in \mathbb{R}$ such that $\forall x \in L_b$, $x \leq_{\mathbb{R}^+} \inf(A)$.

e.g. For $A = [1, 4)$, $Lb = (-\infty, 1]$, $\inf(A) = 1$.

If we substitute $\leq_{\mathbb{R}^+}$ with $\geq_{\mathbb{R}^+}$, the inverse of $\leq_{\mathbb{R}^+}$, we get the upper bound and supremum.

Def. a *upper class* is a set $U_b \subseteq \mathbb{R}$ such that

$\forall a \in A$ and $x \in U_b$, $x \leq_{\mathbb{R}^+} a (a \geq_{\mathbb{R}^+} x)$.

Def. a *supremum* is an element $\sup(A) \in \mathbb{R}$ such that $\forall x \in U_b$, $\sup(A) \leq_{\mathbb{R}^+} x$ $(x \geq_{\mathbb{R}^+} \sup(A))$.

e.g. For $A = [1, 4)$, $Ub = [4, \infty)$, $\sup(A) = 4$.

We may generalize the definition of "largest/smallest", and then "infimum/supremum" into preorders and categories.

For a preorder $(P, \leq)$,

Def. a element $\max(P)$ is *largest* if $\forall a \in P$, $(a, \max(P)) \in \leq$.

Def. a element $\min(P)$ is *smallest* if $\forall a \in P$, $(\min(P), a) \in \leq$.

For a category $\mathbf{C}$,

Def. an object $tmn(\mathbf{C}) \in \mathrm{Obj}(\mathbf{C})$ is *terminal* if $\forall c \in \mathrm{Obj}(\mathbf{C})$, $\exists! f : c \to tmn(\mathbf{C})$.

- if A, then $\exists!$ B is the *universal property*;
- the "largest element" of $\mathbf{C}$.

Def. an object $ini(\mathbf{C}) \in \mathrm{Obj}(\mathbf{C})$ is *initial* if $\forall c \in \mathrm{Obj}(\mathbf{C})$, $\exists! f : ini(\mathbf{C}) \to c$.

- the "smallest element" of $\mathbf{C}$.

e.g. for **Set**, $\varnothing$ is the initial object; $\forall a$,$\{a\}$ is a terminal object.

e.g. a initial/terminal object of $\mathbf{C}$ is the terminal/initial object of $\mathbf{C}^{op}$.

A terminal/initial object may not exist.

e.g. a category $(\{a, b\}, \{1_a, 1_b\})$ has no morphism $f : a \to b$ or $g : b \to a$, hence it's impossible to tell which object is "smaller".

1.4.9 Meet, join, and cone

For a preorder $(P, \leq)$, and a subset $A \subseteq P$:

Def. the *lower class* of A is a set $low(A) = \{\forall l \in P | \forall a \in A, l \leq a\}$.

Def. a *meet* of A is $m(A) = max(low(A))$, an largest element of lower class.
e.g. $\{2\}$ is the meet of $\{\{1,2\}, \{0,2\}\} = \{1,2\} \bigcap \{0,2\} \subset X$, i.e. the intersection of $\{1,2\}$ and $\{0,2\}$.

Def. the *upper class* of A is a set $upp(A) = \{\forall u \in P | \forall a \in A, a \leq u (u \geq_{\mathbb{R}+} a)\}$.

Def. a *join* of A is $j(A) = min(upp(A))$, a smallest element of lower class.
the lower class is $\{\forall (m,n) | m,n \in \mathbb{N}, m \leq 3, n \leq 4\}$, the meet is $(3,5)$;
the upper class $upp(P) = \{\forall (m,n) | m,n \in \mathbb{N}, 3 \leq m, 4 \leq n\}$, the join is $(7,4)$.

$$\begin{tikzcd}
 & & & (9,8) \\
 & (3,5) \arrow[r, "\leq"] \arrow[urr, bend left, "\leq"] & (7,5) \arrow[ur, "\leq"] & \\
 & (3,4) \arrow[u, "\leq"] \arrow[r, "\leq"] & (7,4) \arrow[u, "\leq"] \arrow[uur, bend right, "\leq"'] & \\
(2,4) \arrow[ur, "\leq"] \arrow[uur, bend left, "\leq"] \arrow[urr, bend right, "\leq"'] & & &
\end{tikzcd}$$

e.g. For a set $AB = \{A, B\}$ and preorder (sets, $\subseteq$),
the lower class is $\{\forall S | S \subseteq A \cap B\}$, the meet is $A \cap B$;
the upper class is $\{S \forall | S, A \cup B \subseteq S\}$, the join is $A \cup B$.

$$\begin{tikzcd}
 & & & A \cup B \cup \cdots \\
 & A \arrow[r, "\subseteq"] \arrow[urr, bend left, "\subseteq"] & A \cup B \arrow[ur, "\subseteq"] & \\
 & A \cap B \arrow[u, "\subseteq"] \arrow[r, "\subseteq"] & B \arrow[u, "\subseteq"] \arrow[uur, bend right, "\subseteq"'] & \\
\{\} \arrow[ur, "\subseteq"] \arrow[uur, bend left, "\subseteq"] \arrow[urr, bend right, "\subseteq"'] & & &
\end{tikzcd}$$

To generalize meet into categories, let's state lower bound and meet differently.
For a preorder $(P, \leq)$, a subset $A \in P$, and $\leq_A = \{\forall (a,b) \in \leq | a, b \in A\}$ Def. a *cone* $(x, C(x))$ of A consists of:

- an element $x \in P$;
- a relation $C(x) = \{\forall(x,a)|a \in A\}$, such that $C(x) \in \leq$.
    - the cone and A forms a preorder $A_x = (x \cap A, C(x) \cap \leq_A)$. $\forall a_x \in A_x, x \leq a_x$;
    - e.g. $((2,4), C((2,4)) = ((2,4), \{((2,4),(3,5)), ((2,4),(7,3))\})$,
      $((3,4), C((3,4)) = ((3,4), \{((3,4),(3,5)), ((3,4),(7,3))\})$.

$$\begin{tikzcd}
(3,5) & & (3,5) \\
(2,4) \arrow[u, mapsto, "{((2,4),(3,5))}"'] \arrow[r, mapsto, "{((2,4),(7,3))}"'] & (7,3), \quad (3,4) \arrow[r, mapsto, "{((3,4),(7,3))}"'] \arrow[ur, phantom] & (7,3).
\end{tikzcd}$$

Hence $(x, C(x))$ is a cone if and only if $x \in low(A)$.

the collection of all the cones of A form a set $Cone(A)$.

For two cones $(x, C(x))$ and $(y, C(y))$, if $x \leq y$, then $\forall a \in A$, $x \leq y \leq a = x \leq a$.

We may also interpret $x \leq y$ as a preorder relation $\leq_c cone \subseteq Cone(A) \times Cone(A)$ $\;(x, C(x)) \xmapsto{(x,y)} (y, C(y$

We may construct a preorder $\mathbf{Cone}(A) = (Cone(A), \leq_{cone})$.

e.g. $\;((2,4), C((2,4))) \xmapsto{((2,4),(3,4))} ((3,4), C((3,4)))$,

$$\begin{tikzcd}
 & (3,5) & \\
 & (3,4) \arrow[u, mapsto, "{((3,4),(3,5))}"'] \arrow[r, mapsto, "{((3,4),(7,3))}"'] & (7,3) \; . \\
(2,4) \arrow[uur, mapsto, bend left, "{((2,4),(3,5))}"] \arrow[ur, mapsto, "{((2,4),(3,4))}"'] \arrow[urr, mapsto, bend right, "{((2,4),(7,3))}"'] & &
\end{tikzcd}$$

A largest element of $\mathbf{Cone}(A)$ is $max(Cone(A)) = (m(A), C\,(m(A)))$

Def. a *meet* of A is an element m(A).

1.4.10 Limit and product, colimit and coproduct

To generalize meet into limit of functors, we need to generalize the concepts we used in the following way:

- "subset" becomes $F(\text{Obj}(\mathbf{C}))$, a subcategory of a category $\mathbf{D}$ given by a functor $F : \mathbf{C} \to \mathbf{D}$;
- "smaller" (a,b) becomes a morphism $c(a,b)$;
  For cone $(x, C(x))$,

- $C(x)$ becomes a collection of morphisms (compatible with morphisms in $F(\mathbf{I})$);
- "meet" becomes limit, "largest element" becomes terminal object.

1.Use functor to Define a "subpreorder":
for a small category $\mathbf{I}$, a functor $F : \mathbf{I} \to \mathbf{D}$ called, then $F(\mathbf{I})$ is a subcategory of $\mathbf{D}$.

2. Generalize cone(equip a object x with a selection of morphisms $x \to F(\mathbf{I})$):

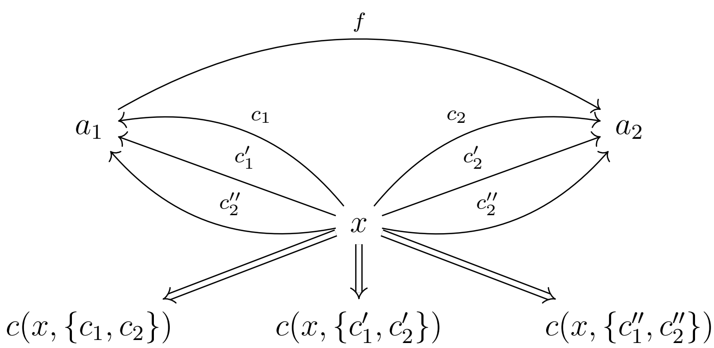
Def. a *cone* $(x, C(x))$ of F consists of:

- an object $x \in \mathbf{D}$, called the base object;
- a collection of morphisms $C(x)$ consists of $c(x, F(a))$:
  $\forall a \in \mathrm{Obj}(\mathbf{I})$, a morphism $c(x, F(a)) : x \to F(a) \in \mathrm{Mor}(\mathbf{D})$.
  - $c(x, F(a))$ is similar to (x, a), x is "smaller" than all the objects in $F(\mathrm{Obj}(\mathbf{I}))$.

  such that
- $\forall f : a \to b \in \mathrm{Mor}(\mathbf{I})$, $F(f) : F(a) \to F(b) \in \mathrm{Mor}(\mathbf{D})$ such that $F(f)c(x, F(a)) = c(x, F(b))$:

$$\begin{tikzcd} x \arrow[r, "{c(x,F(a))}"] \arrow[rr, bend left, "{F(f)c(x,F(a))=c(x,F(b))}"] & F(a) \arrow[r, "F(f)"] & F(b). \end{tikzcd}$$

  - "smaller" is produced uniformly and "$\leq$" in "subset" $F(\mathbf{I})$ preserves "transitivity";
  - $(x, C(x))$ and $F(\mathbf{I})$ forms a category $\mathbf{F_{cone}}(x \cup \mathrm{Obj}\,(F(\mathbf{I}))\,, C(x) \cup \mathrm{Mor}\,(F(\mathbf{I})))$, x is a initial object of $\mathbf{F_{cone}}$, hence a "lower bound";
  - different collection of morphisms $C(x), C'(x), C''(x), \cdots$ define different cones, hence an object x is "decomposed" into many cones, each of them equips with a unique collection of morphisms, e.g.



3. Generalize **Cone**(A)(cones form a "lower class"):

Def. **Cone**(F) is a category consists of:

- *objects*: the cones of F; For a morphism $f : x \to y \in \mathrm{Mor}(\mathbf{D})$,
- *morphisms*: a morphism $f_{cone} : (x, C(x)) \to (y, C(y)) \in \mathrm{Mor}(\mathbf{Cone}(F))$ such that $\forall s \in \mathrm{Obj}(\mathbf{I})$, $c(y, F(a))f = c(x, F(a))$:

$$\begin{tikzcd} x \arrow[r, "f"] \arrow[rr, bend left, "c(y,F(a))f=c(x,F(a))"] & y \arrow[r, "c(y,F(a))"] & F(a) \end{tikzcd} .$$

  – similar to $\leq_{cone}$.

4. Fetch the "maximal" of **Cone**(F):

For a functor $F : \mathbf{I} \to \mathbf{D}$, the terminal object of **Cone**(F) is $\big(\lim(F), c_{\lim(F)}\big)$,
Def. the *limit* of F is $\lim(F)$.
Therefore, meet is a special limit where functor F is replaced by some embedding of subset, and the morphisms in $\mathbf{D}$ are replaced by $\leq$ in P.

Product, an important limit, is another example.

For a category $\mathbf{P} = (\{a, b\}, \{1_a, 1_b\})$, a category $\mathbf{D}$, and a functor $F : \mathbf{P} \to \mathbf{D}$,
Def. the *product* of $F(a), F(b) \in \mathrm{Obj}(\mathbf{D})$ is $\lim(F)$.
$\forall c(x, F(a)), c(x, F(b) \in \mathbf{D})$, $1_a c(x, F(a)) = c(x, F(a))$ and $1_b c(x, F(b)) = c(x, F(b))$,

hence $\forall x \in \mathrm{Obj}(\mathbf{D})$, $(x, \{c(x, F(a)), c(x, F(b))\}$ forms a cone in $Cone(F)$: $\begin{tikzcd} F(a) & \\ X \arrow[u, "c(X,F(a))"] \arrow[r, "c(X,F(b))"'] & F(b) \end{tikzcd}$.

e.g. product of two sets A and B: a set X and any two functions $c(X, A)$, $c(X, B)$ forms a cone $(X, \{c(X, A), c(X, B)\})$: $\begin{tikzcd} A & \\ X \arrow[u, "c(X,A)"] \arrow[r, "c(X,B)"'] & B \end{tikzcd}$.

Def. the *product* of two sets A, B is the the set $A \otimes B$ such that for a set X and any two functions $c(X, A) : S \to A$ and $c(X, B) : X \to B$, $\exists! T : S \to A \otimes B$ such that $p_A T = c(X, A)$ and $p_B T = c(X, B)$:

$$
\begin{tikzcd}
 & A & \\
 & A \otimes B \arrow[u, "p_A"'] \arrow[r, "p_B"'] & B \\
X \arrow[uur, bend left, "c(X{,}A)"] \arrow[ur, dotted, "\exists ! T"] \arrow[urr, bend right, "c(X{,}B)"'] & &
\end{tikzcd}
\qquad
\begin{tikzcd}
 & a & \\
 & (a,b) \arrow[u, mapsto] \arrow[r, mapsto] & b \\
x \arrow[uur, mapsto, bend left] \arrow[ur, mapsto, dotted, "\exists !"] \arrow[urr, mapsto, bend right] & &
\end{tikzcd}
$$

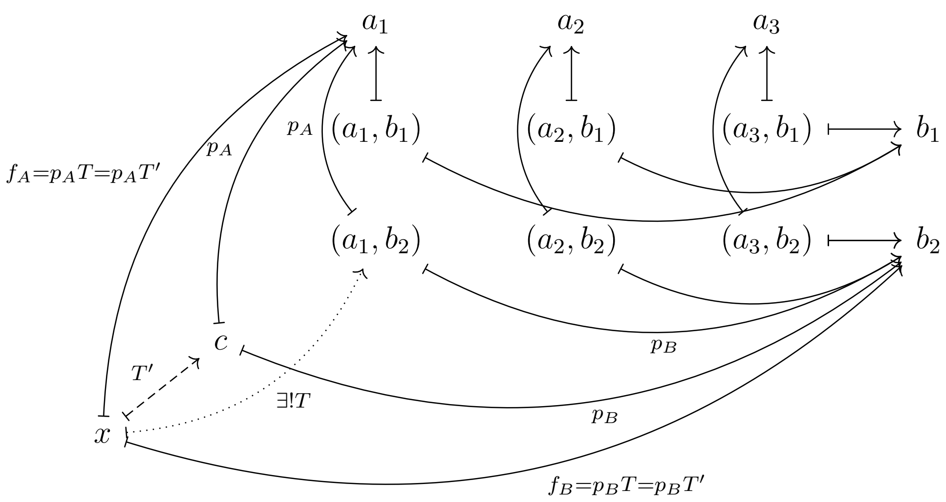
The definition of Cartesian product and product are compatible, i.e.

$A \otimes B = A \times B$, $p_A = pr_A$, $p_B = pr_B$.

$A \times B$ is the range of $(c(X, A), c(A, Y))$. We will use a trivial example to explain the following statements:

- larger sets have extra elements that will kill the uniqueness.
- smaller sets may fall to reach some value.

e.g.



- If there are extra elements in $A \otimes B$ such as c, its value must be the same with an element in (A, B). e.g. if $f_A(x) = a_1$, $f_B(x) = b_2$, there are two ways to get them:
  - $T(x) = (a_1, b_2)$, then $p_A(a_1, b_2) = a_1, p_B(a_1, b_2) = b_2$;
  - $T'(x) = c$, then $p_A(c) = a_1, p_B(c) = b_2$.

  This breaks the uniqueness of L, hence $A \times B \bigcup \{c\}$ is not a terminal object;
- If we delete a element in $A \times B$, e.g. (a_1, b_2), then there's no way to get to $h_A(x) = a_1$, $h_B(x) = b_2$, hence $A \times B$ without (a_1, b_2) is not a terminal object.

Hence $(A \times B, C(A \times B))$ is indeed the terminal object of $\mathbf{Cone}(F)$. Therefore $A \times B$ is the limit.

In conclusion, $A \times B$ is a product of A and B.

On the opposite, we may define colimit(the limit of $F : \mathbf{I}^{op} \to \mathbf{D}^{op}$).

1.For a small category $\mathbf{I}$ such that $\mathrm{Obj}(\mathbf{I})$ is a set and functor $F : \mathbf{I} \to \mathbf{D}$, we get a subcategory $F(\mathbf{I})$ of $\mathbf{D}$.

2. "upper bound":

Def. a *cocone* $(x, Co(x))$ of F consists of:

- an object $x \in \mathbf{D}$, called the base object;
- a collection of morphisms $C(x)$ consists of $c(F(a), x)$:
  $\forall a \in \mathrm{Obj}(\mathbf{I})$, $\exists! c(F(a), x) : F(a) \to x \in \mathrm{Mor}(\mathbf{D})$.

3. "upper class"

Def. $\mathbf{Cocone}(F)$ is a category consists of:

- *objects*: the cocones of F;
  For a morphism $f : x \to y \in \mathrm{Mor}(\mathbf{D})$,
- *morphisms*: a morphism $f_{coc} : (x, Co(x)) \to (y, Co(y)) \in \mathrm{Mor}(\mathbf{Cone}(F))$ such that $\forall a \in \mathrm{Obj}(\mathbf{I})$, $fco(F(a), x) = co(F(a), y)$:

$$\begin{tikzcd} F(a) \arrow[r, "c(F(a){,}x)"] \arrow[rr, bend left, "fco(F(a){,}x)=co(F(a){,}y)"] & x \arrow[r, "f"] & y \end{tikzcd} .$$

4. Fetch the "minimal" of $\mathbf{Cocone}(F)$:

For a functor $F : \mathbf{I} \to \mathbf{D}$, the initial object of $\mathbf{Cone}(F)$ is $\big(\mathrm{col}(F), co_{\mathrm{col}(F)}\big)$, then

Def. the *colimit* of F is $\mathrm{col}(F)$.

Therefore, join is a special colimit where functor F is replaced by some embedding of subset, and the morphisms in $\mathbf{D}$ are replaced by $\leq$ in P.

Coproduct, an important colimit, is another example.

For a category $\mathbf{P} = (\{a, b\}, \{1_a, 1_b\})$, a category $\mathbf{D}$, and a functor $F : \mathbf{P} \to \mathbf{D}$,

Def. the *coproduct* of $F(a), F(b) \in \mathrm{Obj}(\mathbf{D})$ is $\mathrm{col}(F)$.

e.g. coproduct of two sets A and B: a set Y and any two functions $c(A,Y)$, $c(B,Y)$ forms

a cocone:$(Y, \{co(A,Y), co(B,Y)\})$
$$\begin{tikzcd} A \arrow[r, "co(A{,}Y)"] & Y \\ & B \arrow[u, "co(B{,}Y)"'] \end{tikzcd}.$$

Def. the *coproduct* of two sets A, B is the the set $A+B$ such that $\forall S$ and two functions $co(A,S) : A \to S$ and $co(B,S) : B \to S$, $\exists! T : A+B \to S$ such that $Teb_A = co(A,S)$ and $Teb_B = co(B,S)$.

e.g. $y_a = T(a') = co(A,S)(a), y_b = T(b'') = co(B,S)(b)$:

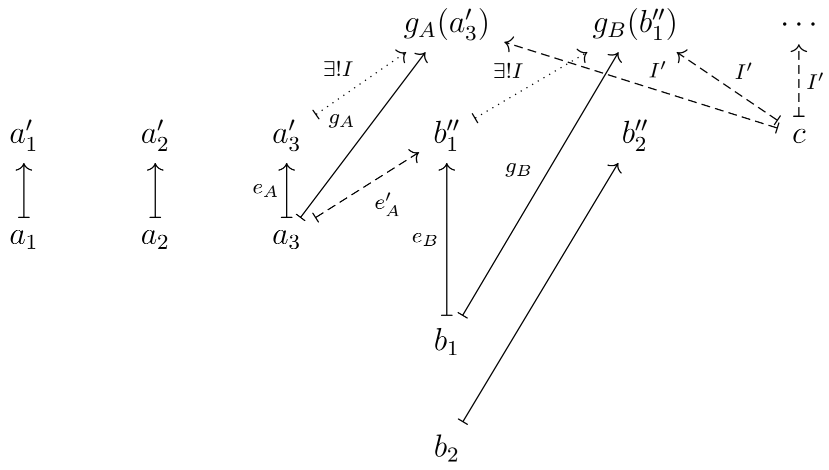
$$\begin{tikzcd} & & Y \\ A \arrow[r, "eb_A"] \arrow[urr, bend left, "co(A{,}Y)"] & A+B \arrow[ur, dotted, "\exists! T"] & \\ & B \arrow[u, "eb_B"'] \arrow[uur, bend right, "co(B{,}Y)"'] & \end{tikzcd} \qquad \begin{tikzcd} & & y_a/y_b \\ a \arrow[r, mapsto] \arrow[urr, mapsto, bend left] & a'/b'' \arrow[ur, dotted, mapsto, "\exists!"] & \\ & b \arrow[u, mapsto] \arrow[uur, mapsto, bend right] & \end{tikzcd}$$

The definition of disjoint union and coproduct are compatible, i.e. $A + B = A \bigsqcup B$.

$A \bigsqcup B$ is the domain of $(co(A,Y), co(B,Y))$. We will use a trivial example to explain the following arguments:

- larger sets have extra elements that will kill the uniqueness;
- smaller sets may lose results and fall to be initial.

e.g. $g_A(a_3), g_B(b_1) \in Y$, $g_A(a_3) \neq g_B(b_1)$,



- If there are extra elements in $A+B$ such as c, then $I(c)$ can be any value in Y, hence breaks the uniqueness of I, $A \bigsqcup B, \{c\}$ is not a initial object;
- If we delete an element in $A \bigsqcup B$, e.g. a_3', then $e_A(a_3)$ have to go eleswhere. If $e_A(a_3) = b_1''$, then $Ie_A(a_3) = Ie_B(b_1)$, contridict to

  $Ie_A(a_3) = g_A(a_3) \neq g_B(b_1) = Ie_B(b_1)$, hence $A \bigsqcup B$ without a_3'' is not an initial object.

Hence $(A \bigsqcup B, Co(A \times B))$ is indeed the initial object of $\mathbf{Cocone}(F)$. Therefore $A \bigsqcup B$ is the colimit.

In conclusion, $A \bigsqcup B$ is a coproduct of A and B.

We may put product and coproduct into one graph:

$$\begin{tikzcd}
& & & Y \\
& A \arrow[r, "eb_A"] \arrow[urr, bend left, "g_A"] & A \bigsqcup B \arrow[ur, dotted, "\exists! I"] & \\
& A \times B \arrow[u, "pr_A"] \arrow[r, "pr_B"] & B \arrow[u, "eb_B"'] \arrow[uur, bend right, "g_B"'] & \\
X \arrow[ur, dotted, "\exists! T"] \arrow[uur, bend left, "f"] \arrow[urr, bend right, "f"'] & & &
\end{tikzcd} .$$

1.5 Galois connection and adjoint functor

To be written...

Chapter 2

Bibliography

[1] H.B. Enderton. *Elements of Set Theory*. Elsevier Science, 1977.

[2] H.B. Enderton. *A Mathematical Introduction to Logic*. Elsevier Science, 2001.

[3] Brendan Fong and David I Spivak. Seven sketches in compositionality: An invitation to applied category theory, 2018.

[4] E. Riehl. *Category Theory in Context*. Aurora: Dover Modern Math Originals. Dover Publications, 2017.