We assume some amount of set theory.

ordered tuples and finite cartesian product

The definitions given here are not good for intuition. The main purpose is to provide (robust?) set-theoretic *implementations* of familiar algebraic constructions. This is a key distinction to be made—I am essentially assuming the reader is familiar with most if not all of the concepts and is able to recognize that the definitions (stroke implementations) adequately capture the necessary functionality.

A mathematical concept in the mind is characterized by some *information*—that being the information necessary to understand the concept. In my usage "information" does not exist in the mathematical world, but rather it exists in each individual mathematicians mind. We hope that this information generally agrees across minds, and in such cases we consider the information characterizing a mathematical object to be the information which is common in the minds of the mathematical community at large. It is this standard of information to which we will hold definitions.

In order to capture the information of a mathematical concept, we will use *data*—something which does exist in the mathematical world. Again, I have my own usage of the word "data," which is luckily more concrete than that of "information." My usage will hopefully become more clear as you read on, but suffice it to say that data is a specific mathematical construction intended to represent the information of some mathematical concept.

To be a bit more opaque, I will employ an analogy and some precise yet non-mathematical terminology. We can consider mathematical concepts like fictional characters. When we gives names to these concepts, there is no actual referent (though Salmon and I would prefer designatum, here) to which the name applies (refers/designates you get the idea). (Unfortunately, my choice of the word "concept" for a mathematical audience now conflicts with the point I am making because we can of course refer to concepts, as I have been doing.) If you're willing to take my word for it—which you absolutely should not—the sort thing I have been calling a "concept" is better described in other terms, namely "sense." For normal sorts of things, you can think of the sense of a name as the tether between the name itself and the referent. The name (word) "goose" has some sense attached to it which allows us to pick out the actual animal, i.e., the goose. One can think of the sense of the word "goose" as that thing which allows you to read and understand that it refers to the goose.

Anyway, definitions are gonna look like this:

A **whimple** W is given by the following data:

(1) an aggle-poogle A,
(2) a buzzo B with impermeable flingos,
(3) a crundo C (called the *crun* inside W),

such that

(i) A and B are equivalent as aggle-poogles,
(ii) each dellon of C is also a dellon of A.

Say W is a whimple or $W = (A, B, C)$ is a whimple.

In other words, definitions of mathematical objects will be given as set-theoretic implementations in terms of collections of data satisfying certain axioms. Then to name such an object, we write it as the ordered tuple of data.

If A and B are sets, a subset $F \subseteq A \times B$ is called **functional** if for each $a \in A$ there is exactly one $b \in B$ such that $(a, b) \in F$.

A **function** f is given by the following data:

(1) a set A called the *domain* of f (denoted "$\operatorname{dom} f$"),
(2) a set B called the *codomain* of f (denoted "$\operatorname{codom} f$"),
(3) a functional subset $F \subseteq A \times B$ called the *graph* of f.

We say that f is a function *from A to B*, denoted $f : A \to B$ or $A \xrightarrow{f} B$.

Given $a \in A$, the unique element $b \in B$ such that $(a, b) \in F$ is called the **value** of f at a, often denoted by $f(a)$, though sometimes denoted by fa or f_a. This is can also be called the evaluation of f at a or the image of a under f.

This definition adapted from [Ronald Brown, TOPOLOGY AND GROUPOIDS: A Geometric Account of General Topology, Homotopy Types and the Fundamental Groupoid]

We make a brief distinction here between our specific *definition* of the word "function" and the (more general) conceptual *notion* of a function. It is impossible to say in words precisely what the latter is, but we can make reference to it by the role it serves in the mathematical world. At one point or another—hopefully rather early in their career—a mathematician will acquire the correct notion of a function. The sense in which the notion is *correct* is that it is consistent with the notions of the vast majority of mathematicians.

We might ask the mathematician, "What is a function?" To which they might respond, "A function from A to B is a rule which assigns, to each element of A, an element of B." You—the reader—can now act as an impartial third party to me—the author—and this other mathematician I have manufactured. With any luck, you are able to read and understand both this definition and my own definition of the word "function." Moreover, I hope you agree that they both refer to the same *thing*—that being the notion of a function.

The goal of the above definition is that any sufficiently experienced mathematician will agree that I am using the word "function" consistently with broader mathematics, even if some of the specifics are different. In other words, my definition is attempting to capture the notion of a function by the set-theoretic data which characterizes it.

To be explicit, we are not defining functions per se, rather we are setting up a correspondence between functions $f : A \to B$ and triples of data (A, B, F) satisfying our definition of function.

In a completely set theoretic construction, we would identify every function f with its corresponding triple of data, i.e., $f = (A, B, F)$. In which case, we are essentially claiming that there is a bijection between functions in the usual sense and their corresponding triples.

$g = (C, D, G)$ is a

The justification for this sort of definition is that it ensures two functions $f : A \to B$ and $f' : A' \to B'$ are equal

---

The domain of f can be recovered from its graph F as follows:

$$\operatorname{dom} f = \{a \mid (a, b) \in F \text{ for some } b\} = A.$$

On the other hand, the codomain of f cannot be recovered in this way. Instead, we find the following subset of the codomain called the **image** of f:

$$\operatorname{im} f := \{b \mid (a, b) \in F \text{ for some } a\} \subseteq B.$$

The domain of f and the evaluation can also be used to find the image as follows:

$$\operatorname{im} f = \{f(a) \mid a \in A\} \subseteq B.$$

Lastly, the graph of f can be recovered from the domain and the evaluations as follows:

$$\Gamma_f := \{(a, f(a)) \mid a \in A\} = F.$$

If desired, B^A is used to denoted the set of all functions from A to B.

Let S be a set. We define the following functions:

the **identity** function $\mathrm{id}_S : S \to S$ where $a \mapsto a$,

the **diagonal** function $\Delta_S : S \to S \times S$ where $a \mapsto (a, a)$,

the unnamed (component interchange) function $\lambda_S : S \times S \to S \times S$ where $(a, b) \mapsto (b, a)$.

Let S be a (nonempty?) set and n be a nonnegative integer.

The **nth cartesian power** of S is the set of all n-tuples of elements of S, denoted

$$S^n = \underbrace{S \times S \times \cdots \times S}_{n \text{ times}} = \{(a_1, \ldots, a_n) \mid a_i \in S\}.$$

By convention, we take $S^0 = \{()\} = \{\varnothing\}$ (the set containing only the empty tuple/set).

An **n-ary operation** on S is a function $S^n \to S$.

Notably, a 0-ary (nullary) operation is a function $f : \{\varnothing\} \to S$. In other words, f picks out a single element of S, namely $f(\varnothing) \in S$. In fact, there is a bijective correspondence between the elements of S and the nullary operations on S, given by

$$\begin{aligned} S &\longleftrightarrow \{\text{nullary operations on } S\}, \\ a &\longleftrightarrow (\varnothing \mapsto a). \end{aligned}$$

Under this correspondence, we can consider distinguished elements of a set to be nullary operations, and vice versa. This is not a super important point, but simply makes the following definitions more concise, maybe. Is it worth it? Who is to say?

We want more than static sets—we want to be able to do stuff with the sets. Similar to our definition of "function," we want to define the term "algebraic structure." Put in an unhelpfully circular way, an algebraic structure is the sort of mathematical construction inside of which we can perform algebraic activities. In the fundamental cases, it consists of a set and some rules for how we can manipulate the elements of the set, e.g., operations. This paragraph is devoid of meaning. I am going to provide a set-theoretic implementation now.

An **algebraic structure** A (simple/on a set?) is given by the following data:

1. a set S, called the *underlying set* of A (denoted "$|A|$"),
2. some operations $\alpha, \beta, \ldots$ of any arity on S.

We say $A = (S, \alpha, \beta, \ldots)$ is an algebraic structure.

There will usually only be finitely many operations on S and in any case there isn't, we will use more compact notation.

A **magma** M is given by the following data:

- a set S,
- a binary operation $f : S \times S \to S$.

We say $M = (S, f)$ is a magma.

A magma M is (naturally) an algebraic structure (S, f).

It is common convention to notate the binary operation with a symbol called a (binary) *operator*—common examples include the following:

$$+ \quad \times \quad \cdot \quad * \quad \star \quad \circ$$

This is of course not a comprehensive catalog, though it covers most of the basic cases. If we choose the star '$\star$' to represent our binary operation, we write '$a \star b$' to mean the image in S under f of the pair $(a, b) \in S \times S$. In other words, f describes a rule which takes two elements $a, b \in A$ and produces a third element of S, denoted

$$a \star b = f(a, b).$$

This new element $a \star b \in S$ might be called many different things depending on the actual context.

We will say things like "S is a set with a binary operation $\star$" or "$\star$ is a binary operation on the set S." For brevity, we will also write "$(S, \star)$ is a magma" or possibly "(S, f) is a magma" if we want to emphasize the fact that the binary operation is a function $S \times S \to S$.

In the broader mathematical world, it is common to consider only a single binary operation on a set at one time—there are many sets with 'canonical' operations—which can lead to the set alone being taken as proxy for the magma. For example, to refer to what we call "the magma $(S, \star)$," one might instead say "the set S *equipped with* a binary operation $\star$." The difference here is subtle and a passing glance is not likely to reveal a discrepancy beyond the obvious linguistic one. And indeed there is no difference to one interfacing with mathematics in its most pure conceptual form. But you and I—dear reader—are confined to the relentlessly imperfect and finite bounds of language. When we do mathematics, we must undertake the impossible challenge of justifying that the sense connecting our mathematical referents to our mathematical language is justified, but can only do so either in the very same mathematical language or in some natural language. (The latter is what a mathematician might call 'intuition' (for the more cultured 'being loosey goosey' or 'feeding the geese')).

I trailed off, but the minor point is that it is unreasonable to be fully explicit all of the time. And while there are sometimes good reasons to 'compress' mathematical data into containers not designed for it, there should be a way to unpack that data when necessary (in my opinion of course). I would also say that neither formality nor explication exist solely on single axis of 'less' or 'more,' where one must simply pick a single point along that axis. A thinner dictionary can be seen as either 'clean and elegant' or 'intuitive and superficial,' while a thicker one as either 'rigorous and complete' or 'cluttered and illegible' (find more creative or meaningful adjectives). A proof on either end of this can also be more or less transparent/opaque/illuminating/aesthetic/motivating. (Hot take things have many aspects and rarely does a single aspect determine its quality.)

We are rarely interested in a general binary operation and will most often require it to have some additional structure. In such cases, however, we will need to define a particular binary operation and prove that it has the desired structure. For this reason—I would argue—it is worth having the language to talk about general binary operations.

(There is a diversion here where I lament about having to prove certain elementary properties of objects without having the proper language to even talk about the way in which those objects possess those properties. To make matters worse, it is often the case that the desired result is essentially some form of "niceness" in the sense that we are showing that some hypothetical bad situation never occurs. In which case, when we are using the object for its intended purpose, we can sweep certain technical nuances under the rug. So because the goal is to be able to ignore the nuance, one has to synthesize a complete model of this nuance only to discard it the moment it reveals its own unimportance.)

An **algebraic structure** A is given by (some or all of) the following data:

(1) an *underlying* algebraic structure, denoted explicitly by $\underline{A}$ unless otherwise specified,
(2) some *auxiliary* algebraic structures/objects,
(3) some homomorphisms between these algebraic structures.

An algebraic structure **class/type** is a *signature* of data for constructing an algebraic structure. The signature prescribes the following: whether or not and from which category an underlying object is required, whether or not and from which category any auxiliary objects are required, whether or not and from which category any morphisms are required. Additionally, the signature prescribes some *coherence conditions* which must be satisfied by the supplied data.

In other words, an algebraic structure class/type describes a format/template for constructing a more narrow sort of object. Any algebraic object which conforms to the template, is said to be of the class/type in question.

Whenever an algebraic object A of type $\mathcal{A}$ has an underlying structure $\underline{A}$, then we very closely identify

hey bud these are just categories. just use categories

A **magma** M is a type of algebraic structure given by the following data:

(1) an underlying set $\underline{M}$,
(2) a binary operation on $\underline{M}$ (a function $\underline{M} \times \underline{M} \to \underline{M}$).

Let $M = (\underline{M}, \odot)$ and $N = (\underline{N}, \otimes)$ be magmas.

In order to notate an expression containing multiple applications of the operation, we use parenthesis, e.g.,

$$a \star (b \star c) = f(a, f(b, c)) \quad \text{and} \quad (a \star b) \star c = f(f(a, b), c).$$

In general, it is not well-defined to write an expression like '$a \star b \star c$,' as it is possible that the result is dependent on the order in which we apply the operation. In cases where the this order does not matter, we can make sense of such notation.

A binary operation $\star$ on a set S is called **associative** if for all $a, b, c \in S$ we have

$$(a \star b) \star c = a \star (b \star c)$$

A magma $(S, \star)$

- (ass) is **associative** if $(a \star b) \star c = a \star (b \star c)$ for all $a, b, c \in S$,
- (id) has **identity** if there exists $e \in S$ such that $e \star a = a \star e = a$ for all $a \in S$,
- (inv) has **inverses** if for every $a \in S$ there exists $b \in S$ such that $ab = ba = e$,
- (comm) is **commutative** if $a \star b = b \star a$ for all $a, b \in S$.

A magma $(S, \star)$ satisfying certain properties typically has a more specific name. We say $(S, \star)$ is a

- **semigroup** if (ass) it is associative,
- **monoid** if (ass, id) it is associative and has an identity,
- **group** if (ass, id, inv) it is associative, has an identity, and has inverses,

Most of these will simply take "commutative" as an adjective, though we usually say **abelian group** to mean a commutative group, i.e.,

- **abelian group** if (ass, id, inv, comm) it is associative, has an identity, has inverses, and is commutative.

Chart:

	ass	id	inv	comm
magma	-	-	-	-
semigroup	x	-	-	-
monoid	x	x	-	-
comm. monoid	x	x	-	x
group	x	x	x	-
abelian group	x	x	x	x

semigroup bad, just add identity to get monoid.

"you can define lots of things, but that doesn't mean you should study them." - Dave Morrison

---

A **monoid** is given by the following data:

(1) s set S,
(2) a distinguished element $e \in S$,

(3) a binary operation $\star$ on S,

such that

(i) (ass) $(a \star b) \star c = a \star (b \star c)$ for all $a, b, c \in S$,
(ii) (id) $e \star a = a \star e = a$ for all $a \in S$.

In which case, we say $(S, e, \star)$ is a **monoid**.

Let $(S, e, \cdot)$ be a monoid. (For simplicity, write $ab = a \cdot b$ for all $a, b \in S$.)

Given $a, b \in S$ such that $ab = e$, we say that a is a **left inverse** of b and b is a **right inverse** of a. If in addition $ba = e$, then we say that a is an **inverse** of b, and vice versa. Equivalently, we might say

- a and b are inverses (of each other),
- a is inverse to b,
- a and b are inverse (to one another).

Additionally, we say that a and b are **invertible** in the monoid.

(One must be careful when using the word "invertible," as it is relative to both the underlying set and the operation—e.g., every element of $\mathbb{Z}$ is invertible with respect to addition, though only 1 and -1 are invertible with respect to multiplication.)

Note a and b are inverses if and only if one is both a left and right inverse of the other.

In general mathematics, it is more common to hear the phrase "a is *the* inverse of b" rather than "a is *an* inverse of b." Indeed, in most nice cases, there is only one inverse. In turns out that this uniqueness holds in any monoid, as we will now demonstrate.

Proposition 1. If $a, b, c \in S$ are such that $ab = ca = e$, then $b = c$.

Proof. $b \overset{\text{(id)}}{=} eb = (ca)b \overset{\text{(ass)}}{=} c(ab) = ce \overset{\text{(id)}}{=} c.$ □

In particular, the common value of b and c is an inverse of a. It follows that an element of a monoid is invertible whenever it has both a left and a right inverse—the proposition implies that the two are equal.

Corollary 1. Inverses are unique.

Proof. If b and c are both inverses of a, the proposition implies $b = c$. □

Hence, every invertible element of the monoid in fact has a unique inverse. If $a \in S$ is invertible, then we denote **the inverse** of a might be denoted by one of the following

$$-a \qquad a^{-1} \qquad \frac{1}{a} \qquad \overline{a} \qquad a'$$

Let $(G, e, \cdot)$ be a group, i.e., a monoid with all elements invertible. Since the inverses in a monoid are unique, so too are the inverses in a group. And since every element has an inverse, there is a well-defined function

$$\begin{aligned} \iota : G &\longrightarrow G, \\ g &\longmapsto g^{-1}, \end{aligned}$$

where g^{-1} is *the* inverse of g in the group. This function has the following properties:

(i) $\iota \circ \iota = \mathrm{id}_G$
(ii) $\iota(g \cdot h) = \iota(h) \cdot \iota(g)$, i.e., $(gh)^{-1} = h^{-1}g^{-1}$

Let $G = (S, e, \cdot)$ be a group. We can define a new binary operation $\odot$ on S as follows:

$$g \odot h := h \cdot g \quad \text{or} \quad \mu_\odot = \mu_\cdot \circ \lambda_G.$$

Denote the algebraic structure $G^{\mathrm{op}} = (S, e, \odot)$. We claim G^{op} is a group. Indeed $(S, \odot)$ is a magma and $e \in S$ is a distinguished element, so we have the required data. (unless i later require the inversion map for groups) We check the axioms now:

- (ass) $(g \odot h) \odot k = k \cdot (h \cdot g) = (k \cdot h) \cdot g = g \odot (h \odot k)$,
- (id) $g \odot e = e \cdot g = g = g \cdot e = e \odot g$,
- (inv) $g \odot g^{-1} = g^{-1} \cdot g = e = g \cdot g^{-1} = g^{-1} \odot g = e$, where g^{-1} is the inverse of g in G.

Hence, G^{op} is a group, called the *opposite group* of G.

If $\iota : S \to S$ is the inversion function of G, then we have

$$\iota(g \cdot h) = \iota(h) \cdot \iota(g) = \iota(g) \odot \iota(h).$$

That is, ι specifies a group homomorphism (what's that bud?) $G \to G^{\mathrm{op}}$. In fact, this is an isomorphism and $(G^{\mathrm{op}})^{\mathrm{op}} = G$.

A **ring** R is an algebraic structure with the following data:

(1) distinguished elements $0, 1 \in |R|$,
(2) a binary operation $+$ called *addition* such that $(|R|, 0, +)$ is an abelian group,
(3) a binary operation $\cdot$ called *multiplication* such that $(|R|, 1, \cdot)$ is a monoid,

such that the two operations satisfy the following distributivity properties:

(i) (dist) $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$,
(ii) (dist) $(a + b) \cdot c = (a \cdot c) + (b \cdot c)$.

By convention, the *order of operations* in a ring puts multiplication before addition. Because of this, we can omit parentheses the order of operations can be deduced. Moreover, we typically write the multiplication without the operator, though sometimes use it for clarity. On the other hand, we always write the operator for addition. For example, we can write the following

$$(a \cdot b) + (c \cdot d) = ab + cd.$$

The additive inverse of $a \in R$ is denoted by $-a$, and we replace the plus sign with a minus sign when adding an inverse, e.g.,

$$a + (-b) = a - b.$$

The multiplicative inverse of $a \in R$, when it exists, is denoted by a^{-1}.

(i) $\mu(a, \alpha(b, c)) = \alpha(\mu(a, b), \mu(a, c))$, i.e.,

$$\begin{tikzcd}
R \times R \times R \arrow[r, "\Delta_R \times \mathrm{id}_R \times \mathrm{id}_R"] \arrow[d, "\mathrm{id}_R \times \alpha"] & R \times R \times R \times R \arrow[r, "\mathrm{id}_R \times \lambda_R \times \mathrm{id}_R"] & R \times R \times R \times R \arrow[d, "\mu \times \mu"] \\
R \times R \arrow[r, "\mu"] & R & R \times R \arrow[l, "\alpha"']
\end{tikzcd}$$

$$\begin{tikzcd}
R^3 \arrow[r, "\Delta \times \mathrm{id} \times \mathrm{id}"] \arrow[d, "\mathrm{id} \times \alpha"'] & R^4 \arrow[r, "\mathrm{id} \times \lambda \times \mathrm{id}"] & R^4 \arrow[d, "\mu \times \mu"] \\
R^2 \arrow[r, "\mu"'] & R & R^2 \arrow[l, "\alpha"]
\end{tikzcd}$$

(ii) $\mu(\alpha(a, b), c) = \alpha(\mu(a, c), \mu(b, c))$, i.e.,

$$\begin{tikzcd}
R^3 \arrow[r, "\mathrm{id} \times \mathrm{id} \times \Delta"] \arrow[d, "\alpha \times \mathrm{id}"'] & R^4 \arrow[r, "\mathrm{id} \times \lambda \times \mathrm{id}"] & R^4 \arrow[d, "\mu \times \mu"] \\
R^2 \arrow[r, "\mu"'] & R & R^2 \arrow[l, "\alpha"]
\end{tikzcd}$$

A **ring** is given by the following data:

(1) a set R,
(2) a binary operation $+$ on R called *addition*,
(3) a binary operation $\cdot$ on R called *multiplication*,

such that

(i) R is an abelian group under addition:

  (a) (ass) $(a+b)+c = a+(b+c)$,

  (b) (id) $0 \in R$ called *zero* such that $0+a = a+0 = a$,

  (c) (inv) for all $a \in R$ there exists an $-a \in R$ such that $a + (-a) = 0$,

(ii) R is a monoid under multiplication, whose identity is called the *one* denoted $1 \in R$.

Proposition 2. In a ring R we have the following properties:

(a) $0 \cdot a = a \cdot 0 = 0$,

(b) $(-1) \cdot a = a \cdot (-1) = -a$,

(c) $(-a) \cdot b = a \cdot (-b) = -(a \cdot b)$,

(d) $(-1) \cdot (-1) = 1$,

(e) $(-a) \cdot (-b) = ab$.

Proof. beebop

(a) $0a \overset{(+\text{id})}{=} (0+0)a \overset{(\text{dist})}{=} 0a + 0a$ implies $0 = 0a$ by cancellation.
$a0 \overset{(+\text{id})}{=} a(0+0) \overset{(\text{dist})}{=} a0 + a0$ implies $0 = a0$ by cancellation.

(b) $a + (-1)a \overset{(\cdot\text{id})}{=} 1a + (-1)a \overset{(\text{dist})}{=} (1-1)a \overset{(+\text{inv})}{=} 0a \overset{(\text{a})}{=} 0$ implies $(-1)a = -a$ by additive inverse uniqueness.
$a + a(-1) \overset{(\cdot\text{id})}{=} a1 + a(-1) \overset{(\text{dist})}{=} a(1-1) \overset{(+\text{inv})}{=} a0 \overset{(\text{a})}{=} 0$ implies $a(-1) = -a$ by additive inverse uniqueness.

(c) $-(ab) \overset{(\text{b})}{=} (-1)ab \overset{(\text{b})}{=} (-a)b \overset{(\text{b})}{=} a(-1)b \overset{(\text{b})}{=} a(-b)$.

(d) $(-1)(-1) \overset{(\text{b})}{=} -(-1)$ but $-(-1) = 1$ by additive inverse uniqueness.

(e) $(-a)(-b) \overset{(\text{b})}{=} a(-1)(-1)b \overset{(\text{d})}{=} a1b \overset{(\cdot\text{id})}{=} ab$.

□

A module is given by the following data:

(1) a ring R,

(2) an abelian group M,

(3) a function $\mu : R \times M \to M$ such that

  (i) $\mu(1, x) = x$ or equivalently $\mu(1, -) = \mu|_{1 \times M} = \mathrm{id}_M$

  (ii) $\mu(ab, x) = \mu(a, \mu(b, x))$ or equivalently the following diagram commutes

$$\begin{tikzcd}
R \times R \times M \arrow[r, "\mathrm{id}_R \times \mu"] \arrow[d, "\mu_R \times \mathrm{id}_M"'] & R \times M \arrow[d, "\mu"] \\
R \times M \arrow[r, "\mu"'] & M
\end{tikzcd}$$

Let R be a ring.

An R-**module** is given by the following data:

(1) an abelian group M written additively, i.e., identity is 0 and binary operation is $+$,
(2) a ring homomorphism $\mu : R \to \mathrm{End}(M)$, called *scalar multiplication*.

We usually write $r \cdot m$ for the evaluation $\mu(r)(m)$.