Chapter 1

Some Elementary Informal Set Theory

Set theory is due to Georg Cantor. “Elementary” in the title above does not apply to the body of his work, since he went into considerable technical depth in this, his new theory. It applies however to *our* coverage as we are going to restrict ourselves to elementary topics only.

Cantor made many technical mistakes in the process of developing set theory, some of considerable consequence. The next section is about the easiest and most fundamental of his mistakes.

How come he made mistakes? The reason is that his theory was not based on axioms and rigid rules of reasoning —a state of affairs for a theory that we loosely characterise as “informal”.

At the opposite end of informal we have the *formal* theories that are based on axioms *and* logic and are thus “safer” to develop (they do not lead to *obvious* contradictions).

One *cannot* fault Cantor for not using logic in arguing his theorems —that process was not invented when he built his theory— but then, *a fortiori*, mathematical logic was not invented in Euclid’s time either, *and yet* he did use axioms that stated how his building blocks, *points*, *lines* and *planes* interacted and behaved!

Guess what: Euclidean Geometry leads to no contradictions.

The problem with Cantor’s set theory is that anything goes as to what sets are and how they come about. He neglected to ask the most fundamental question: “How are sets formed?”[†] He just sidestepped this and simply said that a *set* is any collection. In fact he took the term “set” as just a synonym for “collection”, “class”, “aggregate”, etc.

[†] It’s amazing how much trouble could be avoided if he had done so!

Failure to ask and answer this question leads to “trouble”, which is the subject matter of the next section.

One can still do “safe” set theory —devoid of “trouble”, that is— within an *informal* (non axiomatic) setting, but we have to ask and answer how sets are built *first* and derive from our answer some *principles* that will guide (and protect!) the theory’s development! We will do so.

1.1. Russell’s “Paradox”

Cantor’s *naïve* (this adjective is not derogatory but is synonymous in the literature with *informal* and *non axiomatic*) set theory was plagued by *paradoxes*, the most famous of which (and the *least* “technical”) being pointed out by Bertrand Russell and thus nicknamed “Russell’s paradox”.[†]

His theory is the theory of collections (i.e., sets) of objects, as we mentioned above, terms that were neither defined nor how they were built.[‡]

This theory studies operations on sets, properties of sets, and aims to use set theory as the foundation *of all mathematics*. Naturally, mathematicians “do” set theory of *mathematical object collections* —not collections of birds and other beasts. We have learnt some elementary aspects of set theory at high school. We will learn more in this course.

1. **Variables**. Like any theory, informal or not, informal set theory —a safe variety of which we will develop here— uses *variables* just as algebra does. There is only *one type* of variable that varies over set and over atomic objects too, the latter being objects that have no set structure. For example integers. We use the names $A, B, C, \ldots$ and $a, b, c, \ldots$ for such variables, sometimes with primes (e.g., A'') or subscripts (e.g., x_{23}), or both (e.g., x'''_{22}, Y'_{42}).

2. **Notation**. *Sets given by listing.* For example, $\{1, 2\}$ is a set that contains precisely the objects 1 and 2, while $\{1, \{5, 6\}\}$ is a set that contains precisely the objects 1 and $\{5, 6\}$. The braces $\{$ and $\}$ are used to show the collection/set by outright listing.

3. **Notation**. *Sets given by “defining property”*. But what if we cannot (or will not) explicitly list all the members of a set? Then we may define

[†] From the Greek word “paradoxo” (παράδοξο) meaning against one’s belief or knowledge; a contradiction.

[‡] This is not a problem *in itself.* Euclid too did not say *what* points and lines *were*; but his axioms did characterise their nature and interrelationships: For example, he started from these (among a few others) *a priori truths* (axioms): *a unique line passes through two distinct points*; also, *on any plane, a unique line l can be drawn parallel to another line k on the plane if we want l to pass through a given point A that is not on k.*

The point is:

You cannot leave out *both* what the nature of your objects is and *how* they behave/interrelate and get away with it! Euclid omitted the former but provided the latter, so all worked out.

what objects x get in the set/collection by having them to *pass an entrance requirement*, $P(x)$:

> **An object x gets in the set *iff* (*if and only if*) $P(x)$ is true of said object.**

Let us parse "iff":

(a) The *IF*: So, IF $P(x)$ is true, then x gets in the set (it passed the "admission requirement").

(b) The *ONLY IF*: So, IF x gets in the set, then the ***only** way for this to happen* is for it to pass the "admission requirement"; that is, $P(x)$ is true.

In other words, "iff" (as we probably learnt in high school or some previous university course such as calculus) is the same thing as "is equivalent":

"x is in the set" is equivalent to "$P(x)$ is true".

We denote the collection/set[†] defined by the entrance condition $P(x)$ by

$$\{x : P(x)\} \tag{1}$$

but also as

$$\{x \,|\, P(x)\} \tag{1'}$$

reading it "the set of all x *such that* (this "such that" is the ":" or "|") $P(x)$ is true [or holds]"

4. "$x \in A$" is the assertion that "object x is in the set A". Of course, this assertion may be true or false or "it depends", just like the assertions of algebra $2 = 2$, $3 = 2$ and $x = y$ are so (respectively).

5. $x \notin A$ is the negation of the assertion $x \in A$.

6. **Properties**

   - Sets are *named* by letters of the Latin alphabet (cf. **Variables**, above). Naming is pervasive in mathematics as in, e.g., "let $x = 5$" in algebra.

     So we can write "let $A = \{1, 2\}$" and let "$c = \{1, \{5, 6\}\}$" to give the names A and c to the two example sets above, ostensibly because we are going to discuss these sets, and refer to them often, and it is cumbersome to keep writing things like $\{1, \{5, 6\}\}$. Names are *not permanent*;[‡] they are *local* to a discussion (argument).

[†] We have not yet reached Russell's result, so keeping an open mind and humouring Cantor we still allow ourselves to call said collection a "set".

[‡] OK, there *are* exceptions: $\emptyset$ is the permanent name for the *empty set* —the set with no elements at all— and for that set only; $\mathbb{N}$ is the permanent name of the set of all *natural numbers*.

- **Equality of sets** (repetition and permutation do not matter!)

  Two sets A and B are equal iff they have the same members. Thus order and multiplicity do not matter! E.g., $\{1\} = \{1, 1, 1\}$, $\{1, 2, 1\} = \{2, 1, 1, 1, 1, 2\}$.

- The fundamental equivalence pertaining to definition of sets by "defining property": So, if we name the set in (1) above, S, that is, if we say "let $S = \{x : P(x)\}$", then "$x \in S$ iff $P(x)$ is true"

  By the way, we almost *never say* "is true" unless we want to shout out this fact. We would say instead: "$x \in S$ iff $P(x)$".

  Equipped with the knowledge of the previous bullet, we see that the symbol $\{x : P(x)\}$ defines a *unique* set/collection: Well, say A and B are so defined, that is, $A = \{x : P(x)\}$ and $B = \{x : P(x)\}$. Thus

$$x \in A \overset{A=\{x:P(x)\}}{\text{iff}} P(x) \overset{B=\{x:P(x)\}}{\text{iff}} x \in B$$

  thus

$$x \in A \text{ iff } x \in B$$

  and thus $A = B$.

Let us pursue, as Russell did, the point made in the last bullet above. Take $P(x)$ to be specifically the assertion $x \notin x$. He then gave a name to

$$\{x : x \notin x\}$$

say, R. But then, by the last bullet above,

$$x \in R \text{ iff } x \notin x \tag{2}$$

If we now *believe*,[†] as *Cantor*, the father of set theory did not question and went ahead with it, that every $P(x)$ defines a *set*, *then R is a set.*

What is wrong with that?

Well, if R is a set then this object has the proper *type* to be plugged into the *variable of type "math object"*, namely, x, throughout the equivalence (2) above. But this yields the contradiction

$$R \in R \text{ iff } R \notin R \tag{3}$$

This contradiction is called the Russell's Paradox.

[†] Informal mathematics often relies on "I know so" or "I believe" or "it is 'obviously' true". Some people call "proofs" like this —i.e., "proofs" without justification(s)— "proofs by intimidation". Nowadays, with the ubiquitousness of the qualifier "fake", one could also call them "fake proofs".

This and similar paradoxes motivated mathematicians to develop formal symbolic logic and look to axiomatic set theory[†] as a means to avoid paradoxes like the above.

Other mathematicians who did not care to use mathematical logic and axiomatic theories found a way to do set theory *informally*, yet *safely*.

You see, they asked *and* answered "how are sets formed?"[‡]

Read on!

[†] There are many flavours or axiomatisations of set theory, the most frequently used being the "ZF" set theory, due to Zermelo and Fraenkel.

[‡] Actually, axiomatic set theory —in particular, its axioms are— is built upon the answers this group came up with. This story is told at an advanced level in [Tou03b].

Chapter 2

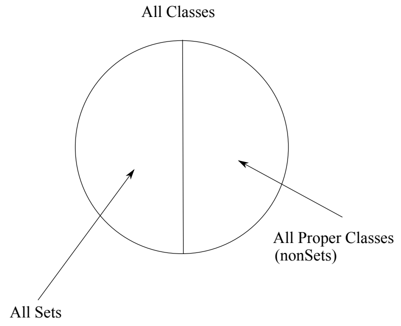
Safe Set Theory

So, *some* collections are *not* —technically— sets, as the Russell Paradox taught us! How do we tell them apart?

From now one we will deal with collections that *may or may not* be sets, with a promise of learning how to create sets if we want to!

The modern literature uses the terminology "**class**" for any such collection (and uses the term "collection" non technically and sparsely).

The above is captured by the following picture:



2.0.1 Definition. (Classes and sets)
From now on we call *all* collections **classes**.

Definitions by defining property "Let $\mathbb{A} = \{x : P(x)\}$" always define a **class**, but as we saw, sometimes —e.g., if "$P(x)$" is specifically "$x \notin x$"— that class is *not* a set (Section 1.1). *Classes that are not sets* are called **proper classes**. We will normally use what is known as "blackboard bold" notation and capital latin letters to denote classes by names such as $\mathbb{A}, \mathbb{B}, \mathbb{X}$. If we determine that some class $\mathbb{A}$ *is* a set, we would rather write it as A, but we make an exception for the following **sets**: Mathematicians use notation and results from set theory in their everyday practice. We call the sets that mathematicians use the "real sets" of our mathematical *intuition*, like the set of natural numbers, $\mathbb{N}$ (also denoted by ω), integers $\mathbb{Z}$, rationals $\mathbb{Q}$ and reals $\mathbb{R}$. □

In forming the class $\{x : P(x)\}$ for any property $P(x)$ we say that we apply *comprehension*. It was the Frege/Cantor who believed (explicitly or implicitly) that comprehension was *safe* —i.e., always produced a set. Russell proved that it was not.

It is known that set theory, using as primitives the notions of *set*, *atom* (an object that is not sub-divisible; not a collection of objects), and the relation *belongs to* ($\in$), is sufficiently strong to serve as the foundation of all mathematics.

Mathematicians use notation and results from set theory in their everyday practice. We call the sets that mathematicians use the "real sets" of our mathematical *intuition*, like the set of natural numbers, $\mathbb{N}$ (also denoted by *omega*), integers $\mathbb{Z}$, rationals $\mathbb{Q}$ and reals $\mathbb{R}$.

2.1. The "real sets"

So, how can we tell, or indeed *guarantee*, that a certain class is a *set*?

Russell proposed this "recovery" from his Paradox:

Make sure that sets are built by stages, where at stage 0 all atoms are available. Atoms are also called *urelements* in the literature from the German *Urelemente*, which in analogy with the word "*urtext*" —meaning *the earliest text*— would mean that they are the "earliest" mathematical objects. Witness that they are available at stage 0!

We may then collect atoms to form all sorts of "first level" *sets*. We may proceed to collect any mix of atoms and first-level sets to build new collections —second-level sets— *and so on.* Much of what set theory does is attempting to remove the ambiguity from this "and so on". See below, **Principles** 0–2.

Thus, at the beginning we have all the level-0, or type-0, objects available to us. For example, atoms such as $1, 2, 13, \sqrt{2}$ are available. At the next level we can include any number of such atoms (from none at all, to all) to build a set, that is, a new mathematical object. Allowing the usual notation, i.e., listing of what is included within braces, we may cite a few examples of level-1 sets:

L1-1. $\{1\}$.

L1-2. $\{1, 1\}$.

L1-3. $\{1, \sqrt{2}\}$.

L1-4. $\{\sqrt{2}, 1\}$.

We already can identify a few level-2 objects, using what (we already know) *is* available:

L2-1. $\{\{\sqrt{2}, 1\}\}$.

Note how the level of nesting of { }-brackets matches the level or stage of the formation of these objects!

2.1.1 Definition. (Class and set *equality*) This definition applies to any classes, hence, in particular, to any *sets* as well.

Two classes $\mathbb{A}$ and $\mathbb{B}$ are *equal* —written $\mathbb{A} = \mathbb{B}$— means

$$x \in \mathbb{A} \text{ iff } x \in \mathbb{B}$$

That is, an object is in $\mathbb{A}$ iff it is also in $\mathbb{B}$.

$\mathbb{A}$ is a *subclass* of $\mathbb{B}$ —written $\mathbb{A} \subseteq \mathbb{B}$— means that every element of the first class occurs also in the second, or

$$\text{If } x \in \mathbb{A}, \text{ then } x \in \mathbb{B}$$

If $\mathbb{A}$ is a set, then we say it is a *subset* of $\mathbb{B}$.

If we have $\mathbb{A} \subseteq \mathbb{B}$ but $\mathbb{A} \neq \mathbb{B}$, then we write $\mathbb{A} \underset{\neq}{\subset} \mathbb{B}$ (some of the literature uses $\mathbb{A} \subsetneq \mathbb{B}$ or even $\mathbb{A} \subset \mathbb{B}$ nstead) and say that $\mathbb{A}$ is a *proper subclass* of $\mathbb{B}$.

Caution. In the terminology "*proper subclass*" the "proper" refers to the fact that $\mathbb{A}$ is not all of $\mathbb{B}$. It does *NOT* say that $\mathbb{A}$ is not a set! It *may* be a set and then we say that it is "*proper subset*" of $\mathbb{B}$. □

If n is an integer-valued variable, then what do you understand by "$2n$ is even"? The normal understanding is that "no matter what the value of n is, $2n$ is even", or "for all values of n, $2n$ is even".

When we get into our logic topic in the course we will see that we *can* write "for all values of n, $2n$ is even" with less English as "$(\forall n)(2n$ is even)". So "$(\forall n)$" says "for all (values of) n".

Mathematicians often prefer to have statements like "$2n$ is even" with the "for all" *always implied*.[†] You can write a whole math book without writing $\forall$ even once, and without overdoing the English.

2.1.2 Remark. Since "iff" between two statements S_1 and S_2 means that we have *both* directions

$$\text{If } S_1, \text{ then } S_2$$

and

$$\text{If } S_2, \text{ then } S_1$$

we have that "$\mathbb{A} = \mathbb{B}$" is the same as (equivalent to) "$\mathbb{A} \subseteq \mathbb{B}$ and $\mathbb{B} \subseteq \mathbb{A}$". □

2.1.3 Example. In the context of the "$\mathbb{A} = \{x : P(x)\}$" notation we should remark that notation-by-listing can be simulated by notation-by-defining-property: For example, $\{a\} = \{x : x = a\}$ —here "$P(x)$" is $x = a$.

[†] An exception occurs in Induction that we will study later, where you *fix* an n (but keep it as a variable, not as 5 or 42) and assume the "induction hypothesis" $P(n)$. But do not worry about this now!

Also $\{A, B\} = \{x : x = A \text{ or } x = B\}$. Let us verify the latter: Say $x \in$ lhs.[†] Then $x = A$ or $x = B$. Thus x must be A or B. But then the entrance requirement of the rhs[‡] is met, so $x \in$ rhs.

Conversely, say $x \in$ rhs. Then the entrance requirement is met so we have (at least) one of $x = A$ or $x = B$. Trivially, in the first case $x \in$ lhs and ditto for the second case. □

We now postulate the principles of formation of sets!

Principle 0. Sets and atoms are *the mathematical objects* of our (safe) set theory.

Sets are formed by stages. At stage 0 we acknowledge the *presence* of atoms. *They are given outright, they are not built.*

At *any* stage Σ we *may* build a *set*, collecting together other *mathematical objects* (sets or atoms) *provided* these (mathematical) objects we put into our set were available at stages before Σ.

Principle 1. *Every set is built at some stage.*

Principle 2. If Σ is a stage of set construction, then *there is* a stage Φ *after* it.

Principle 2 makes clear that we have infinitely many stages of set formation in our toolbox.

2.1.4 Remark. If some set is definable ("buildable") at some stage Σ, then it is also definable at any later stage as well, as **Principle** 0 makes clear.

The informal set-formation-by-stages will guide us to build, safely, all the sets we may need in order to do mathematics. □

2.2. *What caused Russell's paradox*

How would the set-building-by-stages doctrine avoid Russell's paradox?

Recall that *à la Cantor* we get a paradox (contradiction) because we insisted to believe that all classes are sets, that is, following Cantor we "believed" Russell's "R" was a *set*.

Principles 0–2 allow us to know *a priori* that R is a proper class. No contradiction!

How so?

[†] Left Hand Side.

[‡] Right Hand Side.

OK, is $x \in x$ true or false? Is there *any* mathematical object x —say, A— for which it *is* true?

$$A \in A? \tag{1}$$

Well, for atom A, (1) is false since atoms have no set structure, that is, are not collections of objects. An atom A *cannot contain anything*, in particular it cannot contain A.

What if A is a set and $A \in A$? Then in order to build A, the *set*, we have to wait until *after* its member, A is built (Principle 0). So, we need (the left) A to be built before (the right) A in (1).

Absurd!

So (1) is **false**. A being arbitrary, we demonstrated that

$$x \in x \text{ is false}$$

thus $x \notin x$ is true (forall x), therefore R of Section 1.1 is $\mathbb{U}$, the universe of *all sets and atoms.*

$$R = \mathbb{U}$$

So? Well this $\mathbb{U}$ is "far too big" to be built as a *set* and we should never have used $\{x : x \notin x\}$ so recklessly!

"Too Big" is bad in set theory; it intuitively means we ran out of stages after we built all the members of the class! No stages left to build the class as a set!

The "intuition", as always, is vague.

So here is why $\mathbb{U}$ is *not* a set. Well, <u>if it is</u>

- $\mathbb{U} \in \mathbb{U}$ since the rhs contains all sets and we believe the lhs to be a set.
- but we just saw that the above is false if $\mathbb{U}$ is a set!

So $\mathbb{U}$, aka R, is a *proper* class. Thus, the fact that R is not a set is neither a surprise, nor paradoxical. It is just a proper class as we just have recognised.

2.3. Some <u>useful</u> sets

2.3.1 Example. (Pair) By Principle 0, if A and B are sets or atoms, then let A be available at stage Σ and B at stage Σ'. Without loss of generality say Σ' is not later than Σ. Let then pick a stage Σ'' *after* Σ (Principle 2). This will be be after both (cf. Principle 2) Σ, Σ'.

At stage Σ'' we can build

$$\{A, B\} \tag{1}$$

as a *set* (cf. Principle 0).

We call (1) the (unordered) *pair set*.

Pause. Why "unordered"? See 2.1.1.◀ □

We have just proved a theorem above:

2.3.2 Theorem. *If A, B are sets or atoms, then $\{A, B\}$ is a set.*

2.3.3 Exercise. Without referring to stages in your proof, prove that if A is a set or atom, then $\{A\}$ is a set. □

2.3.4 Remark. A very short digression into Boolean Logic —for now. It will be convenient —but not necessary; we are doing fine so far— to use *truth tables* to handle many simple situations that we will encounter where "logical connectives" such as "not", "and", "or", "implies" and "is equivalent" enter into our arguments.

We will put on record here how to compute things such as "S_1 and S_2", "S_1 implies S_2", etc., where S_1 and S_2 stand for two arbitrary statements of mathematics. In the process we will introduce the *mathematical symbols* for "and", "implies", etc.

The symbol translation table from English to symbol is:

NOT	$\neg$
AND	$\wedge$
OR	$\vee$
IMPLIES (IF...,THEN)	$\rightarrow$
IS EQUIVALENT	$\equiv$

The truth table below has a simple reading. For *all possible* truth values —true/false, for short **t**/**f**— of the "simpler" statements S_1 and S_2 we indicate the computed truth value of the compound (or "more complex)" statement that we obtain when we apply one or the other Boolean connective of the previous table.

S_1	S_2	$\neg S_1$	$S_1 \wedge S_2$	$S_1 \vee S_2$	$S_1 \rightarrow S_2$	$S_1 \equiv S_2$	$S_2 \rightarrow S_1$
f	**f**	**t**	**f**	**f**	**t**	**t**	**t**
f	**t**	**t**	**f**	**t**	**t**	**f**	**f**
t	**f**	**f**	**f**	**t**	**f**	**f**	**t**
t	**t**	**f**	**t**	**t**	**t**	**t**	**t**

Comment. All the computations of truth values satisfy our intuition, except perhapsthat for "$\rightarrow$": $\neg$ flips the truth value as it should, $\wedge$ is eminently consistent with common sense, $\vee$ is the "inclusive or" of the mathematician, and $\equiv$ is just equality on the set $\{\mathbf{f}, \mathbf{t}\}$, as it should be.

The "problem" with $\rightarrow$ is that there is no *causality* from left to right. The only "sane" entry is for $\mathbf{t} \rightarrow \mathbf{f}$. The outcome should be false for a "bad implication" and so it is. But look at it this way:

- Looking at → also in the "red column" see how the given table for → is eminently consistent with that for ≡. Intuitively ≡ is → from left to right AND → from right to left. It IS!

- This version of → goes way back to Aristotle. It is the version used by the vast majority of practising mathematicians and is nicknamed "material implication".

Practical considerations. Thus

1. if you want to demonstrate that $S_1 \vee S_2$ is true, for any component statements S_1, S_2, then show that *at least one* of the S_1 and S_2 is true.

2. If you want to demonstrate that $S_1 \wedge S_2$ is true, then show that *both* of the S_1 and S_2 are true.

   Note, incidentally, the if we *know* that $S_1 \wedge S_2$ is true, then the truth table guarantees that each of S_1 and S_2 *must* be true.

3. If now you want to show the implication $S_1 \to S_2$ is true, **then the only real work is to show that if we assume** S_1 **is true, then** S_2 **is true too**.

   If S_1 *is known to be false, then no work is required to prove the implication because of the first two lines of the truth table*!!

4. If you want to show $S_1 \equiv S_2$, then —because the last three columns show that this is equivalent to (same truth values as) $\Big(S_1 \to S_2\Big) \wedge \Big(S_2 \to S_1\Big)$— that is, you just prove **each** of the two implications $S_1 \to S_2$ and $S_2 \to S_1$

   An important variant of → and ≡ Pay attention to this point since almost everybody gets it wrong! In the literature and in the interest of creating a usable shorthand many practitioners of mathematical writing use notation

   $$S_1 \to S_2 \to S_3 \tag{1}$$

   attempting to convey the meaning

   $$(S_1 \to S_2) \wedge (S_2 \to S_3) \tag{2}$$

   Alas, (2) is not the same as (1)! But what about $a < b < c$ for $a < b \wedge b < c$? That is wrong too!

   Back to →-chains like (1) vs. chains like (2): Take S_1 to be $\mathbf{t}$ (true), S_2 to be $\mathbf{f}$ and S_3 to be $\mathbf{t}$. Then (1) is true because in a chain using same Boolean connective *we put brackets from right to left*: (1) is $S_1 \to (S_2 \to S_3)$ and evaluates to $\mathbf{t}$, while (2) evaluates clearly to false ($\mathbf{f}$) since $S_1 \to S_2 = \mathbf{f}$ and $S_2 \to S_3 = \mathbf{t}$.

So we need a special symbol to denote (2) "economically". We need a *conjunctional implies*! Most people use $\Longrightarrow$ for that:

$$S_1 \Longrightarrow S_2 \Longrightarrow S_3 \tag{3}$$

that means, by **definition**, (2) above.

Similarly,

$$S_1 \equiv S_2 \equiv S_3 \tag{4}$$

is **NOT** conjunctional. It is **not** two equivalences —two statements— connected by an *implied* "$\wedge$", rather it says

$$S_1 \equiv (S_2 \equiv S_3)$$

Now if $S_1 = \mathbf{f}$, $S_2 = \mathbf{f}$ and $S_3 = \mathbf{t}$, then (4) evaluates as $\mathbf{t}$ but the conjunctional version

$$(S_1 \equiv S_2) \wedge (S_2 \equiv S_3) \tag{5}$$

evaluates as $\mathbf{f}$ since the second side of $\wedge$ is $\mathbf{f}$.

So how do we denote (5) correctly without repeating the consecutive S_2's and omitting the implied "$\wedge$"? This way:

$$S_1 \Longleftrightarrow S_2 \Longleftrightarrow S_3 \tag{4}$$

By definition, "$\Longleftrightarrow$" is conjunctional: It applies to two statements — S_i and S_{i+1}— only and implies an $\wedge$ before the adjoining next similar equivalence. □

2.3.5 Theorem. (The subclass theorem) *Let $\mathbb{A} \subseteq B$ (B a set). Then $\mathbb{A}$ is a set.*

Proof. Well, B being a set there is a stage Σ where it is built (Principle 1). By Principle 0, all members of B are available or built before stage Σ.

But by $\mathbb{A} \subseteq B$, all the members of $\mathbb{A}$ are among those of B.

Hey! By Principle 0 we can build $\mathbb{A}$ at stage Σ, so *it is a set.* □

Some corollaries are useful:

2.3.6 Corollary. (Modified comprehension I) *If for all x we have*

$$P(x) \to x \in A \tag{1}$$

for some set A, then $\mathbb{B} = \{x : P(x)\}$ is a set.

Proof. I will show that $\mathbb{B} \subseteq A$, that is,

$$x \in \mathbb{B} \to x \in A$$

Indeed (see 3 under **Practical considerations** in 2.3.4), let $x \in \mathbb{B}$. Then $P(x)$ is true, hence $x \in A$ by (1). Now invoke 2.3.5. □

2.3.7 Corollary. (Modified comprehension II) *If A is a set, then so is $\mathbb{B} = \{x : x \in A \wedge P(x)\}$ for any property $P(x)$.*

Proof. The defining property here is "$x \in A \wedge P(x)$". This implies $x \in A$ —by 2 in 2.3.4— that is, we have

$$(x \in A \wedge P(x)) \rightarrow x \in A$$

Now invoke 2.3.6. □

2.3.8 Remark. (*The* empty set) The class $\mathbb{E} = \{x : x \neq x\}$ has no members at all; it is empty. Why? Because

$$x \in \mathbb{E} \equiv x \neq x$$

but the condition $x \neq x$ is always false, therefore so is the statement

$$x \in \mathbb{E} \tag{1}$$

Is the class $\mathbb{E}$ a set?

Well, take $A = \{1\}$. This is a set as the atom 1 is given at stage 0, and thus we can construct the *set* A at stage 1.

Note that, by (1) and 3 in 2.3.4 we have that

$$x \in \mathbb{E} \rightarrow x \in \{1\}$$

is true (for all x). That is, $\mathbb{E} \subseteq \{1\}$.

By 2.3.5, $\mathbb{E}$ *is a set.*

But is it unique so we can justify the use of the definite article "the"? Yes. The specification of the empty set is a class with no members. So if D is another empty set, then we will have $x \in D$ always false. But then

$$x \in \mathbb{E} \equiv x \in D \text{ (both sides of } \equiv \text{ are false)}$$

and we have $\mathbb{E} = D$ by 2.1.1.

The *unique* empty set is denoted by the symbol $\emptyset$ in the literature. □

2.4. Operations on classes and sets

The reader probably has seen before (perhaps in calculus) the operations on sets denoted by $\cap, \cup, -$ and others. We will look into them in this section.

2.4.1 Definition. (Intersection of two classes) We define for any classes $\mathbb{A}$ and $\mathbb{B}$

$$\mathbb{A} \cap \mathbb{B} \overset{Def}{=} \Big\{x : x \in \mathbb{A} \wedge x \in \mathbb{B}\Big\}$$

We call the operator $\cap$ *intersection* and the result $\mathbb{A} \cap \mathbb{B}$ the intersection of $\mathbb{A}$ and $\mathbb{B}$.

If $\mathbb{A} \cap \mathbb{B} = \emptyset$ —which happens precisely when the two classes have no common elements— we call the classes *disjoint*.

$\underline{\text{It is meaningless to have } \cap \text{ operate on atoms.}}$[†] □

We have the easy theorem below:

2.4.2 Theorem. *If B is a set, as its notation suggests, then $\mathbb{A} \cap B$ is a set.*

Proof. I will prove $\mathbb{A} \cap B \subseteq B$ which will rest the case by 2.3.5. So, I want

$$x \in \mathbb{A} \cap B \rightarrow x \in B$$

To this end, let then $x \in \mathbb{A} \cap B$ (cf. 3 in 2.3.4). This says that $x \in \mathbb{A} \wedge x \in B$ is true, so $x \in B$ is true. □

2.4.3 Corollary. *For sets A and B, $A \cap B$ is a set.*

2.4.4 Definition. (Union of two classes) We define for any classes $\mathbb{A}$ and $\mathbb{B}$

$$\mathbb{A} \cup \mathbb{B} \overset{Def}{=} \Big\{x : x \in \mathbb{A} \vee x \in \mathbb{B}\Big\}$$

We call the operator $\cup$ *union* and the result $\mathbb{A} \cup \mathbb{B}$ the union of $\mathbb{A}$ and $\mathbb{B}$.

$\underline{\text{It is meaningless to have } \cup \text{ operate on atoms}}$. □

2.4.5 Theorem. *For any* sets *A and B, $A \cup B$ is a set.*

Proof. By assumption say A is built at stage Σ while B is built at stage Σ'. Without loss of generality (for short, "wlg") say Σ is no later than Σ', that is, $\Sigma \leq \Sigma'$.

By Principle 2 I can pick a state $\Sigma'' > \Sigma'$, thus

$$\Sigma'' > \Sigma' \tag{1}$$

and

$$\Sigma'' > \Sigma \tag{2}$$

Lets us pick examine any item $x \in A \cup B$:

I have two (not necessarily mutually exclusive) cases (by 2.4.5):

[†] The definition expects $\cap$ to *operate on classes.* As we know, atoms (by definition) *have no set/class structure* thus no class and no set is an atom.

- $x \in A$. Then x was available or built[‡] at a stage $< \Sigma$,

$$\text{hence, by (2), } \underline{x \text{ is available } \textit{before } \Sigma''} \tag{3}$$

- $x \in B$. Then x was available or built at a stage $< \Sigma'$,

$$\text{hence, by (1), } \underline{x \text{ is available } \textit{before } \Sigma''} \tag{4}$$

In either case, (3) or (4), the arbitrary x from $A \cup B$ is built before Σ'', so we can collect all those x-values at stage Σ'' in order to form a *set*: $A \cup B$. □

2.4.6 Definition. (Difference of two classes) We define for any classes $\mathbb{A}$ and $\mathbb{B}$

$$\mathbb{A} - \mathbb{B} \stackrel{Def}{=} \Big\{x : x \in \mathbb{A} \wedge x \notin \mathbb{B}\Big\}$$

We call the operator $-$ *difference* and the result $\mathbb{A} - \mathbb{B}$ the difference of $\mathbb{A}$ and $\mathbb{B}$, in that order.

It is meaningless to have $-$ operate on atoms. □

2.4.7 Theorem. *For any* set A *and class* $\mathbb{B}$*,* $A - \mathbb{B}$ *is a set.*

Proof. The reader is asked to verify that $A - \mathbb{B} \subseteq A$. We are done by 2.3.5. □

Notation. The definitions of $\cap$ and $-$ suggest a shorter notation for the rhs for $\mathbb{A} \cap \mathbb{B}$ and $\mathbb{A} - \mathbb{B}$. That is, respectively, it is common to write instead

$$\Big\{x \in \mathbb{A} : x \in \mathbb{B}\Big\}$$

and

$$\Big\{x \in \mathbb{A} : x \notin \mathbb{B}\Big\}$$

2.4.8 Exercise. Demonstrate —using Definition 2.4.1— that for any $\mathbb{A}$ and $\mathbb{B}$ we have $\mathbb{A} \cap \mathbb{B} = \mathbb{B} \cap \mathbb{A}$. □

2.4.9 Exercise. Demonstrate —using Definition 2.4.4— that for any $\mathbb{A}$ and $\mathbb{B}$ we have $\mathbb{A} \cup \mathbb{B} = \mathbb{B} \cup \mathbb{A}$. □

2.4.10 Exercise. By picking two particular very small sets A and B show that $A - B = B - A$ is not true for all sets A and B.

Is it true of all classes? □

Let us generalise unions and intersections next. First a definition:

[‡] As x may be an atom, we allow the *possibility* that it was available *with no building involved*, hence we said "available or built". For A and B though we are told they are *sets*, so they *were* built at some stage, by Principle 1!

2.4.11 Definition. (Family of sets) A class $\mathbb{F}$ is called a *family of sets* iff it contains no atoms. The letter F is here used generically, and a family may be given any name, usually capital. □

2.4.12 Example. Thus, $\emptyset$ is a family of sets; the empty family.

So are $\{\{2\},\{2,\{3\}\}\}$ and $\mathbb{V}$, the latter given by

$$\mathbb{V} \overset{Def}{=} \Big\{x : x \text{ is a set}\Big\}$$

BTW, as $\mathbb{V}$ contains all sets (but no atoms!) it is a proper class! Why? Well, if it is a set, then it is one of the x-values that we are collecting, thus $\mathbb{V} \in \mathbb{V}$. But we saw that this statement is false for sets!

Here are some classes that are *not* families: $\{1\}$, $\{2,\{\{2\}\}\}$ and $\mathbb{U}$, the latter being the universe of all objects —sets and atoms— and equals Russell's "R" as we saw in Section 2.2. These all are disqualified as they contain atoms. □

2.4.13 Definition. (Intersection and union of families) Let $\mathbb{F}$ be a family of sets. Then

(i) the symbol $\bigcap \mathbb{F}$ denotes the class that contains *all the objects* that *are common to all* $A \in \mathbb{F}$.

In symbols the definition reads:

$$\bigcap \mathbb{F} \overset{Def}{=} \Big\{x : \text{for all } A, A \in \mathbb{F} \rightarrow x \in A\Big\} \tag{1}$$

(ii) the symbol $\bigcup \mathbb{F}$ denotes the class that contains *all the objects* that *are found among the various* $A \in \mathbb{F}$. That is, imagine that the members of *each* $A \in \mathbb{F}$ are "emptied" into a single —originally empty— container $\{\ldots\}$. The class we get this way is what we denote by $\bigcup \mathbb{F}$.

In symbols the definition reads (and I think it is clearer):

$$\bigcup \mathbb{F} \overset{Def}{=} \Big\{x : \text{for some } A, A \in \mathbb{F} \wedge x \in A\Big\} \tag{2}$$

□

2.4.14 Example. Let $\mathbb{F} = \{\{1\},\{1,\{2\}\}$. Then emptying all the contents of the members of $\mathbb{F}$ is some (originally) empty container we get

$$\{1, 1, \{2\}\} \tag{3}$$

This is $\bigcup \mathbb{F}$.

Would we get the same answer from the mathematical definition (2)? Of course:

1 *is* in some member of $\mathbb{F}$, indeed in both of the members $\{1\}$ and $\{1,\{2\}\}$, and in order to emphasise this I wrote two copies of 1 —it is empties/contributed

twice. Then $\{2\}$ is the member that only $\{1, \{2\}\}$ of $\mathbb{F}$ contributes.

What is $\bigcap \mathbb{F}$? Well, only 1 is common between the two sets —$\{1\}$ and $\{1, \{2\}\}$— that are in $\mathbb{F}$. So, $\bigcap \mathbb{F} = \{1\}$. □

2.4.15 Exercise.

1. Prove that $\bigcup \Big\{A, B\Big\} = A \cup B$.
2. Prove that $\bigcap \Big\{A, B\Big\} = A \cap B$.

Hint. In each of part 1. and 2. show that lhs $\subseteq$ rhs and rhs $\subseteq$ lhs. For that analyse membership, i.e., "assume $x \in$ lhs and prove $x \in$ rhs", and conversely (cf. 2.1.1 and 2.1.2.) □

2.4.16 Theorem. *If the set F is a family of sets, then $\bigcup F$ is a set.*

Proof. Let F be built at stage Σ. Now,

$$x \in \bigcup F \equiv x \in \overset{\text{some}}{\overset{\downarrow}{A}} \in F$$

Thus x is available or built before A which is built before stage Σ since that is when F was built. x being arbitrary, all members of $\bigcup F$ are available/built before Σ, so we can build $\bigcup F$ as a set at stage Σ. □

2.4.17 Theorem. *If the class $\mathbb{F} \neq \emptyset$ is a family of sets, then $\bigcap \mathbb{F}$ is a set.*

Proof. By assumption there is some set in $\mathbb{F}$. Fix one such and call it D.

First note that

$$x \in \bigcap \mathbb{F} \to x \in D \tag{$*$}$$

Why? Because (1) of Definition 2.4.13 says that

$$x \in \bigcap \mathbb{F} \equiv \text{for all } A \in \mathbb{F} \text{ we have } x \in A$$

Well, D *is* one of those "A" sets in $\mathbb{F}$, so if $x \in \bigcap \mathbb{F}$ then $x \in D$. We established $(*)$ and thus we established

$$\bigcap \mathbb{F} \subseteq D$$

by 2.1.1. We are done by 2.3.5. □

2.4.18 Remark. What if $\mathbb{F} = \emptyset$? Does it affect Theorem 2.4.17? Yes, **badly**!

In Definition 2.4.13 we read

$$\bigcap \mathbb{F} \overset{Def}{=} \Big\{x : \text{for all } A, A \in \mathbb{F} \to x \in A\Big\} \tag{$**$}$$

However, as the hypothesis (i.e., lhs) of the implication in $(**)$ is **false**, the implication itself is **true**. Thus the entrance condition "for all $A, A \in \mathbb{F} \to x \in A$" is true for *all* x and thus allows *ALL* objects x to get into $\bigcap \mathbb{F}$,

Thus $\bigcap \mathbb{F} = \mathbb{U}$, the universe of *all* objects which we saw (cf. Section 2.2 is a proper class. □

2.4.19 Exercise. What is $\bigcup F$ if $F = \emptyset$? Set or proper class? Can you "compute" which class exactly it is? □

2.4.20 Remark. (More notation)

Suppose the family of sets Q is a set of sets A_i, for $i = 1, 2, \ldots, n$ where $n \geq 3$.

$$Q = \{A_1, A_2, \ldots, A_n\}$$

Then we have a few alternative notations for $\bigcap Q$:

(a)

$$A_1 \cap A_2 \cap \ldots \cap A_n$$

or, more elegantly,

(b)

$$\bigcap_{i=1}^{n} A_i$$

or also

(c)

$$\bigcap\nolimits_{i=1}^{n} A_i$$

Similarly for $\bigcup Q$:

(i)

$$A_1 \cup A_2 \cup \ldots \cup A_n$$

or, more elegantly,

(ii)

$$\bigcup_{i=1}^{n} A_i$$

or also

(iii)

$$\bigcup\nolimits_{i=1}^{n} A_i$$

If the family has so many elements that all the natural numbers are need to index the sets in the set family Q we will write

$$\bigcap_{i=0}^{\infty} A_i$$

or

$$\textstyle\bigcap_{i=0}^{\infty} A_i$$

or

$$\bigcap_{i\geq 0} A_i$$

or

$$\textstyle\bigcap_{i\geq 0} A_i$$

for $\bigcap Q$ and

$$\bigcup_{i=0}^{\infty} A_i$$

or

$$\textstyle\bigcup_{i=0}^{\infty} A_i$$

or

$$\bigcup_{i\geq 0} A_i$$

or

$$\textstyle\bigcup_{i\geq 0} A_i$$

for $\bigcup Q$ □ ☡

2.4.21 Example. Thus, for example, $A \cup B \cup C \cup D$ can be seeing —just changing the notation— as $A_1 \cup A_2 \cup A_3 \cup A_4$, therefore it means, $\bigcup\{A_1, A_2, A_3, A_4\}$, or $\bigcup\{A, B, C, D\}$.

Same comment for $\cap$. □

Pause. How come for the case for $n = 2$ we *proved*[†] $A \cup B = \bigcup\{A, B\}$ (2.4.15) but *here* we say ($n \geq 3$) that something like the content of the previous remark and example are *notation* (*definitions*)?

Well, we had *independent* definitions (and associated theorems re set status for each, 2.4.5 and 2.4.16) for $A \cup B$ and $\bigcup\{A, B\}$ so it makes sense to compare the two definitions after the fact and see if we can *prove* that they say the same thing. For $n \geq 3$ we opted to *NOT* give a definition for $A_1 \cup \ldots \cup A_n$ that is independent of $\bigcup\{A_1 \cup \ldots \cup A_n\}$, rather we gave the definition of the former in terms of the latter. No independent definitions, no theorem to compare the two!◀

[†] Well, *you* proved! Same thing :-)

2.5. The powerset

2.5.1 Definition. For any set A the symbol $\mathscr{P}(A)$ —pronounced the *powerset* of A— is defined to be the class

$$\mathscr{P}(A) \overset{Def}{=} \Big\{x : x \subseteq A\Big\}$$

Thus we collect *all* the subsets x of A to form $\mathscr{P}(A)$.

The literature most frequently uses the symbol 2^A in place for of $\mathscr{P}(A)$. □

☡ (1) The term "power*set*" is slightly premature, but it is apt. Under the conditions of the definition —A a set— 2^A is a *set* as we prove immediately below.

(2) We said "*all* the subsets x of A" in the definition. This is correct. As we know from 2.3.5, if $\mathbb{X} \subseteq Y$ and Y is a set, then so is $\mathbb{X}$. ☡

2.5.2 Theorem. *For any set A, its powerset $\mathscr{P}(A)$ is a set.*

Proof. Let A be built at stage Σ. Then each its members y are given or built *before* Σ.

Thus, since *every* subset x of A is a set of y-values, **every such subset x can be built at stage Σ.**

But then, just take any $\Sigma' > \Sigma$. Since all x-values (such that $x \subseteq A$) are built *before* Σ', at stage Σ' we can collect them all and build the *set* 2^A. □

2.5.3 Example. Let $A = \{1, 2, 3\}$. Then

$$\mathscr{P}(A) = \Big\{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{3, 2\}, \{1, 2, 3\}\Big\}$$

Thus the powerset of A has 8 elements.

We will later see that if A has n elements, for any $n \geq 0$, then 2^A has 2^n elements. This observation is at the root of the notation "2^A". □

2.5.4 Remark. For any set A it is trivial (verify!) that we have $\emptyset \subseteq A$ and $A \subseteq A$. Thus, for any A, $\{\emptyset, A\} \subseteq 2^A$. □

2.6. The ordered pair and finite sequences

To introduce the concepts of cartesian product —so that, in principle, plane analytic geometry can be developed within set theory— we need an object "(A, B)" that is *like* the set pair (2.3.1) in that it contains *two* objects, A and B ($A = B$ is a possibility), but in (A, B) order *and* length (here it is 2) matter!

> *We want* $(A, B) = (A', B')$ implies $A = A'$ *and* $B = B'$. *Moreover,* (A, A) *is not* $\{A\}$*! It is still an ordered* pair *but so happens that the first and second component, as we call the members of the ordered pair, are equal in this example.*

So, are we going to accept a new type of object in set theory? *Not at all*! We will build (A, B) so that it is a set!

2.6.1 Definition. (Ordered pair) *By definition*, (A, B) is the abbreviation (short name) given below:

$$(A, B) \overset{Def}{=} \Big\{A, \{A, B\}\Big\} \tag{1}$$

We call "(A, B)" an *ordered pair*, and A its first *component*, while B is its second component. □

2.6.2 Remark.

1. Note that $A \neq \{A, B\}$ and $A \neq \{A, A\}$, because in either case we would otherwise get $A \in A$, which is false for *sets or atoms* A. Thus (A, B) does contain exactly two members, or *has length 2*: A and $\{A, B\}$.

   Pause. We have *not* said in 2.6.1 that A and B are sets or atoms. So what right do we have in the paragraph above to so declare?◀

2. What about the desired property that

   $$(A, B) = (X, Y) \to A = X \wedge B = Y \tag{2}$$

   Well, **assume the lhs** of "$\to$" in (2) and prove the rhs, "$A = X \wedge B = Y$". From our truth table we know that we do the latter by proving *each* of $A = X$ and $B = Y$ true (separately).

   The lhs that we assume translates to

   $$\Big\{A, \{A, B\}\Big\} = \Big\{X, \{X, Y\}\Big\} \tag{3}$$

   By the remark #1 above there are *two* distinct members in each of the two sets that we equate in (3).

   So since (3) is true (by assumption) we have (by definition of set equality) one of:

(a) $A = \{X, Y\}$ and $\{A, B\} = X$, that is, **1st listed element in lhs of "=" equals the 2nd listed in rhs; and 2nd listed element in lhs of "=" equals the 1st listed in rhs**.

(b) $A = X$ and $\{A, B\} = \{X, Y\}$.

Now case (a) above *cannot hold*, for it leads to $A = \{\{A, B\}, Y\}$. This in turn leads to

$$\{A, B\} \in A$$

and thus the set $\{A, B\}$ is built before of its member A, which contradicts Principle 0.

Let's then work with case (b).

We have

$$\{A, B\} = \{A, Y\} \tag{4}$$

Well, all the members on the lhs must also be on the rhs. I note that A is.

- What if B is also equal to A? Then we have $\{B\} = \{A, Y\}$ and thus $Y \in \{B\}$ (why?). Hence $Y = B$. We showed so far $A = X$ (listed in case (b)) and $B = Y$ (proved here); great!

- Here B is *not* equal to A. But B must be in the rhs of (4), so the only way is $B = Y$. *All Done*! □

Worth noting as a theorem what we proved above:

2.6.3 Theorem. *If* $(A, B) = (X, Y)$*, then* $A = X$ *and* $B = Y$*.*

But is (A, B) a set? (atom it is not, of course!) Yes!

2.6.4 Theorem. (A, B) *is a set.*

Proof. Now $(A, B) = \Big\{A, \{A, B\}\Big\}$. By 2.3.1, $\{A, B\}$ is set. Applying 2.3.1 once more, $\Big\{A, \{A, B\}\Big\}$ is a set. □

2.6.5 Example. So, $(1, 2) = \{1, \{1, 2\}\}$, $(1, 1) = \{1, \{1\}\}$, and $(\{a\}, \{b\}) = \{\{a\}, \{\{a\}, \{b\}\}\}$. □

2.6.6 Remark. We can extend the ordered pair to ordered *triple*, ordered *quadruple*, and beyond!

We take this approach in these notes:

$$(A, B, C) \overset{Def}{=} \Big((A, B), C\Big) \tag{1}$$

$$(A, B, C, D) \overset{Def}{=} \Big((A, B, C), D\Big) \tag{2}$$

$$(A, B, C, D) \overset{Def}{=} \Big((A, B, C), D\Big) \tag{3}$$

etc. So suppose we defined what an n-tuple is, for *some fixed unspecified* n, and denote it by $(A_1, A_2, \ldots, A_n)$ for convenience. Then

$$(A_1, A_2, \ldots, A_n, A_{n+1}) \overset{Def}{=} \Big((A_1, A_2, \ldots, A_n), A_n\Big) \tag{$*$}$$

This is an "*inductive*" or "*recursive*" definition, defining a concept ($n+1$-tuple) in terms of *a smaller instance of itself*, namely, in terms of the concept for an n-tuple, and in terms of the case $n = 2$ that we dealt with by *direct* definition (*not* in terms of the concept itself!) in 2.6.1.

Suffice it to say this "case of $n + 1$ in terms of case of n" provides just *shorthand notation* to take the mystery out of the red "etc." above. We **condense**/*codify* infinitely many definitions (1), (2), (3), . . . into just **two**:

- 2.6.1

  and

- $(*)$

The reader has probably seen such recursive definitions before (likely in calculus and/or high school).

The most frequent example that occurs is to define, for any natural number n and any real number $a > 0$, what a^n means. One goes like this:

$$\begin{aligned} a^0 &= 1 \\ a^{n+1} &= a \cdot a^n \end{aligned}$$

The above condenses infinitely many definitions such as

$$\begin{aligned} a^0 &= 1 \\ a^1 &= a \cdot a^0 = a \\ a^2 &= a \cdot a^1 = a \cdot a \\ a^3 &= a \cdot a^2 = a \cdot a \cdot a \\ a^4 &= a \cdot a^3 = a \cdot a \cdot a \cdot a \\ &\vdots \end{aligned}$$

into just two!

We will study *inductive definitions* and *induction* soon!

Before we exit this remark note that $(A, B, C) = (A', B', C')$ implies $A = A', B = B', C = C'$ because it implies

$$C = C' \text{ and } (A, B) = (A', B')$$

That is, (A, B, C) is an **ordered** triple (3-tuple).

We can also prove that $(A_1, A_2, \ldots, A_n, A_{n+1})$ is an **ordered** $n+1$-tuple, i.e.,

$$(A_1, A_2, \ldots, A_{n+1}) = (A'_1, A'_2, \ldots, A'_{n+1}) \rightarrow A_1 = A'_1 \wedge \ldots \wedge A_{n+1} = A'_{n+1}$$

if we have followed the "etc." all the way to the case of $(A_1, A_2, \ldots, A_n)$. We will do the "etc."-argument *elegantly* once we learn induction! □

2.6.7 Definition. (Finite sequences) An n-tuple for $n \geq 1$ is called a finite sequence of length n, where we extend the concept to a *one element sequence* —**by definition**— to be

$$(A) \overset{Def}{=} A$$

□

Note that now we can redefine all sequences of lengths $n \geq 1$ using again $(*)$ above, but this time with starting condition that of 2.6.7. Indeed, for $n = 2$ we rediscover (A_1, A_2):

$$\text{the "new" 2-tuple pair: } (A_1, A_2) \overset{\text{by } (*)}{=} \Big((A_1), A_2\Big) \overset{\text{by 2.6.7 the "old"}}{=} \Big(A_1, A_2\Big)$$

The big red brackets are applications of the ordered pair defined in 2.6.1, just as it was in the general definition $(*)$.

2.7. The Cartesian product

We are ready to define classes of pairs.

2.7.1 Definition. (Cartesian product of classes) Let $\mathbb{A}$ and $\mathbb{B}$ be classes. Then we define

$$\mathbb{A} \times \mathbb{B} \overset{Def}{=} \Big\{(x, y) : x \in \mathbb{A} \wedge y \in \mathbb{B}\Big\}$$

The definition requires both sides of × to be classes. It makes no sense if one or both are atoms. □

2.7.2 Theorem. *If A and B are sets, then so is $A \times B$.*

Proof. By 2.7.1 and 2.6.1

$$A \times B = \Big\{\{x, \{x, y\}\} : x \in A \wedge y \in B\Big\} \tag{1}$$

So, for each $\{x, \{x, y\}\} \in A \times B$ we have $x \in A$ and $\{x, y\} \subseteq A \cup B$, or $x \in A$ and $\{x, y\} \in 2^{A \cup B}$. Thus $\{x, \{x, y\}\} \subseteq A \cup 2^{A \cup B}$ and hence (changing notation) $(x, y) \in 2^{A \cup 2^{A \cup B}}$.

We have established that

$$A \times B \subseteq 2^{A \cup 2^{A \cup B}}$$

thus $A \times B$ is a set by 2.3.5, 2.4.5 and 2.5.2. □

2.7.3 Definition. Mindful of the Remark 2.6.6 where $\Big((A,B),C\Big)$, $\Big((A,B,C),D\Big)$, etc. were defined, we define here $A_1 \times \ldots \times A_n$ for any $n \ge 3$ as follows:

$$\begin{aligned}
&A \times B \times C && \overset{Def}{=} (A \times B) \times C \\
&A \times B \times C \times D && \overset{Def}{=} (A \times B \times C) \times D \\
&\vdots \\
&A_1 \times A_2 \times \ldots \times A_n \times A_{n+1} && \overset{Def}{=} (A_1 \times A_2 \times \ldots \times A_n) \times A_{n+1} \\
&\vdots
\end{aligned}$$

$$\text{We may write } \mathop{\times}\limits_{i=1}^{n} A_i \text{ for } A_1 \times A_2 \times \ldots \times A_n$$

If $A_1 = \ldots = A_n = B$ we may write B^n for $A_1 \times A_2 \times \ldots \times A_n$. □

2.7.4 Remark. Thus, what we learnt in 2.7.3 is, in other words,

$$\mathop{\times}\limits_{i=1}^{n} A_i \overset{Def}{=} \Big\{(x_1, \ldots, x_n) : x_i \in A_i, \text{ for } i = 1, 2, \ldots, n\Big\}$$

and

$$B^n \overset{Def}{=} \Big\{(x_1, \ldots, x_n) : x_i \in B\Big\}$$

□

2.7.5 Theorem. *If A_i, for $i = 1, 2, \ldots, n$ is a set, then so is $\mathop{\times}\limits_{i=1}^{n} A_i$.*

Proof. $A \times B$ is a set by 2.7.2. By 2.7.3, **and in this order**, we verify that so is $A \times B \times C$ and $A \times B \times C \times D$ and … and $A_1 \times A_2 \times \ldots \times A_n$ and … □

If we had inductive definitions available already, then Definition 2.7.3 would simply read

$$\begin{aligned}
&A_1 \times A_2 && \overset{Def}{=} \Big\{(x_1, x_2) : x_1 \in A_1 \wedge x_2 \in A_2\Big\} \\
&\quad \text{and, for } n \ge 2, \\
&A_1 \times A_2 \times \ldots \times A_n \times A_{n+1} && \overset{Def}{=} (A_1 \times A_2 \times \ldots \times A_n) \times A_{n+1}
\end{aligned}$$

Correspondingly, the proof of 2.7.5 would be far more elegant, via induction.

Chapter 3

Relations and functions

The topic of relations and functions is central in all mathematics and computing. In the former, whether it is calculus, algebra or anything else, one deals with relations (notably equivalence relations, order) and all sorts of functions while in the latter one computes relations and functions, in that, one writes programs that given an input to a relation they compute the response (true or false) or given an input to a function they compute a response which is some object (number, graph, tree, matrix, other) or *nothing*, in case there is no response for said input (for example, there is no response to input "x, y" if what we are computing is $\frac{x}{y}$ but $y = 0$.

We are taking an "extensional" point of view in this course, as is customary in set theory, of relations and functions, that is, we view them as sets of (input, output) ordered pairs. It is also possible to take an intentional point of view, especially in computer science and some specific areas of mathematics, viewing relations and functions as *methods* to compute outputs from given inputs.

3.1. Relations

3.1.1 Definition. (Binary relation) A binary relation is a class $\mathbb{R}$[†] of ordered pairs.

The statements $(x, y) \in \mathbb{R}$, $x\mathbb{R}y$ and $\mathbb{R}(x, y)$ are equivalent. $x\mathbb{R}y$ is the "infix" notation —imitating notation such as $A \subset B$, $x < y$, $x = y$ and has notational advantages. □

3.1.2 Remark. $\mathbb{R}$ contains just pairs (x, y), that is, just sets $\{x, \{x, y\}\}$, that is, it is a family of sets. □

3.1.3 Example. Examples of relations:

[†] I write "$\mathbb{R}$" or "R" for a relation, generically, but $\mathbb{P}$, $\mathbb{Q}$, $\mathbb{S}$ are available to use as well. I will avoid specific names such as $<$, $\subseteq$ in a general discussion. These two are apt to bring in in examples.

(i) $\emptyset$

(ii) $\{(1,1)\}$

(iii) $\{(1,1),(1,2)\}$

(iv) $\mathbb{N}^2$, that is $\{(x,y) : x \in \mathbb{N} \land y \in \mathbb{N}\}$. This is a set by the fact that $\mathbb{N}$ is (Why?) and thus so is $\mathbb{N} \times \mathbb{N}$ by 2.7.2.

(v) $<$ on $\mathbb{N}$, that is $\{(x,y) : x < y \land x \in \mathbb{N} \land y \in \mathbb{N}\}$. This is a set since $< \subseteq \mathbb{N}^2$.

(vi) $\in$, that is,

$$\{(x,y) : x \in y \land x \in \mathbb{U} \land y \in \mathbb{V}\} \tag{$*$}$$

This is a proper class (nonSet). Why? Well, if $\in$ *is* a set, then it is built at some stage Σ.

Now examine the arbitrary (x,y) in $\in$. This is $\{x,\{x,y\}\}$ so it is built before Σ, but then so is its member x (available before Σ). Thus we can collect *all* such x into a *set* at stage Σ. But this "set" contains *all* $x \in \mathbb{U}$ due to the middle conjunct in the entrance condition in $(*)$.[†] That is, this "set" is $\mathbb{U}$. This is absurd! □

Here is another way to argue that the relation $\in$ is not a set: If it is, so is $\bigcup \in$. Any $(x,y) \in\in$ is of the form $\{x,\{x,y\}\}$. Thus all x for which there is a y such that $x \in y$ are in $\bigcup \in$. As we said in the footnote, taking $y = \{x\}$ makes clear that "$x \in y$" does not restrict the x's we can get. We get them all: thus they form the proper class $\mathbb{U}$. I argued $\mathbb{U} \subseteq \bigcup \in$, thus $\bigcup \in$ cannot be a set. So, neither can $\in$ (2.4.16).

So, a binary relation $\mathbb{R}$ is a table of pairs:

input: x	output: y
a	b
a'	b'
$\vdots$	$\vdots$
u	v
$\vdots$	$\vdots$

1. Thus one way to view R is as a device that for inputs x, valued $a, a', \ldots, u, \ldots$ one gets the outputs y, valued $b, b', \ldots, v, \ldots$ respectively. It is all right that a given input may yield multiple outputs (e.g., case (iii) in the previous example).

[†] Hmm. Doesn't the first conjunct "$x \in y$" reduce the number of x-values? No: *For every* x out there take $y = \{x\}$ thus the conjunct $x \in y$ is fulfilled for all x-values, as I showed how to find a y that works.

2. Another point of view is to see both x and y as inputs and the outputs are true or false ($\mathbf{t}$ or $\mathbf{f}$). For example, (a, b) is in the table (that is, aRb) hence if both a and b are ordred input values, then the relation outputs $\mathbf{t}$.

Most of the time we will take the point of view in 1 above. This point of view compels us to define *domain* and *range* of a relation $\mathbb{R}$, that is, the class of all inputs that cause an output and the set of all caused outputs respectively.

3.1.4 Definition. (Domain and range) For any relation $\mathbb{R}$ we define *domain*, in symbols "dom" by

$$\mathrm{dom}(\mathbb{R}) \overset{Def}{=} \{x : (\exists y)x\mathbb{R}y\}$$

where we have introduced the notation "$(\exists y)$" as short for "there exists some y such that", or "for some y,"

Range, in symbols "ran", is defined also in the obvious way:

$$\mathrm{ran}(\mathbb{R}) \overset{Def}{=} \{x : (\exists y)y\mathbb{R}x\} \qquad \square$$

We settle the following, before other things:

3.1.5 Theorem. *For a* set *relation R, both* $\mathrm{dom}(R)$ *and* $\mathrm{ran}(R)$ *are sets.*

Proof. For domain we collect all the x such that xRy, for some y, that is, all the x such that

$$\{x, \{x, y\}\} \in R \tag{1}$$

for some y. Since R is a family of sets, we have that $\bigcup R$ is a set. But then each x in the set $\{x, \{x, y\}\}$ in (1) is in $\bigcup R$. But the set of these x is $\mathrm{dom}(R)$ (3.1.4). Thus $\mathrm{dom}(R) \subseteq \bigcup R$. This settles the domain case.

Let A be the set of all atoms in $\bigcup R$ and define

$$S \overset{Def}{=} \left(\bigcup R\right) - A$$

So, S is a set, and it contains just the $\{x, y\}$ parts of all $\{x, \{x, y\}\} \in R$.

Then $\bigcup S$ contains all the y. That is, $\mathrm{ran}(R) \subseteq \bigcup S$, and that settles the range case. $\square$

3.1.6 Definition. In practice we often have an *a priori decision* about what are *in principle* "legal" inputs for a relation $\mathbb{R}$, and where its outputs go. Thus we have two classes, $\mathbb{A}$ and $\mathbb{B}$ for the class of legal inputs and possible outputs respectively. Clearly we have $\mathbb{R} \subseteq \mathbb{A} \times \mathbb{B}$.

We call $\mathbb{A}$ and $\mathbb{B}$ *left field* and *right field* respectively, and instead of $\mathbb{R} \subseteq \mathbb{A} \times \mathbb{B}$ we often write

$$\mathbb{R} : \mathbb{A} \to \mathbb{B}$$

and also

$$\mathbb{A} \xrightarrow{\mathbb{R}} \mathbb{B}$$

pronounced "$\mathbb{R}$ is a relation *from* $\mathbb{A}$ *to* $\mathbb{B}$".

The term *field* —without left/right qualifiers— for $\mathbb{R} : \mathbb{A} \to \mathbb{B}$ refers to $\mathbb{A} \cup \mathbb{B}$.

If $\mathbb{A} = \mathbb{B}$ then we have

$$\mathbb{R} : \mathbb{A} \to \mathbb{A}$$

but rather than pronouncing this as "$\mathbb{R}$ is a relation *from* $\mathbb{A}$ *to* $\mathbb{A}$" we *prefer*[†] to say "$\mathbb{R}$ is on $\mathbb{A}$". □

3.1.7 Remark. Trivially, for any $\mathbb{R} : \mathbb{A} \to \mathbb{B}$, we have $\mathrm{dom}(\mathbb{R}) \subseteq \mathbb{A}$ and $\mathrm{ran}(\mathbb{R}) \subseteq \mathbb{B}$ (give a quick proof of each of these inclusions).

Also, for any relation $\mathbb{P}$ with no *a priori* specified left/right fields, $\mathbb{P}$ is a relation from $\mathrm{dom}(\mathbb{A}) \to \mathrm{ran}(\mathbb{R})$. Naturally, we say that $\mathrm{dom}(\mathbb{P}) \cup \mathrm{ran}(\mathbb{P})$ is the field of $\mathbb{P}$. □

3.1.8 Example. As an example, consider the *divisibility relation* on all integers (their set denoted by $\mathbb{Z}$) denoted by "$|$":

$$x|y \text{ means } x \text{ divides } y \text{ with } 0 \text{ remainder}$$

thus, for $x = 0$ and all y, the division is *illegal*, therefore

> *The input $x = 0$ to the relation "$|$" **produces no output**, in other words, "**for input $x = 0$ the relation is undefined.**"*

We walk away with two things from this example:

1. It **does** make sense for some relations to *a priori* choose left and right fields, here

   $$| : \mathbb{Z} \to \mathbb{Z}$$

   You would not have divisibility on *real numbers*!

2. $\mathrm{dom}(\,|\,)$ is the set of all inputs that produce some output. Thus, it is NOT the case <u>for all relations</u> that their domain is the same as the left field *chosen*! Note the case in this example! And forget the term "codomain"! (Occurs in our text.) □

3.1.9 Example. Next consider the relation $<$ with left/right fields restricted to $\mathbb{N}$. Then $\mathrm{dom}(<) = \mathbb{N}$, but $\mathrm{ran}(<) \underset{\neq}{\subseteq} \mathbb{N}$. Indeed, $0 \in \mathbb{N} - \mathrm{ran}(<)$. □

Let us extract some terminology from the above examples:

[†] Both ways of saying it are correct.

3.1.10 Definition. Given

$$\mathbb{R} : \mathbb{A} \to \mathbb{B}$$

If $\mathrm{dom}(\mathbb{R}) = \mathbb{A}$, then we call $\mathbb{R}$ *total* or totally defined. If $\mathrm{dom}(\mathbb{R}) \subsetneqq \mathbb{A}$, then we say that $\mathbb{R}$ is *nontotal*.

If $\mathrm{ran}(\mathbb{R}) = \mathbb{B}$, then we call $\mathbb{R}$ *onto*. If $\mathrm{ran}(\mathbb{R}) \subsetneqq \mathbb{B}$, then we say that $\mathbb{R}$ is *not onto*. □

So, $|$ above is nontotal, and $<$ is not onto.

In what follows we move away from the full generality of classes (possibly proper) and restrict attention to relations that are sets.

3.1.11 Example. Let $A = \{1, 2\}$.

- The relation $\{(1, 1)\}$ on A is neither total nor onto.
- The relation $\{(1, 1), (1, 2)\}$ on A is onto but not total.
- The relation $\{(1, 1), (2, 1)\}$ on A is total but not onto.
- The relation $\{(1, 1), (2, 2)\}$ on A is total *and* onto. □

3.1.12 Definition. The relation Δ_A on the set A is given by

$$\Delta_A \overset{Def}{=} \{(x, x) : x \in A\}$$

We call it the *diagonal* ("Δ" for "diagonal") *identity* or relation on A.

Consistent with the second terminology, we may also use the symbol $\mathbf{1}_A$ for this relation. □

3.1.13 Definition. A relation R (not *a priori* restricted to have *predetermined* left or right fields) is

1. *Transitive*: Iff $xRy \wedge yRz$ implies xRz.
2. *Symmetric*: Iff xRy implies yRx.
3. *Antisymmetric*: Iff $xRy \wedge yRx$ implies $x = y$.
4. *Irreflexive*: Iff xRy implies $x \neq y$.

Now assume R is *on a set* A. Then we call it reflexive iff $\Delta_A \subseteq R$. □

3.1.14 Example.

(i) *Transitive* examples: $\emptyset$, $\{(1, 1)\}$, $\{(1, 2), (2, 3), (1, 3)\}$, $<$, $\leq$, $=$, $\mathbb{N}^2$.

(ii) *Symmetric* examples: $\emptyset$, $\{(1, 1)\}$, $\{(1, 2), (2, 1)\}$, $=$, $\mathbb{N}^2$.

(iii) *Antisymmetric* examples: $\emptyset$, $\{(1, 1)\}$, $=$, $\leq$, $\subseteq$.

(iv) *Irreflexive* examples: $\emptyset$, $\{(1,2)\}$, $<$, $\subsetneq$, the relation "$\neq$" on $\mathbb{N}$.

(v) *Reflexive* examples: $\mathbf{1}_A$ on A, $\{(1,1)\}$ on $\{1\}$, $\{(1,2),(2,1),(1,1),(2,2)\}$ on $\{1,2\}$, $=$ on $\mathbb{N}$, $\leq$ on $\mathbb{N}$. □

We can compose relations:

3.1.15 Definition. (Relational composition) Let R and S be (set) relations. Then, their composition, *in that order*, denoted by $R \circ S$ is defined for all x and y by:

$$xR \circ Sy \overset{Def}{\equiv} (\exists z)\Big(xRz \wedge zSy\Big)$$

It is customary to abuse notation and write "$xRzSy$" for "$xRz \wedge zSy$" just as one writes $x < y < z$ for $x < y \wedge y < z$. □

3.1.16 Example. Here is whence the emphasis "*in that order*" above. Say, $R = \{(1,2)\}$ and $S = \{(2,1)\}$. Thus, $R \circ S = \{(1,1)\}$ while $S \circ R = \{(2,2)\}$. Thus, $R \circ S \neq S \circ R$ *in general.* □

3.1.17 Example. For any R, we diagrammatically indicate xRy by

$$x \xrightarrow{R} y$$

Thus, the situation where we have that $xR \circ Sy$ means, for some z, $xRzSy$ is depicted as:

$$\begin{tikzcd}
 & z \arrow[dr, "S"] & \\
x \arrow[ur, "R"] \arrow[rr, "R \circ S"'] & & y
\end{tikzcd}$$

□

3.1.18 Theorem. *The composition of two (set) relations R and S in that order is also a set.*

Proof. Trivially, $R \circ S \subseteq \mathrm{dom}(R) \times \mathrm{ran}(S)$ since in

$$xRzSy, \text{ for some } z$$

all the the x-values are in $\mathrm{dom}(R)$ and all the y-values are in $\mathrm{ran}(S)$. Moreover, we proved in 3.1.5 that $\mathrm{dom}(R)$ and $\mathrm{ran}(S)$ are sets. Thus so is $\mathrm{dom}(R) \times \mathrm{ran}(S)$ (2.7.2). □

3.1.19 Corollary. *If we have $R : A \to B$ and $S : B \to C$, then $R \circ S : A \to C$.*

Proof. This is a trivial modification of the argument above. □

The result of the corollary is depicted diagrammatically as

$$\begin{tikzcd} & B \arrow[dr, "S"] & \\ A \arrow[ur, "R"] \arrow[rr, "R \circ S"'] & & C \end{tikzcd}$$

3.1.20 Theorem. (Associativity of composition) *For any relations* $\mathbb{R}, \mathbb{S}$ *and* $\mathbb{T}$*, we have*

$$(\mathbb{R} \circ \mathbb{S}) \circ \mathbb{T} = \mathbb{R} \circ (\mathbb{S} \circ \mathbb{T})$$

We state and prove this central result for any class relations.

Proof. We have two directions:

$\rightarrow$: Fix x and y and let $x(\mathbb{R} \circ \mathbb{S}) \circ \mathbb{T}y$.

Then, for some z, we have $x(\mathbb{R} \circ \mathbb{S})z\mathbb{T}y$ and hence for some w, the above becomes

$$x\mathbb{R}w\mathbb{S}z\mathbb{T}y \tag{1}$$

But $w\mathbb{S}z\mathbb{T}y$ means $w\mathbb{S} \circ \mathbb{T}y$, hence we rewrite (1) as

$$x\mathbb{R}w(\mathbb{S} \circ \mathbb{T})y$$

Finally, the above says $x\mathbb{R} \circ (\mathbb{S} \circ \mathbb{T})y$.

$\leftarrow$: Fix x and y and let $x\mathbb{R} \circ (\mathbb{S} \circ \mathbb{T})y$.

Then, for some z, we have $x\mathbb{R}z(\mathbb{S} \circ \mathbb{T})y$ and hence for some u, the above becomes

$$x\mathbb{R}z\mathbb{S}u\mathbb{T}y \tag{2}$$

But $x\mathbb{R}z\mathbb{S}u$ means $x\mathbb{R} \circ \mathbb{S}u$, hence we rewrite (2) as

$$x(\mathbb{R} \circ \mathbb{S})u\mathbb{T}y$$

Finally, the above says $x(\mathbb{R} \circ \mathbb{S}) \circ \mathbb{T}y$. □

The following is almost unnecessary, but offered for emphasis:

3.1.21 Corollary. *If R, S and T are (set) relations, all on some set A,*[†] *then "$R \circ S \circ T$" has a meaning independent of how brackets are inserted.*

The corollary allows us to just omit brackets in a chain of compositions, even longer that the above. It also leads to the definition of relational exponentiation, below:

[†] Recall that "R is on a set A" means $R \subseteq A^2$, which is the same as $R : A \rightarrow A$.

3.1.22 Definition. (Powers of a binary relation) Let R be a (set) relation. We define R^n, for $n > 0$, as

$$\underbrace{R \circ R \circ \cdots \circ R}_{n\ R} \tag{1}$$

Note that the resulting relation in (1) is independent of how brackets are inserted (3.1.21).

If moreover we have defined R to be on a set A, then we also define the 0-th power: R^0 stands for Δ_A or $\mathbf{1}_A$. □

3.1.23 Exercise. Let R be a relation on A. Then for all $n \geq 0$, R^n is a set.

Hint. You do not need to do induction. A "and so on" argument will be all right. □

3.1.24 Example. Let $R = \{(1,2),(2,3)\}$. What is R^2?

Well, when can we have xR^2y? Precisely if/when we can find x, y, z that satisfy $xRzRy$. The values $x = 1$, $y = 3$ and $z = 2$ are the *only ones* that satisfy $xRzRy$.

Thus $1R^23$, or $(1,3) \in R^2$. We conclude $R^2 = \{(1,3)\}$ by the "only ones" above. □

3.1.25 Exercise. Show that if for a a relation R we know that $R^2 \subseteq R$, then R is transitive and conversely. □

3.1.1. Transitive closure

3.1.26 Definition. (Transitive closure of R) $\underline{\text{A}}$ *transitive closure* of a relation R —if it exists— is the $\subseteq$-*smallest* transitive T that contains R as a subset.

More precisely,

1. T is transitive, and $R \subseteq T$.
2. If S is also transitive and $R \subseteq S$, then $T \subseteq S$. This makes the term "$\subseteq$-*smallest*" precise. □

Note that we hedged twice in the definition, because at this point we do not know yet:

- If every relation has a transitive closure; hence the "if it exists".
- We do not know if it is unique, hence the emphasised indefinite article "$\underline{\text{A}}$".

3.1.27 Remark. Uniqueness can be settled immediately *from the definition above*: Suppose T and T' fulfil Definition 3.1.26, that is,

1. $R \subseteq T$

   and

2. $R \subseteq T'$

since both are closures. But now think of T as a closure and T' as the "S" of 3.1.26 (it includes R all right!)

Hence $T \subseteq T'$.

Now reverse the role playing and think of T' as a closure, while T plays the role of "S". We get $T' \subseteq T$. Hence, $T = T'$. □

3.1.28 Definition. The unique transitive closure, if it exists, is denoted by R^+. □

3.1.29 Exercise. If R is transitive, then R^+ exists. In fact, $R^+ = R$. □

The above exercise is hardly exciting, but learning that R^+ exists for *every* R and also learning how to "compute" R^+ *is* exciting. We do this next.

3.1.30 Lemma. *Given a (set) relation R. Then $\bigcup_{n=1}^{\infty} R^n$ is a transitive (set) relation.*

Proof. We have two things to do.

1. $\bigcup_{n=1}^{\infty} R^n$ is a set.
2. $\bigcup_{n=1}^{\infty} R^n$ is a transitive relation.

Proof of **1.** Note that all positive powers of R, R^{n+1}, for $n \geq 0$, are sets. Indeed, they all are subsets of the same set!

Here is why:

Firstly, $R \subseteq \mathrm{dom}(R) \times \mathrm{ran}(R)$ by Definition 3.1.4.

Let now $n > 0$: We have

$$R^{n+1} = \overbrace{R \circ R \circ \ldots \circ R}^{n+1} = \overbrace{R \circ R \circ \ldots \circ R}^{n} \circ R = R^n \circ R$$

similarly, observing that

$$\overbrace{R \circ R \circ \ldots \circ R}^{n+1} = R \circ \overbrace{R \circ \ldots \circ R}^{n} = R \circ R^n$$

we have $R^{n+1} = R \circ R^n$. Thus, we established

$$R^{n+1} = R \circ R^n \tag{1}$$

and

$$R^{n+1} = R^n \circ R \tag{2}$$

Applying 3.1.18 to (1) we get

$$R^{n+1} \subseteq \mathrm{dom}(R) \times \ldots \tag{$1'$}$$

and applying 3.1.18 to (2) we get

$$R^{n+1} \subseteq \ldots \times \operatorname{ran}(R) \tag{2'}$$

Thus

$$R^{n+1} \subseteq \operatorname{dom}(R) \times \operatorname{ran}(R)$$

for $n \geq 0$.

So

$$X \in \mathbb{F} = \{R^i : i = 1, 2, 3, \ldots\} \rightarrow X \subseteq \operatorname{dom}(R) \times \operatorname{ran}(R) \tag{3}$$

Thus,

$$\bigcup_{i=1}^{\infty} R^i \overset{2.4.20}{=} \bigcup \mathbb{F} \subseteq \operatorname{dom}(R) \times \operatorname{ran}(R)$$

because

$$x \bigcup_{i=1}^{\infty} R^i y \Longrightarrow (x, y) \in \bigcup_{i=1}^{\infty} R^i \Longrightarrow (x, y) \in R^i, \text{ for some } i$$
$$\Longrightarrow (x, y) \in \operatorname{dom}(R) \times \operatorname{ran}(R)$$

hence we are done by 2.3.5 since $\operatorname{dom}(R) \times \operatorname{ran}(R)$ is a set.

Proof of **2.** Of course, $\bigcup_{i=1}^{\infty} R^i$ is a set (by part 1) *relation* since trivially it is a set of ordered pairs.

Next, let

$$x \bigcup_{i=1}^{\infty} R^i \, y \bigcup_{i=1}^{\infty} R^i \, z$$

Thus for some n and m we have

$$x \, R^n \, y \, R^m \, z$$

this says the same thing as

$$x \overbrace{R \circ R \circ \cdots R}^{n} \, y \, \overbrace{R \circ R \circ \cdots R}^{m} \, z$$

or

$$x \overbrace{R \circ R \circ \cdots R}^{n} \circ \overbrace{R \circ R \circ \cdots R}^{m} \, z$$

or

$$x \overbrace{R \circ R \circ \cdots R}^{n+m} \, z$$

that is,

$$x \bigcup_{i=1}^{\infty} R^i \, z$$

□

3.1.31 Remark. Why all this work for Part 1 of the proof above? Why not just *use* 2.4.20 right away? Because 2.4.20 offers *only notation* once we *know* that

$$\mathbb{F} = \{A_0, A_1, A_2, A_3, \ldots\} \tag{3}$$

is a set! Cf. "Suppose the family of sets Q is a set of sets", the opening statement in the passage 2.4.20 on *notation*.

Here we do *not know* (yet) if every family of sets like (3) is indeed a set —but in *this* case it turns out that we *do not care* because *every* member of $\mathbb{F} = \{R^i : i = 1, 2, 3, \ldots\}$ is included (as a subset) in $\mathrm{dom}(R) \times \mathrm{ran}(R)$ (a set), which allows us to sidestep the issue!

Whether *every* family of *sets* like $\mathbb{F}$ in (3) is a set will be answered affirmatively in 3.1.40. For now note that we cannot recklessly say that after *any* sequence of construction by stages there is a stage after all those stages. Why? Well, take *all* the objects in set theory. Each is given outright (atom; stage 0) or is constructed at some stage (set). If we could *prove* there is a stage after all these stages then we could also *prove* that $\mathbb{U}$ is a set, a claim we refuted with two methods so far! □

Since $R \subseteq \bigcup_{i=1}^{\infty} R^i$ due to $R = R^1$, all that remains to show is that $\bigcup_{i=1}^{\infty} R^i$ is a transitive closure of R is to show that

3.1.32 Lemma. *If $R \subseteq S$ and S is transitive, then $\bigcup_{i=1}^{\infty} R^i \subseteq S$.*

Proof. I will just show that for all $n \geq 1$, $R^n \subseteq S$. OK, $R \subseteq S$ is our assumption, thus $R^1 \subseteq S$ is true.

For $R^2 \subseteq S$ let xR^2y, thus (for some z), $xRzRy$ hence $xSzSy$. As S is transitive, the latter gives xSy. Done.

For $R^3 \subseteq S$ let xR^3y, thus (for some z), xR^2zRy hence $xSzSy$. As S is transitive, the latter gives xSy. Done.

You see the pattern: Pretend we proved up to n (fixed but unspecified) and we want to prove for $n+1$ (using the same value, as in our pretense, for n).

$$\text{So, we have } R^n \subseteq S. \tag{1}$$

Thus,

$$xR^{n+1}y \Longleftrightarrow xR^n \circ Ry \Longleftrightarrow xR^n zRy \text{ (some } z \text{)} \overset{(1)}{\Longrightarrow} xSzSy \Longrightarrow xSy \text{ (} S \text{ transitive)}$$

□

We have proved:

3.1.33 Theorem. (The transitive closure exists) *For any relation R, its transitive closure R^+ exists and is unique. We have that $R^+ = \bigcup_{i=1}^{\infty} R^i$.*

An interesting corollary that will lend a computational flavour to 3.1.33 is the following.

3.1.34 Corollary. *If R is on the set $\{a_1, a_2, \ldots, a_n\}$ where, for $i = 1, \ldots, n$, the a_i are distinct, then $R^+ = \bigcup_{i=1}^{n} R^i$.*

Proof. By 3.1.33, all we have to do is prove

$$\bigcup_{i=1}^{\infty} R^i \subseteq \bigcup_{i=1}^{n} R^i \tag{1}$$

since the $\supseteq$ part is obvious.

So let $x \bigcup_{i=1}^{\infty} R^i y$. This means that

$$xR^q y, \text{ for some } q \geq 1 \tag{2}$$

Thus, I have two cases for (2):

Case 1. $q \leq n$. Then $x \bigcup_{i=1}^{n} R^i y$ since $R^q \subseteq \bigcup_{i=1}^{n} R^i$, R^q being one of the "R^i" with i in the $1 \leq i \leq n$ range.

Case 2. $q > n$. In this case I will show that there is also a $k \leq n$ such that $xR^k y$, which sends me back to the "easy **Case 1**".

Well, if there is one $q > n$ that satisfies (2) there are probably more. Let us pretend that our q is *the smallest* $> n$ that gives us (2).

Wait! Why is there a *smallest* q such that

$$xR^q y \text{ and } q > n\,? \tag{3}$$

Because among those "q" that fit (3)[†] imagine we fix attention to one such.

Now, if it is not the smallest such, then go down to the *next smaller* one that still satisfies (3), call it q'.

Now go down to the next smaller, $q'' > n$, if q' is not smallest.

Continue like this. Can I do this forever? That is, can we have the following?

$$n < \ldots < q^{(k)\dagger} < \ldots < q''' < \ldots < q'' < q' < q$$

If yes, then I will have an infinite "descending" chain of distinct numbers between q and n.

Absurd!

Back to the proof. So let the q we are working with be the smallest that satisfies (3). Then we have the configuration

$$xRz_1Rz_2Rz_3 \ldots \boxed{z_iRz_{i+1} \ldots} z_rRz_{r+1} \ldots z_{q-1}Ry \tag{4}$$

[†] There is at least one, else we would **not** be in **Case 2**.

[†] By "$q^{(n)}$" I mean q with k primes.

The above accounts for q copies of R as needed for

$$R^q = \overbrace{R \circ \ldots R}^{q\ R}$$

Now the sequence

$$z_1, z_2, z_3 \ldots z_i, z_{i+1}, \ldots z_r, z_{r+1}, \ldots, z_{q-1}, y$$

in (4) above contains $q > n$ members. As they all come from A, **not all are distinct**. So let $z_i = z_r$ (the z_r could be as late in the sequence as y, i.e., equal to y).

Now omit the boxed part in (4). We obtain

$$x R z_1 R z_2 R z_3 \ldots \underset{\underset{z_i}{\|}}{z_r} R z_{r+1} \ldots z_{q-1} R y \tag{5}$$

which contains at least **one** "R" less than the sequence (4) does —the entry "$z_i R z_{i+1}$" (and everything else in the "..." part) was removed. That is, (5) states

$$x R^{q'} y$$

with $q' < q$. Since the q in (3) was *smallest* $> n$, we *must have* $q' \leq n$ which sends us to **Case 1** and we are done. □

3.1.2. Equivalence relations

Equivalence relations must be on some set A, since we require reflexivity. T hey play a significant role in many branches of mathematics and even in computer science. For example, the minimisation process of finite automata (a topic that we will not cover) relies on the concept of equivalence relations.

3.1.35 Definition. A relation R on A is an equivalence relation, provided it is all of

1. Reflexive
2. Symmetric
3. Transitive □

An equivalence relation on A has the effect, *intuitively*, of "grouping" elements that we view as *interchangeable in their roles*, or "equivalent", into so-called (see Definition 3.1.38 below) "*equivalence classes*" —kind of mathematical clubs!

Why is this intuition *not* applicable to arbitrary relations? There are a few reasons:

- First, not all relations are symmetric, so if element a of A starts up a "club" of "peers" with respect to a (non symmetric) relation R, then a will welcome b in the group as soon as aRb holds. Now since, *conceivably*, bRa may be false, b would *not* welcome a in the club ***it*** belongs! The two groups/clubs would be different! Now that is contrary to the *intuitive* meaning of "club membership" (equivalence) according to which we would like a and b to be indistinguishable, hence club-mates.

  So we have adopted *symmetry* in 3.1.35 for good reason. Is it enough?

- Do all symmetric relations "group" related elements in a way we would intuitively call "equivalence"? NO.

  Consider the symmetric relation $\neq$ on $A = \{(1,2),(2,1)\}$. If it behaved like club membership, then $a \neq b$ and $b \neq c$ would imply that all three a and c belong to the same "club" as b is. In particular, from $1 \neq 2$ and $2 \neq 1$ we expect $1 \neq 1$ (and $2 \neq 2$), which we do *NOT* have. "$\neq$" is not transitive.

  $1 = 1$ says do *not* put 1 in the same club as 1; they are not peers (to be peers requires $1 \neq 1$). But this is contrary to intuition as it says that 1 must be clubless.

  The problem is that $\neq$ is not transitive.

  So we have adopted transitivity in Definition 3.1.35 for good reason!

- This hinges on the previous bullet:

  What do we need *reflexivity* for? Well, without it we would have "clubless" elements (of A), i.e., elements which belong to no clubs at all, and this is undesirable intuitively.

  For example, $R = \{(1,2),(2,1),(1,1),(2,2)\}$ is symmetric and transitive on $A = \{1,2,3\}$, but is not reflexive ($(3,3)$ is missing). We have exactly one club, $\{1,2\}$, and 3 belongs to no club.

  We fix this by adding $(3,3)$ to R —making it reflexive— so that 3 belongs to the club $\{3\}$.

3.1.36 Example. The following are equivalence relations

- $\{(1,1)\}$ on $A = \{1\}$.
- $=$ (or $\mathbf{1}_A$ or Δ_A) on A.
- Let $A = \{1,2,3\}$. Then $R = \{(1,2),(1,3),(2,3),(2,1),(3,1),(3,2),(1,1),(2,2),(3,3)\}$ is an equivalence relation on A.
- $\mathbb{N}^2$ is an equivalence relation on $\mathbb{N}$. □

Here is a longish, more sophisticated example, that is central in number theory. We will have another instalment of it after a few definitions and results.

3.1.37 Example. (Congruences) Fix an $m \geq 2$. We define the relation $\equiv_m$on $\mathbb{Z}$ by

$$x \equiv_m y \text{ iff } m \,|\, (x - y)$$

Recall that "$|$" is the "divides with zero remainder" relation. We verify the required properties for $\equiv_m$ to be an equivalence relation.

A notation that is very widespread in the literature is to split the symbol "$\equiv_m$" into two and write

$$x \equiv y \pmod{m} \text{ instead of } x \equiv_m y$$

"$x \equiv y \pmod{m}$" and $x \equiv_m y$ are read "x is *congruent* to y *modulo* m (or just 'mod m')". Thus "$\equiv_m$" is the congruence (mod m) short symbol, while "$\equiv \ldots$ (mod m)" is the long two-piece symbol. *We will be using the short symbol.*

1. Reflexivity: Indeed, $m \,|\, (x - x)$, hence $x \equiv_m x$.

2. Symmetry: Clearly, if $m \,|\, (x - y)$, then $m \,|\, (y - x)$. I translate: If $x \equiv_m y$, then $y \equiv_m x$.

3. Transitivity: Let $m \,|\, (x - y)$ and $m \,|\, (y - z)$. The first says that, for some k, $x - y = km$. Similarly the second says, for some n, $y - z = nm$. Thus, adding these two equations I get $x - z = (k + n)m$, that is, $m \,|\, (x - z)$. I translate: If $x \equiv_m y$ and $y \equiv_m z$, then also $x \equiv_m z$. □

3.1.38 Definition. (Equivalence classes) Given an equivalence relation R on A. The *equivalence class* of an element $x \in A$ is $\{y \in A : xRy\}$. We use the symbol $[x]_R$, or just $[x]$ if R is understood, for the equivalence class.

3.1.39 Remark. Suppose an equivalence relation R on A is given.

By reflexivity, xRx, for any x. Thus $x \in [x]_R$, hence all equivalence classes are nonempty. □

Be careful to distinguish the brackets $\{\ldots\}$ from these $[\ldots]$. It is NOT a priori obvious that $x \in [x]_R$ until you look at the definition 3.1.38! $[x]_R \neq \{x\}$!!

The symbol A/R denotes the *quotient* class of A with respect to R, that is,

$$A/R \overset{Def}{=} \{[x]_P : x \in A\}$$

□

This is the time to introduce "**Principle 3**"[†] of set formation.

3.1.40 Remark. (Principle 3) Suppose that the *class* family of sets $\mathbb{F}$ *is indexed* by some (or all) members of a *set* A. Then $\mathbb{F}$ is a set.

Being *indexed* by (some) members of a set A means that, for every $X \in \mathbb{F}$, we have attached to it as "*label(s)*" (often depicted as a subscript/superscript)

[†] This is the last Principle, I promise!

some member(s) of A.

We **must** ensure that once a label is used it is *NOT used again* for another (or the same) $X \in \mathbb{F}$.

Thus, if $\mathbb{F} = \{A, B, C\}$, then $\{A_1, B_{13,19,0}, C_{42}\}$ is a valid labelling with members from $\mathbb{N}$.[†]

$\{A_{1,13}, B_{13}, C_{19}\}$ is not correctly labelled (same label twice), the labelling of $\{A_{1,42}, B_{13}, C\}$ is also invalid (C was not labelled): We can label a set of $\mathbb{F}$ with many labels, but we *may NOT use the same label twice* to label two (or the same) sets of $\mathbb{F}$ and *may NOT leave any set of* $\mathbb{F}$ *unlabelled*.

Note that in 3.1.38 we have labelled every $X \in A/R$ by a member of A by virtue of the fact that any X is an $[a]_R$ We can use a or any (or all) $x \in [a]_R$ to label X.

Two things:

1. The presence of a valid (correct) labelling from a *set* A ensures that the *labelled class family* is a *set* as it *has no more members* than the *set* of labels (I can spend many —or even all— of available labels on *one* set of $\mathbb{F}$, but I may not reuse a label, so I have *at least as many labels as there are members in* $\mathbb{F}$.

   Thus $\mathbb{F}$ is as "small" as a *set*, and thus a set itself. Some people call Principle 3 the **size limitation doctrine**.[‡]

2. Why can't I use the Principles 0–2 to argue that $\mathbb{F}$, labelled by A, is a set? Well, because these principles are notorious in not telling me when a stage exists after *infinitely many stages of construction* that I might have if, say, I were to build one set for each natural number:

   $$A_0, A_1, \ldots, A_n, \ldots$$

   Say the nature of *each* A_i is such that after each A_{i+1} is built at stage Σ_{i+1} that is astronomically later than the stage Σ_i at which A_i was built.

   Thus we get an infinite sequence of stages, wildly apart! How can I justify —just from Principles 0-2— the existence of a stage Σ that is *after* all the Σ_i, in order to build the class $\{A_0, A_1, \ldots, A_n, \ldots, \}$ as a *set*?

□

We can now state the obvious:

3.1.41 Theorem. *A/R is a set for any set A and equivalence relation R on A.*

[†] B has three labels attached to it.

[‡] Researchers on the foundations of set theory felt that paradoxes occurred in connection with enormous classes.

Proof. A provides labels for all members of A/R. Now invoke Principle 3. □

3.1.42 Lemma. *Let P be an equivalence relation on A. Then $[x] = [y]$ iff xPy —where we have omitted the subscript $_P$ from the $[\ldots]$-notation.*

Proof. $(\rightarrow)$ part. By reflexivity, $x \in [x]$ (3.1.39). The assumption then yields $x \in [y]$ and therefore yPx by 3.1.38. Symmetry gives us xPy now.
$(\leftarrow)$ part. Let $z \in [x]$. Then xPz. The assumption yields yRx (by symmetry), thus, transitivity yields $y\mathbb{P}z$. That is, $z \in [y]$, proving

$$[x] \subseteq [y]$$

By swapping letters we have proved above that yPx implies $[y] \subseteq [x]$. Now (by symmetry) our original assumption, namely xPy, implies yPx, hence also $[y] \subseteq [x]$. All in all, $[x] = [y]$. □

3.1.43 Lemma. *Let R be an equivalence relation on A. Then*

(i) $[x] \neq \emptyset$*, for all* $x \in A$*.*

(ii) $[x] \cap [y] \neq \emptyset$ *implies* $[x] = [y]$*, for all* x, y *in* A*.*

(iii) $\bigcup_{x \in A} [x] = A$*.*

Proof.

(i) 3.1.39.

(ii) Let $z \in [x] \cap [y]$. Then xRz and yRz, therefore xRz and zRy (the latter by symmetry) hence xRy (transitivity). Thus, $[x] = [y]$ by Lemma 3.1.42.

(iii) The $\subseteq$-part is obvious from $[x] \subseteq A$. The $\supseteq$-part follows from $\bigcup_{x \in A} \{x\} = A$ and $\{x\} \subseteq [x]$. □

The properties *(i)*–*(iii)* are characteristic of the notion of a *partition of a set.*

3.1.44 Definition. (Partitions) Let F be a family of subsets of A. It is a *partition of* A iff all of the following hold:

(i) For all $X \in F$ we have that $X \neq \emptyset$.

(ii) If $\{X, Y\} \subseteq F$ and $X \cap Y \neq \emptyset$, then $X = Y$.

(iii) $\bigcup F = A$. □

3.1.45 Remark. Often a partition F is given as an indexed family of sets denoted by $(F_a)_{a \in I}$, where I is the indexing set.

Less informatively we may write $(F_a)_{a \in I}$ as

$$\{F_a, F_b, F_c, \ldots\}$$

where the F_a are the $X, Y, \ldots$ of the definition above. □

There is a natural affinity between equivalence relations and partitions on a set A. In fact,

3.1.46 Theorem. *Given a partition F on a set A. This leads to the definition of an equivalence relation P whose equivalence classes are precisely the sets of the partition, that is $F = A/P$.*

Proof. First we define P:

$$xPy \overset{Def}{\text{iff}} (\exists X \in F)\{x, y\} \subseteq X \tag{1}$$

Observe that

(i) P is reflexive: Take any $x \in A$. By 3.1.44(iii), there is an $X \in F$ such that $x \in X$, hence $\{x, x\} \subseteq X$. Thus xPx.

(ii) P is, trivially, symmetric since there is no order in $\{x, y\}$.

(iii) P is transitive: Indeed, let $xPyPz$. Then $\{x, y\} \subseteq X$ and $\{y, z\} \subseteq Y$ for some X, Y in F.

Thus, $y \in X \cap Y$ hence $X = Y$ by 3.1.44(ii). Hence $\{x, z\} \subseteq X$, therefore xPz.

So P is an equivalence relation. Let us compare its equivalence classes with the various $X \in F$.

Now $[x]_P$ (denoted without the subscript $_P$ in the remaining proof) is

$$\{y : xPy\} \tag{2}$$

Let us compare $[x]$ with the unique $X \in F$ that contains x —why unique? By 3.1.44(ii). Thus,

$$y \in [x] \overset{(2)}{\Longleftrightarrow} xPy \overset{(1)}{\Longleftrightarrow} x \in X \land y \in X \overset{x\in X \text{ is } \mathbf{t}}{\Longleftrightarrow} y \in X$$

Thus $[x] = X$. □

3.1.47 Example. (Another look at congruences) Euclid's theorem for the division of integers states:

If $a \in \mathbb{Z}$ and $0 < m \in \mathbb{Z}$, then *there are* <u>unique</u> q and r such that

$$a = mq + r \text{ and } 0 \leq r < m \tag{1}$$

There are many proofs, but here is one: The set

$$T = \{x : 0 \leq x = a - mz, \text{ for some z}\}$$

is not empty. For example, if $a > 0$, then take $z = 0$ to obtain $x = a > 0$ in T. If $a = 0$, then take $z = 0$ to obtain $x = 0$. Finally, if $a < 0$, then take $z = -2|a|$[†] to obtain $x = -|a| + 2m|a| = |a|(2m - 1) > 0$. Since $m \geq 1$ we have $2m \geq 2$.

[†] Absolute value.

Let then r be the *smallest* $x \geq 0$ in T. If there is one x that works (as we just showed), then possibly there are more. BUT we *cannot* have an infinite descending sequence of nonnegative integers

$$\ldots < x''' < x'' < x' < x$$

There are just $x+1$ numbers from 0 to x inclusive! *So a smallest x that works one exists.*

The *corresponding* "z" to the smallest $x = r$ let us call q. So we have

$$a = mq + r$$

Can $r \geq m$? If so, them write $r = k + m$, where $k = r - m \geq 0$ and $k < r$. I got

$$a = m(q+1) + k$$

As $k < r$ I have contradicted the minimality of r.

This proves that $r < m$ (the $r \geq 0$ is trivial; why?)

We have proved *existence of at least one pair* q and r that works for (1). How about uniqueness? Well, the worst thing that can happen is to have two representations (1). Here is another:

$$a = mq' + r' \text{ and } 0 \leq r' < m \tag{2}$$

As both r and r' are $< m$, their "distance" (absolute difference) is also $< m$, so from (1) and (2) we get

$$m|q - q'| = |r - r'| \tag{3}$$

This cannot be unless $q = q'$ (in which case $r = r'$, therefore uniqueness is proved).

Wait: Why "it cannot be" if $q \neq q'$? Because then $|q - q'| \geq 1$ thus the lhs of "=" in (3) is $\geq m$ but the rhs is $< m$.

We now take a deep breath!

Now, back to congruences! The above was just a preamble!

Fix an $m > 1$ and consider the congruences $x \equiv_m y$. What are the equivalence classes?

Better question is what representative members are convenient to use for each such class? Given that $a \equiv_m r$ by (1), and using Lemma 3.1.42 we have $[a]_m = [r]_m$.

r is a far better representative than a for the class $[a]_m$ as it is "normalised".

Thus, we have just m equivalence classes $[0], [1], \ldots, [m-1]$.

Wait! Are they distinct? Yes! Since $[i] = [j]$ is the same as $i \equiv_m j$ (3.1.42) and, since $0 < |i - j| < m$, m cannot divide $i - j$ with 0 remainder, we cannot have $[i] = [j]$.

OK. How about missing some? We are not, for any a is uniquely expressible as $a = m \cdot q + r$, where $0 \leq r < m$. Since $m \mid (a - r)$, we have $a \equiv_m r$, i.e., (by 3.1.38) $a \in [r]$. □

3.1.48 Example. (A practical example) Say, I chose $m = 5$. Where does $a = -110987$ belong? I.e., in which $[\ldots]_5$ class out of $[0]_5, [1]_5, [2]_5, [3]_5, [4]_5$?

Well, let's do primary-school-learnt long division of $-a$ divided by 5 and find quotient q and remainder r. We find, in this case, $q = 22197$ and $r = 2$. These satisfy

$$-a = 22197 \times 5 + 2$$

Thus,

$$a = -22197 \times 5 - 2 \tag{1}$$

(1) can be rephrased as

$$a \equiv_5 -2 \tag{2}$$

But easily we check that $-2 \equiv_5 3$ (since $-2 - 3 = 5$). Thus, by transitivity of $\equiv_5$,

$$a \in [-2]_5 = [3]_5$$

□

3.1.49 Exercise. Can you now <u>*easily*</u> write the same a above as

$$a = Q \times 5 + R, \text{ with } 0 \leq R < 5?$$

Show all your work. □

3.1.3. Partial orders

This subsection introduces one of the most important kind of binary relations in set theory and mathematics in general: The *partial order* relations.

We will find the following definitions and notation useful in this subsection:

3.1.50 Definition. (Converse or inverse relation of $\mathbb{P}$) For any relation $\mathbb{P}$, the symbol $\mathbb{P}^{-1}$ stands for the *converse* or *inverse* relation of $\mathbb{P}$ and is defined as

$$\mathbb{P}^{-1} = \{(x, y) : y\mathbb{P}x\} \tag{1}$$

$\underline{x\mathbb{P}^{-1}y \text{ iff } y\mathbb{P}x}$ is an equivalence that says exactly what (1) does. □

3.1.51 Definition. ("$(a)\mathbb{P}$" notation) For any relation $\mathbb{P}$ we write "$(a)\mathbb{P}$" to indicate the *class* —might fail to be a set— of *all outputs* of $\mathbb{P}$ *on (caused by) input a*. That is,

$$(a)\mathbb{P} \overset{Def}{=} \{y : a\,\mathbb{P}\,y\}$$

If $(a)\mathbb{P} = \emptyset$, then $\mathbb{P}$ is *undefined* at a —that is, $\underline{a \notin \mathrm{dom}(\mathbb{P})}$. The underlined statement is often denoted simply by "$\underline{(a)\mathbb{P}\uparrow}$" and is naturally read as "$\mathbb{P}$ is *undefined* at a".

If $(a)\mathbb{P} \neq \emptyset$, then $\mathbb{P}$ is *defined* at a —that is, $\underline{a \in \mathrm{dom}(\mathbb{P})}$. The underlined statement is often denoted simply by "$\underline{(a)\mathbb{P}\downarrow}$" and is naturally read as "$\mathbb{P}$ is *defined* at a". □

3.1.52 Exercise. Give an example of a specific relation $\mathbb{P}$ and <u>one</u> specific object (set or atom) a such that $(a)\mathbb{P}$ is a proper class. □

3.1.53 Remark. We note that for any $\mathbb{P}$ and a,

$$(a)\mathbb{P}^{-1} = \{y : a\mathbb{P}^{-1}y\} = \{y : y\mathbb{P}a\}$$

Thus,

$$(a)\mathbb{P}^{-1}\uparrow \text{ iff } a \notin \mathrm{ran}(\mathbb{P})$$

and

$$(a)\mathbb{P}^{-1}\downarrow \text{ iff } a \in \mathrm{ran}(\mathbb{P})$$

□

3.1.54 Definition. (Partial order) A relation $\mathbb{P}$ is called a *partial order* or just an *order*, iff it is

(1) *irreflexive* (i.e., $x\mathbb{P}y \to x \neq y$ for all x, y), and

(2) *transitive*.

It is emphasised that in the interest of generality —for much of this subsection (until we say otherwise)— $\mathbb{P}$ need not be a set.

Some people call this a *strict order* as it imitates the "<" on, say, the natural numbers. □

3.1.55 Remark. (1) We will normally use the symbol "$<$" in *the abstract setting* to denote *any* unspecified order $\mathbb{P}$, and it will be pronounced "less than".

It is **hoped** that the context will not allow confusion with any concrete use of the symbol $<$ on numbers (say, on the reals, natural numbers, etc.).

(2) If the order $<$ is a subclass of $\mathbb{A} \times \mathbb{A}$ —i.e., it is $<: \mathbb{A} \to \mathbb{A}$— then we say that $<$ *is an order on* $\mathbb{A}$.
(3) Clearly, for any order $<$ and any class $\mathbb{B}$, $< \cap (\mathbb{B} \times \mathbb{B})$ is an order on $\mathbb{B}$. □

3.1.56 Exercise. How clearly? (re (3) above.) Give a simple, short proof. □

3.1.57 Example. The concrete "less than", $<$, on $\mathbb{N}$ is an order, but $\le$ is not (it is *not* irreflexive). The "greater than" relation, $>$, on $\mathbb{N}$ is also an order, but $\ge$ is not. Of course, $>=<^{-1}$.

In general, it is trivial to verify that $\mathbb{P}$ is an order iff $\mathbb{P}^{-1}$ is an order. *Exercise*! □

3.1.58 Example. $\emptyset$ is an order. Since for any $\mathbb{A}$, $\emptyset \subseteq \mathbb{A} \times \mathbb{A}$, $\emptyset$ is also an order *on* $\mathbb{A}$ for the arbitrary $\mathbb{A}$. □

3.1.59 Example. The relation $\in$ is irreflexive by the well known $A \notin A$, for all A. It is not transitive though. For example, if a is a set (or atom), then $a \in \{a\} \in \{\{a\}\}$ but $a \notin \{\{a\}\}$. *So it is not an order.*

Let $M = \Big\{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}\Big\}$. The relation $\varepsilon = \in \cap (M \times M)$ *is* transitive and irreflexive, hence it is an order (on M). *Verify*! □

3.1.60 Example. $\subset$ is an order, $\subseteq$—failing irreflexivity— is not. □

3.1.61 Example. Consider the order $\subset$ again. In this case we have **none** of $\{\emptyset\} \subset \{\{\emptyset\}\}$, $\{\{\emptyset\}\} \subset \{\emptyset\}$ or $\{\{\emptyset\}\} = \{\emptyset\}$. That is, $\{\emptyset\}$ and $\{\{\emptyset\}\}$ are *non comparable* items. This justifies the qualification *partial* for orders in general (Definition 3.1.66).

On the other hand, the "natural" $<$ on $\mathbb{N}$ is such that one of $x = y$, $x < y$, $y < x$ always holds for any x, y. That is, all (unordered) pairs x, y of $\mathbb{N}$ *are* comparable under $<$. This is a concrete example of a *total* order (see the "official definition" below: 3.1.67).

While *all* orders are "partial", some are total ($<$ above) and others are *nontotal* ($\subset$ above). □

3.1.62 Definition. Let $<$ be a partial order on $\mathbb{A}$. We set

$$\le \overset{Def}{=} \boldsymbol{\Delta}_{\mathbb{A}} \cup <$$

We pronounce $\le$ "less than or equal". $\boldsymbol{\Delta}_{\mathbb{A}} \cup >$ is denoted by $\ge$ and is pronounced "greater than or equal".

Let us call $\le$ a *reflexive order.* □

☡ (1) In plain English, given $<$ on $\mathbb{A}$, we define $x \le y$ to mean

$$x < y \lor \overbrace{x = y}^{\text{equality is } \Delta_{\mathbb{A}}}$$

for all x, y in $\mathbb{A}$.
(2) The definition of $\le$ depends on $\mathbb{A}$ due to the presence of $\boldsymbol{\Delta}_{\mathbb{A}}$. **There is no such dependency on a "reference" class in the case of** $<$.
(3) We remind ourselves once more here that the symbols $<$ and $\le$ —and their pronunciations— do *NOT* imply that we are talking about the specific ones on *numbers*. It is just a harmless (I hope) notational devise, but **unless said explicitly otherwise, "$<$" and "$\le$" are <u>any</u> orders**. ☡

3.1.63 Lemma. *For any* $<: \mathbb{A} \to \mathbb{A}$*, the associated relation* $\le$ *on* $\mathbb{A}$ *is* reflexive*,* antisymmetric *and* transitive*.*

Proof. (1) Reflexivity is trivial.

(2) For antisymmetry, let $x \le y$ and $y \le x$. If $x = y$ then we are done, so assume the remaining case $x \ne y$ (i.e., $(x, y) \notin \boldsymbol{\Delta}_{\mathbb{A}}$). Then the hypothesis becomes $x < y$ and $y < x$, therefore $x < x$ by transitivity, contradicting the irreflexivity of $<$.

(3) As for transitivity let $x \le y$ and $y \le z$.

(a) If $x = z$, then $x \le z$ (see the ☡-remark after 3.1.62) and we are done.

(b) The remaining case is $x \ne z$. Now, if it is $x = y$ or $y = z$ (but not both (why?)), then we are done again. So it remains to consider $x < y$ and $y < z$. By transitivity of $<$ we get $x < z$, hence $x \le z$, since $< \subseteq \le$. □

3.1.64 Lemma. *Let* $\mathbb{P}$ *on* $\mathbb{A}$ *be reflexive, antisymmetric and transitive.*
Then $\mathbb{P} - \boldsymbol{\Delta}_{\mathbb{A}}$ *is an order on* $\mathbb{A}$*.*

Proof. Since

$$\mathbb{P} - \boldsymbol{\Delta}_{\mathbb{A}} \subseteq \mathbb{P} \tag{1}$$

it is clear that $\mathbb{P} - \boldsymbol{\Delta}_{\mathbb{A}}$ is *on* $\mathbb{A}$. It is also clear that it is irreflexive. We only need verify that it is transitive.

So let

$$(x, y) \text{ and } (y, z) \text{ be in } \mathbb{P} - \boldsymbol{\Delta}_{\mathbb{A}} \tag{2}$$

By (1) (or (2))

$$(x, y) \text{ and } (y, z) \text{ are in } \mathbb{P} \tag{3}$$

hence

$$(x, z) \in \mathbb{P}$$

by transitivity of $\mathbb{P}$.

Can $(x, z) \in \boldsymbol{\Delta}_{\mathbb{A}}$, i.e., can $x = z$? No, for antisymmetry of $\mathbb{P}$ and (3) would imply $x = y$, i.e., $(x, y) \in \boldsymbol{\Delta}_{\mathbb{A}}$ contrary to (2).

So, $(x, z) \in \mathbb{P} - \boldsymbol{\Delta}_{\mathbb{A}}$. □

3.1.65 Remark. Often in the literature, but decreasingly so, it is the "reflexive order" $\leq : \mathbb{A} \to \mathbb{A}$ that is defined as a "partial order" by the requirements that it is *reflexive*, *antisymmetric* and *transitive*. Then $<$ is obtained as in Lemma 3.1.64, namely, as "$\leq - \boldsymbol{\Delta}_{\mathbb{A}}$". Lemmas 3.1.63 and 3.1.64 show that the two approaches are interchangeable, but the "modern" approach of Definition 3.1.54 avoids the nuisance of having to tie the notion of order to some particular "field" $\mathbb{A}$ (3.1.6).

For us "$\leq$" is the *derived* notion defined in 3.1.62. □

3.1.66 Definition. (PO Class) If $<$ is an order on a class $\mathbb{A}$, we call the *informal* pair $(\mathbb{A}, <)$[†] a *partially ordered class*, or *PO class*.

If $<$ is an order on a *set* A, we call the pair $(A, <)$ a *partially ordered set* or *PO set*. Often, if the order $<$ is understood as being on $\mathbb{A}$ or A, one says that "$\mathbb{A}$ is a PO class" or "A is a PO set" respectively. □

3.1.67 Definition. (Linear order) A relation $<$ on $\mathbb{A}$ is a *total* or *linear* order *on* $\mathbb{A}$ iff it is

(1) An order, and

(2) For any x, y in $\mathbb{A}$ one of $x = y$, $x < y$, $y < x$ holds —this is the so-called "*trichotomy*" property.

If $\mathbb{A}$ is a class, then the informal pair $(\mathbb{A}, <)$ is a *linearly ordered class* —for short, a *LO class*.

If $\mathbb{A}$ is a set, then the pair $(\mathbb{A}, <)$ is a *linearly ordered set* —for short, a *LO set*.

One often calls just $\mathbb{A}$ a LO class or LO set (as the case warrants) when $<$ is understood from the context. □

3.1.68 Example. The standard $< : \mathbb{N} \to \mathbb{N}$ is a total order, hence $(\mathbb{N}, <)$ is a LO set.

3.1.69 Definition. (Minimal and minimum elements) Let $<$ be an order and $\mathbb{A}$ some class.

We are not postulating that $<$ is on $\mathbb{A}$.

An element $a \in \mathbb{A}$ is *a $<$-minimal element in* $\mathbb{A}$, or *a $<$-minimal element of* $\mathbb{A}$, iff $\neg(\exists x \in \mathbb{A})x < a$ —in words, there is nothing below a in $\mathbb{A}$.

$m \in \mathbb{A}$ is *a $<$-minimum element in* $\mathbb{A}$ iff $(\forall x \in \mathbb{A})m \leq x$.

We also use the terminology minimal or minimum *with respect to* $<$, instead of $<$-minimal or $<$-minimum.

[†] Formally, $(\mathbb{A}, <)$ is *not* an ordered pair since $\mathbb{A}$ may be a proper class and we do not allow class *members* —e.g., in $\{\mathbb{A}, \{\mathbb{A}, <\}\}$— to be proper classes. We may think then of "$(\mathbb{A}, <)$" as *informal* notation that simply "ties" $\mathbb{A}$ and $<$ together. Alternatively, if we are really determined to have class pairs (we are not!), we can *define* pairing with proper classes as components, for example as $(\mathbb{A}, \mathbb{B}) =^{Def} (\mathbb{A} \times \{0\}) \cup (\mathbb{B} \times \{1\})$. For our part we will have no use for such formality, and will consider $(\mathbb{A}, <)$ in only the *informal* sense.

If $a \in \mathbb{A}$ is $>$-minimal in $\mathbb{A}$, that is $\neg(\exists x \in \mathbb{A})x > a$, we call a a *$<$-maximal* element in $\mathbb{A}$. Similarly, a $>$-minimum element is called a *$<$-maximum.*

If the order $<$ is understood, then the qualification "$<$-" is omitted. □

3.1.70 Remark. In particular, if a ($\in \mathbb{A}$) is *not* in the *field* $\mathrm{dom}(<) \cup \mathrm{ran}(<)$ (cf. 3.1.6) of $<$, then a is *both* $<$-minimal and $<$-maximal *in* $\mathbb{A}$. For example, $(\exists x \in \mathbb{A})x < a$ is false in this case since if, for some x, we have $x \in \mathbb{A}$ and also $x < a$, then $a \in \mathrm{ran}(<)$; impossible.

Because of the duality between the notions of minimal/maximal and minimum/maximum, we will mostly deal with the $<$-notions whose results can be trivially translated for the $>$-notions.

Note how the notation learnt from 3.1.51 and 3.1.50 and 3.1.53 can *simplify*

$$\neg(\exists x \in \mathbb{A})x < a \tag{1}$$

(1) says that *no x is in **both** $\mathbb{A}$ and $(a)>$*.[†]

That is, a is $<$-minimal in $\mathbb{A}$ iff

$$\mathbb{A} \cap (a) >= \emptyset \tag{2}$$

□

3.1.71 Example. 0 is *minimal*, also *minimum*, in $\mathbb{N}$ with respect to the natural ordering.

In $\mathbf{P}(\mathbb{N})$, $\emptyset$ is both $\subset$-minimal and $\subset$-minimum. On the other hand, all of $\{0\}, \{1\}, \{2\}$ are $\subset$-minimal in $\mathbf{P}(\mathbb{N}) - \{\emptyset\}$ but *none* are *$\subset$-minimum* in that set.

Observe from this last example that minimal elements in a class are *not* unique. □

3.1.72 Remark. (Hasse diagrams) There is a neat pictorial way to depict orders on finite sets known as "*Hasse diagrams*". To do so one creates a so-called "*graph*" of the finite PO set $(A, <)$ where $A = \{a_1, a_2, \ldots, a_n\}$.

How? The graph consists of n *nodes* —which are drawn as points— each labeled by one a_i. The graph also contains 0 or more *arrows* that connect nodes. These arrows are called *edges.*

When we depict an arbitrary R on a finite set like A we draw *one* arrow (edge) from a_i to a_j iff the two *relate*: $a_i R a_j$.

In Hasse diagrams for PO sets $(A, <)$ we are more selective: We say that b *covers* a iff $a < b$, but there is no c such that $a < c < b$. In a Hasse diagram we will

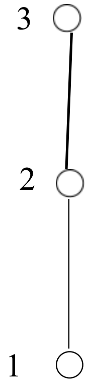
1. draw an edge from a_i to a_j iff a_j covers a_i.

2. by convention we will draw b higher than a on the page if b covers a.

[†] $(a) >= \{x : a > x\} = \{x : x < a\}$ (3.1.53).

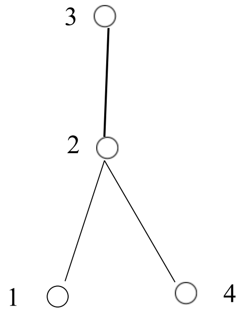
3. given the convention above, using "arrow-heads" is superfluous: our edges are plain line segments.

So, let us have $A = \{1, 2, 3\}$ and $<= \{(1, 2), (1, 3), (2, 3)\}$.



The above has a minimum (1) and a maximum (3) and is clearly a linear order.

A slightly more complex one is this $(A, <)$, where $A = \{1, 2, 3, 4\}$ and $<= \{(1, 2), (4, 2), (2, 3), (1, 3), (4, 3)\}$.



This one has a maximum (3), two minimal elements (1 and 4) but no minimum, and is not a linear order: 1 and 4 are not comparable. □

3.1.73 Lemma. *Given an order* $<$ *and a class* $\mathbb{A}$.
(1) *If* m *is a* minimum *in* $\mathbb{A}$*, then it is also* minimal.
(2) *If* m *is a* minimum *in* $\mathbb{A}$*, then it is unique.*

Proof. (1) Let m be mini*mum* in $\mathbb{A}$. Then

$$m \leq x, \text{ that is, } m = x \vee m < x \qquad (i)$$

for all $x \in \mathbb{A}$. Now, prove that there is no $x \in \mathbb{A}$ such that $x < m$.

OK, let us go by contradiction:

Let
$$\mathbb{A} \ni a < m \tag{ii}$$
By (i) I also have
$$m = a \vee m < a \tag{iii}$$
Now, by irreflexivity, (ii) rules out $a = m$. So, (iii) nets $m < a$. (ii) and (iii) and transitivity yield $a < a$; contradiction ($<$ is irreflexive). Done.

(2) Let m and n both be minima in $\mathbb{A}$. Then $m \le n$ (with m posing as minimum) and $n \le m$ (now n is so posing), hence $m = n$ by antisymmetry (Lemma 3.1.63). □

3.1.74 Example. Let m be $<$-minimal in $\mathbb{A}$.

Let us attempt to "show" that it is also $<$-minimum (this is, of course, doomed to fail due to 3.1.71 and 3.1.73(2) —but the "faulty proof" below is interesting):

By 3.1.69 we have that $\underline{\text{there is no } x \text{ in } \mathbb{A} \text{ such that } x < m}$.

Another way to say this is:

$\underline{\text{For all } x \in \mathbb{A}}$, I have the negation of "$x < m$", $\underline{\text{that is, I have } \neg x < m}$. (1)

But from "our previous math" (high school? university? Netflix?) $\neg x < m$ is equivalent to $m \le x$.

Thus (1) says $(\forall x \in \mathbb{A}) m \le x$, in other words, m is the minimum in $\mathbb{A}$.

Do you believe this? (Don't!) If the order is not total, then I can *fail to have* $\underline{\text{all three of}}$ $x < m, x = m, m < x$ and thus $\neg m < x$ and $x < m \vee x = m$ are *NOT* equivalent. See the counterexample to such expectation in 3.1.61 and also 3.1.71. □

3.1.75 Lemma. *If $<$ is a* linear *order on $\mathbb{A}$, then every minimal element is also minimum.*

Proof. The "false proof" of the previous example is valid under the present circumstances. □

The following type of relation has fundamental importance for set theory, and mathematics in general.

3.1.76 Definition.

1. An order $<$ satisfies the *minimal condition*, for short *it has MC*, iff *every* $\underline{\text{nonempty}}$ $\mathbb{A}$ has $<$-minimal elements.

2. If a *total* order $<: \mathbb{B} \to \mathbb{B}$ has MC, then it is called a *well-ordering*[†] *on* (or *of*) the class $\mathbb{B}$.

[†] The term "well-ordering" is ungrammatical, but it is *the* terminology established in the literature!

3. If $(\mathbb{B}, <)$ is a LO class (or set) with MC, then it is a *well-ordered class* (or set), or *WO class* (or WO set).

□

3.1.77 Remark.

What Definition 3.1.76 says in case 1. is —see (2) in 3.1.70— "*if, for some fixed order* $<$ *the following statement*

$$\emptyset \neq \mathbb{A} \to (\exists a \in \mathbb{A})\mathbb{A} \cap (a) >= \emptyset \tag{1}$$

is provable in set theory, for any $\mathbb{A}$*, then we say that* $<$ *has MC*".

The following observation is very important *for future reference*:

If $\mathbb{A}$ is given via a defining property $F(x)$, as

$$\mathbb{A} \overset{Def}{=} \{x : F(x)\}$$

then (1) translates —in terms of $F(x)$— into

$$(\exists a)F(a) \to (\exists a)\Big(F(a) \wedge \neg(\exists y)\big(y < a \wedge F(y)\big)\Big) \tag{2}$$

Conversely, for each formula $F(x)$ we get a class $\mathbb{A} = \{x : F(x)\}$ and thus —if $\mathbb{A}$ has MC with respect to $<$— we may express this fact as in (2) above.

3.2. Functions

At last! We consider here a special case of relations that we know them as "functions". Many of you know already that a function is a relation with some special properties.

Let's make this official:

3.2.1 Definition. A *function* R is a *single-valued* relation. That is, whenever we have both xRy and xRz, we will also have $y = z$.

It is traditional to use, generically, lower case letters from among f, g, h, k to denote functions but this is by no means a requirement. □

Another way of putting it, using the notation from 3.1.51, is: A relation R is a function iff $(a)R$ is either *empty* or contains *exactly one* element.

3.2.2 Example. The empty set is a relation of course, the empty set of pairs. It is also a function since

$$(x, y) \in \emptyset \land (x, z) \in \emptyset \rightarrow y = z$$

vacuously, by virtue of the left hand side of → being false. □

We now turn to notation and concepts specific to functions.

3.2.3 Definition. (Function-specific notations) Let f be a function. First off, the *concepts* of domain, range, and —in case of a function $f : A \rightarrow B$— total and onto *are inherited from that of relations without change.* Even the notations "aRb" and "$(a, b) \in R$" transfer over to functions. And now we have an annoying *difference* in notation:

It is $f(a)$ that *normally* denotes the set $\{y : afy\}$ *in the literature*, NOT $(a)f$ (compare with 3.1.51). "Normally" allows some to differ: Notably, [Kur63] writes "af" for functions and relations, omitting even the brackets around a.

The reason for the preferred notation "$f(a)$" for functions will become more obvious once we consider composition of *functions*.

Can I use "$(a)f$" for a relation f regardless of whether it is also a function? YES! But once I proved (or I was told) that it is a function I ought to prefer to write $f(a)$.

If b is such that afb or $(a, b) \in f$ and f is a function, then seeing that b is unique we have $f(a) = \{b\}$.

However we will write

$$f(a) = b$$

That is,

$$\underbrace{f(a) = b}_{\text{functional notation}} \quad \text{iff} \quad \underbrace{(a)f = \{b\}}_{\text{relational notation}}$$

The notation "$(a)R\downarrow$" meaning $a \in \mathrm{dom}(R)$ is inherited by functions but for the flipping of the "(a)" part. Thus

Inherited from 3.1.51, $f(a)\downarrow$ iff $a \in \mathrm{dom}(f)$, pronounced "f is defined at a".

and, similarly to the notation $(a)R\uparrow$, we have

Inherited from 3.1.51, $f(a)\uparrow$ iff $a \notin \mathrm{dom}(f)$, pronounced "f is *UN*defined at a".

The set of *all* outputs of a function, *when the inputs come from a particular set* X, is called the *image of* X *under* f and is denoted by $f[X]$. Thus,

$$f[X] \overset{Def}{=} \{f(x) : x \in X\} \tag{1}$$

Note that careless notation (e.g., in our text) like $f(X)$ will *not* do. This means the input *IS* X. If I want the inputs to be *from inside* X I must change the round brackets notation; I did.

Pause. So far we have been giving definitions regarding functions of *one* variable. Or have we?◀

Not really: We have already said that the multiple-input case is subsumed by our notation. If $f : A \to B$ and A is a set of n-tuples, then f is a function of "n-variables", essentially. The binary relation that is the alias of f contains pairs like $\big((\vec{x}_n), x_{n+1}\big)$. However, we usually abuse the notation $f\Big((\vec{x}_n)\Big)$ and write instead $f(\vec{x}_n)$, omitting the brackets of the n-tuple $(\vec{x}_n)$.

The *inverse image* of a set Y under a function is useful as well, that is, the set of *all* inputs that generate f-outputs exclusively in Y. It is denoted by $f^{-1}[Y]$ and is defined as

$$f^{-1}[Y] \overset{Def}{=} \{x : f(x) \in Y\} \tag{2}$$

□

3.2.4 Remark. Regarding, say, the definition of $f[X]$:

What if $f(a)\uparrow$*? How do you "collect" an* undefined value *into a set?*

Well, you don't. Both (1) and (2) have a rendering that is independent of the notation "$f(a)$".

Never forget that a function is no mystery; it is a relation and we have access to relational notation. Thus,

$$f[X] = \{y : (\exists x \in X)xfy\} \tag{1$'$}$$

$$f^{-1}[Y] = \{x : (\exists y \in Y)xfy\} \tag{2$'$}$$

□

3.2.5 Example. Thus, $f[\{a\}] = \{f(x) : x \in \{a\}\} = \{f(x) : x = a\} = \{f(a)\}$.

Let now $g = \{\langle 1,2\rangle, \langle\{1,2\},2\rangle, \langle 2,7\rangle\}$, clearly a function. Thus, $g(\{1,2\}) = 2$, but $g[\{1,2\}] = \{2,7\}$. Also, $g(5)\uparrow$ and thus $g[\{5\}] = \emptyset$.

On the other hand, $g^{-1}[\{2,7\}] = \{1,\{1,2\},2\}$ and $g^{-1}[\{2\}] = \{1,\{1,2\}\}$, while $g^{-1}[\{8\}] = \emptyset$ since no input causes output 8. □

When $f(a)\downarrow$, then $f(a) = f(a)$ as is naturally expected. What about when $f(a)\uparrow$? This begs a more general question that we settle as follows:

3.2.6 Remark. This is the first (and probably last) time that we will view an $(m+n+1)$-ary relation $R(z_1,\ldots,z_m,x,y_1,\ldots,y_n)$ as a *function* with input values entered into *all* the variables $z_1,\ldots,z_m,x,y_1,\ldots,y_n$ and output values belonging to the set $\{\mathbf{t},\mathbf{f}\}$.

Such a relation, as we explained when we introduced relations, is always total, no matter what the input. That is, *any* input $a_1,\ldots,a_m,b,c_1,\ldots,c_n$ either *appears* in the table of the relation, or it does *not*. In other words, $R(a_1,\ldots,a_m,b,c_1,\ldots,c_n)$ is precisely *one* of true or false; there is no "maybe" or "I do not know".

Given such an $(m+n+1)$-ary relation, a function f, and an input u for f,

when is $R(z_1,\ldots,z_m,f(u),y_1,\ldots,y_n)$ true, for any given $z_1,\ldots,z_m,u,y_1,\ldots,y_n$?

Well, what we are saying in the notation (in blue) above is that if $f(u) = w$, for some w, then $R(z_1,\ldots,z_m,w,y_1,\ldots,y_n)$ is true.

Thus,

$$R(z_1,\ldots,z_m,f(u),y_1,\ldots,y_n) \textbf{ iff } (\exists w)\Big(w = f(u) \wedge R(z_1,\ldots,z_m,w,y_1,\ldots,y_n)\Big) \qquad (3)$$

Note that the part "for some w, $w = f(u)$" in (3) entails that $f(u)\downarrow$, so that if *no such w exists* [the case where $f(u)\uparrow$], then the rhs of (3) is *false*; **not** undefined!

This convention is prevalent in the modern literature (cf. [Hin78, p.9]). Contrast with the convention in [Kle43], where, for example, an expression like $f(a) = g(b)$ [and even $f(a) = b$] is allowed to be undefined! □

3.2.7 Example. Thus, applying the above twice, where our "R" is $x = y$, we get that $f(a) = g(b)$ means $(\exists u)(\exists w)(u = f(a) \wedge w = g(b) \wedge u = w)$ which simplifies to $(\exists u)(u = f(a) \wedge u = g(b))$. In particular, $f(a) = g(b)$ entails that $f(a)\downarrow$ and $g(b)\downarrow$ as we noted above.

Furthermore, using $x \neq y$ as R we get that $f(a) \neq g(b)$ means $(\exists u)(\exists w)(u = f(a) \wedge w = g(b) \wedge u \neq w)$. Again, if $f(a) \neq g(b)$ is true, its meaning implies $f(a)\downarrow$ and $g(b)\downarrow$. □

3.2.8 Example. Let $g = \{\langle 1,2\rangle, \langle \{1,2\},2\rangle, \langle 2,7\rangle\}$. Then, $g(1) = g(\{1,2\})$ and $g(1) \neq g(2)$. □

3.2.9 Definition. A function f is 1-1 if for all x and y, $f(x) = f(y)$ implies $x = y$. □

Note that $f(x) = f(y)$ implies that $f(x)\downarrow$ and $f(y)\downarrow$ (3.2.6).

3.2.10 Example. $\{\langle 1,1\rangle\}$ and $\{\langle 1,1\rangle, \langle 2,7\rangle\}$ are 1-1. $\{\langle 1,0\rangle, \langle 2,0\rangle\}$ is not. $\emptyset$ is 1-1 vacuously. □

3.2.11 Exercise. Prove that if f is a 1-1 function, then the relation converse f^{-1} is a function (that is, single-valued). □

3.2.12 Definition. (1-1 Correspondence) A function $f : A \to B$ is called a *1-1 correspondence* iff it is all three: 1-1, total and onto.

Often we say that A and B are *in 1-1 correspondence* writing $A \sim B$, often omitting mention of the function that *is* the 1-1 correspondence. □

The terminology is derived from the fact that every element of A is paired with precisely one element of B and vice versa.

3.2.13 Exercise. Show that $\sim$ is a symmetric and transitive relation on sets. □

3.2.14 Remark. Composition of functions is inherited from the composition of relations. Thus, $f \circ g$ for two functions still means

$$x\, f \circ g\, y \text{ iff, for some } z,\ x\, f\, z\, g\, y \tag{1}$$

In particular,

$f \circ g$ is also a function. Indeed, if we have

$$x\, f \circ g\, y \text{ and } x\, f \circ g\, y'$$

then

$$\text{for some } z, x\, f\, z\, g\, y \tag{1}$$

and

$$\text{for some } w, x\, f\, w\, g\, y' \tag{2}$$

As f is a function, (1) and (2) give $z = w$. In turn, this (g is a function too!) gives $y = y'$. □

The notation (as in 3.1.51) "$(a)f$" for relations is awkward when applied to functions —awkward but correct— where we prefer to use "$f(a)$" instead. The awkwardness manifests itself when we compose functions: In something like

$$x \to \boxed{f} \to z \to \boxed{g} \to y$$

that represents (1) above, note that f **acts first**. Its result $z = f(x)$ is then inputed to g —that is, we do $g(z) = g\Big(f(x)\Big)$ to obtain output y. Thus the first acting function f is "called" first with argument x and then g is called with argument $f(x)$. "Everyday math" notation places the two calls as in the red type above: The first call to the right of the 2nd call —order reversal vis a vis relational notation!

So, set theory heeds these observations and defines:

3.2.15 Definition. (Composition of functions; Notation) We just learnt (3.2.14) that the composition of two functions produces a function. The present definition is *about notation only.*

Let $f : A \to B$ and $g : B \to C$ be two functions. The relation $f \circ g : A \to C$, their *relational composition* is given in 3.1.15.

For composition of *functions*, we have the alternative —so-called *functional notation for composition*: "gf" for "$f \circ g$"; *note the order reversal* and the absence of "$\circ$", the composition symbol. In particular we write $(gf)(a)$ for $(a)(f \circ g)$ —cf. 3.2.3. Thus

$$a(gf)y \overset{Def}{\iff} a\, f \circ g\ y \iff (\exists z)(a f z \wedge z\, g\, y)$$

also

$$a(gf)y \overset{Def}{\iff} a\, f \circ g\ y \overset{Def\ 3.1.51}{\iff} (a)(f \circ g) = \{y\}$$

In particular, we have that $(a)(f \circ g)$ of 3.1.51 is the same as $(gf)(a) = g\Big(f(a)\Big)$ as seen through the "computation"

$$\begin{aligned}(a)(f \circ g) =^{3.2.14} \{y\} &\iff \text{for some } z,\ a\, f\, z \wedge z\, g\, y\\ &\iff^{3.2.3} \text{for some } z,\ f(a) = z \wedge g(z) = y\\ &\iff^{\text{subst. } z \text{ by } f(a)} g\Big(f(x)\Big) = y \qquad (1)\end{aligned}$$

Conclusion:

$$(gf)(a) \overset{\text{blue type above}}{=} (a)(f \circ g) \overset{(1)}{=} g\Big(f(x)\Big)$$

Thus the "reversal" $gf = f \circ g$ **now makes sense! So does** $(gf)(a) = g\big(f(a)\big)$**.** □

3.2.16 Theorem. *Functional composition is associative, that is,* $(gf)h = g(fh)$.

Proof. Exercise!

Hint. Note that by, 3.2.15, $(gf)h = h \circ (f \circ g)$. Take it from here. □

3.2.17 Example. The *identity relation* on a set A is a function since $(a)\mathbf{1}_A$ is the singleton $\{x\}$. □

The following interesting result connects the notions of ontoness and 1-1ness with the "algebra" of composition.

3.2.18 Theorem. *Let $f : A \to B$ and $g : B \to A$ be functions. If*

$$(gf) = \mathbf{1}_A \tag{1}$$

then g is onto while f is total and *1-1*.

We say that g is a *left inverse* of f and f is a *right inverse* of g. "A" because these are not in general unique! Stay tuned on this!

Proof. **About g:** Our goal, ontoness, means that, for each $x \in A$, I can "solve the equation $g(y) = x$ for y". Indeed I can: By definition of $\mathbf{1}_A$,

$$g\Big(f(x)\Big) \overset{3.2.15}{=} (gf)(x) \overset{(1)}{=} \mathbf{1}_A(x) = x$$

So to solve, take $y = f(x)$.

About f: As seen above, $x = g(f(x))$, for each $x \in A$. Since this is the same as "$x\, f \circ g, x$ is true", there must be a z such that $x\, f\, z$ and $z\, g\, x$. The first of these says $f(x) = z$ and therefore $f(x)\downarrow$. This settles totalness.

For the 1-1ness, let $f(a) = f(b)$. Applying g to both sides we get $g(f(a)) = g(f(b))$. But this says $a = b$, by $(gf) = \mathbf{1}_A$, and we are done. □

3.2.19 Example. *The above is as much as can be proved.* For example, say $A = \{1, 2\}$ and $B = \{3, 4, 5, 6\}$. Let $f : A \to B$ be $\{\langle 1, 4\rangle, \langle 2, 3\rangle\}$ and $g : B \to A$ be $\{\langle 4, 1\rangle, \langle 3, 2\rangle, \langle 6, 1\rangle\}$, or in friendlier notation

$f(1) = 4$
$f(2) = 3$
and
$g(3) = 2$
$g(4) = 1$
$g(5)\uparrow$
$g(6) = 1$

Clearly, $(gf) = \mathbf{1}_A$ holds, but note:

(1) f is not onto.
(2) g is neither 1-1 nor total. □

3.2.20 Example. With $A = \{1, 2\}$, $B = \{3, 4, 5, 6\}$ and $f : A \to B$ and $g : B \to A$ as in the previous example, consider also the functions $\tilde{f}$ and $\tilde{g}$ given by

$\tilde{f}(1) = 6$
$\tilde{f}(2) = 3$
and
$\tilde{g}(3) = 2$
$\tilde{g}(4) = 1$
$\tilde{g}(5)\uparrow$
$\tilde{g}(6) = 2$

Clearly, $(\tilde{g}f) = \mathbf{1}_A$ and $(g\tilde{f}) = \mathbf{1}_A$ hold, but note:

(1) $f \neq \tilde{f}$.

(2) $g \neq \tilde{g}$.

Thus, neither left nor right inverses need to be unique. The article "a" in the definition of said inverses was well-chosen. □

The following two partial converses of 3.2.18 are useful.

3.2.21 Theorem. *Let $f : A \to B$ be total and 1-1. Then there is an* onto *$g : B \to A$ such that $(gf) = \mathbf{1}_A$.*

Proof. Consider the converse relation (3.1.50) of f —that is, the *relation* f^{-1}— and call it g:

$$x\,g\,y \overset{\text{Def}}{\text{ iff }} y\,f\,x \tag{1}$$

By Exercise 3.2.11, $g : B \to A$ is a (possibly nontotal) function so we can write (1) as $g(x) = y$ iff $f(y) = x$, from which, substituting $f(y)$ for x in $g(x)$ we get $g(f(x)) = x$, for all $x \in A$, that is $gf = \mathbf{1}_A$, hence g is onto by 3.2.18. We got both statements that we needed to prove. □

3.2.22 Remark. By (1) above, $\mathrm{dom}(g) = \{x : (\exists y) g(x) = y\} = \{x : (\exists y) f(y) = x\} = \mathrm{ran}(f)$. □

3.2.23 Theorem. *Let $f : A \to B$ be onto. Then there is a* total and 1-1 *$g : B \to A$ such that $(fg) = \mathbf{1}_B$.*

Proof. By assumption, $\emptyset \neq f^{-1}[\{b\}] \subseteq A$, for all $b \in B$. To define $g(b)$ choose *one* $c \in f^{-1}[\{b\}]$ and set $g(b) = c$. Since $f(c) = b$, we get $f(g(b)) = b$ for all $b \in B$, and hence g is 1-1 and total by 3.2.18. □

3.3. Finite and Infinite Sets

Broadly speaking (that is, with very little detail contained in what I will say next) we have sets that are *finite* —intuitively meaning that we can count *all* their elements in a finite amount of time (but see the ☡-remark 3.3.3 below)— and those that are not, naturally called *infinite*!

What is a mathematical way to say all this?

Any counting process of the elements of a finite set A will have us say out loud —every time we pick or point at an element of A— "0th", "1st", "2nd", etc., and, once we reach and pick the last element of the set, we finally pronounce "nth", for some appropriate n that we reached in our counting (Again, see 3.3.3.)

Thus, mathematically, we are pairing each member of the set with a member from $\{0, \ldots, n\}$.

So we propose,

3.3.1 Definition. (Finite and infinite sets) A set A is *finite* iff it is either empty, or is in 1-1 correspondence with $\{x \in \mathbb{N} : x \leq n\}$. This "normalized" small set of natural numbers we usually denote by $\{0, 1, 2, \ldots, n\}$.

If a set is *not* finite, then it is *infinite*. □

3.3.2 Example. For any n, $\{0, \ldots, n\}$ is finite since, trivially, $\{0, \ldots, n\} \sim \{0, \ldots, n\}$ using the identity (Δ) function on the set $\{0, \ldots, n\}$. □

3.3.3 Remark. One must be careful when one attempts to explain finiteness via counting by a human.

For example, Achilles[†] could count *infinitely many objects* by constantly accelerating his counting process as follows:

He procrastinated for a *full second*, and then counted the first element. Then, he counted the second object *exactly after* $1/2$ a second from the first. Then he got to the third element $1/2^2$ seconds after the previous, ..., he counted the n th item at exactly $1/2^{n-1}$ seconds after the previous, and so on *forever*.

Hmm! It was *not* "forever", was it? After a total of 2 seconds he was done!

You see (as you can easily verify from your calculus knowledge (limits)),[‡]

$$1 + \frac{1}{2} + \frac{1}{2^2} + \ldots + \frac{1}{2^{n-1}} + \ldots = \frac{1}{1 - 1/2} = 2$$

So "time" is not a good determinant of finiteness! □

3.3.4 Theorem. *If $X \subset \{0, \ldots, n\}$, then there is no onto function $f : X \to \{0, \ldots, n\}$.*

I am saying, no such f, whether total or not; totalness is immaterial.

[†] OK, he was a demigod; but only "demi".

[‡] $1 + \frac{1}{2} + \frac{1}{2^2} + \ldots + \frac{1}{2^{n-1}} = \frac{1-1/2^n}{1-1/2}$. Now let n go to infinity at the limit.

Proof. First off, the claim holds if $X = \emptyset$, since then any such f equals $\emptyset$ and its range is empty.

Let us otherwise proceed by way of contradiction, and assume that the theorem is *wrong*: That is, **assume that** it *is* possible to have such onto functions, for some n and well chosen X.

Since I assume there are such $n > 0$ values, suppose then that the *smallest* n that allows this to happen is, say, n_0, and let X_0 be a *corresponding* set "X" that works, that is,

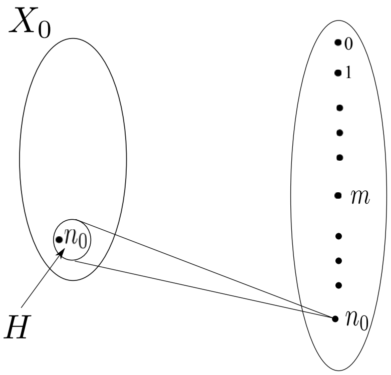
$$\text{Assume that we have an onto } f : X_0 \to \{0, \ldots, n_0\} \tag{1}$$

Thus $X_0 \neq \emptyset$, by the preceding remark, and therefore $n_0 > 0$, since otherwise $X_0 = \emptyset$.

Let us call H be the set of all x such that $f(x) = n_0$, for short, $H = f^{-1}(\{n_0\})$. $\emptyset \neq H \subseteq X_0$; the $\neq$ by ontoness.

Case 1. $n_0 \in H$. Then removing all pairs (a, n_0) from f —all these have $a \in H$— we get a new function $f' : X_0 - H \to \{0, 1, \ldots, n_0 - 1\}$, which *is still onto* as we only removed inputs that cause output n_0.

This contradicts minimality of n_0 since $n_0 - 1$ works too!



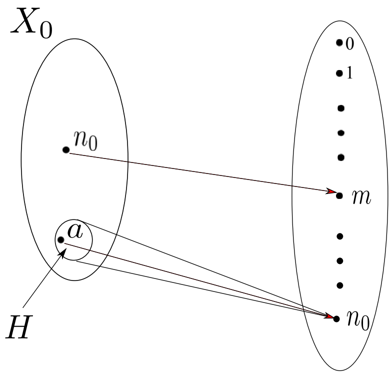
Case 2. $n_0 \notin H$.

If $n_0 \notin X_0$, then we argue exactly as in Case 1 and we just remove the base "H" of the cone (in the picture) from X_0.

Otherwise, we have two subcases:

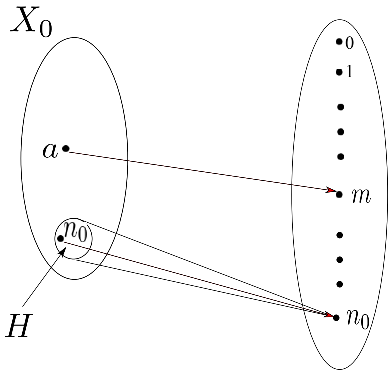
- $f(n_0) \uparrow$. Then (almost) we act as in Case 1: The new "X_0" is $(X_0 - H) - \{n_0\}$, since if we leave n_0 in, then the new "X_0" will not be a subset of $\{0, 1, \ldots, n_0 - 1\}$. We get a contradiction per Case 1.

- The picture below —that is, $f(n_0) = m$ for some m.



We simply transform the picture to the one below, "correcting" f to have $f(a) = m$ and $f(n_0) = n_0$, that is defining a new "f" that we will call f' by

$$f' = \Big(f - \{(n_0, m), (a, n_0)\}\Big) \cup \{(n_0, n_0), (a, m)\}$$



We get a contradiction per Case 1. □

3.3.5 Corollary. (Pigeon-Hole Principle) *If $m < n$, then $\{0, \ldots, m\} \not\sim \{0, \ldots, n\}$.*

Proof. If the conclusion fails then we have an onto $f : \{0, \ldots, m\} \to \{0, \ldots, n\}$, contradicting 3.3.4. □

Important!

3.3.6 Theorem. *If A is finite due to $A \sim \{0, 1, 2, \ldots n\}$ then there is* ***no justification of finiteness via another canonical set*** *$\{0, 1, 2, \ldots m\}$ with $n \neq m$.*

Proof. If $\{0, 1, 2, \ldots n\} \sim A \sim \{0, 1, 2, \ldots m\}$, then $\{0, 1, 2, \ldots n\} \sim \{0, 1, 2, \ldots m\}$ by 3.2.13, hence $n = m$, otherwise we contradict 3.3.5. □

3.3.7 Definition. Let $A \sim \{0, \ldots, n\}$. Since n is uniquely determined by A we say that A has $n+1$ elements and write $|A| = n+1$. □

3.3.8 Corollary. *There is no onto function from* $\{0, \ldots, n\}$ *to* $\mathbb{N}$.

"For all $n \in \mathbb{N}$, there is no…" is, of course, implied.

Proof. Fix an n. By way of contradiction, let $g : \{0, \ldots, n\} \to \mathbb{N}$ be onto. Let

$$Y \overset{Def}{=} \{x \le n : g(x) > n+1\}$$

Now let

$$X \overset{Def}{=} \{0, \ldots, n\} - Y$$

and

$$g' \overset{Def}{=} g - Y \times \mathbb{N}$$

The "$g - Y \times \mathbb{N}$" above is an easy way to say "remove all pairs from g that have their first component in Y".

Thus, $g' : X \to \{0, \ldots, n, n+1\}$ is onto, contradicting 3.3.4 because $X \subseteq \{0, \ldots, n\} \subset \{0, \ldots, n, n+1\}$. □

3.3.9 Corollary. $\mathbb{N}$ *is infinite.*

Proof. By 3.3.1 the opposite case requires that there is an n and a function $f : \{0, 1, 2, \ldots, n\} \to \mathbb{N}$ that is a 1-1 correspondence. Impossible, since any such an f will fail to be *onto.* □

Our mathematical definitions have led to what we hoped they would: That $\mathbb{N}$ *is* infinite as we intuitively understand, notwithstanding Achilles's accelerated counting!

$\mathbb{N}$ is a "canonical" infinite set that we can use to index the members of many infinite sets. Sets that can be indexed using natural number indices

$$a_0, a_1, \ldots$$

are called *countable.*

In the interest of technical flexibility, *we do not insist* that *all* members of $\mathbb{N}$ be used as indices. We might enumerate with gaps:

$$b_5, b_9, b_{13}, b_{42}, \ldots$$

Thus, informally, a set A is *countable* if it is empty or (in the opposite case) if there is a way to index, hence enumerate, all its members in an array, utilizing indices from $\mathbb{N}$. Cf. 3.1.40.

It *is* allowed to repeatedly list any element of A, so that finite sets are countable. For example, the set $\{42\}$:

$$42, 42, 42, \overbrace{\ldots}^{\text{42 forever}}$$

We may think that the enumeration above is done by assigning to "42" *all* of the members of $\mathbb{N}$ as indices, in other words, the enumeration is effected, for example, by the constant function $f : \mathbb{N} \to \{42\}$ given by $f(n) = 42$ for all $n \in \mathbb{N}$. This is consistent with our earlier definition of indexing (3.1.40).

Now, mathematically,

3.3.10 Definition. (Countable Sets) We call a set A *countable* if $A = \emptyset$, or there is an *onto* function $f : \mathbb{N} \to A$. We do NOT require f to be total. This means that some or many indices from $\mathbb{N}$ need not be used in the enumeration If $f(n) \downarrow$, then we say that $f(n)$ is the nth element of A in the enumeration f. We often write f_n instead of $f(n)$ and then call n a "subscript" or "index". □

Thus a nonempty set is countable iff it is the *range* of some function that has $\mathbb{N}$ as its *left field.*

BTW, since we allow f to be non total, the hedging "nonempty" is unnecessary: $\emptyset$ is the range of the empty function that has $\mathbb{N}$ as its left field.

We said that the f that proves countability of a set A need not be total. But such an f can always be "completed", by adding pairs to it, to get an f' such that $f' : \mathbb{N} \to A$ is onto *and* total. Here is how:

3.3.11 Proposition. *Let $f : \mathbb{N} \to A \neq \emptyset$*[†] *be onto. Then we can extend f to f' so that $f' : \mathbb{N} \to A$ is onto* and *total.*

Proof. Pick an $a \in A$ —possible since $A \neq \emptyset$— and keep it fixed. Now, our sought f' is given for all $n \in \mathbb{N}$ by cases as below:

$$f'(n) = \begin{cases} f(n) & \text{if } f(n) \downarrow \\ a & \text{if } f(n) \uparrow \end{cases}$$

□

Some set theorists also define sets that can be enumerated using *all* the elements of $\mathbb{N}$ as indices *without repetitions.*

3.3.12 Definition. (Enumerable or denumerable sets) A set A is *enumerable* iff $A \sim \mathbb{N}$. □

3.3.13 Example. Every enumerable set is countable, but the converse fails. For example, $\{1\}$ is countable but not enumerable due to 3.3.8. $\{2n : n \in \mathbb{N}\}$ is enumerable, with $f(n) = 2n$ effecting the 1-1 correspondence $f : \mathbb{N} \to \{2n : n \in \mathbb{N}\}$. □

[†] Since we are constructing a *total* onto function to A we need to assume the case $A \neq \emptyset$ as we cannot put any outputs into $\emptyset$.

3.3.14 Theorem. *If A is an infinite subset of $\mathbb{N}$, then $A \sim \mathbb{N}$.*

Proof. We will build a 1-1 and total enumeration of A, presented in a finite manner as a (pseudo) program below, which enumerates all the members of A in strict ascending order and arranges them in an array

$$a(0), a(1), a(2), \ldots \qquad (1)$$

```
$n$        $\leftarrow 0$
while      $A \neq \emptyset$
$a(n)$     $\leftarrow \min A$  Comment. Inside the loop $\emptyset \neq A \subseteq \mathbb{N}$, hence min exists.
$A$        $\leftarrow A - \{a(n)\}$
$n$        $\leftarrow n + 1$
end while
```

☡ Note that the sequence $\{a(0), a(1), \ldots, a(m)\}$ is **strictly increasing** for any m, since $a(0)$ is smallest in A, $a(1)$ is smallest in $A - \{a(0)\}$ and hence the next higher than $a(0)$ in A, etc. ☡

Will this loop ever exit? **Say, yes**, when it starts (but does not complete) the k-th pass through the loop. Thus A became empty when we did $A \leftarrow A - \{a(k-1)\}$ in the previous pass, that is $A = \{a(0), a(1), \ldots, a(k-1)\}$ and thus, since

$$a(0) < a91) < \ldots < a(k-1)$$

we have that the function $f : \{0, \ldots, k-1\} \to A$ given by

$$f = \{(0, a(0)), (1, a(1)), \ldots (k-1, a(k-1))\}$$

is total, 1-1 and onto, thus, $A \sim \{0, \ldots, k-1\}$ **contradicting that A is infinite**!

Thus, we never exit the loop!

Thus, by the remark in the ☡ paragraph above, (1) enumerates A in strict ascending order, that is,

if we define $f : \mathbb{N} \to A$ by $f(n) = a(n)$, for all n

then f is 1-1 (by strict increasing property: distinct inputs cause distinct outputs), and is trivially total, and onto. Why the latter? Every $a \in A$ is reached in ascending order, and assigned an "n" from $\mathbb{N}$. □

3.3.15 Theorem. *Every infinite countable set is enumerable.*

Proof. Let $f : \mathbb{N} \to A$ be onto and total (cf. 3.3.11), where A is infinite. Let $g : A \to \mathbb{N}$ such that $(fg) = \mathbf{1}_A$ (3.2.23). Thus, if we let $B = \mathrm{ran}(g)$, we have that g is onto B, and thus by 3.2.18 is also 1-1 and total. Thus it is a 1-1 correspondence $g : A \to B$, that is,

$$A \sim B \qquad (1)$$

B *must* be infinite, otherwise (3.3.1), for some n, $B \sim \{0, \ldots, n\}$ and by (1) via Exercise 3.2.13 we have $A \sim \{0, \ldots, n\}$, contradicting that A is infinite. Thus, by 3.3.14, $B \sim \mathbb{N}$, hence (again, Exercise 3.2.13 and (1)) $A \sim \mathbb{N}$. That is, A is enumerable. □

So, if we can enumerate an infinite set at all, then we can enumerate it without repetitions.

We can linearise an infinite square matrix of elements in each location (i, j) by devising a traversal that will go through each (i, j) entry *once*, and will *not miss any entry*!

In the literature one often sees the method diagrammatically, see below, where arrows *clearly* indicate the sequence of traversing, with the understanding that we use the arrows by pick the first unused chain of arrows from left to right.

$$\begin{tikzcd}
(0,0) & (0,1) & (0,2) & (0,3) & \ldots \\
(1,0) \arrow[ur] & (1,1) \arrow[ur] & (1,2) \arrow[ur] & & \\
(2,0) \arrow[ur] & (2,1) \arrow[ur] & & & \\
(3,0) \arrow[ur] & & & & \\
\vdots & & & &
\end{tikzcd}$$

So the linearisation induces a 1-1 correspondence between $\mathbb{N}$ *and the linearised sequence of matrix entries, that is, it shows that* $\mathbb{N} \times \mathbb{N} \sim \mathbb{N}$. For short,

3.3.16 Theorem. *The set* $\mathbb{N} \times \mathbb{N}$ *is countable. In fact, it is enumerable.*

Is there a "mathematical" way to do this? Well, the above IS mathematical, don't get me wrong, but is given in *outline*. It is kind of an argument in geometry, where we rely on drawings (figures).

Here are the algebraic details:

Proof. (of 3.3.16 with an algebraic argument). Let us call $i + j + 1$ the "*weight*" of a pair (i, j). The weight is the number of elements in the group:

$$(i + j, 0), (i + j - 1, 1), (i + j - 2, 2), \ldots, (i, j), \ldots, (0, i + j)$$

Thus the diagrammatic enumeration proceeds by enumerating *groups* by increasing weight

$$1, 2, 3, 4, 5, \ldots$$

and in each group of weight k we enumerate in *ascending order of the second component.*

Thus the (i, j) th entry occupies position j *in its group* —the first position in the group being the 0 th, e.g., in the group of $(3, 0)$ the first position is the 0 th— and this position *globally* is the number of elements in all groups *before* group $i+j+1$, *plus* j. Thus the first available position for the first entry of group (i,j) members is just after this many occupied positions:

$$1+2+3+\ldots(i+j)=\frac{(i+j)(i+j+1)}{2}$$

That is,

$$\text{global position of } (i,j) \text{ is this: } \frac{(i+j)(i+j+1)}{2}+j$$

The function f which for all i,j is given by

$$f(i,j)=\frac{(i+j)(i+j+1)}{2}+j$$

is the algebraic form of the above enumeration. □

There is an easier way to show that $\mathbb{N}\times\mathbb{N}\sim\mathbb{N}$ without diagrams:

By the unique factorisation of numbers into products of primes (Euclid) the function $g:\mathbb{N}\times\mathbb{N}\to\mathbb{N}$ given for all m,n by $g(m,n)=2^m3^n$ is 1-1, since Euclid proved that $2^m3^n=2^{m'}3^{n'}$ implies $m=m'$ and $n=n'$. It is not onto as it never outputs, say, 5, but $\mathrm{ran}(g)$ is an *infinite* subset of $\mathbb{N}$ (Exercise!).

Thus, trivially, $\mathbb{N}\times\mathbb{N}\sim\mathrm{ran}(g)\sim\mathbb{N}$, the latter "$\sim$" by 3.3.14.

3.3.17 Exercise. If A and B are enumerable, so is $A\times B$.

Hint. So, $\mathbb{N}\sim A$ and $\mathbb{N}\sim B$. Can you show now that $\mathbb{N}\times\mathbb{N}\sim A\times B$? □

With little additional effort one can generalise to the case of $\bigtimes_{i=1}^{n} A_i$.

3.3.18 Remark.

1. Let us collect a few more remarks on countable sets here. Suppose now that we start with a countable set A. Is every subset of A countable? Yes, because the composition of onto functions is onto.

2. **3.3.19 Exercise.** What does composition of onto functions have to do with this? Well, if $B\subseteq A$ then there is a *natural* onto function $g:A\to B$. Which one? Think "natural"! Get a *natural* total and 1-1 function $f:B\to A$ and then use f to get g. □

3. As a special case, if A is countable, then so is $A\cap B$ for any B, since $A\cap B\subseteq A$.

4. How about $A\cup B$? If both A and B are countable, then so is $A\cup B$. Indeed, and without inventing a new technique, let

$$a_0,a_1,\ldots$$

be an enumeration of A and

$$b_0, b_1, \ldots$$

for B. Now form an infinite matrix with the A-enumeration as the 1st row, while each remaining row is the same as the B-enumeration. Now linearise this matrix!

Of course, we may alternatively adapt the unfolding technique to an infinite matrix of just two rows. **How?**

5. **3.3.20 Exercise.** Let A be enumerable and an enumeration of A

$$a_0, a_1, a_2, \ldots \tag{1}$$

is given.

So, this is an enumeration without repetitions.

Use techniques we employed in this section to propose a new enumeration in which *every* a_i is listed *infinitely many times* (this is useful in some applications of logic). □

3.4. Diagonalisation and uncountable sets

3.4.1 Example. Suppose we have a 3 × 3 matrix

$$\begin{matrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{matrix}$$

and we are asked: Find a sequence of three numbers, *using only* 0 *or* 1, that does not *fit* as a row of the above matrix —i.e., is *different from all rows.*

Sure, you reply: Take 1 1 1. Or, take 0 0 0.

That is correct. But what if the matrix were big, say, $10^{350000} \times 10^{350000}$, or even *infinite*?

Is there a *finitely describable technique* that can produce an "unfit" row for any square matrix, even an infinite one? Yes, it is Cantor's *diagonal method* or technique.

He noticed that any row that fits in the matrix as the, say, i-th row, intersects the main diagonal at the same spot that the i-th column does.

$\underline{\text{That is, at entry } (i, i).}$

Thus if we take the main diagonal —a sequence that has the same length as any row— and *change every one of its entries*, then it will not fit *anywhere* as a row! *Because no row can have an entry that is different than the entry at the location where it intersects the main diagonal!*

This idea would give the answer 0 1 0 to our original question. While 1000 11 3 also follows the principle “change all the entries of the diagonal” and works, we are constrained here to “use only 0 or 1” as entries. More seriously, in a case of a very large or infinite matrix it is best to have a simple technique that works even if we do not know much about the elements of the matrix. Read on! □

3.4.2 Example. We have an infinite matrix of 0-1 entries. Can we produce an infinite sequence of 0-1 entries that does not match *any* row in the matrix? Yes, take the main diagonal and *flip every entry* (0 to 1; 1 to 0).

If we think that, yes, it fits as row i, then we get a contradiction:

Say the original row has an a as entry (i,i). But, by our construction, the *new* row has an $1-a$ in as entry (i,i), so it will not fit as row i after all. So it fits nowhere, i being arbitrary. □

☡ **3.4.3 Example. (Cantor)** Let S denote the set of all infinite sequences of 0s and 1s.

Pause. What is an *infinite sequence*? Our intuitive understanding of the term is captured mathematically by the concept of a total function f with left field (and hence domain) $\mathbb{N}$. The n-th member of the sequence is $f(n)$.◀

Can we arrange *all* of S in an infinite matrix —one element per row? No, since the preceding example shows that we would miss at least one infinite sequence (i.e., we would fail to list it as a row), for a sequence of infinitely many 0s and/or 1s can be found, that does not match any row!

But arranging all members of S as an infinite matrix —one element per row— is tantamount to saying that we can enumerate all the members of S using members of $\mathbb{N}$ as indices.

So we cannot do that. S is not countable! □ ☡

3.4.4 Definition. (Uncountable Sets) A set that is not countable is called *uncountable.* □

So, an uncountable set is neither finite, nor enumerable. The first observation makes it infinite, the second makes it “more infinite” than the set of natural numbers since it is not in 1-1 correspondence with $\mathbb{N}$ (else it would be enumerable, hence countable) nor with a subset of $\mathbb{N}$: If the latter, our uncountable set would be finite or enumerable (which is absurd) according as it is in 1-1 correspondence with a finite subset or an infinite subset (cf. 3.3.14 and Exercise 3.2.13).

Example 3.4.3 shows that uncountable sets exist. Here is a more interesting one.

☡ **3.4.5 Example. (Cantor)** The set of real numbers in the interval

$$(0,1)\overset{\text{Def}}{=}\{x\in\mathbb{R} : 0<x<1\}$$

is uncountable. This is done via an elaboration of the argument in 3.4.3.

Think of a member of $(0,1)$, *in form*, as an infinite sequence of numbers from the set $\{0,1,2,3,4,5,6,7,8,9\}$ prefixed with a dot; that is, think of the number's decimal notation.

Some numbers have representations that end in 0s after a certain point. We call these representations *finite*. Every such number has also an "infinite representation" since the non zero digit d immediately to the left of the infinite tail of 0s can be converted to $d-1$, and the infinite tail into 9s, without changing the value of the number.

Allow only infinite representations.

Assume now by way of contradiction that a listing of all members of $(0,1)$ exists, listing them via their infinite representations

$$\begin{array}{c}
.a_{00}a_{01}a_{02}a_{03}a_{04}\cdots\\
.a_{10}a_{11}a_{12}a_{13}a_{14}\cdots\\
.a_{20}a_{21}a_{22}a_{23}a_{24}\cdots\\
.a_{30}a_{31}a_{32}a_{33}a_{34}\cdots\\
\vdots
\end{array}$$

The argument from 3.4.3 can be easily modified to get a "row that does not fit", that is, a representation

$$.d_0d_1d_2\cdots$$

not in the listing.

Well, just let

$$d_i = \begin{cases} 2 & \text{if } a_{ii} = 0 \lor a_{ii} = 1\\ 1 & \text{otherwise}\end{cases}$$

Clearly $.d_0d_1d_2\cdots$ does not fit in any row i as it differs from the expected digit at the i-th decimal place: should be a_{ii}, but $d_i \neq a_{ii}$. It is, on the other hand, an infinite decimal expansion, being devoid of zeros, and thus *should* be listed. This contradiction settles the issue. □

3.4.6 Example. (3.4.3 Revisited) Consider the set of all total functions from $\mathbb{N}$ to $\{0,1\}$. Is this countable?

Well, if there is an enumeration of these one-variable functions

$$f_0, f_1, f_2, f_3, \ldots \tag{1}$$

consider the function $g : \mathbb{N} \to \{0,1\}$ given by $g(x) = 1 - f_x(x)$. Clearly, this *must* appear in the listing (1) since it has the correct left and right fields, and is total.

Too bad! If $g = f_i$ then $g(i) = f_i(i)$. By definition, it is also $1 - f_i(i)$. A contradiction.

This is just version of 3.4.3; as already noted there, an infinite sequence of 0s and 1s is just a total function from $\mathbb{N}$ to $\{0,1\}$. □

The same argument as above shows that the set of all functions from $\mathbb{N}$ to itself is uncountable. Taking $g(x) = f_x(x)+1$ also works here to "systematically change the diagonal" $f_0(0), f_1(1), \ldots$ since we are not constrained to keep the function values in $\{0, 1\}$.

3.4.7 Remark. Worth Emphasizing. Here is how we constructed g: We have a list of *in principle available f-indices* for g. We want to make sure that *none of them applies.*

A convenient method to do that is to inspect each available index, i, and using the diagonal method do this: Ensure that g differs from f_i at input i, by setting $g(i) = 1 - f_i(i)$.

This ensures that $g \neq f_i$; period. We say that *we cancelled the index i* as a *possible* "f-index" of g.

Since the process is applied *for each i*, *we have cancelled* all *possible indices for g*: For no i can we have $g = f_i$. □

3.4.8 Example. (Cantor) What about the set of all subsets of $\mathbb{N}$ —$\mathcal{P}(\mathbb{N})$ or $2^{\mathbb{N}}$?

Cantor showed that this is uncountable as well: If not, we have an enumeration of its members as

$$S_0, S_1, S_2, \ldots \tag{1}$$

Define the set

$$D \overset{\text{Def}}{=} \{x \in \mathbb{N} : x \notin S_x\} \tag{2}$$

So, $D \subseteq \mathbb{N}$, thus it must appear in the list (1) as an S_i. But then $i \in D$ iff $i \in S_i$ by virtue of $D = S_i$. However, also $i \in D$ iff $i \notin S_i$ by (2). This contradiction establishes that a legitimate subset of $\mathbb{N}$, namely D, is *not* an S_i. That is, $2^{\mathbb{N}}$ *cannot* be so enumerated; it is uncountable. □

3.4.9 Example. (Characteristic functions) Let $S \subseteq \mathbb{N}$. We can represent S as an infinite 0/1 array:

$$\begin{array}{rccccc}
\text{array position} & \ldots & i & \ldots & j & \ldots \\
\text{array content} & \ldots & 0 & \ldots & 1 & \ldots \\
 & \ldots & \uparrow & \ldots & \uparrow & \ldots \\
\text{means} & \multicolumn{5}{c}{\ldots i \notin S \ldots j \in S \ldots}
\end{array}$$

This array faithfully represents S —tells all we need to know about what S contains— since it contains a "1" in location x iff $x \in S$; contains "0" otherwise.

The array viewed as a total function from $\mathbb{N}$ to $\{0, 1\}$ is called the *characteristic function of S*, denoted by c_S:

$$c_S(x) = \begin{cases} 1 & \textbf{if } x \in S \\ 0 & \textbf{if } x \in \mathbb{N} - S \end{cases}$$

Note that there is a 1-1 correspondence, let's call it F, between subsets of $\mathbb{N}$ and the total 0-1-valued functions from $\mathbb{N}$ simply given by $F(S) = c_S$. (Exercise!)

Thus

$$\{f : f : \mathbb{N} \to \{0,1\} \text{ and } f \text{ is total}\} \sim 2^{\mathbb{N}}$$

In particular, the concept of characteristic functions shows that Example 3.4.8 fits the diagonalization methodology. Indeed, the argument in 3.4.8 sets $c_D(x) = 1 - c_{S_x}(x)$, for all x, because

$$c_D(x) = 1 \text{ iff } x \in D \text{ iff } x \notin S_x \text{ iff } c_{S_x}(x) = 0 \text{ iff } 1 - c_{S_x}(x) = 1$$

But then, the argument in 3.4.8 essentially applies the diagonal method to the list of 0/1 functions c_{S_x}, for $x = 0, 1, 2, \ldots$, to show that some 0/1 function, namely, c_D cannot be in the list. □

Chapter 4

A Tiny Bit of Informal Logic

We have come somewhat proficient in using informal logic in our arguments about aspects of discrete mathematics.

Although we have used quantifiers, ∃ and ∀ we did so mostly viewing them as symbolic abbreviations of English texts about mathematics. In this chapter we will expand our techniques in logic, extending them to include manipulation of quantifiers including the versatile Induction —or mathematical induction— technique used to prove properties of the natural numbers.

We know how to detect fallacious statements formulated in Boolean logic: Simply show by a truth table that the statement is not a tautology. (talk about tautological implication too)

We will show in the domain of quantifiers not only how to prove statements that include quantifiers but also how to disprove false statements that happen to include quantifiers.

4.1. Enriching our proofs to manipulate quantifiers

Manipulation of quantifiers boils down to "how can I remove a quantifier from the beginning of a formula?" and "how can I add a quantifier at the beginning of a formula?" Once we learn this technique we will be able to reason within mathematics with ease.

But first let us define once and for all what a mathematical proof *looks like*: its *correct*, *expected syntax* or *form*.

We will need some concepts to begin with.

1. The alphabet and structure of formulas. Formulas are strings. The alpha-

bet of *symbols* that we use contain, *at a minimum*,

$$=, \neg, \wedge, \vee, \rightarrow, \equiv, (,), \forall, \exists, \text{object variables}^\dagger$$

We finitely generate the infinite set of object variables using single letters, if necessary with primes and/or subscripts: $A, x, y'', w'''_{23}, u_{501}$.

2. One normally works in a mathematical area of interest, or *mathematical theory* —such as Geometry, Set Theory, Number Theory, Algebra, Calculus— where one needs additional symbols to write down formulas, like

$$0, \emptyset, \in, \subset, \int, \circ, +, \times$$

and many others.

3. Mathematicians as a rule get to recognise the *formulas* and *terms* in the math areas of their interest without being necessarily taught the recursive definition of the syntax of these. We will not give the syntax in these notes either (but see [Tou08] if you want to know!). Thus one learns to be content with getting to know formulas and terms by their behaviour and through use, rather than by their exact definition of syntax.

   - *Terms* are "function calls", in the jargon of the computer savvy person. These calls take math objects as inputs and return math objects as outputs. *Examples* are: $x + y$, $x \times 3$, $0 \times x + 1$ (one is told that $\times$ is stronger than $+$, so, notwithstanding the bracket-parsimonious notation "$0 \times x + 1$", we know it means "$(0 \times x) + 1$", so this call returns 1, no matter what we plugged into x).
   - *Formulas* are also function calls, but their output is *restricted* (by their syntax that I will not define carefully!) to be one or the other of the truth values true or false (**t** or **f**) but nothing else! Their input, just as in the case for terms, is any math object. *Examples* are: $2 < 3$ (**t**), $(\forall x)x = x$ (**t**), $(\forall x)x = 0$ (**f**), $(\exists x)x = 0$ (**t**), $x = 0$ neither true nor false; answer depends on the input in x!

     More: $x = x$ (**t**) answer is independent of input; $x = 0 \rightarrow x = 0$ (**t**) answer is independent of input; $x = 0 \rightarrow (\forall x)x = 0$ neither true nor false; answer depends on input in x! The input variable is the *leftmost* x; the other two are *bound* and *unavailable* to accept inputs. See below.
   - If an **occurrence** of formula variable *is* available for input it would normally be called an occurrence as an *input variable.* Rather, such occurrences are called *free occurrences* in the literature.

     At the expense of writing style, "occurrence" occurred no less than four times in the short passage above. The aim is *emphasis*: It is not a *variable* x that is free or bound in a formula, but it is the occurrences

[†] That is, variables that denote *objects* such as numbers, arrays, matrices, sets, trees, etc.

of said variable that we are speaking of, as the immediately preceding example makes clear.

4. In $(\forall x)x = 0$ the variable x is non input, it is "bound" we say. Just like this: $\Sigma_{i=1}^{4} i$, which means $1 + 2 + 3 + 4$ and "i" is not available for input: Something like $\Sigma_{3=1}^{4} 3$ is nonsense! Similar comment for $\exists$.

5. We call $\forall, \exists, \neg, \wedge, \vee, \rightarrow, \equiv$ the "*logical connectives*", the last 5 of them being called *Boolean connectives*. People avoid cluttering notation with a lot of brackets by agreeing that the first 3 have the same "strength" or "priority"; the *highest*. The remaining connectives have priorities decreasing as we walk to the right.

   Thus, if A and B are (denote) formulas, then $\neg A \vee B$ means $(\neg A) \vee B$; $\neg$ wins the claim for A. If we want $(\forall x)$ to apply to the entire $A \rightarrow B$ we must write $(\forall x)(A \rightarrow B)$.

   What about $A \rightarrow B \rightarrow C$ and $A \equiv B \equiv C$? Brackets are *implied from right to left*: $A \rightarrow (B \rightarrow C)$ and $A \equiv (B \equiv C)$. And this? $(\exists y)(\forall x)\neg A$. Brackets are *implied, again, from right to left*: $\Big((\exists y)\big((\forall x)(\neg A)\big)\Big)$.

   BTW, the part where a $\forall x$ or $\exists x$ *acts* —the "$(\ldots)$" in $(\forall x)(\ldots)$ and $(\exists x)(\ldots)$— is called their scope.

6. **Boolean deconstruction**. A formula like $(\forall x)A \rightarrow B$ can be *deconstructed* Boolean-wise into $(\forall x)A$ and B. If I knew more about B —say, it is $x = 3 \rightarrow x = 7$, then I can deconstruct further.

   So, now I have got

   $$(\forall x)A, \quad x = 3, \quad x = 7$$

   The last two have NO Boolean structure so deconstructing stops with them. How about $(\forall x)A$? This cannot be deconstructed either, even if A had Boolean structure! Such structure is locked up in the scope of $(\forall x)$.

   We call the formulas where deconstruction stops "*prime*". A prime formula is one with no Boolean structure, e.g., $x < 8$, or one of the form $(\forall x)A$ (A is the scope) or $(\exists x)A$ (A is the scope).

   Every formula is either prime or can be deconstructed into prime components.

4.1.1 Remark. (Tautologies) A formula A is a *tautology* iff it **is true due to its Boolean structure**, according to truth tables (2.3.4) no matter what the values of its prime formulas into which it is deconstructed are assumed to be. **Assumed to be**: We do **NOT** compute the *intrinsic* truth value of a prime formula when we check whether A is a tautology or not.

For example, $x = x$ is a prime formula and thus its assumed value could be ANY of **t** or **f**. Thus it is NOT a tautology, even though, *intrinsically IS true*, no matter what the value of x. □

4.1.2 Example.

1. $(\forall x)A$ is not a tautology as it has two possible truth values (being a prime formula) in principle.

2. $x = 0 \to x = 0$ is a tautology. Which are its prime (sub) formulas? □

3. $(\forall x)x = 0 \to x = 0$ is not a tautology. I repeat (**once**): To determine tautologyhood we *DO NOT evaluate prime formulas*; we just consider *each* of the two scenarios, **t** or **f**, for each prime formula and use truth tables to compute the overall truth value.

If we DID evaluate $(\forall x)x = 0$ we would see that (say over the natural numbers, or reals, or complex numbers) it is false.[†] So the implication is true! But we DON'T do that! **Not true** as a Boolean formula!

So, how do we show that $(\forall x)A$ is true (if it is)? Well, in easy cases we try to see if A is true for all values of x. That failing, we will use a proof (see 4.1.9).

Similarly for $(\exists x)A$. To show it is true (if it is) we try to see if A is true for some value of x. Often we just guess one such value that works, say c, and verify the truth of A when $x = c$. That failing, we will use a proof.

4.1.3 Definition. (Tautological implication)
We say that the formulas $A_1, A_2, \ldots, A_n$ *tautologically imply* a formula B —in symbols $A_1, A_2, \ldots, A_n \models_{taut} B$— meaning

"the truth of $A_1 \wedge A_2 \wedge \ldots \wedge A_n$ implies the truth of B"

that is, that

$A_1 \wedge A_2 \wedge \ldots \wedge A_n \to B$ is a tautology

□

4.1.4 Remark. Note that we do NOT care to *check*, or even *state*, what happens if $A_1 \wedge A_2 \wedge \ldots \wedge A_n$ is false.

The implication in blue type is true regardless of the truth value of B

So, a tautological implication $A_1, A_2, \ldots, A_n \models_{taut} B$ says that B is true provided we proved (or accepted) that the lhs of $\models_{taut}$ is true.

$\models_{taut}$ propagates truth from left to right. □

4.1.5 Example. Here are some easy and some involved tautological implications. They can all be verified using truth tables, either building the tables in full, or taking shortcuts.

[†] If we are doing our mathematics restricted to the set $\{0\}$, then, in this "theory" the formula IS true!

1. $A \models_{taut} A$

2. $A \models_{taut} A \vee B$

3. $A \models_{taut} B \to A$

4. $A, \neg A \models_{taut} B$ —any B. Because I do *work* only if $A \wedge \neg A$ is true! See above.

5. $\mathbf{f} \models_{taut} B$ —any B. Because I do *work* only if lhs is true! See above.

6. Is this a valid tautological implication? $\underline{B, A \to B \models_{taut} A}$, where A and B are distinct.

   No, for if A is false and B is true, then the lhs is true, but the rhs is false!

7. Is this a valid tautological implication? $\underline{A, A \to B \models_{taut} B}$? Yes! Say $A = \mathbf{t}$ and $(A \to B) = \mathbf{t}$. Then, from the truth table of →, it *must* be $B = \mathbf{t}$.

8. How about this? $\underline{A, A \equiv B \models_{taut} B}$? Yes! Verify!

9. How about this? $\underline{A \vee B \equiv B \models_{taut} A \to B}$? Yes! I verify:

   First off, **assume** lhs of $\models_{taut}$ —that is, $A \vee B \equiv B$— is true.

   Two cases:

   - $B = \mathbf{f}$. Then I need the lhs of $\equiv$ to be true to satisfy the bolded "assume". So $A = \mathbf{f}$ as well and clearly the rhs of $\models_{taut}$ is true with these values.
   - $B = \mathbf{t}$. Then I need not worry about A on the lhs. The rhs of $\models_{taut}$ is true by truth table of →.

10. $A \wedge (\mathbf{f} \equiv A) \models_{taut} B$, for any B. Well, just note that the lhs of $\models_{taut}$ is $\mathbf{f}$ so we need to do no work with B to conclude that the implication is valid.

11. $$A \to B, C \to B \models_{taut} A \vee C \to B$$

    This is nicknamed "proof by cases" for the obvious reasons. Verify this tautological implication! □

The job of a mathematical proof is to start from established (previous theorems) truths, or assumed truths (axioms) and unfailingly preserve truths in all its steps as it is developed. Thus, it will have produced, in particular, a truth at its very last step. A theorem.

What are our axioms, our starting assumptions, when we do proofs?

4.1.6 Definition. First off, in *any proof that we will write in math* there are axioms that are independent of the type of math that we do, whether it is set theory, number theory, algebra, calculus, etc.

Our logical axioms are

1. All tautologies; these need no defence as “start-up truths”.

2. Formulas of the form $(\forall x)A[x] \to A[t]$, for any formula A, variable x and “object” t.

   This object can be as simple as a variable y (might be same as x), constant c, or as complex as a “*function call*”, $f(t_1, t_2, \ldots, f_n)$ where f accepts n-inputs, and the inputs shown here are already available objects.

   Two comments: This is a *bona fide* start-up *truth* as its says “if $A[x]$ is true for all x-values,[†] then it is true also if we plug a specific value/object into x”.

   The other comment is that I write $A[x]$ to indicate *a variable of interest*. This may or may not occur in A, which may also have other variables that it depends on. I would write $A(x, y, z)$ —round brackets— if I knew that x, y, z are *all* the variables on which A depends.

3. Formulas of the form $A[x] \to (\forall x)A[x]$, **for any formula A where the variable x does not occur in it.** For example say A is $3 = 3$. This axiom says then, “if $3 = 3$ is true, then so is $(\forall x)3 = 3$”. Sure! $3 = 3$ does not depend on x. So saying “for all values of x we have $3 = 3$” is the same as saying just “we have $3 = 3$”.

4. Formulas of the form $A[t] \to (\exists x)A[x]$, for any formula A, variable x and “object t. This is a good start-up *truth*: It says that if we know that some object plugged into x makes $A[x]$ true, then it is correct to say “there is some value x that makes $A[x]$ true —in symbols $(\exists x)A[x]$.”

5. $x = x$ is the *identity* axiom, no matter what “x” I use to express it. So, $y = y$ and $w = w$ are also instances of the axiom.

6. $x = y \to y = x$ and $x = y \wedge y = z \to x = z$ are the *equality* axioms.

   They can be expressed equally well using variables other than x and y (e.g., u, v and w).

7. The $\exists$ vs. $\forall$ axiom. For any formula A, $(\exists x)A[x] \equiv \neg(\forall x)\neg A[x]$ is an axiom. □

The “rules of proving”, or rules of inference. These are two up in front —you will find I am grossly miscounting:

4.1.7 Definition. (Rules)

1. From $A[x]$ I may infer $(\forall x)A[x]$. Logicians write the up-in-front (“primary”) rules as fractions without words:

$$\frac{A[x]}{(\forall x)A[x]}$$

   this rule we call *generalisation*, or Gen for short.

[†] People usually say “for all x”, meaning for all **values** of x.

2. I may *construct* (and use) using any tautological implication *that I verified*, say, this one

$$A_1, A_2, \ldots, A_n \models_{taut} B$$

   the rule

$$\frac{A_1, A_2, \ldots, A_n}{B}$$

   Seeing readily that $A, A \rightarrow B \models_{taut} B$, we have the rule

$$\frac{A, A \rightarrow B}{B}$$

   This is a very popular rule, known as *modus ponens*, for short MP.

□

1. **HOW** do you *use* rules? See Definition 4.1.9 below. If in a proof you are writing you have reached the numerator of a rule, then *it is correct* to write next (or later) the denominator of the rule. We say that you applied the rule.

2. The second "rule" above is a rule constructor. Any tautological implication we come up with is fair game: It leads to a *valid rule* since the name of the game (in a proof) is *preservation/propagation of truth.*

   This is NOT an invitation to learn and memorise infinitely many rules (!) but is rather a license to build your own rules as you go, *as long as you bothered to verify* the validity of the tautological implication they are derived from.

3. Gen is a rule that indeed propagates truth: If $A[x]$ is true, that *means* that it is so for all values of x and all values of any other variables on which A depends but I did not show in the $[\ldots]$ notation. But then so is $(\forall x)A[x]$ true, as it says precisely the same thing: "$A[x]$ is true, for all values of x and all values of any other variables on which A depends but I did not show in the $[\ldots]$ notation".

   The only difference between the two notations is that I added some notational *emphasis* in the second —$(\forall x)$.

   For example, if I know that B has just two variables, u and v, I can write it as $B(u, v)$. Then

$$B(u, v) \ \mathbf{t} \text{ iff } (\forall u)B(u, v) \ \mathbf{t} \text{ iff } (\forall v)B(u, v) \ \mathbf{t} \text{ iff } (\forall u)(\forall v)B(u, v) \ \mathbf{t}$$

4. Hmm. So is $\forall x$ redundant? Yes, but *only as a formula prefix.* In something like this

$$x = 0 \rightarrow (\forall x)x = 0 \tag{1}$$

it is NOT redundant!

Dropping $\forall$ we change the meaning of (1).

As is, (1) is *not* a true statement. For example, for $x = 0$ it is false. However dropping $\forall x$, (1) changes to $x = 0 \to x = 0$ which is a tautology; always true.

5. The axioms in 4.1.6 are indispensable to do just logic; that is why we call them *logical axioms.*

   You also use them in *all* math reasoning no matter what type of math it is. However, the latter has its own ***additional*** axioms! These are called *special*, but most often "*mathematical axioms*".

   We are not going to list them. Why? Because every math branch, or "theory" as we say, has different axioms!

4.1.8 Example. Here is a *sample* of axioms from *math* (*theories*):

1. Number theory for $\mathbb{N}$:
   - $x < y \lor x = y \lor x > y$ (*trichotomy*)
   - $\neg x < 0$ this axiom indicates that 0 is *minimal* in $\mathbb{N}$. Adding the previous one makes $<$ a total order, so 0 is also *minimum.*
   - Many others that we omit.
2. Euclidean geometry:
   - From two distinct points passes one and only one line.
   - ("Axiom of parallels") From a point A off a line named k —both A and k being on the same plane— passes a unique line on said plane that is parallel to k.
   - Many others that we omit.
3. Axiomatic set theory:
   - For any set A,

     $$(\exists y)y \in A \to (\exists x)\Big(x \in A \land \neg(\exists z)(z \in x \land z \in A)\Big)$$

     This is the axiom of "foundation" from which one can prove things like $A \in A$ is always *false.*

     It says that *IF* there is *any* element in A *at all* —this is the hypothesis part "$(\exists y)y \in A$"— *THEN* there is some element —this is the part "$(\exists x)\Big(x \in A$"— *below which*, if you follow "$\in$" backwards from it, you will *not* find a z ("$\neg(\exists z)$") that is *both* below x along $\in$ backwards, *and* also a member of A —this part is "$(z \in x \land z \in A)$".

4. And a few others that we omit. □

So what is the *shape* of proofs?

4.1.9 Definition. (Proofs and theorems) A proof is a finite sequence of formulas

$$F_1, F_2, \ldots, F_i, \ldots, F_n \tag{1}$$

such that, for *each* $i = 1, 2, \ldots, n$, F_i is obtained as ONE of:

1. It is an axiom from among the ones we listed in 4.1.6.
2. It is an axiom of the theory (area of Math) that we are working in.
3. It is a PREVIOUSLY proved theorem.
4. It is the result of "Gen" applied to a previous formula F_j. That is, $F_i = (\forall x)F_j$, for some x and $j < i$.
5. It is the result of "$\models_{taut}$" applied to previous formulas F_{j_k}, $k = 1, 2, \ldots, m$. That is, $F_{j_1}, F_{j_2}, F_{j_3}, \ldots, F_{j_m} \models_{taut} F_i$, and all j_r for $r = 1, 2, \ldots, m$ are $< i$.

Such proofs are known as "Hilbert-style proofs". We write them vertically, ONE formula per line, every formula consecutively numbered, with annotation to the right of formulas (the "why did I write this?"). Like this

$$\begin{array}{lll}
1) & F_1 & \langle \text{because} \rangle \\
2) & F_2 & \langle \text{because} \rangle \\
\vdots & \vdots & \vdots \\
n) & F_n & \langle \text{because} \rangle
\end{array}$$

Every F_n in (1) is called a theorem. Thus we define

<u>A theorem is a formula that **appears** in a proof.</u>

Often one writes $\vdash A$ to symbolically say that A is a theorem. If we must indicate that we worked in some specific theory, say ZFC (set theory), then we may indicate this as

$$\vdash_{ZFC} A$$

If moreover we have had some "*non-axiom* assumptions" (read on to see when this happens!) that form a set Σ, then we may indicate so by writing

$$\Sigma \vdash_{ZFC} A$$

□

Why Σ for a set of (*non-axiom*) assumptions? Because we reserve upper case latin letters for formulas. For *sets* of formulas we use a *distinguishable* capital letter, so, we chose distinguishable Greek capital letters, such as $\Gamma, \Sigma, \Delta, \Phi, \Theta, \Psi, \Omega$. Obviously, Greek capital letters like A, B, E, Z will not do!

4.1.10 Example. (New (derived) rules) A derived rule is one we were not given —in 4.1.7— to bootstrap logic, but we can still prove they propagate truth.

1. We have a new (derived) rule: $(\forall x)A[x] \vdash A[t]$.

   This is called *Specialisation*, or *Spec.*

   Aha! We used a non-axiom assumption here! I write a Hilbert proof to show that $A[t]$ is a theorem if $(\forall x)A[x]$ is a (non-axiom) hypothesis (assumption) —shortened to "hyp".

1)	$(\forall x)A[x]$	$\langle$hyp$\rangle$
2)	$(\forall x)A[x] \to A[t]$	$\langle$axiom$\rangle$
3)	$A[t]$	$\langle 1 + 2 + \text{MP}\rangle$

2. Taking t to be x we have $(\forall x)A[x] \vdash A[x]$, simply written as $(\forall x)A \vdash A$.

3. The Dual Spec derived rule: $A[t] \vdash (\exists x)A[x]$. We prove it:

1)	$A[t]$	$\langle$hyp$\rangle$
2)	$A[t] \to (\exists x)A[x]$	$\langle$axiom$\rangle$
3)	$(\exists x)A[x]$	$\langle 1 + 2 + \text{MP}\rangle$

Taking t to be x we have $A[x] \vdash (\exists x)A[x]$, simply written as $A \vdash (\exists x)A$. □

There are two principles of proof that we state without proving them (see [Tou08] if curious).

4.1.11 Remark. (Deduction theorem and proof by contradiction)

1. The *deduction theorem* (also known as "proof by assuming the antecedent") states, if

   $$\Gamma, A \vdash B \tag{1}$$

   then also $\Gamma \vdash A \to B$, provided that in the proof of (1), all free variables of A were treated as constants: That is we neither used them to do a Gen, nor substituted objects into them.

   The notation "Γ, A" is standard for the more cumbersome $\Gamma \cup \{A\}$.

   In practice, this principle is applied to prove $\Gamma \vdash A \to B$, by doing instead the "easier" (1). Why easier? We are helped by an extra hypothesis, A, and the formula to prove, B, is less complex than $A \to B$.

2. Proof by contradiction. To prove $\Gamma \vdash A$ is equivalent to proving the "constant formula" $\mathbf{f}$ from hypothesis $\Gamma, \neg A$.

3. Why the burden of the non-axiom hypotheses Γ? Because in applying the deduction theorem we usually start with a task like "do $\vdash A \to B \to C \to D$".

   So we go like this:

- By DThm, it suffices to prove $A \vdash B \to C \to D$ instead (here "Γ"was $\emptyset$).
- Again, by DThm, it suffices to prove $A, B \vdash C \to D$ instead (here "Γ" was A).
- Again, by DThm, it suffices to prove $A, B, C \vdash D$ instead (here "Γ" was A, B).

□ ☡

☡☡ I referred you to [Tou08] for some things. However, the short intro here adopted the so-called "strong generalisation", which has the side-effect of making the deduction theorem to hedge: In proving B from Γ, A one *must* ensure that no variable of A was subject to generalisation or substitution. [Tou08] trades some power of generalisation in order to get an easier to apply deduction theorem, with no hedging.

So this is a choice on what we want to be "easy", and what we want to "not be so easy". There are two options! ☡☡

4.1.12 Remark. (Ping-Pong) For any formulas A and B, the formula —where I am using way more brackets than I have to, ironically, to *improve* readability—

$$(A \equiv B) \equiv \Big((A \to B) \land (B \to A)\Big)$$

is a tautology (draw up a truth table with one row for each of the possible values of A and B and verify that the equivalence is always $\mathbf{t}$).

Thus to prove the lhs of the $\equiv$ suffices to prove the rhs:

	$\vdots$	$\vdots$
1)	$(A \to B) \land (B \to A)$	⟨suppose I proved this⟩
2)	$(A \equiv B) \equiv \Big((A \to B) \land (B \to A)\Big)$	⟨axiom⟩
3)	$A \equiv B$	⟨1 + 2 + tautological implication⟩

In turn, to prove the rhs it suffices to prove each of $A \to B$ and $B \to A$ separately. This last idea encapsulates the ping-pong approach to proving equivalences.

Here are a few applications. □

4.1.13 Example. 1. Establish $\vdash (\forall x)(A \land B) \equiv (\forall x)A \land (\forall x)B$.

By ping-pong.

- Prove $\vdash (\forall x)(A \land B) \to (\forall x)A \land (\forall x)B$. By DThm suffices to do $(\forall x)(A \land B) \vdash (\forall x)A \land (\forall x)B$ instead.

1)	$(\forall x)(A \land B)$	⟨hyp⟩
2)	$A \land B$	⟨1 + Spec⟩

3)	A	$\langle$2 + tautological implication$\rangle$
4)	B	$\langle$2 + tautological implication$\rangle$
5)	$(\forall x)A$	$\langle$3 + Gen; OK: x is not free in line 1$\rangle$
6)	$(\forall x)B$	$\langle$4 + Gen; OK: x is not free in line 1$\rangle$
7)	$(\forall x)A \wedge (\forall x)B$	$\langle$5 + 6 + tautological implication$\rangle$

- Prove $\vdash (\forall x)A \wedge (\forall x)B \to (\forall x)(A \wedge B)$. By DThm suffices to do $(\forall x)A \wedge (\forall x)B \vdash (\forall x)(A \wedge B)$ instead.

1)	$(\forall x)A \wedge (\forall x)B$	$\langle$hyp$\rangle$
2)	$(\forall x)A$	$\langle$1 + tautological implication$\rangle$
3)	$(\forall x)B$	$\langle$1 + tautological implication$\rangle$

Complete the above proof!

2. Prove $\vdash (\forall x)(\forall y)A \equiv (\forall y)(\forall x)A$. By ping-pong.

- Prove $\vdash (\forall x)(\forall y)A \to (\forall y)(\forall x)A$.
  By DThm suffices to do $(\forall x)(\forall y)A \vdash (\forall y)(\forall x)A$ instead.

1)	$(\forall x)(\forall y)A$	$\langle$hyp$\rangle$
2)	$(\forall y)A$	$\langle$1 + Spec$\rangle$
3)	A	$\langle$2 + Spec$\rangle$
4)	$(\forall x)A$	$\langle$3 + Gen; OK, no free x in line 1$\rangle$
5)	$(\forall y)(\forall x)A$	$\langle$4 + Gen; OK, no free y in line 1$\rangle$

- Prove $\vdash (\forall y)(\forall x)A \to (\forall x)(\forall y)A$.
  Exercise! □

We have seen how to *add* an $(\exists x)$ in front of a formula (4.1.10 3.).

How about *removing* an $(\exists x)$-prefix? This is much more complex than removing a $(\forall x)$-prefix:

The technique can be *proved* to be correct (eg., [Tou03a]) but I will omit the proof here as I did omit the proof of the deduction theorem technique and the proof by contradiction technique. I could say "see [Tou03a] if you want to learn the proof", but this reference is too advanced for a first year course on discrete math. So, why not look at [Tou08]? These two books have chosen *incompatible* "generalisation" rules, which results to *incompatible* deduction theorem versions.

The proof of the technique of eliminating $\exists$-prefixes *relies on the deduction theorem.*

Technique of removing an $\exists$-prefix: Suppose I have that $(\exists x)A[x]$ is true — either as an **assumption** or a **theorem I proved earlier**— and I want to prove B.

Then I **assume** that —for *some* constant c that does not occur in B— $A[c]$ is true.

That is, I **add** $A[c]$ for an unknown c *NOT* in B as a non-axiom hypothesis.

People annotate this step in a proof as “aux. hyp. caused by $(\exists x)A[x]$.”

Now proceed to prove B using all that is known to you —that is, the axioms of the theory $\mathcal{T}$ that you work in, perhaps some non-axiom hypotheses Γ, and $(\exists x)A[x]$, *and the non-axiom hypothesis* $A[c]$.

Do so by using all free (input-) variables of $A[c]$ as constants in your proof![†]

The *technique of removing an* $\exists$*-prefix* guarantees that you did better than

$$\Gamma, (\exists x)A[x], \boxed{A[c]} \vdash_{\mathcal{T}} B$$

that actually you achieved

$$\Gamma, (\exists x)A[x] \vdash_{\mathcal{T}} B$$

as if you never assumed nor used $A[c]$!

That is why they call it “auxiliary hypothesis”. Once it helps you prove B it drops out; it does not stay around to get credit!

4.1.14 Example. Prove $\vdash (\exists y)(\forall x)A[x,y] \to (\forall x)(\exists y)A[x,y]$.

By the DThm it suffices to prove $(\exists y)(\forall x)A[x,y] \vdash (\forall x)(\exists y)A[x,y]$ instead.

1)	$(\exists y)(\forall x)A[x,y]$	$\langle$hyp$\rangle$
2)	$(\forall x)A[x,c]$	$\langle$aux. hyp. caused by 1; for some constant c not in the conclusion$\rangle$
3)	$A[x,c]$	$\langle$2 + Spec$\rangle$
4)	$(\exists y)A[x,y]$	$\langle$3 + Dual Spec$\rangle$
5)	$(\forall x)(\exists y)A[x,y]$	$\langle$4 + Gen; OK, no free x in lines 1 and 2$\rangle$

□

4.1.15 Example. Can I also prove the converse of the above? That is $\vdash (\forall x)(\exists y)A[x,y] \to (\exists y)(\forall x)A[x,y]$.

I will try.

By the DThm it suffices to prove $(\forall x)(\exists y)A[x,y] \vdash (\exists y)(\forall x)A[x,y]$ instead.

1)	$(\forall x)(\exists y)A[x,y]$	$\langle$hyp$\rangle$
2)	$(\exists y)A[x,y]$	$\langle$1 + spec$\rangle$
3)	$A[x,c]$	$\langle$aux. hyp. for 2; c not in the conclusion$\rangle$
4)	$(\forall x)A[x,c]$	$\langle$3 + Gen; Hmmm! Illegal: I should treat the free x of aux. hyp. as a constant!$\rangle$

Still, can anyone PROVE this even if I cannot?

[†] This is a side-effect of using the deduction theorem in the proof of correctness of the technique.

A question like this, if you are to answer "NO", must be resolved by offering a *counterexample.* That is, a special case of A for which I can clearly see that the claim is not true.

Here is one such:

$$(\forall x)(\exists y)x = y \to (\exists y)(\forall x)x = y \tag{1}$$

Say we work in $\mathbb{N}$. The lhs of → is true, but the rhs is false as it claims that there is a number such that *all* numbers are equal to it. □

Another useful principle that can be proved, but we will not do so, is that one can *replace equivalents-by-equivalents.* That is, if C is some formula, and if I have

1. $A \equiv B$, via proof, or via assumption, and also
2. A is a subformula of C

then I can *replace* one (or more) occurrence(s) of A in C (as subformula(s)) by B and call the resulting formula C', and be guaranteed the conclusion $C \equiv C'$. That is, from $A \equiv B$, I can prove $C \equiv C'$.

This principle is called the *equivalence theorem.*

Let's do a couple of ad hoc additional examples before we move to the section on Induction.

4.1.16 Example. $A \to B \vdash (\forall x)A \to (\forall x)B$.

By the DThm it suffices to prove $A \to B, (\forall x)A \vdash (\forall x)B$ instead.

1)	$A \to B$	$\langle$hyp$\rangle$
2)	$(\forall x)A$	$\langle$hyp$\rangle$
3)	A	$\langle 2$ + Spec$\rangle$
4)	B	$\langle 1 + 3$ + MP$\rangle$
5)	$(\forall x)B$	$\langle 4$ + Gen; OK as the DThm hyp. (line 2) has no free $x\rangle$

□

4.1.17 Example. Refer to 4.1.6(7). Let us apply it to $\neg A$ for arbitrary A. We get

$$\vdash (\exists x)\neg A \equiv \neg(\forall x)\neg\neg A \tag{1}$$

Pause. Why "$\vdash$"?◀

Since $A \equiv \neg\neg A$ is a tautology, hence a theorem

Pause. Why "hence a theorem"?◀

we apply the equivalence theorem above and tautological implication to obtain from (1):

$$\vdash (\exists x)\neg A \equiv \neg(\forall x)A \tag{2}$$

Applying another tautological implication to (2) we get

$$\vdash (\forall x)A \equiv \neg(\exists x)\neg A$$

which is of the same form as 4.1.6(7) with the roles of $\exists$ and $\forall$ reversed. □

4.1.18 Example. $A \equiv B \vdash (\forall x)A \equiv (\forall x)B$.

True due to the equivalence theorem! "C" is "$(\forall x)A$". We replaced (one occurrence of) A by B in C, and we have assumed as starting point that $A \equiv B$. □

4.1.19 Exercise. Prove $A \equiv B \vdash (\forall x)A \equiv (\forall x)B$ without relying on the equivalence theorem. Rather use 4.1.16 in your proof, remembering the ping-pong tautology (4.1.12). □

4.2. Induction

In Remark 3.1.77 we concluded with a formulation of the *minimal condition* (MC) for any order $<$ as follows:

> An order $<$ on a class $\mathbb{A}$ has MC is captured by the statement
>
> *For any "property", that is, formula $F[x]$ —recall that this notation,* square brackets*, indicates our interest in* one *among the, possibly many, free variables of F— we have that the following is true*
>
> $$(\exists a)F[a] \to (\exists a)\Big(F[a] \wedge \neg(\exists y)\big(y < a \wedge F[y]\big)\Big) \tag{1}$$

So let $<$ be the standard order on $\mathbb{N}$. We have used the fact that it is a total order (satisfies trichotomy) and that every nonempty subset of $\mathbb{N}$ has a minimal —hence unique minimum— element.

Pause. Why *unique* and *minimum*?◀

So let us fix in the rest of this section $<$ to be the "less than" order on $\mathbb{N}$, until we indicate otherwise.

Let us rewrite (1) for $\neg P[x]$ where $P[x]$ is arbitrary. We get the theorem

$$(\exists x)\neg P[x] \to (\exists x)\Big(\neg P[x] \wedge \neg(\exists y)\big(y < x \wedge \neg P[y]\big)\Big) \tag{2}$$

Using the equivalence theorem (p.90) and the 7, we obtain from (2)

$$\neg(\forall x)P[x] \to \neg(\forall x)\neg\Big(\neg P[x] \wedge (\forall y)\neg\big(y < x \wedge \neg P[y]\big)\Big)$$

and then (the tautology known as "contrapositive" is used) also

$$(\forall x)\neg\Big(\neg P[x] \wedge (\forall y)\neg\big(y < x \wedge \neg P[y]\big)\Big) \to (\forall x)P[x]$$

Using the tautology

$$\neg(A \wedge B) \equiv \neg A \vee \neg B$$

and the equivalence theorem, we transform the above to this theorem:

$$(\forall x)\Big(P[x] \vee \neg(\forall y)\big(\neg y < x \vee P[y]\big)\Big) \to (\forall x)P[x]$$

Again, this time using the tautology

$$\neg A \vee B \equiv A \to B$$

(twice) and the equivalence theorem, we transform the above to this theorem:

$$(\forall x)\Big((\forall y)\big(y < x \to P[y]\big) \to P[x]\Big) \to (\forall x)P[x] \tag{3}$$

(3) is the principle of *strong induction*, or *complete induction*, or *course-of-values induction* that you probably encountered at school, and the above work shows that *it is equivalent to the least principle*! (Clearly we can reverse all the steps we took above as all were equivalences!)

Let us render (3) more recognisable: By applying MP (elaborate this!) I can transform (3) in "rule of inference form", indeed I will write it like a rule that says, like all rules do, "if you proved my numerator, then my denominator is also proved!"

$$\frac{(\forall x)\Big((\forall y)\big(y < x \to P[y]\big) \to P[x]\Big)}{(\forall z)P[z]}$$

Dropping the ∀-prefix we have the rule in the form:

$$\frac{(\forall y)\big(y < x \to P[y]\big) \to P[x]}{P[z]} \tag{CVI}$$

"(CVI)" for Course-of-Values Induction. (CVI) says

To prove $P[x]$ (for all x is implied!) **do as follows**:

Step (a) Fix an **arbitrary** x-value. Now, **assume** $(\forall y)\big(y < x \to P[y]\big)$ for said x. We call the assumption **Induction Hypothesis**, for short, **I.H.**

Step (b) Next **prove** $P[x]$, for the same fixed unspecified x. This proof step we call the **Induction Step** or **I.S.**

Note that what is described by (a) and (b) is precisely an application of the Deduction theorem towards proving "If, for all $y < x$, $P[y]$ is true, then $P[x]$ is true", that is, **proving the implication on the numerator of (CVI) for any given** x.

Step (c) If you have done **Step** (a) and **Step** (b) above, then you **have proved** $P[x]$ (for all x is implied!)

Important.

- **Step** (a) above says "**arbitrary** x".

  So, I should *not* leave any x-value out of the proof!

  But how do I prove the I.S. for $x = 0$? There is no I.H. to rely one (no numbers below $x = 0$). No problem: The numerator implication in (CVI) now reads

  $$(\forall y)\big(y < 0 \to P[y]\big) \to P[0]$$

  The lhs of "→" is true since $y < 0$ is false. Thus, to ensure the truth of the *implication* I must prove $P[0]$.

  This step was hidden in **Steps** (a) – (b) above. It is called the **Basis** of the induction!

- The I.H. is usually stated in English: **Assume** $P[y]$ (true), for *all* $y < x$.

Above we admitted much less than what we actually proved. $\mathbb{N}$ does *not* have the monopoly of the CVI methodology in proofs! So let us shift gear and have $<$ indicate in the corollary below an arbitrary order with MC on an arbitrary set A —*not* a set of numbers necessarily.

4.2.1 Corollary. *If $(A, <)$ is a POset with MC, then we can prove a property $P[x]$, for all $x \in A$, by doing precisely the steps of CVI:*

1. *Prove/verify $P[a]$, for every $<$-minimal member of A. This is the* Basis.
2. *Fix an arbitrary b and assume $P[x]$, for all $x < b$. This is the* I.H.
3. *Finally, do the* I.S.: *For the fixed b in 2. prove $P[b]$ using 1. and 2.*

Proof. Nothing changes in the derivation of the equivalence between MC and CVI above. Just forget the opening line "So let $<$ be the standard order on $\mathbb{N}$."!

The only change is in *applying* CVI in the general case is in the *Basis* step: Instead of proving/verifying $P[0]$ for the (unique) *minimum* element of $\mathbb{N}$, we prove/verify $P[x]$ for *all minimal* elements of A, which may be infinitely many! □

There is another simpler induction principle that we call it, well, *simple* induction:

$$\frac{P[0], P[x] \to P[x+1]}{P[x]} \qquad (SI)$$

"(SI)" for Simple Induction. That is, to prove $P[x]$ for all x (denominator) do *three* things:

Step 1. Prove/verify $P[0]$

Step 2. **Assume** $P[x]$ for fixed ("frozen") x (unspecified!).

Step 3. **prove** $P[x+1]$ for that same x. The assumption is the I.H. for simple induction. The I.S. is the step that proves $P[x+1]$.

Note that what is described here is precisely an application of the Deduction theorem towards proving "$P[x] \to P[x+1]$", that is, **proving the implication for any given** x.

Step 4. If you have done **Step** 1. through **Step** 3. above, then you **have proved** $P[x]$ (for all x is implied!)

Is the principle (SI) *correct*? I.e., if I do all that the numerator of (SI) asks me to do (or **Steps** 1. – 3.), then do I *really* get that the denominator is true (for all x implied)?

4.2.2 Theorem. *The validity of (SI) is a consequence of MC on $\mathbb{N}$.*

Proof. Suppose (SI) is *not* correct. Then, for some property $P[x]$, despite having completed **Steps** 1. – 3., yet, $P[x]$ is *not true* for all x!

Well, if so, let $n \in \mathbb{N}$ be *smallest* such that $P[n]$ is *false*. Now, $n > 0$ since I *did* verify the truth of $P[0]$ (**Step** 1.). Thus, $n - 1 \geq 0$. But then, when I proved "$P[x] \to P[x+1]$ for all x (in $\mathbb{N}$)" —in **Steps** 2. and 3.— this includes **proving** the case

$$P[n-1] \to P[n] \tag{4}$$

But by the smallest-ness of n, $P[n-1]$ is *true*, hence $P[n]$ is true by the truth table of "$\to$". I have just got a contradiction! I conclude that no such smallest n exists, i.e., $P[x]$ is true (for all $x \in \mathbb{N}$). (SI) works! □

How do the simple and course-of-values induction relate? They are equivalent tools! Here is why:

4.2.3 Theorem. *From the validity of (SI) I can obtain the validity of (CVI).*

Proof. Suppose that I have

verified the numerator of (CVI), for $P[x]$, via **Steps** (a) and (b) p.93 (†)

but let me pretend that

I do not know if doing so guarantees the truth of the denominator, $P[x]$ (‡)

Let me show that it does, by doing simple induction SI using a related property, $Q[x]$.

I define $Q[x]$, for all x in $\mathbb{N}$, by

$$Q[x] \overset{Def}{\equiv} P[0] \wedge P[1] \wedge \ldots \wedge P[x] \tag{5}$$

Now, as we emphasised on p.92, "property" is colloquial for *formula*. But formulas do *not* have variable length! The length of $Q[x]$ above increases or decreases with the value of its input n. Well, (5) is also a colloquialism to keep things intuitively clear! The mathematically correct definition of Q is the following,

$$Q[x] \overset{Def}{\equiv} (\forall z)(z < x \to P[z]) \tag{5$'$}$$

but now that the point has been made, I will continue using the form (5).

So, my job is to show that

if for some property $P[x]$ I proved the truth of the numerator of (CVI), then

it is guaranteed that $P[x]$ is *true*, for all x (6)

I prove this by showing property $Q[x]$ is true, for all x, using SI.

To this end I have to do

SI 1) Verify $Q[x]$ for $x = 0$ (Basis). But $Q[0]$ —by (5)— is just $P[0]$, which I proved *true* as part of my due Basis for CVI (blue underlined if-clause above).

SI 2) For $x > 0$, show,

$$Q[x-1] \to Q[x] \text{ is true} \tag{7}$$

I argue that I already showed (7) by proving the CVI numerator:

- I proved
$$P[0] \wedge P[1] \wedge \ldots \wedge P[x-1] \to P[x]$$
- By tautological implication from the above I get also
$$P[0] \wedge P[1] \wedge \ldots \wedge P[x-1] \to P[0] \wedge P[1] \wedge \ldots \wedge P[x-1] \wedge P[x]$$
- But the above says $Q[x-1] \to Q[x]$ is true. This is (7).

By SI, I have proved $Q[x]$ is true, for all x. But by (5), this trivially implies that $P[x]$ is true, for all x. I proved (6). □

4.2.4 Remark.

1. So, for $\mathbb{N}$, MC, CVI and SI **are all equivalent**. We have already indicated that MC and CVI are equivalent. The work on CVI vs. SI (4.2.3) and SI vs. MC (4.2.2) is summarised as
$$MC \Longrightarrow SI \Longrightarrow CVI \Longrightarrow MC$$
which establishes the equivalence claim about all three.

2. When do I use CVI and when SI? SI is best to use when to prove $P[x]$ (in the I.S.) I only need to know $P[x-1]$ is true. CVI is used when we need a more flexible I.H. that $P[n]$ is true for all $n < x$. See the examples below!

3. "0" is the boundary case if the claim we are proving is valid "for all $n \in \mathbb{N}$", or simply put, "for $n \geq 0$". If the claim is "for all $n \geq a$, $P[n]$ is true" then usually $P[n]$ is meaningless for $x < a$ and thus the Basis is for $n = a$. □

4.2.5 Example. This is the "classical first example of induction use" in the discrete math bibliography! Prove that

$$0 + 1 + 2 + \ldots + n = \frac{n(n+1)}{2} \tag{1}$$

So, the property to prove is the entire expression (1). On must learn to not have to rename the "properties to use" as "$P[n]$".

I will use SI. So let us do the *Basis*. Boundary case is $n = 0$. We verify: $lhs = 0$. $rhs = (0.1)/2 = 0$. Good!

Fix n and tale the expression (1) as I.H.

Do the I.S. Prove:

$$0+1+2+\ldots+n+(n+1)=\frac{(n+1)(n+2)}{2}$$

Here it goes

$$\begin{aligned}0+1+2+\ldots+n+(n+1) &\overset{\text{using I.H.}}{=} \frac{n(n+1)}{2}+(n+1)\\ &=(n+1)(n/2+1)\\ &=\frac{(n+1)(n+2)}{2}\end{aligned}$$

□

I will write more concisely in the examples that follow.

4.2.6 Example. Same as above but doing away with the "0+". Again, I use SI.

$$1+2+\ldots+n=\frac{n(n+1)}{2} \tag{1}$$

- *Basis.* $n=1$: (1) becomes $1=(1.2)/2$. True.
- Take (1) as I.H. with fixed n.
- I.S.:

$$\begin{aligned}1+2+\ldots+n+(n+1) &\overset{\text{using I.H.}}{=} \frac{n(n+1)}{2}+(n+1)\\ &=(n+1)(n/2+1)\\ &=\frac{(n+1)(n+2)}{2}\end{aligned}$$

□

4.2.7 Example. Prove

$$1+2+2^2+\ldots 2^n=2^{n+1}-1 \tag{1}$$

By SI.

- Basis. $n=0$. $1=2^0=2^1-1$. True.
- As I.H. take (1) for fixed n.
- I.S.

$$\begin{aligned}1+2+2^2+\ldots+2^n+2^{n+1} &\overset{\text{using I.H.}}{=} 2^{n+1}-1+2^{n+1}\\ &=2\cdot 2^{n+1}-1\\ &=2^{n+2}-1\end{aligned}$$

□

4.2.8 Example. An inequality! I prove that

$$n < 2^n \tag{1}$$

for all $n \geq 0$.

I do SI on n.

- *Basis.* $0 < 2^0 = 1$ is true.
- As I.H. fix n and assume (1).
- For the I.S. we have $2^{n+1} = 2^n + 2^n$. By the I.H. $2^n > n$ but also $2^n \geq 1$. Thus, adding these two inequalities I get

$$2^{n+1} = 2^n + 2^n > n + 1$$

□

4.2.9 Example. (Euclid) Every natural number $n \geq 2$ is expressible as a product of primes.

A "product" includes the trivial case of **one** factor.

I do CVI (as you will see why!)

- *Basis*: For $n = 2$ we are done since 2 is a prime.[†]
- I.H. Fix an n and assume the claim for all k, such that $2 \leq k < n$.
- I.S.: Prove for n: Two subcases:
  1. If n is prime, then nothing to prove! Done.
  2. If not, then $n = a \cdot b$, where $a \geq 2$ **and** $b \geq 2$. By I.H.[‡] each of a and b are products of primes, thus so is $n = a \cdot b$. □

4.2.10 Example. (Euclid) Every natural number $n \geq 0$ is expressible base-10 as an expression

$$n = a_m 10^m + a_{m-1} 10^{m-1} + \cdots + a_1 10 + a_0 \tag{1}$$

$$\text{where each } a_i \text{ satisfies } 0 \leq a_i < 10 \tag{2}$$

Proof by CVI again. You will see why.

- *Basis.* For $n = 0$ the expression "0" has the form of the rhs of (1) *and* satisfies inequality (2).

[†] You will recall that a number $\mathbb{N} \ni n > 1$ is a *prime* iff its **only** factors are 1 and n.

[‡] You see? a and b cannot be both $n - 1$ to apply SI's I.H. In fact, if $n = (n-1)^2$, then $n = n^2 - 2n + 1$ or $n^2 - 3n + 1 = 0$. This equation has no natural number roots! So SI would *not* help with its rigid I.H.

- Fix an $n > 0$ and assume (I.H.) that if $k < n$, then k can be expressed as in (1).

- For the I.S. express the n of the I.H. using Euclid's theorem (3.1.47) as

  $$n = 10q + r$$

  where $0 \le r < 10$. By the I.H. —since $q < n$— let

  $$q = b_t 10^t + b_{t-1} 10^{t-1} + \cdots + b_1 10 + b_0$$

  with $0 \le b_j < 10$.

  Then

  $$\begin{aligned} n &= 10q + r \\ n &= 10\Big(b_t 10^t + b_{t-1} 10^{t-1} + \cdots + b_1 10 + b_0\Big) + r \\ n &= b_t 10^{t+1} + b_t 10^t + \cdots + b_1 10^2 + b_0 10 + r \end{aligned}$$

We see n has the right form since $0 \le r < 10$. □

4.2.11 Example. Another inequality. Let p_n denote the n-th prime number, for $n \ge 0$. Thus $p_0 = 2$, $p_1 = 3$, $p_2 = 5$, etc.

We prove that

$$p_n \le 2^{2^n} \tag{1}$$

I use CVI on n. This is a bit of a rabbit out of a hat if you never read Euclid's proof that there are infinitely many primes.

- Basis $p_0 = 2 \le 2^{2^0} = 2^1 = 2$.

- Fix $n > 0$ and take (1) as I.H.

- The I.S.: I will work with the fixed n above and the expression (product of primes, plus 1; this is inspired from Euclid's proof quoted above).

  $$p_0 p_1 p_2 \cdots p_n + 1$$

  By the I.H. I have

  $$\begin{aligned} p_0 p_1 p_2 \cdots p_n + 1 &\le 2^{2^0} 2^{2^1} 2^{2^2} \cdots 2^{2^n} + 1 && \text{by I.H.} \\ &= 2^{2^0+2^1+2^2+\cdots+2^n} + 1 && \text{algebra} \\ &= 2^{2^{n+1}-1} + 1 && \text{by 4.2.7} \\ &= 2^{2^{n+1}-1} + 2^{2^{n+1}-1} && \text{smallest } n \text{ possible is } n = 1 \\ &= 2^1 \cdot 2^{2^{n+1}-1} \\ &= 2^{2^{n+1}} \end{aligned}$$

  Now we have two cases on $q = p_0 p_1 p_2 \cdots p_n + 1$

1. q is a prime. Because of the "$+1$" q is different from all p_i in the product, so q is p_{n+1} or p_{n+2} or p_{n+3} or ...

   Since the sequence of primes is strictly increasing, p_{n+1} is the least that q can be.

   Thus
   $$p_{n+1} \le p_0 p_1 p_2 \cdots p_n + 1 \le 2^{2^n}$$
   in this case.

2. q is composite. By 4.2.9 some prime r divides q. Now, none of the
   $$p_0, p_1, p_2, \cdots, p_n$$
   divides q because of the "$+1$". Thus r is different from all of them, so it must be one of p_{n+1} or p_{n+2} or p_{n+3} or ...

   Thus,
   $$p_{n+1} \le r < q = p_0 p_1 p_2 \cdots p_n + 1 \le 2^{2^n}$$

Done! □

4.2.12 Example. Let

$$\begin{aligned} b_1 &= 3,\ b_2 = 6 \\ b_k &= b_{k-1} + b_{k-2},\ \text{for } k \ge 3 \end{aligned}$$

Prove by induction that b_n $\underline{\text{is divisible by 3 for } n \ge 1}$. (Be careful to distinguish between what is *basis* and what are *cases* arising from the **induction step**! As you know, our text is careless about this.)

Proof. So the boundary condition is (from the underlined part above) $n = 1$. This is the *Basis*.

1. *Basis*: For $n = 1$, I have $a_1 = 3$ and this is divided by 3. We are good.

2. *I.H.* Fix n and **assume claim** for all $k < n$.

3. *I.S.* **Prove claim** for the above fixed n. There are two cases, as the I.H. is *not useable* for $n = 2$. Why? Because it would require entries b_0 and b_1. The red entry does not exist since the sequence starts with b_1. So,

   Case 1. $n = 2$. Then I am OK as $b_2 = 6$; it *is* divisible by 3.

   Case 2. $n > 2$. Is b_n divisible by 3? Well, $b_n = b_{n-1} + b_{n-2}$ in this case. By I.H. (valid for all $k < n$) I have that $b_{n-1} = 3t$ and $b_{n-2} = 3r$, for some integers t, r. Thus, $b_n = 3(t + r)$. Done! □

Here are a few additional exercises for you to try —please do try!

4.2.13 Exercise.

1. Prove that $2^{2n+1} + 3^{2n+1}$ is divisible by 5 for all $n \ge 0$.

2. Using induction prove that $1^3 + 2^3 + \ldots + n^3 = \left[\dfrac{n(n+1)}{2}\right]^2$, for $n \geq 1$.

3. Using induction prove that $\sum_{i=1}^{n+1} i2^i = n2^{n+2} + 2$, for $n \geq 0$.

4. Using induction prove that $\sqrt{n} < \dfrac{1}{\sqrt{1}} + \dfrac{1}{\sqrt{2}} + \ldots + \dfrac{1}{\sqrt{n}}$, for $n \geq 2$.

5. Let

$$\begin{aligned} b_0 &= 1,\, b_1 = 2,\, b_3 = 3 \\ b_k &= b_{k-1} + b_{k-2} + b_{k-3}, \text{ for } k \geq 3 \end{aligned}$$

Prove by induction that $b_n \leq 3^n$ for $n \geq 0$. (Once again, be careful to distinguish between what is *basis* and what are *cases* arising from the **induction step**!) □

4.3. Inductive definitions

Inductive definitions are increasingly being renamed to “recursive definitions” in the modern literature, thus using “recursive” for *definitions*, and “induction” for *proofs*. I will not go out of my way to use this dichotomy of nomenclature.

4.3.1 Example.

$$\begin{aligned} a^0 &= 1 \\ a^{n+1} &= a \cdot a^n \end{aligned}$$

is an example of an inductive (recursive) definition of the non-negative integer powers of a non zero number a. □

4.3.2 Example. Another example is the Fibonacci sequence,[†] given by

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1 \\ &\text{and for } n \ge 1 \\ F_{n+1} &= F_n + F_{n-1} \end{aligned}$$

Unlike the function (sequence) $a^0, a^1, a^2, a^3, \ldots$, for which we only need the value at n to compute the value at $n+1$, the Fibonacci function needs two previous values, at $n-1$ and at n, to compute the value at $n+1$. □

This section looks at inductive/recursive definitions in general, but for functions whose left field is $\mathbb{N}$ or $\mathbb{N}^{n+1}$ for some fixed n.

4.3.3 Definition. We consider in this section a general recursive definition of a function $G : \mathbb{N}^{n+1} \to A$, for a given $n \ge 0$ and set A.

This definition has the form (1) below.

Two total functions are *given*.

1. $H : \mathbb{N}^n \to A$, where A is some set. The typical *call* to H looks like $H(\mathbf{b})$ where $\mathbf{b} \in \mathbb{N}^n$. If $n = 0$, then we do *not* have any arguments for H. In this case H is just a *constant* (i.e., a fixed element of A).
2. $K : \mathbb{N}^{n+1} \times 2^A \to A$. The typical *call* to K looks like $K(m, \mathbf{b}, z)$ where $m \in \mathbb{N}$, $\mathbf{b} \in \mathbb{N}^n$ and z is a subset of A. If $n = 0$ then we do *not* have the argument $\mathbf{b}$.

   We will explore below whether the following definition (1) indeed yields a *function* $G : \mathbb{N}^{n+1} \to A$ of arguments a and $\mathbf{b}$ where $a \in \mathbb{N}$ and $\mathbf{b} \in \mathbb{N}^n$. If $n = 0$, then we do *not* have the argument $\mathbf{b}$, rather we will have just one argument in G: $a \in \mathbb{N}$.

[†] The “sequence” $F_0, F_0, F_0, \ldots$ is, of course, a total function from $F : \mathbb{N} \to \mathbb{N}$.

$$\begin{aligned} G(a,\mathbf{b}) &= H(\mathbf{b}) \\ G(a+1,\mathbf{b}) &= K\Big(a,\mathbf{b},\Big\{G(0,\mathbf{b}),G(1,\mathbf{b}),\ldots,G(a,\mathbf{b})\Big\}\Big) \end{aligned} \tag{1}$$

□

4.3.4 Remark. The notation of the set-argument

$$\Big\{G(0,\mathbf{b}),G(1,\mathbf{b}),\ldots,G(a,\mathbf{b})\Big\} \tag{2}$$

in (1) above is *way less* informative than the notation implies! Its members —listed again in (2)— can be put in *any* order and *there are no markings on any of these members of* A that will reveal the 1st argument of G (the position of the call $G(i,\mathbf{b})$ in the sequence as presented in (2)). So we should not read (2) as if it conveys position!

Pause. Well, why not instead of using a *set*-argument write instead

$$K\Big(a,\mathbf{b},G(0,\mathbf{b}),G(1,\mathbf{b}),\ldots,G(a,\mathbf{b})\Big)$$

that is, have each call to $G(i,\mathbf{b})$ explicitly "coded" in the function K? Because I cannot have a variable number of arguments!◄

This is *no problem in practise*. In any specific application of the **definition form** (1) the structure of K can be chosen/built so that it will "know and choose" what recursive calls it needs to make —in which order and for which arguments— to compute $G(a+1,\mathbf{b})$.

For example, the specific use of principle (1) to the Fibonacci function definition 4.3.2 has chosen that to compute F_{n+1} it will always call just F_n and F_{n-1} from the entire "history at input n" —namely, $\{F_0,F_1,F_2,\ldots,F_n\}$— and then return the sum of the call results.

So the notation (1) (via (2)) simply conveys —for the benefit of our two theorems coming up below— that *in general* an inductive definition (1) might call recursively as many as all the $G(i,\mathbf{b})$ in (2) to compute $G(a+1,\mathbf{b})$.

BTW, there are complicated inductive definitions such that the recursive calls are not always at fixed (argument-)positions to the left of "$a+1$", unlike the Fibonacci recursive definition that computes F_{n+1}, for any $n\geq 1$, by always calling the function recursively with arguments *at precisely the numbers before* $n+1$. These complicated cases will choose which $G(i,\mathbf{b})$ from among the history (2) to call, depending on the value of $a+1$ □

4.3.5 Lemma. *Let $n\geq 1$. If we define the order $\prec$ on $\mathbb{N}^{n+1}$ by $(a,\mathbf{b})\prec(a',\mathbf{b}')$ iff $a<a'$ and $\mathbf{b}=\mathbf{b}'$, then $\prec$ is an order that has MC on $\mathbb{N}^{n+1}$.*

Proof.

1. $\prec$ is an order:
   - Indeed, if $(a,\mathbf{b})\prec(a,\mathbf{b})$, then $a<a$ which is absurd.

- If $(a, \mathbf{b}) \prec (a', \mathbf{b}') \prec (a'', \mathbf{b}'')$, then $\mathbf{b} = \mathbf{b}' = \mathbf{b}''$ and $a < a' < a''$. Thus $a < a''$ and hence $(a, \mathbf{b}) \prec (a'', \mathbf{b}'')$.

2. $\prec$ has MC: So let $\emptyset \neq A \subseteq \mathbb{N}^{n+1}$. Let a be $<$-minimum in $S = \{x : (\exists \mathbf{b})(x, \mathbf{b}) \in A\} \subseteq \mathbb{N}$.

   Pause. Why is $S \neq \emptyset$?◄

   Let $\mathbf{c}$ be such that $(a, \mathbf{c}) \in A$. This $(a, \mathbf{c})$ is $\prec$-minimal in A. Otherwise for some d, $A \ni (d, \mathbf{c}) \prec (a, \mathbf{c})$. Hence $d < a$, but this is a contradiction since $d \in S$ (why?). □

The minimal elements of $\prec$ are of the form $(0, \mathbf{b}), (0, \mathbf{b}'), (0, \mathbf{b}''), \ldots$, which are not comparable if they have distinct "**b**-parts". Thus they are infinitely many.

4.3.6 Lemma. *Let $(Y, <)$ be a POset with MC —where I use "$<$" generically, not as the one on $\mathbb{N}$.*

Then, for any subset $\emptyset \neq B$ of Y, $(B, <)$ is a POset with MC.

Proof. We show two things:

1. $(B, <)$ is a POset.

   $<$ is irreflexive on Y, hence it is trivially so on any subset of Y. Transitivity too is inherited from that of $<$ on Y, since if x, y, z are in B and we have $x < y < z$, then x, y, z are in Y and we still have $x < y < z$. Hence $x < z$ is true.

2. Let $\emptyset \neq S \subseteq B$. Now S —viewed as a subset of Y— has a $<$-*minimal* member m. We cannot have $x < m$ with $x \in S$ in $(B, <)$ since then we have $x < m$ with $x \in S$ in $(Y, <)$. □

4.3.7 Theorem. *If there is a function $G : \mathbb{N}^{n+1} \to A$ satisfying (1) of 4.3.3, then it is unique.*

Proof. Suppose we have two such functions, G and G' that satisfy (1) for **given** H and K. If G and G' differ, then there is an argument $(a, \mathbf{b})$ such that $G(a, \mathbf{b}) \neq G'(a, \mathbf{b})$ then there is —by Lemma 4.3.5— a $\prec$-*minimal* such argument, say, $(m, \mathbf{c})$, in the set $T = \{(a, \mathbf{b}) : G(a, \mathbf{b}) \neq G'(a, \mathbf{b})\}$. So

$$G(m, \mathbf{c}) \neq G'(m, \mathbf{c}) \tag{$*$}$$

Now, $(m, \mathbf{c})$ is *not* $\prec$-minimal in $\mathbb{N}^{n+1}$ since on such inputs we have $G(0, \mathbf{d}) = H(\mathbf{d}) = G'(0, \mathbf{d})$. Thus, in particular, $m > 0$.

But then, by (1) of 4.3.3, we compute each of $G(m, \mathbf{c})$ and $G'(m, \mathbf{c})$ by the *second equation* as

$$K\Big(m - 1, \mathbf{c}, \{G(0, \mathbf{c}), G(1, \mathbf{c}), \ldots, G(m - 1, \mathbf{c})\}\Big)$$

since minimality of $(m, \mathbf{c})$ in the set T entails

$$G(i, \mathbf{c}) = G'(i, \mathbf{c}), \text{ for } i = 0, 1, \ldots m - 1$$

Since K is single-valued (function!) we have $G(m, \mathbf{c}) = G'(m, \mathbf{c})$, contradicting $(*)$. Thus $T = \emptyset$ and therefore $G(a, \mathbf{b}) = G'(a, \mathbf{b})$, for all $(a, \mathbf{b}) \in \mathbb{N}^{n+1}$. For short, the functions G and G' are the same. □

4.3.8 Theorem. *There is a function $G : \mathbb{N}^{n+1} \to A$ satisfying (1) of 4.3.3.*

Proof. The idea is simple: Build the function by stages as an infinite set of building blocks. Each block is a *restriction* of G —that is, a partial table for G— so that the domain of the restriction is an "*initial segment*" of $\mathbb{N}^{n+1}$ determined by some point ("point" is synonymous to "element") $(m, \mathbf{b})$. Thus the "general" segment is the set

$$S_{(m,\mathbf{b})} \overset{Def}{=} \{(a, \mathbf{b}) : (a, \mathbf{b}) \prec (m, \mathbf{b})\} \cup \{(m, \mathbf{b})\} \tag{†}$$

The notation "$S_{(m,\mathbf{b})}$" reflects "S" for *segment*, subscripted with the defining point $(m, \mathbf{b})$. Once you have *all* the building blocks, you put them together to get the G you want.

Let us call $G_{(m,\mathbf{b})}$ *the* function (if it exists) from $S_{(m,\mathbf{b})} \to A$ that satisfies (1) of 4.3.3 if we replace the G there by $G_{(m,\mathbf{b})}$ everywhere.

Why am I emphasising "the"? Because $S_{(m,\mathbf{b})}$ inherits MC from N^n. Cf. 4.3.6. And then 4.3.7 applies to $G_{(m,\mathbf{b})} : S_{(m,\mathbf{b})} \to A$ as the proof of 4.3.7 applies unchanged (just change $\mathbb{N}^{n+1}$ and G to $S_{(m,\mathbf{b})}$ and $G_{(m,\mathbf{b})}$ respectively; all else is the same in the proof).

We have one more **important** (for this proof) observation related to uniqueness: If $\boxed{(x, \mathbf{b}) \prec (y, \mathbf{b}), \text{ then } G_{(x,\mathbf{b})}(u, \mathbf{b}) = G_{(y,\mathbf{b})}(u, \mathbf{b}), \text{ for all } u \leq x}$.[†]

Indeed, if $G_{(x,\mathbf{b})}$ and $G_{(y,\mathbf{b})}$ exist, then they both satisfy (1) of 4.3.3 on the subset $S_{(x,\mathbf{b})}$ of $S_{(y,\mathbf{b})}$.

Our next task is simply to show that for each $(m, \mathbf{b}) \in \mathbb{N}^{n+1}$,

$$\text{the function } G_{(m,\mathbf{b})} : S_{(m,\mathbf{b})} \to A \text{ that satisfies (1) in 4.3.3 } \textit{exists} \tag{‡}$$

where we changed $\mathbb{N}^{n+1}$ and G into $S_{(m,\mathbf{b})}$ and $G_{(m,\mathbf{b})}$ respectively.

We do so *constructively* —that is, show how each $G_{(m,\mathbf{b})} : S_{(m,\mathbf{b})} \to A$ is *built*— by CVI on the variable $(m, \mathbf{b})$ along the order $\prec$ over $\mathbb{N}^{n+1}$.

1. *Basis*: For *any* minimal $(0, \mathbf{b})$,[‡] we have $S_{(0,\mathbf{b})} = \{(0, \mathbf{b})\}$. Thus, using the first equation of (1) in 4.3.3, we set

$$G_{(0,\mathbf{b})} = \Big\{\big((0, \mathbf{b}), H(\mathbf{b})\big)\Big\}^{§}$$

[†] Here "$\leq$" is, of course, the "less-than-or-equal" on $\mathbb{N}$.

[‡] We remarked that the $(0, \mathbf{b})$ for various $\mathbf{b} \in \mathbb{N}^n$ *are* the $\prec$-minimal points in $\mathbb{N}^{n+1}$.

[§] We still remember that a function is a set of pairs! This *one* has just one pair.

2. *I.H.* Assume that for all $(x, \mathbf{b}) \prec (m, \mathbf{b})$[†] we have built $G_{(x,\mathbf{b})} : S_{(x,\mathbf{b})} \to A$ all of which satisfy (the two equations of) (1) of 4.3.3.

   In view of the boxed statement above, $G_{(m,\mathbf{b})}$ coincides with each $G_{(x,\mathbf{b})}$ —for $(x, \mathbf{b}) \prec (m, \mathbf{b})$— on the latter's domain. Thus I need only add one input/output pair to $\bigcup_{(x,\mathbf{b})\prec(m,\mathbf{b})} G_{(x,\mathbf{b})} = G_{(m-1,\mathbf{b})}$

   Why is this last "=" correct?

   at input $(m, \mathbf{b})$ to obtain $G(m, \mathbf{b})$.

   To do so I simply use (1) of 4.3.3, second equation. The I/O pair added to obtain $G_{(m,\mathbf{b})}$ is

$$\Big((m-1, \mathbf{b}),\ K\big(m-1, \mathbf{b}, \{G_{(m-1,\mathbf{b})}(0, \mathbf{b}), \ldots, G_{(m-1,\mathbf{b})}(m-1, \mathbf{b})\}\big)\Big)$$

   It is clear that on *any* input $(u, \mathbf{b})$, whether the just *constructed relation* $G_{(m,\mathbf{b})}$ "thinks" that it is $G_{(x,\mathbf{b})}$ or $G_{(y,\mathbf{b})}$ it will give *the same output* due the boxed statement above. Thus, the relation $G_{(x,\mathbf{b})}$ is a *function.*

It is now time to put all the $G_{(x,\mathbf{b})}$ together to form $G : \mathbb{N}^{n+1} \to A$. Just define G by

$$G \overset{Def}{=} \bigcup_{(x,\mathbf{b})\in\mathbb{N}^{n+1}} G_{(x,\mathbf{b})} \qquad (*)$$

Observe regarding G:

1. As a *relation* it is total on the left field $\mathbb{N}^{n+1}$ because it is *defined* on the arbitrary $(x, \mathbf{b}) \in \mathbb{N}^{n+1}$ since $G_{(x,\mathbf{b})} : S_{(x,\mathbf{b})} \to A$ is.

2. $\mathrm{ran}(G) \subseteq A$. Because it is so for each $G_{(x,\mathbf{b})} : S_{(x,\mathbf{b})} \to A$.

3. G is single-valued, hence a *function* from $\mathbb{N}^{n+1}$ to A, since the value $G(u, \mathbf{b})$ does not depend on which $G_{(x,\mathbf{b})} : S_{(x,\mathbf{b})\to A}$ we used to obtain it as $G_{(x,\mathbf{b})}(u, \mathbf{b})$ (by boxed statement above).

   Finally,

4. G satisfies (1) of 4.3.3 since by $(*)$, for any $(x, \mathbf{b}) \in \mathbb{N}^{n+1}$, $G(x, \mathbf{b}) = G_{(x,\mathbf{b})}(x, \mathbf{b})$, and $G_{(x,\mathbf{b})}(x, \mathbf{b})$ is constructed to obey the two equations of (1) of 4.3.3, for all $x \geq 0$ and $\mathbf{b} \in \mathbb{N}^n$. □

Let us see some examples:

4.3.9 Example. We know that 2^n means

$$\overbrace{2 \times 2 \times 2 \times \ldots \times 2}^{n\ 2s}$$

[†] Recall that for $\mathbf{b} \neq \mathbf{c}$, $(x, \mathbf{b})$ and $(y, \mathbf{c})$ are not comparable.

But "…", or "etc.", is *not* MATH! That is why we gave at the outset of this section the definition 4.3.1.

Applied to the case $a = 2$ we have

$$\begin{aligned} 2^0 &= 1 \\ 2^{n+1} &= 2 \times 2^n \end{aligned} \tag{1}$$

We know from 4.3.8 and 4.3.7 that both (1) above and the definition in 4.3.1 define a unique function, each satisfying its defining equations.

For the function that for each n outputs 2^n we can give an alternative definition that uses "+" rather than "×":

$$\begin{aligned} 2^0 &= 1 \\ 2^{n+1} &= 2^n + 2^n \end{aligned}$$

□

4.3.10 Example. Let $f : \mathbb{N}^{n+1} \to \mathbb{N}$ be given. How can I define $\sum_{i=0}^{n} f(i, \mathbf{b})$ —for any $\mathbf{b} \in \mathbb{N}^n$— other than by the sloppy

$$f(0, \mathbf{b}) + f(1, \mathbf{b}) + f(2, \mathbf{b}) + \ldots + f(i, \mathbf{b}) + \ldots + f(n, \mathbf{b})?$$

By induction/recursion, of course:

$$\begin{aligned} \textstyle\sum_{i=0}^{0} &= f(0, \mathbf{b}) \\ \textstyle\sum_{i=0}^{n+1} &= \Big(\textstyle\sum_{i=0}^{n} f(i, \mathbf{b})\Big) + f(n+1, \mathbf{b}) \end{aligned} \tag{1}$$

□

4.3.11 Example. Let $f : \mathbb{N}^{n+1} \to \mathbb{N}$ be given. How can I define $\prod_{i=0}^{n} f(i, \mathbf{b})$ —for any $\mathbf{b} \in \mathbb{N}^n$— other than by the sloppy

$$f(0, \mathbf{b}) \times f(1, \mathbf{b}) \times f(2, \mathbf{b}) \times \ldots \times f(i, \mathbf{b}) \times \ldots \times f(n, \mathbf{b})?$$

By induction/recursion, of course:

$$\begin{aligned} \textstyle\prod_{i=0}^{0} &= f(0, \mathbf{b}) \\ \textstyle\prod_{i=0}^{n+1} &= \Big(\textstyle\prod_{i=0}^{n} f(i, \mathbf{b})\Big) + f(n+1, \mathbf{b}) \end{aligned} \tag{2}$$

Again, by 4.3.8 and 4.3.7, each of (1) and (2) define a unique function, $\sum$ and $\prod$ that behaves as required. Really? For example, the first equation of (1) gives us the one-term sum, $f(0, \mathbf{b})$. It is correct. Assume (I.H. by simple induction on n) that the term $\sum_{i=0}^{n} f(i, \mathbf{b})$ correctly captures the sloppy

$$f(0, \mathbf{b}) + f(1, \mathbf{b}) + f(2, \mathbf{b}) + \ldots + f(i, \mathbf{b}) + \ldots + f(n, \mathbf{b})$$

that indicates the sum of the first $n + 1$ terms of the type $f(i, \mathbf{b})$ for $i = 0, 1, 2, \ldots, n$. But then, clearly the second equation of (1) correctly defines the sum of the first $n + 2$ terms of the above type, by adding $f(n + 1, \mathbf{b})$ to $\sum_{i=0}^{n} f(i, \mathbf{b})$. □

4.3.12 Example. Here is a function with huge output! Define $f : \mathbb{N} \to \mathbb{N}$ by

$$\begin{aligned} f(0) &= 1 \\ f(n+1) &= 2^{f(n)} \end{aligned} \tag{3}$$

What does $f(n)$ look like in sloppy notation? Well,

$$f(0) = 1, \quad f(1) = 2^{f(0)} = 2, \quad f(2) = 2^{f(1)} = 2^2, \quad f(3) = 2^{f(2)} = 2^{2^2}$$

Hmm! Is the guess that $f(n)$ is a ladder of n 2s? Yes! Let's verify by induction:

1. *Basis.* $f(0) = 1$. A ladder of zero 2s. Correct.
2. *I.H.* Fix n and assume that

$$f(n) = \left.2^{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}}\right\} n \text{ 2s}$$

   A ladder of n 2s.
3. I.S. Thus $f(n+1) = 2^{f(n)}$, so we put the ladder of n 2s of the I.H. as the exponent of 2 —forming a ladder of $n+1$ 2s— to obtain $f(n+1)$. Done! □

4.3.13 Example. (Fibonacci; a comment) This short example is to be clear, as in the case of induction proofs, that the "Basis" case is for minimal elements (compare with Exercise 4.2.13, case 5).

$$\begin{aligned} F_0 &= 0 \\ F_1 &= 1 \\ &\text{and for } n \geq 1 \\ F_{n+1} &= F_n + F_{n-1} \end{aligned}$$

In the above "$F_1 = 1$" is *NOT* a "Basis case" because 1 is *not* minimal in $\mathbb{N}$! ("$F_0 = 0$" *is* the Basis case, corresponding to the first equation in (1) of 4.3.3). So what is "$F_1 = 1$"? It is a boundary case of the second equation in the general Definition 4.3.3. This equation, in the Fibonacci case, can be rewritten as

$$\begin{aligned} F_{n+1} = \text{if } n = 0 \text{ then } 1 \\ \text{else } F_n + F_{n-1} \end{aligned}$$

□

Chapter 5

Inductively defined sets; Structural induction

This chapter looks at a generalisation of the inductive definitions of the last section. An example of an inductively defined set is the following.

Suppose you want to define *by finite means*, and define *precisely*, the set of all *"simple" arithmetical expressions* that use the numbers $1, 2, 3$, the operations $+$ and $\times$, and round brackets. Then you would do it like this:

The set of said *simple arithmetical expressions* is the *smallest* set ($\subseteq$-smallest) that

1. Contains each of $1, 2$ and 3.

2. If it contains expressions E and E', then it also contains $(E + E')$ and $(E \times E')$.

Some folks would add a 3rd requirement "nothing else is in the set unless so demonstrated using 1. 2. above" and omit "smallest". *Really*?

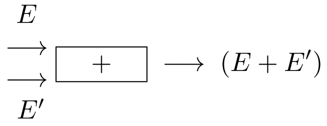
How exactly would you so "demonstrate"? In a recursive definition you ought to be able to make your recursive calls and not have to trace back why the object you constructed exists!

We will prove in Section 5.2.5 that indeed there *is* an iterative way to show that a *particular* simple arithmetic expression was formed correctly by our recursion, but that defeats the beauty of recursion. Besides, until we reach said section we don't *know* what "nothing else is in the set unless so demonstrated using 1. 2. above" *means* or *how* to "use" 1. and 2. do it! So it is nonsense to stick such a statement in the bottom of the definition as a (redundant) afterthought.

Before we get to the general definitions, let us finesse our construction and propose some terminology.

(a) First off, in step 1. above we say that $1, 2$ and 3 are *the initial objects* of our recursive/inductive definition.

(b) In step 2. we say that $(E+E')$ is obtained by an *operation* (on strings) that is available to us, depicted as a “blackbox” below, which we named “+”.



In words, the operation *concatenates from left to right the strings*

“(”, “E”, “+”, “E'”, and “)”

Similar comments for the operation “×”.

(c) Both operations in this example are single-valued, that is, functions. It is preferable to be slightly more general and allow operations that are just relations, but not necessarily functions. Such an operation $O(x_1,\ldots,x_n,y)$ is n-ary —n inputs, $x_1,\ldots,x_n$— with output variable y.

(d) We say that a set of objects S is *closed under* a **relation** (operation) —it could be a function— $O(x_1,\ldots,x_n,y)$ meaning that for *all* input values $x_1,\ldots,x_n$ $\underline{\text{in } S}$, *all* the obtained values y $\underline{\text{are also in } S}$.

We are ready for the general definition:

5.0.1 Definition. Given a set of *initial objects* $\mathcal{I}$ and a *set* of *operations* $\mathcal{O} = \{O_1, O_2, O_2, \ldots\}$, the object $\mathrm{Cl}(\mathcal{I},\mathcal{O})$ is called the *closure of* $\mathcal{I}$ *under* $\mathcal{O}$ —or the set *inductively defined by the pair* $(\mathcal{I},\mathcal{O})$— and *denotes* the $\subseteq$-smallest *class*[†] S that satisfies

1. $\mathcal{I} \subseteq S$.

2. S is *closed under all operations in* $\mathcal{O}$, or simply, $\underline{\text{closed under } \mathcal{O}}$.

3. The “smallest” part: Any class T that satisfies 1. and 2. also satisfies $S \subseteq T$.

The set $\mathcal{O}$ may be infinite. Each operation O_i is a set. □

Nice definition, but does $\mathrm{Cl}(\mathcal{I},\mathcal{O})$ *exist* given $\mathcal{I}$ and $\mathcal{O}$? Yes. But first,

5.0.2 Theorem. *For any choice of* $\mathcal{I}$ *and* $\mathcal{O}$*, if* $\mathrm{Cl}(\mathcal{I},\mathcal{O})$ *exists, then it is unique.*

Proof. Say $S = \mathrm{Cl}(\mathcal{I},\mathcal{O}) = T$. Then, letting S pose as closure, we get $S \subseteq T$ from 5.0.1. Then, letting T pose as closure, we get $T \subseteq S$, again from 5.0.1. Thus $S = T$. □

[†] Let’s say “class” until we learn that it is actually a *set*.

5.0.3 Theorem. *For any choice of $\mathcal{I}$ and $\mathcal{O}$ with the restrictions of Definition 5.0.1 the set* $\mathrm{Cl}(\mathcal{I},\mathcal{O})$ *exists.*

Proof. We have to check and note a few things.

1. By 3.1.5, for each O_i, $\mathrm{ran}(O_i)$ is a set.
2. The class $F = \{\mathrm{ran}(O_i) : i = 1, 2, 3 \ldots\}$ is a set. This is so by **Principle** 2, since I can index all members of F by assigning unique indices from $\mathbb{N}$ to each of its members (and $\mathbb{N}$ is a set by **Principle** 0).
3. By 2. above and 2.4.16, $\bigcup F$ is a set, and so is $T = \mathcal{I} \cup \bigcup F$ □
4. T contains $\mathcal{I}$ as a subset (by the way T was defined) and is $\mathcal{O}$-closed since any O_i-output —no matter where the inputs come from— is in $\mathrm{ran}(O_i) \subseteq \bigcup F$.
5. The family $\mathbb{G} = \{S : \mathcal{I} \subseteq S \ \&\ S \text{ is } \mathcal{O}\text{-closed}\}$ contains the set T as a member. Thus (cf. 2.4.17)
$$C \overset{Def}{=} \left(\bigcap \mathbb{G}\right) \subseteq T$$
is a set. Since all sets S in $\mathbb{G}$ contain $\mathcal{I}$ and are $\mathcal{O}$-closed, so is C. But $C \subseteq S$ for all such sets S the way it is defined. So it is $\subseteq$-smallest.

Thus, $C = \mathrm{Cl}(\mathcal{I},\mathcal{O})$. We proved existence. □

5.1. Induction over a closure

5.1.1 Definition. Let a pair $(\mathcal{I},\mathcal{O})$ be given as above.

We say that a property $P[x]$ $\underline{\text{propagates with } \mathcal{O}}$ *iff* for each $O_i(x_1, \ldots, x_n, y) \in \mathcal{O}$, $\underline{\text{if whenever all the inputs in the } x_i}$ satisfy $P[x]$ (i.e., $P[x_i]$ is true for each argument x_i), then all output values returned by y —for said inputs— satisfy $P[x]$ as well. Recall that for each assignment of values to the inputs $x_1, \ldots, x_n$ we may have more than one output values in y; for all such values $P[y]$ is true. □

5.1.2 Lemma. *For all* $(\mathcal{I},\mathcal{O})$ *and a property* $P[x]$*, if the latter propagates with* $\mathcal{O}$*, then the class* $\mathbb{A} = \{x : P[x]\}$ *is closed under* $\mathcal{O}$ *(is* $\mathcal{O}$*-closed).*

Proof. So let $O_i(x_1, \ldots, x_n, y) \in \mathcal{O}$. Let $a_1, \ldots, a_n$ be all in $\mathbb{A}$. Thus
$$P[a_i], \text{ for all } i = 1, \ldots, n$$
By assumption, if $O_i(a_1, \ldots, a_n, b)$, then $P[b]$ is true, hence $b \in \mathbb{A}$. □

5.1.3 Theorem. *Let* $\mathrm{Cl}(\mathcal{I},\mathcal{O})$ *and a property* $P[x]$ *be given. Suppose we have done the following steps:*

1. *We showed that for each $a \in \mathcal{I}$, $P[a]$ is true.*
2. *We showed that $P[x]$ propagates $\mathcal{O}$.*

Then $\underline{\textit{every } a \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})}$ *has property $P[x]$.*

Naturally, the technique encapsulated by 1. and 2. of 5.1.3 is called "induction over $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$" or "structural induction" over $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$.

Note that for each $O_i \in \mathcal{O}$ the "propagation of property $P[x]$" will take the form of an I.H. followed by an I.S.:

- **Assume** for the unspecified fixed inputs $a_1, \ldots, a_n$ of O_i that *all* satisfy $P[x]$. This is the *I.H.* for O_i.
- Then **prove** that any output b of O_i caused by said input also satisfies the property.

Proof. (of 5.1.3) Let us write

$$\mathbb{A} \overset{Def}{=} \{x : P[x]\}$$

Thus, 1. in 5.1.3 translates to

$$\mathcal{I} \subseteq \mathbb{A} \tag{$*$}$$

2. and 5.1.3 yield

$$\mathbb{A} \text{ is } \mathcal{O}\text{-closed} \tag{$**$}$$

If $\mathbb{A}$ were a set —a hypothesis we *cannot* make because of Russell's paradox— then $(*)$ and $(**)$ would immediately yield $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \subseteq \mathbb{A}$ and we would be done. So we have a tiny bit more work to do:

By 5.0.3, item 4, the set T built for our $\mathcal{I}$ and $\mathcal{O}$ contains $\mathcal{I}$ and is $\mathcal{O}$-closed. Thus so is $T \cap \mathbb{A}$! Moreover the latter is a set, as we know (2.4.2). Hence, by 5.0.1,

$$\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \subseteq T \cap \mathbb{A} \subseteq \mathbb{A}$$

The last implication immediately translates to

$$\underline{\text{“}x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O}) \text{ implies } P[x] \text{ is true”}} \qquad \square$$

5.1.4 Example. Let $S = \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ where $\mathcal{I} = \{0\}$ and $\mathcal{O}$ contains just one operation, $x + 1 = y$, where y is the output variable. That is,

$$n \longrightarrow \boxed{x + 1 = y} \longrightarrow n + 1 \tag{1}$$

is our only operation. By induction over S, I can show $S \subseteq \mathbb{N}$.

The "$P[x]$" is "$x \in \mathbb{N}$".

So $P[0]$ is true. I verified the property for $\mathcal{I}$. That the property propagates with our operation is captured by (1) above (if $n \in \mathbb{N}$, then $n + 1 \in \mathbb{N}$). Done!

Can we show also $\mathbb{N} \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$? **Yes**: In this direction I do SI over $\mathbb{N}$ on variable n. The property, let's call it $Q[x]$, now is "$x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$".

For $n = 0$, $n \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ since $0 \in \mathcal{I} \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ by 5.0.1.

Now, say (*I.H.*) $n \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. Since $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$ is closed under the operation $x + 1 = y$, we have $n + 1 \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ by 5.0.1.

So,

$$\mathrm{Cl}(\mathcal{I}, \mathcal{O}) = \mathbb{N} \qquad \square$$

Thus the induction over a closure generalises *SI.*

5.2. Closure vs. definition by stages

We will see in this section that there is also a *by-stages* or *by-steps* way to obtain $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$.

5.2.1 Definition. (Derivations) An $(\mathcal{I}, \mathcal{O})$-*derivation* —or just *derivation* if we know which $(\mathcal{I}, \mathcal{O})$ we are talking about— is a *finite sequence of objects*

$$d_1, d_2, d_3, \ldots, d_i, \ldots, d_n \tag{1}$$

satisfying:

Each d_i is

1. A member of $\mathcal{I}$,

   or

2. For some j, one of the results of $O_j(x_1, \ldots, x_k, y)$ with inputs $a_1, \ldots, a_k$ that are found in the derivation (1) *to the left of* d_i.

n is called the *length of the derivation.* Every d_i is called an $(\mathcal{I}, \mathcal{O})$-*derived* object, or just *derived*, if the $(\mathcal{I}, \mathcal{O})$ is understood. □

Clearly, the concept of a derivation abstracts, thus generalises, the concept of *proof*, while a derived object abstracts the concept of a *theorem.*

5.2.2 Example. For the $(\mathcal{I}, \mathcal{O})$ of 5.1.4, here are some derivations:

$$0$$

$$0, 0, 0$$

$$0, 1, 0, 1, 0, 1, 1, 1, 1, 0$$

Nothing says we cannot repeat a d_i in a derivation! Lastly here is an "efficient" derivation with no redundant steps: $0, 1, 2, 3, 4, 5$. □

5.2.3 Proposition. *If* $d_1, d_2, d_3, \ldots, d_i, \ldots, d_n, d_{n+1}, \ldots, d_m$ *is a* $(\mathcal{I}, \mathcal{O})$*-derivation, then so is* $d_1, d_2, d_3, \ldots, d_i, \ldots, d_n$.

Proof. Each d_i is validated in a derivation either outright (i.e., is in $\mathcal{I}$) or by looking to the left! What we may remove to the *right* of d_i does not affect the validity of that entry. □

5.2.4 Proposition. *If $d_1, d_2, \ldots, d_n$ and $e_1, e_2, \ldots, e_m$ are $(\mathcal{I}, \mathcal{O})$-derivations, then so is $d_1, d_2, \ldots, d_n, e_1, e_2, \ldots, e_m$.*

Proof. Traversing $d_1, d_2, \ldots, d_n$ and $e_1, e_2, \ldots, e_m$ in

$$d_1, d_2, \ldots, d_n, e_1, e_2, \ldots, e_m$$

from left to right we validate each d_i and each e_j giving precisely the same validation *reason* as we would in each sequence $d_1, d_2, \ldots, d_n$ and $e_1, e_2, \ldots, e_m$ separately. These reasons are local to each sequence. □

We now prove that defining a set S as a $(\mathcal{I}, \mathcal{O})$-closure is equivalent with defining S as the set of all $(\mathcal{I}, \mathcal{O})$-derived objects.

5.2.5 Theorem. *For any initial sets of objects and operations on objects ($\mathcal{I}$ and $\mathcal{O}$) we have that* $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) = \{x : x$ *is* $(\mathcal{I}, \mathcal{O})$*-derived*$\}$.

Proof. Let us write $D = \{x : x \text{ is } (\mathcal{I}, \mathcal{O})\text{-derived}\}$ and prove that $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) = D$. We have two directions:

1. $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \subseteq D$: By induction over $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$. The property to prove is "$x \in D$".

    - Let $x \in \mathcal{I}$. Then x is derived via the one-member derivation

    $$x$$

    So $x \in D$. Thus all $x \in \mathcal{I}$ have the property.

    - The property "$x \in D$" propagates with each $O_k(\vec{x}_n, y) \in \mathcal{O}$: So let each of the x_i have a derivation $\boxed{\ldots, x_i}$. We show that so does y.

    Concatenating all these derivations we get a derivation (5.2.4)

    $$\boxed{\ldots, x_1}, \ldots, \boxed{\ldots, x_i}, \ldots, \boxed{\ldots, x_n} \tag{1}$$

    But then so is

    $$\boxed{\ldots, x_1}, \ldots, \boxed{\ldots, x_i}, \ldots, \boxed{\ldots, x_n}, y \tag{2}$$

    by 5.2.1, case 2. That is, y is *derived*, hence $y \in D$ is proved (I.S.).

2. $D \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$: Let $x \in D$. This time we do good old-fashioned CVI over $\mathbb{N}$ on the length n of a derivation of x, toward showing that $x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ —this is the "property of x" that we prove.

    Basis. $n = 1$. The only way to have a 1-element derivation is that $x \in \mathcal{I}$. Thus, $x \in \mathcal{I} \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ by 5.0.1.

I.H. Assume the claim for x derived with length $k < n$.

I.S. Prove that the claim holds when x has a derivation of length n.

Consider such a derivation

$$a_1, \ldots a_i, \ldots, a_k, \ldots, \overset{\displaystyle a_n}{\overset{\|}{x}}$$

If $x \in \mathcal{I}$, then we are done by the *Basis*. Otherwise, say x is the result of an operation (relation) $O_r \in \mathcal{O}$, applied on entries to the left of x, that is, say that $O_r(\ldots, x)$ is true —where we did not (have to) specify the inputs.

By the I.H. the inputs of O_r all are in $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$. Now, since this closure is closed under $O_r(\ldots, x)$, we have that the output x is in $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$ too. □

So now we have two *equivalent* (5.2.5) approaches to defining inductively defined sets S: As $S = \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ or as $S = \{x : x \text{ is } (\mathcal{I}, \mathcal{O})\text{-derived}\}$.

The first approach is best when you want to prove properties of all members of the set S. The second is best when you want to show $x \in S$, for some specific x.

5.2.6 Example. Let $A = \{a, b\}$. We call A an "*alphabet*".

Let $\mathcal{I} = \{\lambda\}$, λ being (the name of) the *empty string*. Let us denote string concatenations by putting the strings we want to concatenate next to each other. E.g., concatenate *aaa* and *bbbaa* to obtain *aaabbbaa*. Also, if X denotes a string, and so does Y, then XY denotes the concatenation of the strings (denoted by) X and Y in that order. Similarly, Xa means the result of concatenating string X with the (length-1) string a, in that order. The *length* of a string over A is the number of occurrences in the string (with repetitions) of a and b.

We denote by A^+ the set of all strings of non zero length formed using the symbols a and b. A^* is defined to be $A^+ \cup \{\lambda\}$. Let $\mathcal{O}$ consist of the operations O_a and O_b:

$$X \longrightarrow \boxed{O_a} \longrightarrow Xa \tag{1}$$

and

$$X \longrightarrow \boxed{O_b} \longrightarrow Xb \tag{2}$$

We claim that $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) = A^*$.

1. For $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \subseteq A^*$ we do induction over the closure to prove that any $x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ satisfies $x \in A^*$ ("the property").

   - Well, if $x \in \mathcal{I}$ then $x = \lambda$. But $\lambda \in A^*$.
   - The property propagates with each of O_a and O_b. For example, if $X \in A^*$, then since Xa is also a string over the alphabet A, we have $Xa \in A^*$. Similarly for O_b. Done.

2. For $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \supseteq A^*$ we do induction over $\mathbb{N}$ on $n = |Y|$ —the length of Y— to prove that any $Y \in A^*$ satisfies $Y \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ ("the property").
   - Basis. $n = 0$. Then $Y = \lambda \in \mathcal{I} \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. Done.
   - *I.H.* **Assume** claim for fixed n.
   - *I.S.* **Prove** for $n + 1$. If $|Y| = n + 1$ then $Y = Xa$ or $Y = X'b$ for some X or X' of length n. Say, it is $Y = Xa$. By I.H. $X \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. But since $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$ is $\mathcal{O}$-closed, we have $Y = Xa \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ by (1). The $Y = X'b$ case is entirely similar. □

5.2.7 Example. Let $A = \{a, b\}$ again.

Let $\mathcal{I} = \{\lambda\}$, let $\mathcal{O}$ consist of one operation R:

$$X \longrightarrow \boxed{R} \longrightarrow aXb \tag{3}$$

We claim that $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) = \{a^n b^n : n \geq 0\}$, where for any string X,

$$X^n \overset{Def}{=} \underbrace{XX \ldots X}_{n \text{ copies of } X}$$

If $n = 0$, "0 copies of X" means λ.

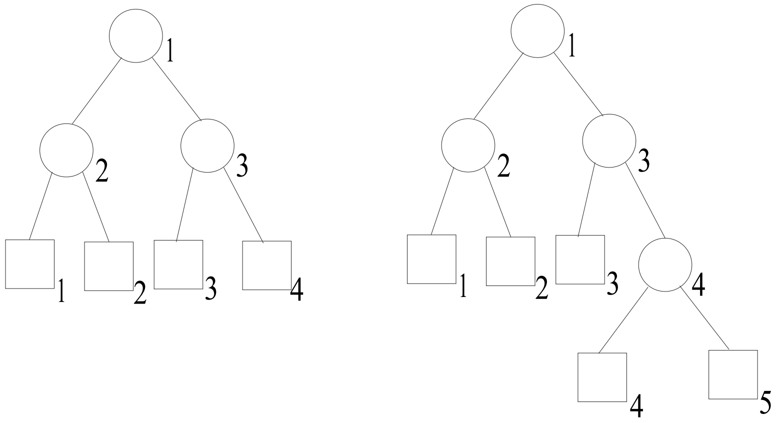
Let us write $S = \{a^n b^n : n \geq 0\}$.

1. For $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \subseteq S$ we do induction over the closure to prove that any $x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ satisfies $x \in S$ ("the property").
   - Well, if $x \in \mathcal{I}$ then $x = \lambda = a^0 b^0$. Done.
   - The property propagates with each of R. For example, say $x = a^n b^n \in S$. Using (3) we see that the output, axb, is $a^{n+1} b^{n+1} \in S$. The property does propagate! Done.
2. For $\mathrm{Cl}(\mathcal{I}, \mathcal{O}) \supseteq S$ we do induction over $\mathbb{N}$ on n of $x = a^n b^n$ (arbitrary member of S) to prove that any $x \in S$ satisfies $x \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ ("the property").
   - Basis. $n = 0$. Then $x = \lambda \in \mathcal{I} \subseteq \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. Done.
   - *I.H.* **Assume** claim for fixed n.
   - *I.S.* **Prove** for $n + 1$. Thus $x = a^{n+1} b^{n+1} = a a^n b^n b$. By the I.H., $a^n b^n \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. By (3) —recall that $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$ is $\mathcal{O}$-closed— we get the output $a a^n b^n b = a^{n+1} b^{n+1} \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$. □

5.2.8 Example. (Extended Binary Trees) This is a longish example with some preliminary discussion up in front. We want to define the term known as "Tree". This term refers to a structure, which uses as building blocks —called **nodes**— the members of the enumerable set below

$$A = \{\bigcirc_0, \bigcirc_1, \bigcirc_2, \ldots ; \square_0, \square_1, \square_2, \ldots\}$$

Trees look something like this:



The qualifier "extended" is due to the presence of square nodes. We will not define simple trees (they have round nodes only).

These *nodes* are made *distinct* by the use of *subscripts*. The symbols in the set A are *distinguished* by their *type*, "round" vs. "square", and *within each type* by their natural number index. Thus, $\bigcirc_i \neq \bigcirc_j$ iff $i \neq j$, $\square_i \neq \square_j$ iff $i \neq j$, and $\bigcirc_i \neq \square_j$, *for all* i, j.

One feature in both of the above drawings is essential to note (blue type below):

Circular or **square** nodes are connected by line segments. Walking in the vertical direction from the top of the page towards the bottom, **no nodes are *ever* shared.** In particular, in all the examples above where we have more than one node, you will notice that the two *sets* of nodes that "hang below" the top node (left and right of it) are *disjoint*. **We need to include this requirement in our definition**.

But clearly these sets of notes have "geometric structure" (*position*: left/right; and *connections*: via line segments)! They are not "flat" sets like $\{\bigcirc_5, \square_{11}\}$.

And yet, in the *mathematical definition below* we will need to *state* the blue condition: the left and right, when you "forget" the lines and positions, become disjoint flat sets. This observation is what *imposes* some complexity in the definition, which defines the "structure" *and* the "flat" set that supports the structure (the set of nodes in the tree) *simultaneously*.

We define an *extended binary tree* as a *member* of the inductively defined set of *E-Trees*. It is intended that each e-tree of the inductively defined set of all trees is an *ordered pair*:

(flat set of its nodes, geometric tree structure)

The "geometric tree structure" is **mathematically given** in a *one-dimensional depiction* of the trees.

For example, the first tree in the figure above is linearly represented by

$$\Big((\Box_1, \bigcirc_2, \Box_2), \bigcirc_1, (\Box_3, \bigcirc_3, \Box_4)\Big)$$

To appreciate the issue of "structure vs. flat set of nodes" let us first write the above as

$$\Big((a, b, c), d, (f, g, h)\Big) \tag{2}$$

How easy is to obtain the flat set of nodes a–h? Easy via naked eye for very small trees, hard for large ones.

So, why not forget flat structure and just say "left and right parts of a tree must be disjoint"? Because such parts are not sets (sets do not have "structure", like "edges"), and the term "disjoint" refers to sets.

Here (2) is shorthand for something really complex, namely

$$(a, b, c) = \Big\{\{a, \{a, b\}\}, \{a, c, \{a, b\}\}\Big\}$$

Suppose now that $b = g$, but all other letters (a, c, d, f, h) are distinct. Thus $(f, g, h) = (f, b, h) = \Big\{\{f, \{f, b\}\}, \{f, h, \{f, b\}\}\Big\}$ and hence

$$(a, b, c) \cap (f, g, h) = \emptyset \tag{t}$$

So test (t) does *NOT* give me the information I *need* before I build the tree in (2). Apparently it is wrong to do so, as $b = g$.

I do need the information the flat set of nodes gives me, for the decision. See definition below for the details!

Thus our definition below builds the flat set —called the *support* of the tree— of nodes of a tree *at the same time as it builds the structure of the tree.*

5.2.9 Definition. We define the *set* of *all extended trees* —or just *trees*— ET, as $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$ where:

1. First, chose as the set of initial objects

$$\mathcal{I} = \Big\{(\emptyset, \Box_0), (\emptyset, \Box_1), (\emptyset, \Box_2), \ldots\}$$

2. $\mathcal{O}$ has just one rule with a constraint on the input: If $F_X \cap F_Y = \emptyset$ **and** $\bigcirc_i \notin F_X \cup F_Y$, then

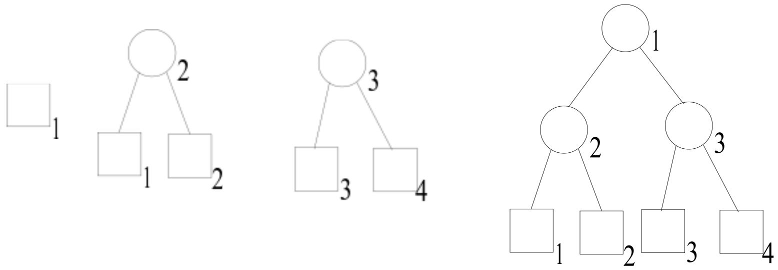
$$\left.\begin{array}{r}(F_X, X) \longrightarrow \\ \bigcirc_i \longrightarrow \\ (F_Y, Y) \longrightarrow\end{array}\right\} \boxed{\text{form tree}} \longrightarrow \Big(F_X \cup F_Y \cup \{\bigcirc_i\},\ (X, \bigcirc_i, Y)\Big)$$

3. For *each* $(S, T) \in \mathrm{Cl}(\mathcal{I}, \mathcal{O})$ we say that T is an extended tree, and S is its *support*, that is, the "flat" set nodes from the set A used to build T. We indicate this relationship by

$$S = \sup(T)^{\dagger}$$

If $T = (X, \bigcirc_i, Y)$, then we say that $\bigcirc_i$ is the **root** of T, while X is its **left** and Y is its **right subtree**. □

Thus, some immediate examples of trees are



Indeed, using 5.2.5, the leftmost example is a tree since it is the right component of the pair $(\emptyset, \Box_1)$. The next tree is built via the derivation —written linearly,

$$(\emptyset, \Box_1), (\emptyset, \Box_2), \Big(1, (\Box_1, \bigcirc_2, \Box_2)\Big)$$

The next derivation builds both the 2nd and 3rd trees:

$$(\emptyset, \Box_1), (\emptyset, \Box_2), \Big(1, (\Box_1, \bigcirc_2, \Box_2)\Big), (\emptyset, \Box_3), (\emptyset, \Box_4), \Big(1, (\Box_3, \bigcirc_3, \Box_4)\Big)$$

The 4th tree has this as a derivation:[‡]

$$(\emptyset, \Box_1), (\emptyset, \Box_2), \Big(1, (\Box_1, \bigcirc_2, \Box_2)\Big), (\emptyset, \Box_3), (\emptyset, \Box_4), \Big(1, (\Box_3, \bigcirc_3, \Box_4)\Big),$$
$$\bigg(3, \Big(1, (\Box_1, \bigcirc_2, \Box_2)\Big), \bigcirc_1, \Big(1, (\Box_3, \bigcirc_3, \Box_4)\Big)\bigg)$$

The support of the 4th tree is the flat set $\{\bigcirc_1, \bigcirc_2, \bigcirc_3\}$. □

5.2.10 Example. (Trees —continued) Hmm! Seems like we are not including square nodes in the support. See how the *support* of all nodes in $\mathcal{I}$ is $\emptyset$ for each entry. Why so?

In the words of Knuth ([Knu73]) "trees is the most important nonlinear structure arising in computing algorithms". The extended tree is an abstraction of trees that we implement with computer programs, where round nodes are the

[†] **Caution**: As for many other symbols, "sup" means something else in the context of POsets. We will not get into this!

[‡] Derivations are not unique as is clear from Example 5.2.2.

only ones that can carry data. The lines are (implicitly) pointing downwards. They are *pointers*, in computer jargon. For example, the topmost leftmost line in the fourth tree above points to the node $\bigcirc_2$. Practically it means that if your program is processing node $\bigcirc_1$, then it can transfer to and process node $\bigcirc_2$ if it wishes. It knows the address of $\bigcirc_2$. The pointer holds this address as value.

Which brings me to square nodes! Together with the line planted on them, they are notation for *null* pointers! They point nowhere. So square nodes cannot hold information, *that is why they do not contribute to the support of the tree.*

The computer scientist calls round nodes "internal" and calls square nodes "external".

Finally, how do the lines —called *edges*— get inserted? We defined "root" for trees, as well as "left subtree" and "right subtree". So, to draw lines and draw a tree that is given mathematically as $(X, \bigcirc_r, Y)$, we *call* recursively the process that does it on (inputs) X and Y. Then add two more edges: One from $\bigcirc_r$ to the root of X and one from $\bigcirc_r$ to the root of Y.

How does the recursion terminate? Well, if your tree is just $\Box_j$, then there is nothing to draw. $\Box_j$ is the root. This is the basis of the recursive procedure: do nothing. □

Here is something interesting about all extended trees:

5.2.11 Proposition. *In any extended tree, the number of square nodes exceeds by one the number of round nodes.*

Proof. Induction over the set of all trees (5.2.9) $\mathrm{Cl}(\mathcal{I}, \mathcal{O})$.

1. *Basis.* For any $(\emptyset, \Box_i)$, the tree-part (structure-part) is just $\Box_i$. One square node, 0 round nodes. Done.

2. The property propagates with the only tree-builder operation:

   $$\left.\begin{array}{c}(F_X, X) \longrightarrow\\ \bigcirc_i \longrightarrow\\ (F_Y, Y) \longrightarrow\end{array}\right\} \boxed{\text{form tree}} \longrightarrow \Big(F_X \cup F_Y \cup \{\bigcirc_i\},\ (X, \bigcirc_i, Y)\Big)$$

   Indeed, *suppose* that X has ϕ internal (round) and ε external (square) nodes. Let also Y have ϕ' internal and ε' external nodes.

   The assumption *on the input side* is then (I.H.) that

   $$\phi + 1 = \varepsilon \tag{1}$$

   and

   $$\phi' + 1 = \varepsilon' \tag{2}$$

The *output side* of the operation has the tree $(X, \bigcirc_i, Y)$. This has $\Phi = \phi + \phi' + 1$ internal nodes and $E = \varepsilon + \varepsilon'$ external ones. Using (1) and (2) we have

$$\Phi = \varepsilon + \varepsilon' - 1 = E - 1$$

Seeing that this is the property we want to prove on the output side, indeed the property propagates with the rule. Done. □

Chapter 6

Recurrence relations and their closed-form solutions

In "divide and conquer" algorithms one usually ends up with a recurrence relation that "defines" the "timing function", $T(n)$. For example, it might look like

$$T(n) = \begin{cases} 1 & \textbf{if } n = 1 \\ T(n/2) + 1 & \textbf{otherwise} \end{cases}$$

In order to assess the "goodness" of the proposed algorithm by comparison to either our expectations or to another algorithm, we need to know $T(n)$ in "closed" form in terms of known functions, for example, n^r for $r > 0$, c^n for $c > 1$, $\log_b n$ for some integer $b > 1$.

Often, a preliminary analysis need only worry about the "asymptotic behaviour" of the algorithm, i.e., the behaviour for *large* inputs (n is the input size). "Big-O" notation is an excellent tool in this case, therefore the solution of recurrences is often sought in such notation. On occasion one requires an "exact" solution (this is much harder to achieve in general).

There is a big variety of recurrence relations and an equally big variety of solution techniques. Some restricted cases are handled well by packages such as *Mathematica* or *Maple V*. For the mathematical reasons that make the solutions tick the best reference is perhaps Knuth *et al.* "*Concrete Mathematics*" (Addison-Wesley).

In this chapter we restrict attention to simple classes of recurrences taken from both the "additive" and "multiplicative" cases. These characterizations in quotes refer to the manner of handling the argument of the recurrence. E.g., the recurrence above is multiplicative as the recursive call is to an argument obtained by *halving* the original argument n. On the other hand, the Fibonacci recurrence is additive.

6.1. Big-O, small-o, and the "other" $\sim$

This notation is due to the mathematician E. Landau and is in wide use in number theory, but also in computer science in the context of measuring (bounding above) computational complexity of algorithms for all "very large inputs".

6.1.1 Definition. Let f and g be two total functions of one variable, where $g(x) > 0$, for all x. Then

1. $f = O(g)$ —also written as $f(x) = O(g(x))$— read "f is big-oh g", means that there are positive constants C and K in $\mathbb{N}$ such that

$$x > K \text{ implies } |f(x)| \le Cg(x)$$

2. $f = o(g)$ —also written as $f(x) = o(g(x))$— read "f is small-oh g", means that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 0$$

3. $f \sim g$ —also written as $f(x) \sim g(x)$— read "f is of the same order as g", means that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 1$$

□

"$\sim$" between two sets A and B, as in $A \sim B$, means that there is a 1-1 correspondence $f : A \to B$. Obviously, the context will protect us from confusing this $\sim$ with the one introduced just now, in 6.1.1.

Both definitions 2. and 3. require some elementary understanding of differential calculus. Case 2. says, intuitively, that as x gets extremely large, then the fraction $f(x)/g(x)$ gets extremely small, infinitesimally close to 0. Case 3. says, intuitively, that as x gets extremely large, then the fraction $f(x)/g(x)$ gets infinitesimally close to 1; that is, the function outputs are infinitesimally close to each other.

6.1.2 Example.

1. $x = O(x)$ since $x \le 1 \cdot x$ for $x \ge 0$.

2. $x \sim x$, since $x/x = 1$, and stays 1 as x gets very large.

3. $x = o(x^2)$ since $x/x^2 = 1/x$ which trivially goes to 0 as x goes to infinity.

4. $2x^2 + 1000^{1000}x + 10^{350000} = O(x^2)$. Indeed

$$\frac{2x^2 + 1000^{1000}x + 10^{350000}}{3x^2} = 2/3 + 1000^{1000}/x + 10^{350000}/x^2 < 1$$

for $x > K$ for some well chosen K. Note that $1000^{1000}/x$ and $10^{350000}/x^2$ will each be $< 1/6$ for all sufficiently large x-values: we will have $2/3 + 1000^{1000}/x + 10^{350000}/x^2 < 2/3 + 1/6 + 1/6 = 1$ for all such x-values. Thus $2x^2 + 1000^{1000}x + 10^{350000} < 3x^2$ for $x > K$ as claimed.

In many words, in a polynomial, the order of magnitude is determined by the highest power term. □

The last example motivates

6.1.3 Proposition. *Suppose that $f(x) \geq 0$ for all $x > L$, hence $|f(x)| = f(x)$ for all $x > L$. Now, if $f(x) \sim g(x)$, then $f(x) = O(g(x))$.*

Proof. The assumption says that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 1$$

From "calculus 1" (1st year differential calculus) we learn that this implies that for some K, $x > K$ entails

$$\left|\frac{f(x)}{g(x)} - 1\right| < 1$$

hence

$$-1 < \frac{f(x)}{g(x)} - 1 < 1$$

therefore, $x > \max(K, L)$ implies $f(x) < 2g(x)$. □

6.1.4 Proposition. *Suppose that $f(x) \geq 0$ for all $x > L$, hence $|f(x)| = f(x)$ for all $x > L$. Now, if $f(x) = o(g(x))$, then $f(x) = O(g(x))$.*

Proof. The assumption says that

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = 0$$

From calculus 1 we learn that this implies that for some K, $x > K$ entails

$$\left|\frac{f(x)}{g(x)}\right| < 1$$

hence

$$-1 < \frac{f(x)}{g(x)} < 1$$

therefore, $x > \max(K, L)$ implies $f(x) < g(x)$. □

These two propositions add to our toolbox:

6.1.5 Example.

1. $\ln x = o(x^r)$ for any positive real r. Here “ln” stands for $\log_e$ where e is the Euler constant

   $$2.718281828459045235360287471352662497757247093 7\ldots$$

   Seeing that both numerator and denominator

   $$\lim_{x\to\infty} \frac{\ln x}{x^r}$$

   go to ∞, we have here (if we do not do anything to mitigate) an impasse: We have a “limit” that is indeterminate:

   $$\frac{\infty}{\infty}$$

   So, we will use “l’Hôpital’s rule” (the limit of the fraction is equal to the limit of the fraction of the derivatives):

   $$\lim_{x\to\infty} \frac{\ln x}{x^r} = \lim_{x\to\infty} \frac{1/x}{rx^{r-1}} = \lim_{x\to\infty} \frac{1}{rx^r} = 0$$

2. $\ln x = O(\log_{10}(x))$. In fact, you can go from one log-base to the other:

   $$\log_e(x) = \frac{\log_{10}(x)}{\log_{10}(e)}$$

   The claim follows from 6.1.3 since trivially $\ln x \sim \log_{10}(x)/\log_{10}(e)$. For that reason —and since multiplicative constants are hidden in big-O notation— complexity- and algorithms-practitioners omit the base of the logarithm and write things like $O(\log n)$ and $O(n \log n)$. □

6.2. Solving recurrences; the additive case

The general case here is of the form[†]

$$\begin{aligned} &T_0 = k \\ &s_n T_n = v_n T_{n-1} + f(n) \textbf{ if } n > 0 \end{aligned}$$

a recurrence defining the *sequence* T_n, or equivalently, the *function* $T(n)$ (both jargons and notations spell out the same thing), in terms of the *known* functions (sequences) $s_n, v_n, f(n)$.

For the general case see Knuth cited above. Here we will restrict attention to the case $s_n = 1$ for all n and $v_n = a$ (a constant) for all n.

Subcase 1. $(a = 1)$ Solve

$$\begin{aligned} &T_0 = k \\ &T_n = T_{n-1} + f(n) \textbf{ if } n > 0 \end{aligned} \tag{1}$$

[†] Note the “additivity” in the relation between indices/arguments: n vs. $n - 1$.

From (1), $T_n - T_{n-1} = f(n)$, thus

$$\sum_{i=1}^{n}(T_i - T_{i-1}) = \sum_{i=1}^{n} f(i)$$

the lower summation value dictated by the lowest valid value of $i-1$ according to (1).

☡ **6.2.1 Remark.** The summation in the lhs above is called a "*telescoping (finite) series*" because the terms $T_1, T_2, \ldots, T_{n-1}$ appear both positively and negatively and pairwise cancel. Thus the series "contracts" into $T_n - T_0$ like a (hand held) telescope. □ ☡

Therefore

$$\begin{aligned} T_n &= T_0 + \textstyle\sum_{i=1}^{n} f(i) \\ &= k + \textstyle\sum_{i=1}^{n} f(i) \end{aligned} \tag{2}$$

If we know how to get the sum in (2) in closed form, then we solved the problem!

6.2.2 Example. Solve

$$p_n = \begin{cases} 2 & \textbf{if } n = 1 \\ p_{n-1} + n & \textbf{otherwise} \end{cases} \tag{3}$$

Here

$$\sum_{i=2}^{n}(p_i - p_{i-1}) = \sum_{i=2}^{n} i$$

Note the lower bound of the summation: It is here 2, to allow for the lowest $i-1$ value possible. That is 1 according to 3, hence $i = 2$.

Thus,

$$p_n = 2 + \frac{(n+2)(n-1)}{2}$$

(Where did I get the $(n+2)(n-1)/2$ from?) The above answer is the same as (verify!)

$$p_n = 1 + \frac{(n+1)n}{2}$$

obtained by writing

$$2 + \sum_{i=2}^{n} i = 1 + \sum_{i=1}^{n} i$$

Subcase 2. $(a \neq 1)$ Solve

$$\begin{aligned} T_0 &= k \\ T_n &= aT_{n-1} + f(n) \textbf{ if } n > 0 \end{aligned} \tag{4}$$

(4) is the same as

$$\frac{T_n}{a^n} = \frac{T_{n-1}}{a^{n-1}} + \frac{f(n)}{a^n}$$

To simplify notation, set

$$t_n \stackrel{\text{Def}}{=} \frac{T_n}{a^n}$$

thus the recurrence (4) becomes

$$\begin{aligned} t_0 &= k \\ t_n &= t_{n-1} + \frac{f(n)}{a^n} \textbf{ if } n > 0 \end{aligned} \tag{5}$$

By subcase 1, this yields

$$t_n = k + \sum_{i=1}^{n} \frac{f(i)}{a^i}$$

from which

$$T_n = ka^n + a^n \sum_{i=1}^{n} \frac{f(i)}{a^i} \tag{6}$$

6.2.3 Example. As an illustration solve the recurrence below.

$$T_n = \begin{cases} 1 & \textbf{if } n = 1 \\ 2T_{n-1} + 1 & \textbf{otherwise} \end{cases} \tag{7}$$

To avoid trouble, note that the lowest term here is T_1, hence its "translation" to follow the above methodology will be "$t_1 = T_1/2^1 = 1/2$". So, the right hand side of (6) applied here will have "ka^{n-1}" instead of "ka^n" (Why?) and the indexing in the summation will start at $i = 2$ (Why?)

Thus, by (6),

$$\begin{aligned} T_n &= 2^n(1/2) + 2^n \textstyle\sum_{i=2}^n \frac{1}{2^i} \\ &= 2^{n-1} + 2^n\left(\frac{(2^{-1})^{n+1}-1}{2^{-1}-1} - 1 - \frac{1}{2}\right) \\ &= 2^{n-1} + 2^n(2 - 2^{-n} - 1 - \tfrac{1}{2}) \\ &= 2^n - 1 \end{aligned}$$

In the end you will probably agree that it is easier to redo the work with (7) directly, first translating it to

$$t_n = \begin{cases} 1/2 & \textbf{if } n = 1 \\ t_{n-1} + 1/2^n & \textbf{if } n > 1 \end{cases} \tag{8}$$

rather than applying (6)!

We immediately get from (8)

$$T_n = 2^n t_n = 2^n \Big(1/2 + \sum_{i=2}^{n} 1/2^i\Big) = 2^n \Big(1/2 + \frac{(2^{-1})^{n+1} - 1}{2^{-1} - 1} - 1 - 1/2\Big)$$

etc.

The red terms are subtracted as they are missing from our $\sum$. The blue formula used is for

$$\sum_{i=0}^{n} 1/2^i \qquad \square$$

6.3. *Solving recurrences; the multiplicative case*

Subcase 1.

$$T(n) = \begin{cases} k & \textbf{if } n = 1 \\ aT(n/b) + c & \textbf{if } n > 1 \end{cases} \tag{1}$$

were a, b are positive integer constants ($b > 1$) and k, c any constants. Recurrences like (1) above arise in *divide and conquer* solutions to problems. For example, *binary search* has timing governed by the above recurrence with $b = 2, a = c = k = 1$.

Why does (1) with the above-mentioned parameters —$b = 2, a = c = k = 1$— capture the run time of binary search? First off, regarding “run time” let us be *specific*: we mean number of comparisons.

OK, to do such a search on a sorted (ascending order, say) array of length n, you first check the mid point (for a match with what you are searching for). If you found what you want, exit. If not, you know (due to the ordering) whether you should search the left half or the right half. So you call the procedure recursively on an arrow of length about $n/2$. This decision *and* call took $T(n/2) + 1$ comparisons. This equals $T(n)$. If the array has length 1, then you spend just one comparison, $T(1) = 1$.

We seek a general solution in big-O notation.

First convert to an “additive case” problem: To this end, seek a solution in the *restricted* set $\{n \in \mathbb{N} : n = b^m \text{ for some } m \in \mathbb{N}\}$. Next, set

$$t(m) = T(b^m) \tag{2}$$

so that the recurrence becomes

$$t(m) = \begin{cases} k & \textbf{if } m = 0 \\ at(m-1) + c & \textbf{if } m > 0 \end{cases} \tag{3}$$

hence, from the work in the previous section,

$$\sum_{i=1}^{m} \Big(\frac{t(i)}{a^i} - \frac{t(i-1)}{a^{i-1}}\Big) = c \sum_{i=1}^{m} a^{-i}$$

therefore

$$t(m) = a^m k + ca^m \begin{cases} m & \textbf{if } a = 1 \\ a^{-1}\dfrac{(a^{-1})^m - 1}{a^{-1} - 1} & \textbf{if } a \neq 1 \end{cases}$$

or, more simply,

$$t(m) = \begin{cases} k + cm & \textbf{if } a = 1 \\ a^m k + c\dfrac{a^m - 1}{a - 1} & \textbf{if } a \neq 1 \end{cases}$$

Using O-notation, and going back to T we get:

$$T(b^m) = \begin{cases} O(m) & \textbf{if } a = 1 \\ O(a^m) & \textbf{if } a \neq 1 \end{cases} \tag{4}$$

or, *provided we remember that this solution relies on the assumption that* n has the form b^m:

$$T(n) = \begin{cases} O(\log n) & \textbf{if } a = 1 \\ O(a^{\log_b n}) & \textbf{if } a \neq 1 \end{cases} = \begin{cases} O(\log n) & \textbf{if } a = 1 \\ O(n^{\log_b a}) & \textbf{if } a \neq 1 \end{cases} \tag{5}$$

☡ If $a > b$ then we get slower than linear "run time" $O(n^{\log_b a})$. If on the other hand $b > a > 1$ then we get a sublinear run time, since then $\log_b a < 1$. ☡

☡☡ The symbol ☡☡ means "can be omitted with loss of continuity".

Now an important observation. For functions $T(n)$ that are *increasing*,[†] i.e., $T(i) \le T(j)$ if $i < j$ the restriction of n to have form b^m proves to be *irrelevant* in obtaining the solution. The solution is still given by (5) *for all* n. Here's why:

In the general case, n satisfies

$$b^{m-1} < n \le b^m \text{ for some } m \ge 0 \tag{6}$$

Suppose now that $a = 1$ (upper case in (4)). We want to establish that $T(n) = O(\log n)$ for the general n (of (6)). By monotonicity of T and the second inequality of (6) we get

$$T(n) \overset{\text{by (6) right}}{\le} T(b^m) \overset{\text{by (4)}}{=} O(m) \overset{\text{by (6) left}}{=} O(\log n)$$

The last invocation of (6) above used the first inequality therein.

The case where $a > 1$ is handled similarly. Here we found an answer $O(n^r)$ (where $r = \log_b a > 0$) *provided* $n = b^m$ (some m). Relax this proviso, and assume (6).

Now

$$T(n) \overset{\text{by (6) right}}{\le} T(b^m) \overset{\text{by (4)}}{=} O(a^m) = O((b^m)^r) \overset{\text{Why?}}{=} O((b^{m-1})^r) \overset{\text{by (6) left}}{=} O(n^r)$$

where again the last invocation of (6) above used the first inequality therein. ☡☡

[†] Such are the "complexity" or "timing" functions of algorithms.

Subcase 2.

$$T(n) = \begin{cases} k & \textbf{if } n = 1 \\ aT(n/b) + cn & \textbf{if } n > 1 \end{cases} \tag{$1'$}$$

were a, b are positive integer constants ($b > 1$) and k, c any constants. Recurrences like ($1'$) above also occur in divide and conquer solutions to problems. For example, *two-way merge sort* has timing governed by the above recurrence with $a = b = 2$ and $c = 1/2$. Quicksort has *average* run time governed, essentially, by the above with $a = b = 2$ and $c = 1$. Both lead to $O(n \log n)$ solutions. Also, *Karatsuba integer multiplication* has a run time recurrence as above with $a = 3, b = 2$.

These examples are named for easy look up, in case the trigger your interest or curiosity. It is not in the design of this course to expand on them. Merge Sort and Quicksort you might see in a course on data structures (e.g., EECS 2011) while Karatsuba's "fast multiplication" of natural numbers might appear in a course on algorithms like EECS 3101.

Setting at first (our famous *initial* restriction on n) $n = b^m$ for some $m \in \mathbb{N}$ and using (2) above we end up with a variation on (3):

$$t(m) = \begin{cases} k & \textbf{if } m = 0 \\ at(m-1) + cb^m & \textbf{if } m > 0 \end{cases} \tag{$3'$}$$

thus we need do

$$\sum_{i=1}^{m} \left(\frac{t(i)}{a^i} - \frac{t(i-1)}{a^{i-1}}\right) = c \sum_{i=1}^{m} (b/a)^i$$

therefore

$$t(m) = a^m k + ca^m \begin{cases} m & \textbf{if } a = b \\ (b/a)\dfrac{(b/a)^m - 1}{b/a - 1} & \textbf{if } a \neq b \end{cases}$$

Using O-notation, and using cases according as to $a < b$ or $a > b$ we get:

$$t(m) = \begin{cases} O(b^m m) & \textbf{if } a = b \\ a^m O(1) = O(a^m) & \textbf{if } b < a \qquad /^*\ (b/a)^m \to 0 \text{ as } m \to \infty\ ^*/ \\ O(b^m - a^m) = O(b^m) & \textbf{if } b > a \end{cases}$$

or, in terms of T and n, which is *restricted* to form b^m (using same calculational "tricks" as before):

$$T(n) = \begin{cases} O(n \log n) & \textbf{if } a = b \\ O(n^{\log_b a}) & \textbf{if } b < a \\ O(n) & \textbf{if } b > a \end{cases} \tag{$4'$}$$

The above solution is valid for *any* n without restriction, *provided* T is increasing. The proof is as before, so we will not redo it (you may wish to check the "new case" $O(n \log n)$ as an exercise).

In terms of complexity of algorithms, the above solution says that in a divide and conquer algorithm (governed by $(1')$) we have the following cases:

- The total size of all subproblems we solve (recursively) is *equal* to the original problem's size. Then we have a $O(n \log n)$ algorithm (e.g., merge sort).

- The total size of all subproblems we solve is *more* than the original problem's size. Then we go worse than linear ($\log_b a > 1$ in this case). An example is Karatsuba multiplication that runs in $O(n^{\log_2 3})$ time.

- The total size of all subproblems we solve is *less* than the original problem's size. Then we go in linear time (e.g., the problem of finding the k-th smallest in a *set* of n elements).

☡☡

6.4. Generating Functions

We saw some simple cases of recurrence relations with additive and multiplicative index structure (we reduced the latter to the former). Now we turn to a wider class of additive index structure problems where our previous technique of utilizing a "telescoping sum"

$$\sum_{i=1}^{n}(t(i) - t(i-1))$$

does not apply because the right hand side still refers to $t(i)$ for some $i < n$. Such is the case of the well known Fibonacci sequence F_n given by

$$F_n = \begin{cases} 0 & \textbf{if } n = 0 \\ 1 & \textbf{if } n = 1 \\ F_{n-1} + F_{n-1} & \textbf{if } n > 1 \end{cases}$$

The method of *generating functions* that solves this harder problem also solves the previous problems we saw.

Here's the method in *outline*. We will then embark on a number of fully worked out examples.

Given a recurrence relation

$$t_n = \ldots t_{n-1} \ldots t_{n-2} \ldots t_{n-3} \ldots \tag{1}$$

with the appropriate "starting" (initial) conditions. We want t_n in "closed form" in terms of known functions. Here are the steps:

1. Define a *generating function* of the *sequence* $t_0, t_1, \ldots, t_n, \ldots$

$$\begin{aligned} G(z) &= \sum_{i=0}^{\infty} t_i z^i \\ &= t_0 + t_1 z + t_2 z^2 + \cdots + t_n z^n + \cdots \end{aligned} \tag{2}$$

   (2) is a *formal power series*, where *formal* means that we only are interested in the *form* of the "infinite sum" and *not* in any issues of convergence[†] (therefore "meaning") of the sum. It is stressed that our disinterest in convergence matters is *not* a simplifying convenience but it is due to the fact that convergence issues are *irrelevant* to the problem at hand.

☡ In particular, we will *never* have to consider values of z or make substitutions into z. ☡

2. Using the recurrence (1), find a *closed form* of $G(z)$ as a function of z (this *can* be done *prior* to knowing the t_n in closed form!)

3. Expand the closed form $G(z)$ back into a power series

$$\begin{aligned} G(z) &= \sum_{i=0}^{\infty} a_i z^i \\ &= a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n + \cdots \end{aligned} \tag{3}$$

   But now we *do have* the a_n's in terms of known functions, because we know $G(z)$ in closed form! We only need to compare (2) and (3) and proclaim

$$t_n = a_n \quad \text{for } n = 0, 1, \ldots$$

   The problem has been solved.

Steps 2. and 3. embody all the real work. We will illustrate by examples how this is done in practice, but first we need some "tools":

From here on we will put our ☡☡ in use to advise the reader of what can be omitted.

The derivation of these formulas is trivial, but really long, so let us concentrate on 2 or 3 "boxed" <u>results</u> —forgetting the arithmetic!— that we will be employing!
<u>These will be boxed and provided as aids in, e.g., an exam situation.</u>

☡☡ **The Binomial Expansion.** For our purposes we will be content with just one tool, the "binomial expansion theorem" of calculus:

For any *real* m,
$$\begin{aligned} (1+z)^m &= \sum_{r=0}^{\infty} \binom{m}{r} z^r \\ &= \cdots + \binom{m}{r} z^r + \cdots \end{aligned} \tag{4}$$

[†] In Calculus one learns that power series converge in an interval like $|z| < r$ for some real $r \geq 0$. The $r = 0$ case means the series diverges for *all* z.

where for any $r \in \mathbb{N}$ and $m \in \mathbb{R}$

$$\binom{m}{r} \overset{\text{def}}{=} \begin{cases} 1 & \textbf{if } r = 0 \\ \dfrac{m(m-1)\cdots(m-[r-1])}{r!} & \textbf{otherwise} \end{cases} \tag{5}$$

The expansion (4) terminates with last term

$$\binom{m}{m} z^m \overset{\text{by (5)}}{=} z^m$$

as the "binomial theorem of *Algebra* says, iff m is a *positive integer*. In *all* other cases (4) is non-terminating (infinitely many terms). As we remarked before, we will not be concerned with when (4) converges.

Note that (5) gives the familiar

$$\begin{aligned} \tbinom{m}{r} &= \frac{m(m-1)\cdots(m-[r-1])}{r!} \\ &= \frac{m(m-1)\cdots(m-[r-1])(m-r)\cdots 2\cdot 1}{r!(m-r)!} \\ &= \frac{m!}{r!(m-r)!} \end{aligned}$$

when $m \in \mathbb{N}$. *In all other cases we use* (5) for if $m \notin \mathbb{N}$, then "$m!$" is meaningless.

Let us record the very useful special case when m is a *negative integer*, $-n$ $(n > 0)$.

$$\begin{aligned} (1+z)^{-n} &= \cdots + \tfrac{-n(-n-1)\cdots(-n-[r-1])}{r!} z^r + \cdots \\ &= \cdots + (-1)^r \tfrac{n(n+1)\cdots(n+[r-1])}{r!} z^r + \cdots \\ &= \cdots + (-1)^r \tfrac{(n+[r-1])\cdots(n+1)n}{r!} z^r + \cdots \\ &= \cdots + (-1)^r \tbinom{n+r-1}{r} z^r + \cdots \end{aligned} \tag{6}$$

$$(1-z)^{-n} = \cdots + \binom{n+r-1}{r} z^r + \cdots \tag{7}$$

Finally, let us record in "boxes" some important special cases of (6) and (7)

$$\boxed{\begin{aligned} (1-z)^{-1} = \frac{1}{1-z} &= \cdots + \tbinom{r}{r} z^r + \cdots \\ &= \cdots + z^r + \cdots \end{aligned}} \tag{8}$$

The above is the familiar "converging geometric progression" (converging for $|z| < 1$, that is, but this is the last time I'll raise **irrelevant** convergence issues). Two more special cases of (6) will be helpful:

$$\boxed{\begin{aligned} (1-z)^{-2} = \tfrac{1}{(1-z)^2} &= \cdots + \tbinom{r+1}{r} z^r + \cdots \\ &= 1 + 2z + \cdots + (r+1)z^r + \cdots \end{aligned}} \tag{9}$$

and

$$\boxed{\begin{aligned}(1-z)^{-3} = \tfrac{1}{(1-z)^3} &= \cdots + \tbinom{r+2}{r} z^r + \cdots \\ &= 1 + 3z + \cdots + \tfrac{(r+2)(r+1)}{2} z^r + \cdots\end{aligned}} \tag{10}$$

6.4.1 Example. Solve the recurrence

$$\begin{aligned} a_0 &= 1 \\ a_n &= 2a_{n-1} + 1 \qquad \textbf{if } n > 0 \end{aligned} \tag{i}$$

Write (i) as

$$a_n - 2a_{n-1} = 1 \tag{ii}$$

Next, form the generating function for a_n, and a "shifted" copy of it (multiplied by $2z$; z does the shifting) underneath it (this was "inspired" by (ii)):

$$\begin{aligned} G(z) &= a_0 + a_1 z + a_2 z^2 + \cdots + a_n z^n + \cdots \\ 2zG(z) &= \phantom{a_0 +{}} 2a_0 z + 2a_1 z^2 + \cdots + 2a_{n-1} z^n + \cdots \end{aligned}$$

Subtract the above term-by-term to get

$$\begin{aligned} G(z)(1-2z) &= 1 + z + z^2 + z^3 + \cdots \\ &= \frac{1}{1-z} \end{aligned}$$

Hence

$$G(z) = \frac{1}{(1-2z)(1-z)} \tag{iii}$$

(iii) is $G(z)$ in closed form. To expand it back to a (known) power series we first use the "*partial fractions*" method (familiar to students of calculus) to write $G(z)$ as the sum of *two* fractions with *linear* denominators. I.e., find constants A and B such that (iv) below is true for all z:

$$\frac{1}{(1-2z)(1-z)} = \frac{A}{(1-2z)} + \frac{B}{(1-z)}$$

or

$$1 = A(1-z) + B(1-2z)$$

Setting in turn $z \leftarrow 1$ and $z \leftarrow 1/2$ we find $B = -1$ and $A = 2$, hence

$$\begin{aligned} G(z) &= \frac{2}{1-2z} - \frac{1}{1-z} \\ &= 2\big(\cdots (2z)^n \cdots\big) - \big(\cdots z^n \cdots\big) \\ &= \cdots (2^{n+1} - 1) z^n \cdots \end{aligned}$$

Comparing this known expansion with the original power series above, we conclude that

$$a_n = 2^{n+1} - 1$$

Of course, we solved this problem much more easily in Section 6.2. However due to its simplicity it was worked out here again to illustrate this new method. Normally, you apply the method of generating functions when there is *no other simpler way to do it.*

6.4.2 Example. Solve

$$\begin{aligned} p_1 &= 2 \\ p_n &= p_{n-1} + n \qquad \textbf{if } n > 1 \end{aligned} \qquad (i)$$

Write (i) as

$$p_n - p_{n-1} = n \qquad (ii)$$

Next, form the generating function for p_n, and a "shifted" copy of it underneath it (this was "inspired" by (ii)).
Note how this sequence starts with p_1 (rather than p_0). Correspondingly, the constant term of the generating function is p_1.

$$\begin{aligned} G(z) &= p_1 + p_2 z + p_3 z^2 + \cdots + p_{n+1} z^n + \cdots \\ zG(z) &= \qquad p_1 z \; + p_2 z^2 + \cdots + p_n z^n \quad + \cdots \end{aligned}$$

Subtract the above term-by-term to get

$$\begin{aligned} G(z)(1-z) &= 2 + 2z + 3z^2 + 4z^3 + \cdots + (n+1)z^n + \cdots \\ &= 1 + \frac{1}{(1-z)^2} \qquad \text{by (9)} \end{aligned}$$

Hence

$$\begin{aligned} G(z) &= \tfrac{1}{1-z} + \tfrac{1}{(1-z)^3} \\ &= \big(\cdots z^n \cdots\big) + \big(\cdots \tfrac{(n+2)(n+1)}{2} z^n \cdots\big) \qquad \text{by (10)} \\ &= \cdots \left(1 + \frac{(n+2)(n+1)}{2}\right) z^n \cdots \end{aligned}$$

Comparing this known expansion with the original power series above, we conclude that

$$p_{n+1} = 1 + \frac{(n+2)(n+1)}{2}, \text{ the coefficient of } z^n$$

or

$$p_n = 1 + \frac{(n+1)n}{2}$$

6.4.3 Example. Here is one that cannot be handled by the techniques of Section 6.2.

$$\begin{aligned} s_0 &= 1 \\ s_1 &= 1 \\ s_n &= 4s_{n-1} - 4s_{n-2} \qquad \textbf{if } n > 1 \end{aligned} \qquad (i)$$

Write (i) as

$$s_n - 4s_{n-1} + 4s_{n-2} = 0 \qquad (ii)$$

to "inspire"

$$\begin{array}{lllll} G(z) & = s_0 + s_1 z + s_2 z^2 & + \cdots + s_n z^n & + \cdots \\ 4zG(z) & = \quad 4s_0 z + 4s_1 z^2 & + \cdots + 4s_{n-1} z^n & + \cdots \\ 4z^2 G(z) & = \qquad\quad 4s_0 z^2 & + \cdots + 4s_{n-2} z^n & + \cdots \end{array}$$

By (ii),

$$\begin{aligned} G(z)(1-4z+4z^2) &= 1 + (1-4)z \\ &= 1 - 3z \end{aligned}$$

Since $1-4z+4z^2 = (1-2z)^2$ we get

$$\begin{aligned} G(z) &= \tfrac{1}{(1-2z)^2} - 3z\tfrac{1}{(1-2z)^2} \\ &= \big(\cdots(n+1)(2z)^n\cdots\big) - 3z\big(\cdots(n+1)(2z)^n\cdots\big) \\ &= \big(\cdots\big[(n+1)2^n - 3n2^{n-1}\big]z^n\cdots\big) \end{aligned}$$

Thus,

$$\begin{aligned} s_n &= (n+1)2^n - 3n2^{n-1} \\ &= 2^{n-1}(2n+2-3n) \\ &= 2^n(1-n/2) \end{aligned}$$

6.4.4 Example. Here is another one that cannot be handled by the techniques of Section 6.2.

$$\begin{aligned} s_0 &= 0 \\ s_1 &= 8 \\ s_n &= 2s_{n-1} + 3s_{n-2} \quad \textbf{if } n > 1 \qquad (i) \end{aligned}$$

Write (i) as

$$s_n - 2s_{n-1} - 3s_{n-2} = 0 \qquad (ii)$$

Next,

$$\begin{array}{lllll} G(z) & = s_0 + s_1 z + s_2 z^2 & + \cdots + s_n z^n & + \cdots \\ 2zG(z) & = \quad 2s_0 z + 2s_1 z^2 & + \cdots + 2s_{n-1} z^n & + \cdots \\ 3z^2 G(z) & = \qquad\quad 3s_0 z^2 & + \cdots + 3s_{n-2} z^n & + \cdots \end{array}$$

By (ii),

$$G(z)(1-2z-3z^2) = 8z$$

The roots of $1-2z-3z^2=0$ are

$$z = \frac{-2 \pm \sqrt{4+12}}{6} = \frac{-2 \pm 4}{6} = \begin{cases} -1 \\ 1/3 \end{cases}$$

hence $1-2z-3z^2 = -3(z+1)(z-1/3) = (1-3z)(1+z)$, therefore

$$G(z) = \frac{8z}{(1-3z)(1+z)} = \frac{A}{1-3z} + \frac{B}{1+z} \text{ splitting into partial fractions}$$

By a calculation as in the previous example, $A = 2$ and $B = -2$, so

$$\begin{aligned}G(z) &= \tfrac{2}{1-3z} - \tfrac{2}{1+z}\\ &= 2\big(\cdots(3z)^n\cdots\big) - 2\big(\cdots(-z)^n\cdots\big)\\ &= (\cdots[2\cdot 3^n - 2(-1)^n]z^n\cdots)\end{aligned}$$

hence $s_n = 2\cdot 3^n - 2(-1)^n$

6.4.5 Example. The Fibonacci recurrence.

$$\begin{aligned}F_0 &= 0\\ F_1 &= 1\\ F_n &= F_{n-1} + F_{n-2} \quad \textbf{if } n > 1 \qquad (i)\end{aligned}$$

Write (i) as

$$F_n - F_{n-1} - F_{n-2} = 0 \qquad (ii)$$

Next,

$$\begin{aligned}G(z) &= F_0 + F_1z + F_2z^2 + \cdots + F_nz^n + \cdots\\ zG(z) &= \quad F_0z + F_1z^2 + \cdots + F_{n-1}z^n + \cdots\\ z^2G(z) &= \quad\quad F_0z^2 + \cdots + F_{n-2}z^n + \cdots\end{aligned}$$

By (ii),

$$G(z)(1 - z - z^2) = z$$

The roots of $1 - z - z^2 = 0$ are

$$z = \frac{-1 \pm \sqrt{1+4}}{2} = \begin{cases}\dfrac{-1+\sqrt{5}}{2}\\ \dfrac{-1-\sqrt{5}}{2}\end{cases}$$

For convenience of notation, set

$$\phi_1 = \frac{-1+\sqrt{5}}{2}, \qquad \phi_2 = \frac{-1-\sqrt{5}}{2} \qquad (iii)$$

Hence

$$\begin{aligned}1 - z - z^2 &= -(z-\phi_1)(z-\phi_2)\\ &= -(\phi_1 - z)(\phi_2 - z) \qquad (iv)\end{aligned}$$

therefore

$$G(z) = \frac{z}{1-z-z^2} = \frac{A}{\phi_1 - z} + \frac{B}{\phi_2 - z} \text{ splitting into partial fractions}$$

from which (after some arithmetic that I will not show),

$$A = \frac{\phi_1}{\phi_1 - \phi_2}, \qquad B = \frac{\phi_2}{\phi_2 - \phi_1}$$

so

$$\begin{aligned}G(z) &= \tfrac{1}{\phi_1-\phi_2}\left[\tfrac{\phi_1}{\phi_1-z} - \tfrac{\phi_2}{\phi_2-z}\right]\\ &= \tfrac{1}{\phi_1-\phi_2}\left[\tfrac{1}{1-z/\phi_1} - \tfrac{1}{1-z/\phi_2}\right]\\ &= \tfrac{1}{\phi_1-\phi_2}\left(\left(\cdots [\tfrac{z}{\phi_1}]^n \cdots\right) - \left(\cdots [\tfrac{z}{\phi_2}]^n \cdots\right)\right)\end{aligned}$$

therefore

$$F_n = \frac{1}{\phi_1 - \phi_2}\left(\frac{1}{\phi_1^n} - \frac{1}{\phi_2^n}\right) \qquad (v)$$

Let’s simplify (v):

First, by brute force calculation, or by using the “known” relations between the roots of a 2nd degree equation, we find

$$\phi_1\phi_2 = -1, \qquad \phi_1 - \phi_2 = \sqrt{5}$$

so that (v) gives

$$\begin{aligned}F_n &= \tfrac{1}{\sqrt{5}}\left(\tfrac{\phi_2^n}{(\phi_1\phi_2)^n} - \tfrac{\phi_1^n}{(\phi_1\phi_2)^n}\right)\\ &= \tfrac{1}{\sqrt{5}}\left((-1)^n \tfrac{\left((1+\sqrt{5})/2\right)^n}{(-1)^n} - (-1)^n \tfrac{\left((1-\sqrt{5})/2\right)^n}{(-1)^n}\right)\\ &= \tfrac{1}{\sqrt{5}}\left(\left[\tfrac{1+\sqrt{5}}{2}\right]^n - \left[\tfrac{1-\sqrt{5}}{2}\right]^n\right)\end{aligned}$$

In particular, we find that

$$F_n = O\left(\left[\frac{1+\sqrt{5}}{2}\right]^n\right)$$

since

$$\left[\frac{1-\sqrt{5}}{2}\right]^n \to 0 \text{ as } n \to \infty$$

since $(1-\sqrt{5})/2$ is about -0.62.

That is, F_n grows exponentially with n, since $|\phi_2| > 1$.

Bibliography

[Dav65] M. Davis, *The undecidable*, Raven Press, Hewlett, NY, 1965.

[Hin78] P. G. Hinman, *Recursion-theoretic hierarchies*, Springer-Verlag, New York, 1978.

[Kle43] S.C. Kleene, *Recursive predicates and quantifiers*, Transactions of the Amer. Math. Soc. **53** (1943), 41–73, [Also in [Dav65], 255–287].

[Knu73] Donald E. Knuth, *The Art of Computer Programming; Fundamental Algorithms*, 2nd ed., vol. 1, Addison-Wesley, 1973.

[Kur63] A.G. Kurosh, *Lectures on General Algebra*, Chelsea Publishing Company, New York, 1963.

[Tou03a] G. Tourlakis, *Lectures in Logic and Set Theory, Volume 1: Mathematical Logic*, Cambridge University Press, Cambridge, 2003.

[Tou03b] ______, *Lectures in Logic and Set Theory, Volume 2: Set Theory*, Cambridge University Press, Cambridge, 2003.

[Tou08] ______, *Mathematical Logic*, John Wiley & Sons, Hoboken, NJ, 2008.