0. Preliminaries on Sets and Maps

A *set* X is a collection of *elements*. When x is an element of X, we write $x \in X$. When a set X consists of specified elements $x, y, \ldots$, we write $X = \{x, y, \ldots\}$. What follows is a list of sets whose notation and name we fix throughout the sequel.

- the *empty* or *null* set $\varnothing = \{\}$
- the generic *singleton* $\star = \{*\}$
- the *Booleans* $\mathbb{B} = \{\top, \bot\}$, where $\top$ is "true" and $\bot$ is "false"
- the generic n-element set $\mathbf{n} = \{0, 1, \ldots n-1\}$
- the *natural numbers* $\mathbb{N} = \{0, 1, \ldots\}$
- the *integers* $\mathbb{Z} = \{\cdots - 1, 0, 1, \ldots\}$

Two sets A, B are said to be equal, in which case we may write $A = B$, when $x \in A$ if and only if $x \in B$. We note that $\varnothing = \mathbf{0}$, and hence these can be used interchangeably. The *cardinality* of a set is its number of elements. Although $\star$ and $\mathbf{1}$ are not equal, they both aspire towards being "generic" sets of cardinality 1 and are hence interchangeable since their respective elements are mere placeholders. In both cases, will choose the notation whose evocation best matches the context.

We say A is a *subset* of X and write $A \subseteq X$ when $x \in A$ implies $x \in X$. When we wish to define a subset B from a set X that keeps only the elements of X that satisfy some *condition* P, we write $B = \{x \in X \mid P(x)\}$. One reads each term in this expression using the following dictionary:

$\{\}$	"the set of"
$x \in X$	"x's in X"
$\mid$	"such that"
$P(x)$	"x satisfies the condition P"

Given two subsets $A, B \subseteq X$, we can use this notation to define two new subsets

- the *union* $A \cup B = \{x \in X \mid x \in A \text{ or } x \in B\}$
- the *intersection* $A \cap B = \{x \in X \mid x \in A \text{ and } x \in B\}$

This style of specifying sets is called *set-builder* notation. More generally, we can replace the element $x \in X$ with some expression consisting of elements, in which case we use the convention that the set to which each element in the expression belongs is written to the right of the '$|$' symbol. For example, we can define the set $\mathbb{Q}$ of *rational numbers* using this notation:

$$\mathbb{Q} = \{\tfrac{p}{q} \mid p, q \in \mathbb{Z}, q \neq 0\}$$

In this case our set consists of expression given by the fractions p/q, whose constituent terms p, q are subject to the condition that they are integers, and that q is subject to the condition that it does not equal 0. Note that we have thus far employed commas as an informal "and."

A function, or henceforth, *map* f of type $X \to Y$, for sets X and Y, is a rule that assigns each element $x \in X$ precisely one element $fx \in Y$. We call X the *domain* and Y the *codomain* of f. We often refer to an element of the domain as an *argument* and an element of the codomain as a *value.* We say f *takes arguments* in X and *takes values* in Y. When applying a map f to an argument x, we say it *evaluates* to the value fx. When we specify a map by both its type and its rule, we use the notation

$$f : X \to Y :: x \mapsto fx,$$

which should be read using the following dictionary

$$\begin{array}{c||c} f & \text{“the map } f\text{”} \\ : & \text{“of type”} \\ X \to Y & \text{“}X \text{ to } Y\text{”} \\ :: & \text{“given by”} \\ x \mapsto fx & \text{“}x \text{ maps to } fx\text{”} \end{array}$$

We write $X \xrightarrow{f} Y$ as shorthand for $f : X \to Y$. Sometimes, instead of inserting '::' between the type and the rule, we simply place the rule below the type so as to align elements with their sets:

$$X \xrightarrow{f} Y$$
$$x \mapsto fx$$

Two maps $f, g : X \to Y$ are said to be equal precisely when $fx = gx$ for all arguments $x \in X$.

It may be worthwhile to name a map by its rule. We call such a specification an *anonymous declaration.* If the intricacy of a situation calls for careful formality, we will use λ-calculus syntax, which would express our above map via the notation

$$\lambda x.fx : X \to Y$$

This is particularly useful in expressing situations in which an argument maps to a function. For example, consider the map $I : \mathcal{C}^\infty \to \mathcal{C}^\infty$ that sends a smooth map f to the smooth map given by mapping x to $\int_0^x f(t)dt$. We can write this as

$$I : \mathcal{C}^\infty \to \mathcal{C}^\infty :: f \mapsto \lambda x. \int_0^x f(t)dt \tag{1}$$

More often, however, we will use the less formal notation $f(-)$, which should evaluate an argument x by substituting x in place of the $-$. For example:

$$[\textstyle\int_0^x (-)dt](1) = x$$

If our map involves arithmetic, we will use $\square$ in place of $(-)$ to avoid ambiguity.

For every set X, its *identity map* is defined as

$$\mathbb{1}_X : X \to X :: x \mapsto x.$$

Given maps

$$f : X \to Y :: x \mapsto fx \text{ and } G : Y \to Z :: y \mapsto gy$$

we can form their *composition* $g \circ f$, or simply gf, as

$$gf : X \to Z :: x \mapsto gfx.$$

The composition and identity satisfy two important axioms.

- *(associativity)* $[f \circ g] \circ h = f \circ [g \circ h]$, whenever these are composable.
- *(identity)* $f \circ \mathbb{1}_X = f = \mathbb{1}_Y \circ f$, for $f \colon X \to Y$.

The following elementary examples serve to demonstrate that map composition and identity generalize familiar arithmetic operations and identities.

$$(\square + n) \circ (\square + m) = \square + (n + m)$$
$$(\square \cdot n) \circ (\square \cdot m) = \square \cdot (nm)$$
$$(\square + n) \circ (\square + 0) = (\square + 0) \circ (\square + n) = \square + n$$
$$(\square \cdot n) \circ (\square \cdot 1) = (\square \cdot 1) \circ (\square \cdot n) = \square \cdot n$$

We would be remiss not to point out the following frustration. When written as a string of arrows, our situation is represented by the *diagram*

$$X \xrightarrow{f} Y \xrightarrow{g} Z,$$

which automatically renders in many minds as fg instead of gf. This unfortunate convention is a consequence of the '$f(x)$' notation, due to Euler, for the evaluation of an argument x by a map f. When treating historically ossified contexts like linear algebra, we will bend to this convention. In situations for which it would heretical to do so, however, we write the above composition in *diagrammatic order* as $f \mathbin{/\!\!/} g$, to be read as "f then g." In such contexts, we will also write $x \mathbin{/\!\!/} f$ in place of $f(x)$. This is justified by interpreting each element $x \in X$ as the map

$$x : \star \to X :: * \mapsto x.$$

This notation, with equal precision yet more succinctness and or evocativeness, recaptures familiar facts in terms of map algebra. For example, we can reconceptualize "completing the square" as factoring—via $\circ$ not $\times$—a quadratic map

$$q : \mathbb{R} \to \mathbb{R} :: x \mapsto ax^2 + bx + c$$

into the map composition

$$q = L_v \circ \square^2 \circ L_h$$

for some linear polynomial maps L_v, L_h. For example, the equality on values

$$q(x) = x^2 - 6x + 16 = (x-3)^2 + 7$$

can now be recast as the equality on maps

$$q = (\square + 7) \circ \square^2 \circ (\square - 3).$$

This has the advantage of assessing q directly without considering its evaluation on arguments. Furthermore, this modularizes q into composable sub-processes. This decomposition, although obscured by the classic notation, is, as we shall soon see, precisely why completing the square solves quadratic equations.

Sometimes it's meaningful to consider a map which takes a pair of arguments; for example addition $+$ or multiplication $\times$ of numbers. To define this we need to construct a domain that consists of pairs of arguments. Given two sets X, Y, we define their *Cartesian product*

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

as the set of ordered pairs whose first entry is in X and second entry in Y. Then, a binary operation $\odot$ taking pairs of X elements as argument and outputting Y values can be conceived of as a map $\odot : X \times X \to Y$. We use the *infix* notation $x \odot y$ as shorthand for $\odot(x, y)$. For example, addition and multiplication of pairs of integers yields an operations of type $\mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$. Suppose we wish to iterate $\odot$ on more than two arguments, as so:

$$(((w \odot x) \odot y) \odot z).$$

This operation takes 4 inputs. More generally, we define an n-ary operation as one that takes n arguments. Such an operation has as domain the *Cartesian power*

$$X^n = \{(x_0, \ldots, x_{n-1}) \mid \forall i \in \mathbf{n}, x_i \in X\}.$$

Given a binary operation, the number of possible n-ary operations we can form from it grows wildly with n—in fact, any binary tree of arguments corresponds to a possible n-ary operation. This dramatically simplifies if we suppose further that $\odot$ is *associative*:

$$x \odot (y \odot z) = (x \odot y) \odot z.$$

We can then simply write $x \odot y \odot z$ without concern for bracketing. Iteratively applying associativity allows for the unique extension of $\odot$ to the following n-ary operator that takes as argument any finite non-empty sequence $(x_j)_{j=0}^{n-1} \in X^n$.

$$\bigodot_{j=0}^{n-1} : X^n \to Y :: (x_j)_{j=0}^{n-1} \mapsto x_0 \odot \cdots \odot x_{n-1}$$

Note that, by convention, we simply write $\bigodot_{j=0}^{n-1} x_i$ in place of $\bigodot_{j=0}^{n-1}(x_i)_{i=0}^{n-1}$. When $\odot$ is invariant under swapping arguments, we say it is *commutative*:

$$x \odot y = y \odot x.$$

In this situation, since the order does not matter, $\odot$ can be extended to an operation that takes any non-empty finite set J's worth of arguments. This can be encoded by the *indexed tuple* $(x_j)_{j\in J}$, where we call J the *indexing set*. We call the extension of $\odot$ to such arguments a (finitely) *indexed* operation. What exactly is the domain of this map? Convince yourself that we may conceive of the indexed tuple $(x_j)_{j\in J}$ as a map $J \to X$. In this case the domain should be the set of such maps, which we will for now denote as $[J \to X]$. We then denote the finitely indexed operation as follows.

$$\bigodot_{j\in J} : [J \to X] \to Y.$$

Note that, since the n-ary version can be seen as the special case of the above, we have that X^n is in some sense "the same" as $[\mathbf{n} \to X]$. We will come back to this later in the lecture.

Sometimes, for the sake of encoding structure more systematically, we may also be interested in the *nullary* version of $\odot$, i.e. on that takes zero arguments. What exactly could this mean? The idea is that this should be the X element, if one such exists, that is inert under $\odot$.

Towards further formally, we make the following aside. Consider sets J, I. We form their *disjoint union* or *sum* $J + I$ as the set consisting of the J-elements alongside the I-elements. This is not the same as the union! One takes the union of two *subsets* $A, B \subseteq X$ of a fixed *ambient set* X by considering the set of X-elements that are present in at least one of A or B. This means that an X-element that appears in both A and B is not double counted. However, we need this ambient set X in order to even compare elements of A with those of B, since these a priori have nothing to do with each other. In practice, these two concepts are difficult to separate since most of the time we express the elements of our two sets J, I using a shared "alphabet of symbols," which plays the role of ambient set. Generically, however, if no interaction is pre-specified between two sets J, I, we cannot even think about what it would mean to form the classic union, since the elements of each exist beyond comparison with each other.

Now conceive of J, I as two indexing sets with indexed tuples $(x_j)_{j\in J}$ and $(x_j)_{j\in I}$ in X. These can be combined into the indexed tuple $(x_j)_{j\in J+I}$ with indexing set

$J+I$. Applying this situation to our operators yields the following identity.

$$\bigodot_{j\in J+I} x_i = \left[\bigodot_{j\in J} x_i\right] \odot \left[\bigodot_{j\in I} x_i\right]$$

Note that, in the situation that I is the empty set $\varnothing$, $J+I$ is essentially just J. This reduces the above to:

$$\bigodot_{j\in J} x_i = \left[\bigodot_{j\in J} x_i\right] \odot \left[\bigodot_{j\in \varnothing} x_i\right]$$

When J is a mere singleton j and we denote x_j simply as x, we have the following identity.

$$x = x \odot \left[\bigodot_{j\in \varnothing} x_i\right]$$

Note that, by commutativity of $\odot$, this also implies the other order. However, even when $\odot$ is not commutative, the evaluation of the nullary operation still commutes with any other element. This is true since $J+\varnothing$ and $\varnothing+J$ are both equivalent to J. Therefore we *define* the nullary $\odot$ operation to be the constant map that picks out the inert or *unit* element ϵ of $\odot$. More formally, we say that $\odot$ satisfies *unitality* with *unit* ϵ when for any $x \in X$:

$$\epsilon \odot x = x = x \odot \epsilon.$$

Let's apply this principle to the operations $+$ and $\times$, whose n-ary extensions are denoted with the symbols $\sum$ and $\prod$, respectively.

$$\sum_{j\in\varnothing} x_j = 0 \qquad \prod_{j\in\varnothing} x_j = 1.$$

In the special case that $x_j = j$, the latter reduces to the fact that $0! = 1$.

We now return to the context of general maps $f : X \to Y$. Given a subset $A \subseteq Y$, we define its *preimage* under f as

$$f^*(A) = \{x \in X \mid fx \in A\}.$$

We would like to conceive of f^* as a map, but it takes arguments and values in subsets. To make f^* a map we must define the *powerset* of a set X as

$$\mathcal{P}X = \{A \subseteq X\}.$$

We then conceive of f^* as a map of type $\mathcal{P}Y \to \mathcal{P}X$. For any set Y, there is a map $\iota_Y : Y \to \mathcal{P}Y :: y \mapsto \{y\}$. Then, given a map $F : \mathcal{P}Y \to \mathcal{P}X$, we can take the composition $\hat{F} = i_Y \mathbin{/\!\!/} F : Y \to \mathcal{P}X$ to yield the restriction of F from subsets to elements. In turn, given a map $f : Y \to \mathcal{P}X$, we can uniquely recover a map $\overline{f} : \mathcal{P}Y \to \mathcal{P}X$ by taking the union of the evaluations of f on every element of its argument. More precisely:

$$\overline{f} : \mathcal{P}Y \to \mathcal{P}X :: A \mapsto \cup_{x\in A} fx$$

Given this correspondence, we will consider f^* as having either type $Y \to \mathcal{P}X$ or type $\mathcal{P}Y \to \mathcal{P}X$, depending on which suits the context. There is more to say about this, but it will have to wait until we are equipped with more linguistic technology.

We use the preimage $f^* : Y \to \mathcal{P}X$ to define the *jectivity* properties of f; in particular, we say that f is

- *surjective* if for all $y \in Y$, $f^*(y)$ has *at least* 1 element
- *injective* if for all $y \in Y$, $f^*(y)$ has *at most* 1 element

These properties have an interpretation in the context of equations. An *equation*

$$f(x) = y$$

is merely a prompt, given a map f and a value $y \in Y$, to compute the preimage $f^*(y)$. We say that f has the

- *existence property* if for all $y \in Y$, there exists a solution to $f(x) = y$
- *uniqueness property* if for all $y \in Y$, any solution to $f(x) = y$ is unique

To write expressions for solutions to equations $[g \circ f](x) = z$ that involve map compositions $X \xrightarrow{f} Y \xrightarrow{g} Z$, we simply apply iterated preimages:

$$\mathcal{P}Z \xrightarrow{g^*} \mathcal{P}Y \xrightarrow{f^*} \mathcal{P}X$$

As an example, consider the equation $\tan^2(x) = 3$, which can be rewritten as

$$[\square^2 \circ \tan](x) = 3.$$

Before solving our equation, we compute the relevant reduced preimages

$$\begin{aligned} \tan^* : \mathbb{R} \to \mathcal{P}\mathbb{R} :: x \mapsto \{\arctan x + 2\pi n \mid n \in \mathbb{Z}\} &=: \arctan x + 2\pi\mathbb{Z} \\ [\square^2]^* : \mathbb{R} \to \mathcal{P}\mathbb{R} :: x \mapsto \{\sqrt{x}, -\sqrt{x}\} &=: \pm\sqrt{x}. \end{aligned}$$

We now compute our solution set A:

$$\begin{aligned} A &= [(\tan)^* \circ (\square^2)^*](3) \\ &= (\tan)^*(\pm\sqrt{3}) \\ &= \arctan(\pm\sqrt{3}) + 2\pi\mathbb{Z} \\ &= \pm\frac{\pi}{3} + 2\pi\mathbb{Z} \end{aligned}$$

Applying this to the context of a quadratic map q, its roots can be expressed as

$$[L_h^* \circ [\square^2]^* \circ L_v^*](0).$$

We say that f is *bijective* when it is both surjective and injective. In this case, for each $y \in Y$, there is always precisely one element $x \in X$ such that $fx = y$. This allows us to construct a map $f^{-1} : Y \to X$ sending y to this unique element. In other terms, $f^*(y)$ is a singleton $\{x\}$ and, since it is precisely this sole element x to which y gets sent, we write $f^{-1} = x$. We say the map f^{-1} is the *inverse* map to f. Given a map $f : X \to Y$, its inverse map $g : Y \to X$ is a map such that

$$\begin{aligned} g \circ f &= \mathbb{1}_X \\ f \circ g &= \mathbb{1}_Y. \end{aligned}$$

We say $f : X \to Y$ is *invertible*, or an *isomorphism* when it has an inverse map. We then write $f : X \xrightarrow{\cong} Y$ or, when we wish to speak directly about the sets in consideration, we write $X \cong Y$ and say that "X is isomorphic to Y." This language allows us to formalize our remark about singleton sets: $\star \cong \mathbf{1}$; or in general any pair of sets of the same cardinality, e.g. $\mathbb{B} \cong \mathbf{2}$.

Just as map composition generalizes numeric addition and multiplication, map inverse generalizes additive and multiplicative inverse.

$$(\square + n) \circ (\square + [-n]) = \square + 0$$
$$(\square \cdot n) \circ (\square \cdot n^{-1}) = \square \cdot 1$$

In other words: $(\square{+}n)^{-1} = (\square{+}[-n])$ and $(\square{\cdot}n)^{-1} = (\square{\cdot}n^{-1})$. This also sheds light on the adage "you can't divide by 0"—the map $(\square \cdot 0)$ amounts to the rule $x \mapsto 0$, which is surely neither injective nor surjective, and hence non-invertible.

This language inspires a reprisal of the fundamental theorems of calculus. We encode the derivative as $D : \mathcal{C}^\infty \to \mathcal{C}^\infty :: f \mapsto f'$, the integral as the map I in Equation (1), and the needed shifting adjustment as

$$\sigma : \mathcal{C}^\infty \to \mathcal{C}^\infty :: f \mapsto \lambda x.[f(x) - f(0)].$$

Then the first and second fundamental theorems can be rewritten as follows.

$$I \circ D = \sigma$$
$$D \circ I = \mathbb{1}_{\mathcal{C}^\infty}$$

These equations encode precisely how the derivative and integral are "inverse" to each other: I is a right inverse but not a left inverse to D—t is however rather close to being a left inverse since it is only off by a shift. This shift is of course the consequence of the arbitrary constant that gets lost in the process of differentiating and hence, instead of being recovered, is replaced by a default constant $f(0)$.

The preimage notion also facilitates a more algebraic repackaging of set-builder notation. We can consider a condition P on X as a map $P : X \to \mathbb{B}$, in which case we should write $P(x) = \top$ in place of just $P(x)$ for "x satisfies P." In turn, this added detail allows us to encode the set as a preimage of $\top$ under P:

$$P^*(\top) = \{x \in X \mid P(x) = \top\}.$$

Aside from aesthetics, why would one care to do this? This notation throws in relief the correspondence between logical and set theoretic constructions. We list some examples in the following table, whose first two columns correspond to the informal and symbolic logical expressions.

P implies Q	$P \Rightarrow Q$	$P^*(\top) \subseteq Q^*(\top)$
P if and only if Q	$P \Leftrightarrow Q$	$P^*(\top) = Q^*(\top)$
P or Q	$P \vee Q$	$P^*(\top) \cup Q^*(\top)$
P and Q	$P \wedge Q$	$P^*(\top) \cap Q^*(\top)$

Making these relations explicit in the notation can allow for more succinct proofs via immediate symbolic reasoning. I'm of the belief that a goal of mathematical notation should be the excising of indirect, seemingly necessary, verbal fillers like "such that" from one's inner monologue.

We end the section by noting an important isomorphism.

Proposition 0.1. *The following is an isomorphism*

$$\lambda\varphi.\varphi^*(\top) : [X \to \mathbb{B}] \to \mathcal{P}X.$$

Proof. The inverse of this map is given by the map $\top_{(-)} : \mathcal{P}X \to [X \to \mathbb{B}]$, which maps a subset $S \subseteq X$ to the following so called *indicator* map,

$$\top_S : X \to \mathbb{B} :: x \mapsto \begin{cases} \top & x \in S \\ \bot & x \notin S \end{cases}$$

By construction $[\top_S]^*(\top) = S$ and $\top_{\varphi^*(\top)} = \varphi$. $\square$