ITCS 531 Mathematics for Computer Science Notes

Rob Egrot

Contents

1 Preface

These notes are for the first 8 weeks of a course taught to first year PhD students at the faculty of ICT at Mahidol university. Incoming students here do not usually have a strong background in mathematics or computation theory, so this course is intended to be a rigorous but relatively gentle introduction to formal mathematics and proofs. These notes have 4 sections which are mostly independent and can be taught or read in different orders or concurrently.

The section on number theory builds the necessary theory to properly understand the theory behind RSA encryption, and also to see proofs in action in the relatively familiar setting of integers and natural numbers.

The section on logic introduces the fundamental concepts of propositional and first-order logic, both because it is generally useful [6], and also because it is necessary to understand some of the computation theory material on the subsequent 532 Foundations of CS course (in particular Turing's solution to Hilbert's entscheidungsproblem and the Cook–Levin theorem). We state and prove the soundness and completeness of propositional logic. We also introduce the concept of first-order languages, theories and models, and we sketch a proof of soundness and completeness. For both propositional and first-order logic we use the natural deduction proof system, because it to some extent mirrors the intuitions governing mathematical thinking.

The section on linear algebra uses an abstract approach modeled on that of [1]. In some ways this is not ideal in computer science, as computation details are often very important, but it is a nice way to develop a deeper understanding and more abstract thinking that will hopefully come in useful to students later. Unfortunately we do not have time in this course to cover very much, but we build up to demonstrating how the abstract approach can lead to elegant proofs of theorems in Euclidean geometry.

The section titled 'Counting' covers the basics of cardinal arithmetic, because it is good general knowledge for a person in a mathematical science to have, and also some general combinatorics techniques such as pigeon hole principle, with many examples. The first version of these notes also contained a subsection on ordinal arithmetic, but this was omitted from later versions as the ratio of useful material to technical details was judged to be too low for the purposes of the course.

Each subsection ends with a small number of exercises, and students are generally advised to attempt all of them as solving problems is essential for developing mathematical skills. If this alone were not enough, results and ideas from the exercises may also be needed elsewhere. Full solutions are provided at the end of the notes. Some exercises are marked 'optional'. This is used when assigning homework to avoid overly long problem sets in some weeks, but students with time available should still attempt them, for the reasons just described. Sometimes the optional questions are more difficult then the standard exercises. Each section ends with some pointers to further reading on the material covered. Where possible the focus is on material that is freely available online.

2 Number Theory

2.1 Prime numbers

Prime numbers are like the elementary particles of arithmetic, in the sense that they cannot be non-trivially divided into smaller pieces, and they form the building blocks from which the other numbers are constructed. Mathematicians have been fascinated by prime numbers for thousands of years, and there are many simple questions about them that need very advanced techniques from abstract mathematics to solve, or are even unsolved to this day. For example, do you know if there are an infinite number of primes p such that $p+2$ is also prime? Well, nobody does at the time of writing, and this is known as the *twin prime conjecture*. In fact, it was only as recently as 2013 that mathematicians were able to prove that there is any finite number k with an infinite number of pairs of primes whose difference is less than k. The first proof of this (published by Yitang Zhang) has this bounding number k set at 70,000,000, but collaborative work building on this proof quickly reduced the possible value of k to 246.

More relevant in computer science, prime numbers and their properties give us important techniques for encryption. Understanding this will be the focus of this course, but to do this we will need some abstract theory.

Notation

- $\mathbb{N}$ is the set *natural numbers*, so $\mathbb{N} = \{0, 1, 2, \ldots\}$.
- $\mathbb{Z}$ is the set of *integers*, so $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$.
- $\mathbb{Q}$ is the set of *rational numbers*. $\mathbb{Q}$ can be thought of as the set of fractions of two integers.
- $\mathbb{R}$ is the set of *real numbers*. $\mathbb{R}$ can be thought of as the set of all numbers expressible as a (possibly infinite) decimal. Every real number that is not rational is *irrational*.
- If X is a set and x is an element, we use $x \in X$ to say that x is a member of X. Note that in ZF set theory, all objects are sets, and there are various rules saying which sets exist and when they can be elements of other sets. We don't need to worry about this now.
- Given two integers $a, b \in \mathbb{Z}$, we say a divides b if there is $c \in \mathbb{Z}$ with $b = ac$. We write $a \mid b$ if a divides b. If a does not divide b we write $a \nmid b$.

Definition 2.1.1 (Prime number)**.** *$n \in \mathbb{N}$ is* prime *if $n > 1$ and, whenever $a, b \in \mathbb{N}$, if $ab = n$ then either $a = 1$ and $b = n$ or vice-versa. We use $\mathbb{P}$ for the set of prime numbers. So $\mathbb{P} = \{2, 3, 5, 7, 11, \ldots\}$. Numbers that are not prime are* composite*.*

The aim in this section is to prove the following two important results about prime numbers. Both these theorems were known to the ancient Greeks.

Theorem 2.1.2 (Fundamental Theorem of Arithmetic)**.** *Every natural number greater than 1 can be expressed as a product of primes. Moreover, this product is unique up to reordering.*

Theorem 2.1.3. *The set of prime numbers is infinite.*

We will prove these at the end of the section, but first we need some facts about numbers.

Lemma 2.1.4. *Let $a, b_1, \ldots, b_n \in \mathbb{Z}$. Then, if $a|b_i$ for all $i \in \{1, \ldots, n\}$, we have $a|(b_1 + \ldots + b_n)$.*

Proof. For each $i \in \{1, \ldots, n\}$ there is k_i with $b_i = k_i a_i$ (by definition of $a|b_i$). So $b_1 + \ldots + b_n = k_1 a + \ldots + k_n a = (k_1 + \ldots + k_n)a$, and so $a|(b_1 + \ldots + b_n)$ as claimed. □

We might wonder if the converse to lemma 2.1.4 is true. In other words, if $a|(b_1 + \ldots + b_n)$ is it always true that $a|b_i$ for all $i \in \{1, \ldots n\}$? Before we get excited and try to prove this, we should test it out in simple cases. Doing this potentially saves us some time and wasted effort, because we notice that, for example $2|(1 + 3)$, but 2 doesn't divide either 1 or 3, and so the converse is not true in general.

This is a good principle to bear in mind when you're not sure if something is true or not. Before trying to prove it, first try to find a simple counterexample. If you can't find one then try to understand why your attempts so far don't work. Often by doing this you see a general principle that you can turn into a proof of the original conjecture. If that doesn't work then you should at least get a better understanding of what a counterexample would have to look like, which can often help you find one. Real mathematics research usually works something like this. You go back and forth between trying to find proofs and counterexamples until, hopefully, you settle the question one way or the other.

Anyway, returning to basic number theory, we're going to need some fairly boring technical lemmas. These are minor results that seem unimportant, but will be necessary to prove the big theorems we are interested in. Some of these seem obviously true, but in mathematics, something isn't obvious unless you know how to prove it. This isn't just pedantry. Sometimes things that are 'obviously true' but difficult to prove turn out to be false.

Lemma 2.1.5. *Let $a, b, c \in \mathbb{Z}$. Then if $a|b$ and $a|(b + c)$ then $a|c$.*

Proof. By definition there are $x, y \in \mathbb{Z}$ with $xa = b$ and $ya = b+c$. So combining these we get $ya = xa + c$, and so $(y - x)a = c$, and so $a|c$ by definition. □

Lemma 2.1.6. *Given $a, b \in \mathbb{N}$ with $a < b$, if c is the highest common factor of a and b, then c is also the highest common factor of $b - a$ and a.*

Proof. By definition there are $x, y \in \mathbb{N}$ with $xc = a$ and $yc = b$. So $(y - x)c = b - a$, and so $c|(b - a)$. In other words, c is a common factor of $b - a$ and a, and we must show it is the largest such factor. If $d|(b - a)$ and $d|a$, then by lemma 2.1.4 we must have $d|b$, and so $d \leq c$ as c is the highest common factor of a and b. So c is the highest common factor of $b - a$ and a as required. □

Proposition 2.1.7 (Euclid's algorithm)**.** *Given $a, b \in \mathbb{N}$ with $a < b$ we can find* $\mathbf{HCF}(a, b)$ *by computing:*

$$\begin{aligned} b &= x_0 a + r_0 \text{ where } r_0 < a \\ a &= x_1 r_0 + r_1 \text{ where } r_1 < r_0 \\ r_0 &= x_2 r_1 + r_2 \text{ where } r_2 < r_1 \\ &\vdots \\ r_{n-3} &= x_{n-1} r_{n-2} + r_{n-1} \text{ where } r_{n-1} < r_{n-2} \\ r_{n-2} &= x_n r_{n-1} + r_n \text{ where } r_n < r_{n-1} \\ r_{n-1} &= x_{n+1} r_n \end{aligned}$$

In which case the HCF is r_n.

Proof. First note that this algorithm is well defined, as, for example, since $r_0 < a$ there are unique x_0 and r_0 such that $b = x_0 a + r_0$. Moreover, the algorithm must terminate, because $r_i < r_{i-1}$, so at some point must reach zero.

The result now follows from lemma 2.1.6, as the remainder r_0, for example, is found by subtracting a from b multiple times. So, if c is the HCF of a and b, then it is also the HCF of a and $b - a$, and of a and $b - 2a$ etc., and so also of a and r_0, as $r_0 = b - x_0 a$. By the same logic, the HCF of a and r_0 must also be the HCF of r_0 and r_1. Continuing this thought process we see that the HCF of a and b must also be the HCF of r_{n-1} and r_n, which can only be r_n, as $r_n < r_{n-1}$.

We can also prove this result by studying the algorithm and applying lemma 2.1.4 to prove that r_n divides a and b, and applying lemma 2.1.5 to show that it is the largest such common factor. □

Corollary 2.1.8 (Bézout's identity)**.** *If $a, b \in \mathbb{N}$ and* $\mathbf{HCF}(a, b) = d$*, then there are $x, y \in \mathbb{Z}$ such that $d = xa + yb$.*

Proof. Euclid's algorithm gives us a method to compute x and y (just start with $d = r_n = r_{n-2} - x_n r_{n-1}$ in the last step and work backwards). For example, the first two steps of this calculation give us:

$$\begin{aligned} r_n &= r_{n-2} - x_n r_{n-1} \\ &= r_{n-2} - x_n (r_{n-3} - x_{n-1} r_{n-2}). \end{aligned}$$

For the sake of convenient notation lets define $b = r_{-2}$, and $a = r_{-1}$. Then, for all $i \in \{0, \dots n\}$, the process we described above replaces occurrences of r_i with a term containing r_{i-1} and r_{i-2}. Thus, ultimately this process produces a complicated expression involving only a and b, and no other r_i values. □

The use of the Euclidean algorithm in the proof above is sometimes called the *extended Euclidean algorithm.* Bézout's identity is not obvious, at least, it's

not obvious to me. However, it is an easy consequence of some simple, maybe even obvious, lemmas. This is the power of mathematics. By systematically collecting facts, we can combine them into proofs of surprising new facts.

As I have described it above, to find x and y we first work forward through the algorithm to find the HCF d, then work backwards to find expressions for x and y involving only a and b. This works, but it is not efficient. There is a computation trick we can use the find the values of x and y simultaneously with d. I will describe this now. Again, use the convention that $b = r_{-2}$ and $a = r_{-1}$. Define also $s_{-2} = 1$, $s_{-1} = 0$, $t_{-2} = 0$ and $t_{-1} = 1$. Notice that $b = r_{-2} = s_{-2}b + t_{-2}a$, and $a = r_{-1} = s_{-1}b + t_{-1}a$. Can we find formulas for general s_n and t_n such that $r_n = s_n b + t_n a$? It turns out that, yes, we can, and by using these formulas in the case where $r_k = d$ we can calculate x and y on the *forward* pass through the algorithm, alongside the calculation of d.

To see this, suppose we have formulas for s_k and t_k so that $r_k = s_k b + t_k a$ for all $k \leq n$ (we know we have these for $n = 1$, as we defined them a moment ago). From Euclid's algorithm we know that $r_{n+1} = r_{n-1} - r_n q_n$, so, using our formulas for s_{n-1}, s_n, t_{n-1} and t_n, we have

$$\begin{aligned} r_{n+1} &= r_{n-1} - r_n q_n \\ &= s_{n-1}b + t_{n-1}a - (s_n b + t_n a)q_n \\ &= (s_{n-1} - s_n q_n)b + (t_{n-1} - t_n q_n)a. \end{aligned}$$

In other words, $s_{n+1} = (s_{n-1} - s_n q_n)$, and $t_{n+1} = (t_{n-1} - t_n q_n)$, and we can compute these recursively on the forward pass through the algorithm as claimed.

Returning to prime numbers, the following lemma gives us an important property. In fact, in some abstract number systems it's used to *define* prime numbers, but we don't need to worry about that now.

Lemma 2.1.9. *Let $p \in \mathbb{P}$ and let $a, b \in \mathbb{N} \setminus \{0\}$. Then, if $p|ab$, either $p|a$ or $p|b$.*

Proof. Suppose $p|ab$ and $p \nmid a$. Then $\mathbf{HCF}(p, a) = 1$, so by corollary 2.1.8 there are $x, y \in \mathbb{Z}$ with $xp + ya = 1$. But since $xp + ya = 1$ it follows that $xpb + yab = b$, and since $p|xpb$ and $p|yab$, by lemma 2.1.4 we must have $p|b$. A similar argument proves that if $p \nmid b$ then we must have $p|a$. □

Note that lemma 2.1.9 generalizes to $p|a_1 \dots a_n \implies p|a_i$ for some $i \in \{1, \dots, n\}$. You can prove this using induction and lemma 2.1.9.

Proof of theorem 2.1.2. There are two parts to this (existence and uniqueness). First we show existence of a prime factorization. We will use something called the well-ordering principle.

Lemma 2.1.10 (Well-ordering principle)**.** *If $X \subseteq \mathbb{N}$ and $X \neq \emptyset$, then X has a smallest element. In other words, every non-empty subset of natural numbers has a smallest member.*

Proof. Since X has at least one element we can pick $x \in X$. Then X has a finite number of elements less than or equal to x. One of these must be smaller than all the others. □

The well-ordering principle is really another way of looking at the principle of induction for natural numbers. This says that if you can prove something is true for 0, and if you can also prove that whenever that thing is true for a number n it must also be true for $n+1$, then it must be true for every natural number. The relationship is that the well-ordering principle says that if a statement is *not* true for some natural number, then there must be a smallest natural number k where it is not true. The way people generally use well-ordering principle arguments is to prove that it's impossible for this smallest k to exist for the some statement. Then they can conclude that the set of natural numbers for which the statement they are interested in is true is empty (i.e. the negation of the statement is true for all natural numbers).

Returning to the proof of existence of a prime factorization, suppose for a contradiction that $n \in \mathbb{N}$ and has no prime factorization. Then by the well-ordering principle (lemma 2.1.10) we can assume without loss of generality that n is the smallest such number. If n is prime then n is its own prime factorization, which would be a contradiction. So n is composite. But then $n = ab$ for some non-trivial factors a and b (non-trivial here means not equal to either 1 or n). But then, by minimality of n, both a and b have prime factorizations, and these combine to give a prime factorization of n. I.e. if $a = p_1 \dots p_k$ and $b = q_1 \dots q_m$ then $n = p_1 \dots p_k q_1 \dots q_m$. This contradicts the assumption that n has no prime factorization.

Now we show uniqueness. Suppose there is $n \in \mathbb{N}$ that has two non-trivially distinct prime factorizations. Appealing to the well-ordering principle we assume that n is minimal with this property.

Suppose n can be factored as $p_1 \dots p_k$, and as $q_1 \dots q_m$. Here p_i and q_j are primes (which may be repeated) for all $1 \leq i \leq k$ and $1 \leq j \leq m$. Then these two factorizations cannot have a prime factor in common, as if they did we could divide both factorizations by this common prime to obtain a number smaller than n. But unique factorization would fail for this new number, and this would contradict minimality of n. So we know that p_1 is not equal to q_i for any $i \in \{1, \dots, m\}$. But $p_1 | n$, and so $p_1 | q_1 \dots q_m$, and so by lemma 2.1.9 we must have $p_1 | q_j$ for some j. But as q_j is prime this is a contradiction, as the only way $p_1 | q_j$ is if $p_1 = q_j$, which we know cannot happen.

Proof of theorem 2.1.3. Suppose there are only a finite number of primes, and that the set of primes is $\{p_1, \dots, p_n\}$. Then consider the number $k = (\prod_{1=1}^{n} p_i) + 1$. By the existence part of theorem 2.1.2 we know there must be a prime number p dividing k. Since $\{p_1, \dots, p_n\}$ contains all the primes we must have $p = p_j$ for some $j \in \{1, \dots, n\}$. But $p_j | k$ and $p_j | \prod_{i=1}^{n} p_i$, and so by lemma 2.1.5 we must have $p_j | 1$, which is a contradiction. So the set of primes must be infinite.

Exercises

Exercise 2.1. *Consider the following (false) theorem:*

Theorem. *If $a, b \in \mathbb{N}$ and $a = b$ then $a = 0$.*

Proof.

$$\begin{aligned} a &= b \\ a^2 &= ab \\ a^2 - b^2 &= ab - b^2 \\ (a-b)(a+b) &= (a-b)b \\ a+b &= b \\ a &= 0 \end{aligned}$$

□

What is wrong with this proof?

Exercise 2.2. *Use the well-ordering principle to show that*

$$2 + 4 + 6 + \ldots + 2n = n(n+1).$$

HINT: If there is a value of n there must be a smallest such value. Show that the existence of this smallest value leads to a contradiction.

Exercise 2.3. *Let $n \in \mathbb{N}$. If n^2 is even must n also be even? Give a proof or a counterexample. HINT: think about the fundamental theorem of arithmetic, specifically the existence of a prime factorization, and also lemma 2.1.9.*

Exercise 2.4. *Let $n \in \mathbb{N} \setminus \{0\}$. Then using theorem 1.2 prove that $\log_5(n)$ is either a natural number or irrational. HINT: Suppose $5^{\frac{a}{b}} = n$. What does this tell us about the ratio $\frac{a}{b}$? The fact that 5 is prime is important. HINT: Suppose $5^{\frac{a}{b}} = n$. What does this tell us about the ratio $\frac{a}{b}$? The fact that 5 is prime is important.*

Exercise 2.5. *Is the result from exercise 2.4 still true if we replace 5 with 4? Provide a proof or a counterexample.*

2.2 Modular arithmetic

If it's 14:00 now, what time will it be in 24 hours? Most of us will be able to answer without much thought that the time will still be 14:00. We are so used to clocks and the way we use them to divide up time that there's nothing mysterious about this calculation at all, but there is some interesting and important mathematics behind it. This is a simple example of what we call *modular arithmetic*. While we don't need to understand the theory of modular arithmetic to tell the time, combining this theory, which we will introduce in this section, with what we learned about prime numbers in the previous section will be the key to unlocking one of the most important developments of the last century, the theory of RSA encryption. We will get to this in the final section of this course, but first we need introduce a mathematical notion of 'equivalence'.

Equivalence relations and modular arithmetic.

Definition 2.2.1 (Equivalence relation)**.** *A binary relation R on a set X is an* equivalence relation *if it has the following three properties.*

1. *$R(x, x)$ for all $x \in X$ (reflexive).*
2. *$R(x, y) \iff R(y, x)$ for all $x, y \in X$ (symmetric).*
3. *$R(x, y)$ and $R(y, z) \implies R(x, z)$ for all $x, y, z \in X$ (transitive).*

If R is an equivalence relation on X, and $x \in X$, then $\{y \in X : R(x, y)\}$ is the *equivalence class* of x. We often write $[x]$ for the equivalence class of x when it's clear what equivalence relation we're talking about. Sometimes we write e.g. $[x]_R$ when we want to make it explicit. Equivalence relations give us a way of grouping objects that are 'essentially the same' together. What 'essentially the same' means depends on the context. For example, it is a principle of monetary systems that, e.g. one $10 bill is, for the purpose of normal use, essentially the same as any other $10 bill. So all $10 bills are equivalent to each other in normal use. On the other hand, photographs are not usually equivalent to each other. For example, a photograph of my family will not usually have the same value to me as a photograph of someone else's family, or even necessarily a different photograph of my own family. However, identical copies of the same photograph will normally be equivalent in everyday use, even though they are physically different objects (or e.g. stored on different computers). We don't need a formal concept of equivalence to handle examples like this, but it will be very useful when things get more abstract.

Example 2.2.2. *Let X be a set of balls. Then 'being the same colour' is an equivalence relation on X. Every ball is the same colour as itself (reflexive), and if x is the same colour as y then y is obviously the same colour as x (symmetric). Similarly, if x and y are the same colour, and also y and z are the same colour, then clearly x and z are the same colour (transitive).*

Example 2.2.3. *'Being friends' is not an equivalence relation on a group of people. We can assume, for the sake of argument, that it's reflexive, though 'being friends with yourself' may sound a bit strange, and it's symmetric by definition. However, it's not usually transitive.*

The equivalence classes of an equivalence relation on a set divide the set into pieces. We can formalize this concept with another definition.

Definition 2.2.4 (Partition)**.** *If X is a set then a* partition *of X is a set of pairwise disjoint subsets of X whose union is equal to X. In other words, a partition of a set divides it into pieces that don't overlap at all.*

Partitions and equivalence relations are different ways of talking about the same thing.

Proposition 2.2.5. *If R is an equivalence relation on X then $\{[x] : x \in X\}$ is a partition of X.*

Proof. We must show that $\{[x] : x \in X\}$ satisfies the two conditions required to be a partition of X. We will use the properties of equivalence relations to make the argument work.

1. We need to show that the union of all the equivalence classes is equal to X. We have $\bigcup_{x\in X}[x] \subseteq X$ because $[x] \subseteq X$ for all x (by definition of $[x]$). Conversely, if $y \in X$ then $y \in [y]$ by reflexivity of R, so $X \subseteq \bigcup_{x\in X}[x]$ and so $\bigcup_{x\in X}[x] = X$ as required.

2. Now we need to show that the equivalence classes a pairwise disjoint, i.e. they don't have any common elements. Suppose $[x] \cap [y] \neq \emptyset$. Then there is $z \in X$ with $R(x, z)$ and $R(y, z)$. But then $R(z, y)$, by symmetry, and so $R(x, y)$ by transitivity. By symmetry again we also have $R(y, x)$. Now, using transitivity and the fact that $R(x, y)$ and $R(y, x)$ we have

$$\begin{aligned} z \in [x] &\iff R(x, z) && \text{(by definition)} \\ &\iff R(y, z) && \text{(by transitivity with } R(x, z) \text{ and } R(y, x)) \\ &\iff z \in [y] && \text{(by definition)} \end{aligned}$$

   So $[x] = [y]$. I.e. x and y define the same equivalence class. In other words, the only way $[x]$ and $[y]$ can fail to be disjoint is if they are actually the same set. This proves that $\{[x] : x \in X\}$ satisfies the 2nd partition condition.

□

The above proposition also has a converse, which you can find in the exercises.

Now we've taken a detour through the concept of equivalence, which we will return to in the exercises, we can start taking modular arithmetic seriously.

Definition 2.2.6 (Modular equality)**.** *Given $x, y \in \mathbb{Z}$, we say $x \equiv y \mod n$ if there is $k \in \mathbb{Z}$ with $x - y = kn$. I.e. if the difference between x and y is a multiple of n. We also write $x \equiv_n y$.*

So, for example, a 24 hour clock uses numbers modulo 24, and if we add 24 to a number on the clock then we get back the same number. In other words, 14:00 is, according to the clock, 'essentially the same' as 38:00, which is 'essentially the same' as 52:00 etc. Since equivalence relations are supposed to be a way of handling things that are 'essentially the same, we might expect to be able to view modular equality as a kind of equivalence relation, and indeed we can.

Proposition 2.2.7. *Let $n \in \mathbb{N}$. Then $\equiv_n$ is an equivalence relation on $\mathbb{Z}$.*

Proof. We must check each condition from definition 2.2.1. Let $x, y \in \mathbb{Z}$.

1. $x - x = 0 = 0n$, so $x \equiv_n x$.

2. If $x - y = kn$ then $y - x = -kn$, and vice versa, so $x \equiv_n y \iff y \equiv_n x$.

3. If $x - y = kn$ and $y - z = ln$, then $x - z = kn + ln = (k + l)n$, so $x \equiv_n y$ and $y \equiv_n z \implies x \equiv_n z$.

□

Properties of modular arithmetic. If the number x is 'essentially the same' as x', and the number y is 'essentially the same as y', then we should expect e.g. $x + y$ to be 'essentially the same' as $x' + y'$, because numbers which are 'essentially the same' should arguably behave in the same way with respect to the ordinary operations of arithmetic. Fortunately, modular equality does satisfy this intuitive condition, which we formalize in the proposition below.

Proposition 2.2.8. *Suppose $x \equiv_n x'$, and $y \equiv_n y'$. Then:*

1. *$x + y \equiv_n x' + y'$, and*
2. *$xy \equiv_n x'y'$.*
3. *For all $k \in \mathbb{N}$, $x^k \equiv_n x'^k$.*

Proof. For the first part suppose $x - x' = kn$ and suppose $y - y' = ln$. Then $(x + y) - (x' + y') = (k + l)n$. I.e. $(x + y) \equiv_n x' + y'$. The second part will be an exercise, and the 3rd part follows from the 2nd part. □

Note that it's not true that $x^y \equiv_n x^{y'}$ when $y \equiv_n y'$. E.g. $5 \equiv_4 1$, but $2^5 = 32 \equiv_4 0$, and $2^1 = 2 \equiv_4 2$.

Despite the obvious differences, modular arithmetic behaves in many ways like ordinary arithmetic. The next proposition summarizes this good behaviour.

Proposition 2.2.9. *Let $n \in \mathbb{N}$. Then the following familiar properties of arithmetic carry over to arithmetic* $\mod n$.

(1) $(x + y) + z \equiv_n x + (y + z)$ for all $x, y, z \in \mathbb{Z}$ (Associativity of addition).

(2) $(xy)z \equiv_n x(yz)$ for all $x, y, z \in \mathbb{Z}$ (Associativity of multiplication).

(3) $x + y \equiv_n y + x$ for all $x, y \in \mathbb{Z}$ (Commutativity of addition).

(4) $xy \equiv_n yx$ for all $x, y \in \mathbb{Z}$ (Commutativity of multiplication).

(5) $x(y + z) \equiv_n (xy) + (xz)$ for all $x, y, z \in \mathbb{Z}$ (Distributivity).

Proof. Because $(x + y) + z = x + (y + z)$, we have

$$((x + y) + z) - (x + (y + z)) = 0 = 0 \times n.$$

This proves (1), and similar simple arguments prove all the other claims too. □

Combining propositions 2.2.8 and 2.2.9 we can also say e.g. that $(x + y \mod n) + z \equiv_n x + (y + z \mod n)$ for all $x, y, z \in \mathbb{Z}$. In other words, it doesn't matter at what point we calculate remainders modulo n. We can wait till the end or do it as we go along, and we will still get the same answer.

Calculations in modular arithmetic. Using the properties of modular arithmetic we can simplify complex seeming expressions, and perform calculations with large numbers without using a computer (or perform calculations with very large numbers on a computer without running out of memory).

Example 2.2.10.

$$2^{345} \equiv_{31} (2^5)^{69} \equiv_{31} 32^{69} \equiv_{31} 1^{69} \equiv_{31} 1$$

We often want to evaluate exponentials in modular arithmetic. We won't always be able to makes things as easy as they are in example 2.2.10, but we definitely want to do better than the naive approach (i.e. calculating x^y then finding the answer mod n). We need to do better than this because, in practical applications, x^y could be too big for our computer to handle. Fortunately, the properties of modular arithmetic we have discovered allow us to break exponentials down into small parts, so the numbers never get too large.

$$\text{If } x \equiv_n x' \text{ and } (x')^{y-1} \equiv_n z, \text{ then } x^y \equiv_n zx'.$$

I.e. to work out $x^y \mod n$, first find $x \mod n$, then find $x(x \mod n) \mod n$ etc. Using this method the numbers never get too big, but we need to perform $y-1$ multiplications, which can take a lot of time. We can speed up the algorithm with a trick. Every number can be written in binary, which represents a sum of powers of 2. So, in particular, we can rewrite x^y so that it is a product of x to the power of various powers of 2. E.g.

$$x^{25} = xx^8x^{16},$$

which corresponds to the fact that 25 is 11001 in binary.

So, to evaluate $x^{25} \mod n$ we can calculate $x \mod n$, then calculate $x^2 \mod n$, then calculate $x^4 \equiv_n (x^2)^2$, then $x^8 \equiv_n (x^4)^2$, then $x^{16} \equiv_n (x^8)^2$. Finally we can multiply them together (mod n) step by step to get the answer. So we only need to perform 6 multiplications (x^2, $(x^2)^2$, $((x^2)^2)^2$, $(((x^2)^2)^2)^2$, $x.x^8$, and $(x.x^8).x^{16}$).

Using this method, in the worst case (i.e. when the binary representation is a string of ones), if l is the length of y when written in binary, we have to perform $(l-1)+(l-1) = 2l-2$ multiplications. This is linear in the length of the binary form of y. With a little thought, we can turn this idea into a neat recursive function. Moreover, evaluating this function does not take a large amount of time or space, so it is practical from a computational perspective.

$$\exp(x,y,n) = \begin{cases} 1 \text{ if } y = 0 \\ (\exp(x, \lfloor \frac{y}{2} \rfloor), n))^2 \mod n \text{ if } y \text{ is even} \\ x(\exp(x, \lfloor \frac{y}{2} \rfloor), n))^2 \mod n \text{ if } y \text{ is odd} \end{cases}$$

This algorithm is not mysterious. The key observation is that, for $y > 0$, we have

$$x^y = \begin{cases} (x^{\frac{y}{2}})^2 \text{ when } y \text{ is even} \\ x.(x^{\frac{y-1}{2}})^2 \text{ when } y \text{ is odd.} \end{cases}$$

So, for example,

$$x^{25} = x(x^{12})^2 = x((x^6)^2)^2 = x(((x^3)^2)^2)^2 = x(((x(x)^2)^2)^2)^2 = xx^8x^{16}.$$

To illustrate how the algorithm works in practice we will go through it step by step in the case where $x = 3$ and $n = 4$.

$$\begin{aligned}
3^{25} \mod 4 &= 3(3^{12} \mod 4)^2 \mod 4 \\
&= 3((3^6 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3^3 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3(3 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((3 \cdot 3^2 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(((27 \mod 4)^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3((3^2 \mod 4)^2 \mod 4)^2 \mod 4 \\
&= 3(1^2 \mod 4)^2 \mod 4 \\
&= 3(1^2 \mod 4)^2 \mod 4 \\
&= 3(1^2) \mod 4 \\
&= 3
\end{aligned}$$

Exercises

Exercise 2.6. *Suppose $x \equiv_n y$, and suppose $m|n$. Show that $x \equiv_m y$.*

Exercise 2.7. *Complete the proof of proposition 2.2.8. HINT: Use the fact that $xy - x'y' = xy - xy' + xy' - x'y'$.*

Exercise 2.8. *Calculate $2^{2^{13543}} \mod 3$.*

Exercise 2.9. *Let p and q be distinct primes, and let $x \in \mathbb{Z}$. Prove that if $p|x$ and $q|x$, then $pq|x$.*

Exercise 2.10.

a) Prove that $4 = 9 = -1 \mod 5$.

b) Prove that $4^{1536} \equiv_7 9^{4824}$ (HINT: $9 \equiv_7 2$ and $8 \equiv_7 1$).

c) Using exercise 2.9, and your answers to a) and b), prove that $4^{1536} \equiv_{35} 9^{4824}$.

Exercise 2.11. *Let X be a set and let $\{Y_i : i \in I\}$ be a partition of X (here I is an* indexing set*, i.e. a non-empty set we use to label something, in this case elements of the partition). Prove that the binary relation R, defined by $R(x, y) \iff x$ and y are in Y_i for some $i \in I$, is an equivalence relation.*

Exercise 2.12. *(Optional)*

a) Given an equivalence relation R on a set X, define P_R to be the partition obtained from R in proposition 2.5. Let R_{P_R} be the equivalence relation obtained from P_R as in exercise 2.11. Prove that $R(x,y) \iff R_{P_R}(x,y)$ for all $x, y \in X$.

b) State and prove a similar conjecture on converting from partitions to equivalence relations and back to partitions.

2.3 Primality testing

In the previous section, we showed that arithmetic modulo n 'makes sense'. In other words, we can define operations of addition, subtraction and multiplication on equivalence classes modulo n for all $n \in \mathbb{N} \setminus \{0\}$. The definition below is used to pick out the number system defined by looking at integers mod n for some n.

Definition 2.3.1 ($\mathbb{Z}_n$)**.** *If $n \in \mathbb{N} \setminus \{0\}$ then $\mathbb{Z}_n$ is the set of integers mod n.*

Modular inverses. In standard arithmetic over $\mathbb{R}$, every number except 0 has an inverse under multiplication. That is, for all $x \in \mathbb{R} \setminus \{0\}$ there is $y \in \mathbb{R} \setminus \{0\}$ with $xy = 1$. We write x^{-1} or $\frac{1}{x}$ for the multiplicative inverse of x. In the integers $\mathbb{Z}$, only the numbers 1 and -1 have an inverse, but in $\mathbb{Z}_n$ this is not usually true.

Definition 2.3.2 (Modular multiplicative inverse)**.** *For $a \in \mathbb{Z}$ we define $b \in \mathbb{Z}$ to be the multiplicative inverse, or just the* inverse*, of $a \mod n$ if $ab \equiv_n 1$. We write a^{-1} for the multiplicative inverse (when it exists - see proposition 2.3.5).*

Soon we will prove a result that tells us exactly when integers have an inverse mod n, but first we will need a quick definition and a technical lemma.

Definition 2.3.3 (Coprime)**.** *Integers a and b are* coprime *if their highest common factor (***HCF***) is* 1

Lemma 2.3.4. *Let $a, b, c \in \mathbb{Z}$, let $a|bc$, and let a and b be coprime. Then $a|c$.*

Proof. This is exercise 2.13. □

Now we have all we need to prove the first important result in this section.

Proposition 2.3.5. *Let $a \in \mathbb{Z}$ and let $n \in \mathbb{N} \setminus \{0\}$. Then a has multiplicative inverse $\mod n$ if and only if a and n are coprime. Moreover, the multiplicative inverse of $a \mod n$ is unique in $\mathbb{Z}_n$, whenever it exists.*

Proof. This proof has three parts. We must show that *if* a and n are coprime, *then* a has an inverse in $\mathbb{Z}_n$, and also *if* a has an inverse in $\mathbb{Z}_n$, *then* a and n are coprime, and finally that if b and c are both inverses to $a \mod n$, then $b \equiv_n c$.

First we show that, if a and n are coprime, the multiplicative inverse of $a \mod n$ exists. Since a and n are coprime it follows from corollary 2.1.8 (Bézout's

identity) that there are $x, y \in \mathbb{Z}$ with $xa + yn = 1$. So $xa - 1 = -yn$, but this means that $xa \equiv_n 1$ by definition. So a has an inverse in $\mathbb{Z}_n$ as required.

Now suppose that a has an inverse, and call it x. Then we have $xa \equiv_n 1$, or, in other words, there is y with $xa - 1 = yn$. We can rewrite this as $xa - yn = 1$. Suppose $d|a$ and $d|n$. Then $d|(ax - yn)$, and so $d|1$, by lemma 2.1.4. The only way this can be true is if $d = \pm 1$, and this means $HCF(a, n) = 1$, and so a and n are coprime.

Finally we show that if the inverse exists it is unique mod n. If an inverse to a exists then we have just shown that (a, n) must be coprime. Let $ab \equiv_n 1$ and $ac \equiv_n 1$. Then there are $k, l \in \mathbb{Z}$ with $ab - 1 = kn$ and $ac - 1 = ln$. So $a(b - c) = (k - l)n$. Now, we obviously have $a|a(b - c)$, so by lemma 2.3.4 we must have $a|(k - l)$, and so $b - c = \frac{k-l}{a}n$, and $\frac{k-l}{a} \in \mathbb{Z}$, and thus $b \equiv_n c$. □

Primality testing with Fermat's little theorem Now we know the basics of modular arithmetic, we can start to seriously study prime numbers and prime factorizations. The computational difficulty of finding the prime factors of large numbers is the basis for much of modern cryptography, particularly the RSA encryption system we study in the next section.

An old result about prime numbers known as *Fermat's little theorem* will be important. This neat theorem gives us a kind of detector for numbers which are not prime (i.e. composite numbers), and so with some ingenuity can be turned into a powerful probabilistic method for testing whether a number is prime. First we will need another small technical lemma.

Lemma 2.3.6. *Let $a \in \mathbb{Z} \setminus \{0\}$ and $n \in \mathbb{N} \setminus \{0\}$ be coprime. Then, for all $b, c \in \mathbb{Z}$, if $ab \equiv_n ac$, we have $b \equiv_n c$.*

Proof. Since a and n are coprime, by proposition 2.3.5 we know a has a multiplicative inverse $a^{-1} \pmod{n}$. So $a^{-1}ab \equiv_n a^{-1}ac$, and so $b \equiv_n c$ by definition of the inverse. □

Theorem 2.3.7 (Fermat's little theorem)**.** *If p is prime then $a^{p-1} \equiv_p 1$ whenever a and p are coprime.*

Proof. By lemma 2.3.6 we have

$$\{1, 2, 3, \ldots, p-1\} = \{a \mod p, 2a \mod p, 3a \mod p, \ldots, (p-1)a \mod p\}.$$

This is because the set on the right is obtained by multiplying every element of the set on the left by a, then taking the result mod n. The lemma says that no two distinct elements in the right hand set will produce the same result (mod n) when multiplied by a, so multiplying everything by a and taking the result mod n doesn't change the set. So

$$(p-1)! \equiv_p a^{p-1}(p-1)!, \qquad (\dagger)$$

as the left hand side is obtained by multiplying all elements of $\{1, 2, 3, \ldots, p-1\}$, and the right hand side is obtained by multiplying all elements of $\{a \mod p, 2a$

mod $p, 3a \mod p, \ldots, (p-1)a \mod p\}$. Now, since p is prime, it follows from lemma 2.1.9 that p cannot divide $(p-1)!$, and so p and $(p-1)!$ are coprime. Thus, by proposition 2.3.5 it follows that $(p-1)!$ has an inverse modulo p. Multiplying (†) by this inverse gives $a^{p-1} \equiv_p 1$ as required. □

Fermat's *little* theorem should not be confused with Fermat's *last* theorem.

Theorem 2.3.8 (Fermat's last theorem)**.** *Let $a, b, c \in \mathbb{N} \setminus \{0\}$, and let $n \in \mathbb{N}$ with $n > 2$. Then $a^n + b^n \neq c^n$.*

Proof. Exercise. HINT: See Wiles, A. *Modular elliptic curves and Fermat's last theorem* (1995). □

Fermat's little theorem gives us a computationally efficient way we can test whether a number is prime. Given $n \in \mathbb{N}$ we pick a with $1 < a < n$, then calculate $a^{n-1} \mod n$. If this is not 1 then n is not prime, by Fermat's little theorem (as if n is prime then a would automatically be coprime with n). However, if $a^{n-1} \equiv_n 1$ then we cannot conclude that n is prime. This is because Fermat's little theorem only tells us that *if* p is prime *then* $a^{p-1} \equiv_p 1$. It doesn't say that if p is *not* prime then $a^{p-1} \not\equiv_p 1$. For example, $341 = 11 \times 31$, but $2^{340} \equiv_{341} 1$. Passing Fermat's test does give us evidence that a number is prime though, because of the following result.

Lemma 2.3.9. *Let $n \in \mathbb{N}$ and suppose there is $1 \leq a < n$ such that a is coprime with n and $a^{n-1} \not\equiv_n 1$. Then the modular inequality $b^{n-1} \not\equiv_n 1$ must hold for at least half the natural numbers b less than n.*

Proof. Suppose $b < n$ and b passes Fermat's test (i.e. $b^{n-1} \equiv_n 1$). Then ab fails Fermat's test, because $(ab)^{n-1} = a^{n-1}b^{n-1} \equiv_n a^{n-1} \not\equiv 1$. Moreover, if $ab \equiv_n ac$ for $1 < b, c < n$ then $b = c$ (by lemma 2.3.6). What this means is that every element that passes Fermat's test has a partner that doesn't, and these partners are all distinct, so there are at least as many elements that fail as that pass. □

It follows from lemma 2.3.9 that, if n is not prime, then so long as there is at least one coprime $a < n$ that fails Fermat's test, the test will fail at least 50% of the time. This gives us a reliable, but not infallible, test for determining whether a number n is prime: We repeat Fermat's test k times with different numbers a with $1 < a < n$ (we can choose a randomly each time). If the test fails for any a we conclude with certainty that n is not prime (by the little theorem), and if every test is passed we conclude that the probability that n is not prime must be at most $\frac{1}{2^k}$ (because, if n is not prime, assuming there is at least one value a that is coprime with n with $a^{n-1} \not\equiv_n 1$, every randomly picked a provides at least a 50% chance of making n fail the test). So, if we choose a value of k such that $\frac{1}{2^k}$ is 'small enough', if n passes every round of this testing procedure we can conclude with high probability that n is prime. This test is always correct when it says a number is composite, but it occasionally says a number is prime when actually it is not. In other words, if n is prime, then the test will give the correct answer, but if n is composite, then there is a small chance it will get the answer wrong.

Carmichael numbers. There is a small problem with Fermat's test as we have described it. Lemma 2.3.9 relies on the existence of at least one a that is coprime with n and fails Fermat's test (i.e. $a^{n-1} \not\equiv_n 1$). Unfortunately, there are composite numbers where every coprime a passes Fermat's test. These numbers are called *Carmichael numbers*. The smallest Carmichael number is 561. This is not prime as $561 = 3 \times 11 \times 17$, but for every $1 < a < 561$ that is coprime to 561 we have $a^{560} \equiv_{561} 1$.

So, lemma 2.3.9 does not apply to 561, and the probability calculation we used for Fermat's test is not correct. There are an infinite number of Carmichael numbers, but fortunately they are quite rare, so we can use Fermat's test naively and most of the time we will not have a problem. Alternatively, we can use more advanced methods, like the Rabin-Miller test (which is based on Fermat's test), that give correct probability bounds, taking Carmichael numbers into account.

Lagrange's theorem. Lagrange's theorem, at least, the one we're going to talk about here (there are several important results named after Lagrange), concerns the number of roots of polynomial equations with integer coefficients. Remember that a polynomial with variable x and degree n is a function

$$a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n,$$

where $a_0, \ldots, a_n$ are fixed parameters, usually taken from $\mathbb{R}$ or some subset like $\mathbb{Q}$ or $\mathbb{Z}$ (here we are interested in $\mathbb{Z}$). It is well known that a polynomial can have, at most, the same number of real roots as its degree (remember a *root* of a single variable function f is a value x such that $f(x) = 0$). What is known as the Fundamental Theorem of Algebra tells us that we can always factorize a polynomial over the complex numbers using its roots, but this is not in the scope of this course. We will show soon that the limit on the number of roots of a polynomial we have just described also applies to polynomials over $\mathbb{Z}_p$, when p is prime. First note the following observation, expressed as a lemma.

Lemma 2.3.10. *If $x, y \in \mathbb{R}$ then*

$$\frac{x^n - y^n}{x - y} = x^{n-1} + x^{n-2}y + x^{n-3}y^2 + \ldots + xy^{n-2} + y^{n-1}.$$

Proof. Direct calculation of $(x-y)(x^{n-1}+x^{n-2}y+x^{n-3}y^2+\ldots+xy^{n-2}+y^{n-1})$ shows it is equal to $x^n - y^n$. □

The point of this lemma is that if x and y are variables, or if one is a variable and the other is a constant, the polynomial $x-y$ divides the polynomial x^n-y^n. Now, you might be thinking, if x and/or y are variables, isn't it possible that $x = y$, and then we end up dividing by zero? This is a good question, but it's not actually a problem here, because we're dealing with polynomials. Writing $\frac{x^n-y^n}{x-y} = x^{n-1} + x^{n-2}y + x^{n-3}y^2 + \ldots + xy^{n-2} + y^{n-1}$ is just another way to say $(x - y)(x^{n-1} + x^{n-2}y + x^{n-3}y^2 + \ldots + xy^{n-2} + y^{n-1}) = x^n - y^n$, for all values of x and y. Note that $x^{n-1} + x^{n-2}y + x^{n-3}y^2 + \ldots + xy^{n-2} + y^{n-1}$ is

the unique polynomial that makes this true. In the case where $x = y$ both sides are just zero. Contrast this to the case of numbers. We can't write e.g. $\frac{5}{0} = x$ because there's no value of x that makes $5 = 0 \times x$ true. Now some notation. Let f be a polynomial over $\mathbb{Z}$ of degree n. I.e.

$$f(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_n x^n,$$

where $a_i \in \mathbb{Z}$ for all i and $a_n \neq 0$. Let p be a prime number. We define $f_p(x) = a'_0 + a'_1 x + a'_2 x^2 + \ldots + a'_n x^n$ where each $a'_i = a_i \mod p$ for all i. So f_p is f converted to being a polynomial over $\mathbb{Z}_p$. For example, if $f(x) = 8 + 14x + 3x^2$, then $f_5(x) = 3 + 4x + 3x^2$.

Theorem 2.3.11 (Lagrange)**.** *Let p be prime, let $f(x) = a_0 + a_1 x + \ldots + a_m x^m$ be a polynomial over $\mathbb{Z}$, and let f_p be as above. Suppose the degree of f_p is n. Then, unless every coefficient of f_p is zero, f_p has at most n distinct roots modulo p.*

Proof. The proof of this theorem is a little difficult, but we include it for completeness. You don't need to learn it, but you should remember the statement of the theorem. First note that the degree of f_p must be less than or equal to the degree of f, i.e. we must have $n \leq m$. We induct on n, the degree of f_p. The result is clearly true when $n = 1$, because here we have $f_p = a'_0 + a'_1 x$, and the root occurs when $x \equiv_p -a'_0 a'^{-1}_1$.

Suppose now that the result is true for all $n \leq k$. Let the degree of f_p be $k + 1$. Suppose that f_p has a root b modulo p. In other words $f_n(b) \equiv_p 0$. If such a root does not exist then we are already finished, as if f_p has no roots modulo p it certainly has at most n roots.

Consider the polynomial

$$f_p(x) - f_p(b) = a'_1(x - b) + a'_2(x^2 - b^2) + \ldots + a'_{k+1}(x^{k+1} - b^{k+1}).$$

By lemma 2.3.10, $(x - b)$ divides $(x^l - b^l)$ for all $1 \leq l \leq k + 1$, so we can define a polynomial $g(x) = \frac{f_p(x) - f_p(b)}{x - b}$ over $\mathbb{Z}_p$. Moreover, g has degree at most k.

By definition of g we have $f_p(x) - f_p(b) = (x - b)g(x)$. Let c be a root of $f_p(x)$ modulo p. Then, setting $x = c$ and remembering that b is also a root of f_p modulo p, we get $0 \equiv_p (c - b)g(c)$. I.e. $p|(c - b)g(c)$. Since p is prime this means either $p|(b - c)$, which happens if and only $c \equiv_p b$, or $p|g(c)$, in which case c is a root of $g(x)$ modulo p. But, by the inductive hypothesis, there are at most k roots of g modulo p. So there at most $k + 1$ roots of f_p modulo p, which is what we're trying to prove. □

Exercises

Exercise 2.13. *Prove lemma 2.3.4.*

Exercise 2.14. *Find all solutions to $x^2 - 1 \equiv_8 0$. What does this tell us about Lagrange's theorem in the case where p is not prime?*

Exercise 2.15. *Calculate* $5^{30,000} - 6^{123,456} \mod 31$. *HINT: Fermat's little theorem is useful here.*

Exercise 2.16 (Wilson's theorem)**.**

a) Prove that when $n = 2$ *we have* $(n-1)! \equiv_n -1$.

b) Let p *be an odd prime. Define* $g(x) = (x-1)(x-2)\dots(x-(p-1))$.

i) What are the roots of g *modulo* p*?*

ii) What is the degree of g*?*

iii) What is the leading term of g*? (The leading term is the one with the highest power of* x*).*

c) Define $h(x) = x^{p-1} - 1$. *What are the roots of* h *modulo* p*? HINT: Fermat's little theorem.*

d) Define $f(x) = g(x) - h(x)$. *Prove that* f_p *must be the constant function* $f(x) \equiv_p 0$ *for all* x. *HINT: Lagrange's theorem.*

e) Using the conclusion to part d), prove that n *is prime if and only if*

$$(n-1)! \equiv_n -1$$

(this is known as Wilson's theorem).

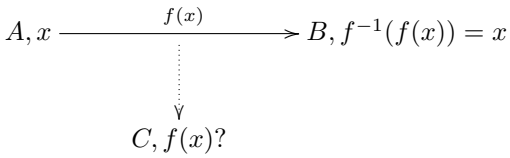
2.4 RSA encryption

Private key encryption. If you want to send someone a message, and you don't want other people to be able to read it, a simple thing you can do is to write the message in code. You agree a code system with the other person, then they can translate your coded messages back into something that makes sense, and you can do the same for theirs. People who don't know the system will have to spend a lot of time and effort cracking your code if they want to read your messages. For convenience we assume that our messages are just numbers. This is reasonable because there are lots of ways we can use numbers to represent strings of, say, English words. We avoid worrying about the details of this translation between words and numbers by just working directly with the numbers. This way we can focus on the mathematics, which is the important part. Formally we can proceed by making the following definition.

Definition 2.4.1 (Encryption function)**.** *An* encryption function *is a bijection between two subsets of* $\mathbb{N}$.

The idea is that there's a set of natural numbers that represent 'meaningful' messages, and the encryption function maps these bijectively with another set of natural numbers representing the encrypted forms of these messages. Since the encryption function is a bijection, going from messages to their encrypted forms and back is well defined. For a simple coded conversation with a predefined set

of meaningful message numbers, two people A and B can agree an encryption function f. Both A and B know f and f^{-1}. If A wants to send a message x to B, she calculates $f(x)$ and sends it to B. To read this message, B calculates $f^{-1}(f(x))$, recovering x. Since f is a bijection it must be invertible, so this is possible. Similarly, B can use f to send coded messages to A, who can read them using f^{-1}. A third person C can intercept the message, but they will only have $f(x)$, from which, if f is a well chosen encryption function, it should be extremely difficult to recover x.



The problem with this system is that A and B, and anyone else who should be able to legitimately read the coded messages, must all know the function f (and its inverse f^{-1}). If C knows f^{-1} then they can read any f-encrypted message easily. So A and B have to keep f^{-1} a secret, but they must also reveal f and f^{-1} to anyone they want to communicate with using this encryption system. The more often they reveal f^{-1}, the more likely that information will leak out to people they don't want to have it. This is not practical for situations where large numbers of coded messages must be sent to a large number of different people.

Public key encryption. Private key cryptography is *symmetrical*, i.e. every person in the conversation has all the information used for encryption. Public key encryption is *asymmetrical*, that is, the sender has less information about the encryption system than the receiver. The typical situation is this. A wants to send B a message x in encrypted form. She calculates $f(x)$ and sends it to B. B then uses a function $g = f^{-1}$ to calculate $g(f(x)) = x$. The idea is that B can broadcast the function f, but keep g a secret, so f must be chosen so that its inverse cannot be easily found. Then anyone can send B a message encrypted with f, but only B will be able to read it, as g is kept secret. This seems almost like magic, but it is possible through the number theory we have studied. More specifically, Fermat's little theorem will play a crucial role.

$$A \xrightarrow{f(x)} B \xleftarrow{f(y)} C$$

RSA encryption. The RSA encryption system, invented in the 1970s and the basis for e-commerce, operates as follows. In this situation A wants to send a message x to B in encrypted form.

1. B chooses two (large) primes p and q. He defines $N = pq$, and he chooses some number $e < (p-1)(q-1)$ that is coprime to $(p-1)(q-1)$. The easiest way to do this is to make e prime, for example, B can set $e = 3$ so

long as $3 \nmid (p-1)(q-1)$. The pair (N, e) is B's *public key*. He makes this information freely available to anyone who wants to send him an encrypted message. We assume that p and q are large, so that $x < N$. If $x \geq N$ there is a problem as we will be working modulo N, so x will be confused with another message.

2. A calculates $x^e \mod N$. This is what she sends to B. There is a number, d, such that $(x^e)^d = x \mod N$ (see lemma 2.4.3 below). This number d is B's *private key*. B keeps d secret. The idea is that it is extremely difficult to calculate d from knowledge of (N, e), but easy to calculate it from p and q. So B can recover x from x^e in a reasonable amount of time, but nobody else can.

$$A \xrightarrow{x^e \mod N} B$$

We now know the idea behind RSA, but we still have some details to go through. In particular, what is d and how does B calculate it? To answer this we need some technical lemmas, which is where Fermat's little theorem comes in.

Lemma 2.4.2. *Let p be prime, and let $a, m \in \mathbb{N}$. Then $a \equiv_{p-1} 1 \implies m^a \equiv_p m$.*

Proof. $a \equiv_{p-1} 1 \iff a - 1 = k(p-1)$ for some $k \in \mathbb{N}$. Assuming this is true, we must prove that $m^a - m \equiv_p 0$. This is obviously true if m and p are not coprime (because then $p|m$), so suppose they are coprime (and thus that $m \neq 0$). Now,

$$\begin{aligned} m^a - m &= m(m^{a-1} - 1) \\ &= m(m^{k(p-1)} - 1). \end{aligned}$$

By assumption of coprimality, Fermat's little theorem says that $m^{p-1} \equiv_p 1$. So, by exercise 4.2 we have $m^{k(p-1)} \equiv_p 1$, and so

$$\begin{aligned} m^a - m &= m(m^{k(p-1)} - 1) \\ &\equiv_p m(1-1) \\ &\equiv_p 0. \end{aligned}$$

□

Using lemma 2.4.2, we can show that d is actually just the inverse of e modulo $(p-1)(q-1)$, as we show in lemma 2.4.3 below. This is good, because we can calculate inverses in modular arithmetic quickly using the extended Euclidean algorithm (which is what we used to prove Bézout's identity in corollary 2.1.8). I.e. since e and $(p-1)(q-1)$ are coprime, we can find y and z so that $ye + z(p-1)(q-1) = 1$, and then y is the inverse of e modulo $(p-1)(q-1)$.

Lemma 2.4.3. *If d is the inverse of e modulo $(p-1)(q-1)$ then $x^{ed} \equiv_N x$ for all $x \in \{0, 1, \ldots, N-1\}$.*

Proof. If d is the inverse of e modulo $(p-1)(q-1)$, then, by definition, we have $ed - 1 = k(p-1)(q-1)$ for some k. Consequently we have $ed \equiv_{p-1} 1$ and $ed \equiv_{q-1} 1$. So, by lemma 2.4.2 we have $x^{ed} \equiv_p x$ and $x^{ed} \equiv_q x$. By exercise 2.4, this means $x^{ed} \equiv_N x$.

□

This system allows A to send B a message x encrypted as x^e, which B can decrypt by calculating $(x^e)^d$. If a third party C wants to read this message they must do one of two things.

1. C can calculate $y^e \mod N$ for all $y < N$. This is linear in the size of N, but computer scientists care about running times in terms of the lengths of inputs in binary. To represent the numbers up to N in binary we need approximately $\log_2 N$ bits, so, as $N = 2^{\log_2 N}$, checking N numbers actually uses exponential operations as a function of the binary length of N.

2. C can factor N into p and q and calculate the inverse of e mod $(p-1)(q-1)$. We don't know for sure, but we believe that there is no efficient algorithm for factoring numbers into their prime factors. At least on classical computers. Peter Shor found an efficient algorithm for integer factorization, but this requires large scale quantum computers, which do not currently exist. Certainly, if such an algorithm exists for 'normal' computers then it is a closely guarded secret.

Cryptography in practice. RSA is not the only public key encryption method in use. There are others, but the common theme is that there must be a process that is easy to perform, but very hard to reverse. RSA, uses the fact that it's easy to multiply two primes, but (probably) very hard to factor a number into two prime factors. The security of the system is based on the idea that B can broadcast the information needed to encrypt messages, but only B can efficiently decrypt them. For real world applications, people usually use a combination of public and private key encryption. Public key encryption is used to transmit information about a private encryption function, which is freshly generated for each interaction, between parties, then further messages are exchanged using that.

What we have described here is sometimes called *textbook RSA*. This is a clean exposition of the mathematical ideas involved, but ignores some issues that are very important if you want to actually implement a secure RSA based system. For example, if you choose a small value of e, say $e = 3$, and you encrypt a message x such that $x < N^{\frac{1}{3}}$, then an attacker could decode the encrypted message (which is just x^3 in this case), simply by calculating the cube root of x^3. There's nothing magic about the number 3 here. This is potentially a problem for all values of e, though larger values of e make it rarer to have $x < N^{\frac{1}{e}}$.

Another potential threat comes from the Chinese remainder theorem, which you can prove as exercise 4.4. Suppose that A wants to send the same message x to B, C and D. Suppose also that B, C and D are all using $e = 3$, and N values N_B, N_C and N_D respectively. Then, if we can intercept all three of the messages A sends, we know the value of

$$\begin{aligned} y_B &= x^3 \mod N_B, \\ y_C &= x^3 \mod N_C, \text{ and} \\ y_D &= x^3 \mod N_D. \end{aligned}$$

If N_B, N_C and N_D are not pairwise coprime, then we can use Euclid's algorithm to efficiently find a common factor for two of them, and then use this to find p and q for one of the N values. Using these we can decode the message directly.

Alternatively, if they *are* pairwise coprime, then the Chinese remainder theorem (exercise 4.4) provides a method for finding y such that $y \equiv_{N_B} y_B$, $y \equiv_{N_C} y_C$ and $y \equiv_{N_D} y_D$. I.e. $N_B|(y-x^3)$, $N_C|(y-x^3)$ and $N_D|(y-x^3)$. Since we are assuming N_B, N_C and N_D are pairwise coprime, it follows from exercise 4.3 that $N_BN_CN_D|(y-x^3)$. Since $x < N_B$, $x < N_C$ and $x < N_D$, we must have $x^3 < N_BN_CN_D$, and since y is unique modulo $N_BN_CN_D$, we must have $y = x^3$. So we can efficiently find x by calculating the cube root of y.

Implementations of RSA usually mitigate these problems by *padding* the encrypted message. That is, the message x is padded with additional random elements to distort the exploitable rigid mathematical structure involved in textbook RSA. There are various ways to do this, and we will not discuss them here.

Exercises

Exercise 2.17. *Let $p = 11$ and $q = 13$. Choose suitable e and d for use in RSA encryption.*

Exercise 2.18. *Prove that if $a \equiv_n b$ then $a^k \equiv_n b^k$ for all $k \in \mathbb{N}$.*

Exercise 2.19. *Let a and b be coprime. Prove that if $a|c$ and $b|c$ then $ab|c$.*

Exercise 2.20 (Chinese remainder theorem)**.** *Let $n_1, \ldots, n_k \in \mathbb{N}$ all be greater than 1 and such that n_i and n_j are coprime for all $i \neq j$. Define $N = \prod_{i=1}^{k} n_i$. For each $i \in \{1, \ldots, k\}$ let $a_i \in \{0, 1, 2, \ldots, n_i - 1\}$.*

a) Let x and y be integers with $x \equiv_{n_i} a_i$ and $y \equiv_{n_i} a_i$ for all i. Prove that $x \equiv_N y$.

b) Find $z \in \mathbb{Z}$ with $z \equiv_{n_1} a_1$ and $z \equiv_{n_2} a_2$. HINT: Bézout.

c) Extend part b) to prove that there is z with $z \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k\}$. HINT: Induction.

Combining parts a) and c) we get that there is a number z such that $z_i \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k\}$, and that this z is unique mod N. This result is known as the Chinese remainder theorem. It is attributed to the 3rd century Chinese mathematician Sunzi, though his version was presented very differently.

2.5 Further reading

More about number theory and cryptography can be found in [2, chapter 1], on which some of the material here is based, and also in [8, chapter 9]. An interesting discussion of the proof of the Fundamental Theorem of Arithmetic, and proofs in general, written by one of the most prominent mathematicians alive today, can be found in [3].

3 Logic

3.1 Semantics for propositional formulas

The logic of mathematical proofs. Mathematical proofs, presented in a formal style, start with assumptions (axioms), and proceed by making a sequence of logical deductions till the desired conclusion is reached. This axiom-theorem-proof style dates back to the Ancient Greeks, particularly the geometric results collected by Euclid in the famous Elements around 300 BCE. This neat picture of mathematics does not really correspond to how mathematicians actually work as, in reality, mathematicians do a lot of work based on informal ideas and intuitions. The modern style of being very explicit about assumptions and definitions that you will see if you read a mathematics paper or advanced text book (or these notes!) really started in the late 19th century. The reason for the change to a more formal style of presentation was that, as mathematics became more advanced, particularly after the development of Calculus by Newton and Leibniz, mathematicians started proving results that seemed to contradict each other. To resolve these apparent contradictions, mathematicians found it was necessary to state very precisely what they were trying to prove and what assumptions they were making. By doing this they were able to see that the contradictions often came from people starting from slightly different assumptions about what they were talking about[1].

So, while the formal style does not correspond to how mathematicians *think*, it is an important part of mathematical communication, as without it mathematicians are not sure whether they are actually talking about the same things. The formal style also helps mathematicians prevent logic errors in their own reasoning. What this means in practice is that mathematicians usually come up with ideas using intuitive reasoning, but they *write them down* in a kind of formal style as a protection against making mistakes, and also so other people can, with effort, understand exactly what they're talking about.

The result of this is that while it is debatable whether the formal style captures the true essence of mathematics, all mathematics should be, in principle, capable of being expressed as a formal procession of axioms and deductions. In other words, mathematics can, in the abstract, be treated as a formal system, and can therefore itself be a subject of mathematical reasoning! This realization opens the door to doing mathematics about mathematics (metamathematics). This abstract work was a crucial step in the development of computers, which we will study next semester. But, before we can understand the role of formal logic in the theory of computation, and also the modern role of formal logic as a tool for reasoning about computer systems, we need to understand the basics, and that is what this course is about. To properly describe mathematical and computational ideas symbolically we need a complex language, but we can think about the abstract structure of logical arguments with a relatively simple formal system.

[1] The philosopher of science and mathematics Imre Lakatos explored the process of mathematical argument, proof and discovery in his famous book *proofs and refutations* [7].

The Ancient Greeks thought a lot about this. For example, Aristotle gave the following example of a logical deduction:

1. *All humans are mortal.*
2. *All Greeks are human.*
3. *Therefore, all Greeks are mortal.*

This is an example of something call a *syllogism*. The conclusion here is true in reality, but also, if we accept the truth of the preceding statements, we must also accept the truth of the conclusion, just because of its form. I.e.

1. *All X have property Y.*
2. *Z is X.*
3. *Therefore, Z has property Y.*

Whatever the values of X, Y and Z, if statements one and two are true, then statement three must be true too. Medieval Christian scholars loved syllogisms, and they studied them intensively for about 300 years starting around the early 12th century. We won't spend any more time on this 'Aristotelian' style of logic though, as syllogisms are not flexible enough to cover all the deductions we understand today as 'logical'. Instead we will develop formal tools for reasoning about the logic of propositions in an essentially mathematical way. The key idea is that, like the syllogism above, we are interested in arguments that are correct or not based only on their logical form, and not on what the basic statements actually mean. We develop a logical theory of propositions by abstracting away the meaning of the propositions, so we can investigate the structure of arguments in a pure form.

Propositional logic For our formal system of propositional logic we need three things:

- A collection of basic propositions (also called *propositional variables*), $\{p_0, p_1, p_2, \ldots\}$. These are used to represent statements that can be either true or false (but not both!). We can also use individual letters, e.g. p, q, r, to stand for basic propositions.
- A set of logical connectives $\{\wedge, \vee, \neg, \rightarrow, \leftrightarrow\}$. These represent ways we can combine true/false statements to create new ones.
  - $\wedge$: $p \wedge q$ is supposed to mean "p and q".
  - $\vee$: $p \vee q$ is supposed to mean "p or q".
  - $\neg$: $\neg p$ is supposed to mean "not p".
  - $\rightarrow$: $p \rightarrow q$ is supposed to mean "p implies q" (we need to be a bit careful here as there are different forms of implication. The one we're interested in is technically known as *material implication*, and we will see what this means soon).

$\leftrightarrow$: $p \leftrightarrow q$ is supposed to mean "p implies q, and q implies p".

- Brackets '(' and ')'. We use these to delimit formulas. In other words, we use brackets to tell us where one formula ends and another begins, so we can make sense of them.

If we assign meaning to some of the basic propositions we can combine them into new statements using the logical connectives and brackets.

Example 3.1.1. *Let $a, b, c \in \mathbb{N}$, and suppose p means "$a|b$", q means "$a|(b+c)$", and r means "$a|c$". Then $(p \wedge q) \rightarrow r$ means "If a divides b, and a divides $(b+c)$, then a divides c". This statement is true, which we proved during the number theory course.*

Example 3.1.2. *Again let $a, b, c \in \mathbb{N}$, and suppose p means "$a|b$", q means "$a|c$", and r means "$a|bc$". Then $(p \wedge q) \leftrightarrow r$ means "a divides b, and a divides b, if and only if a divides bc". This is not true (why?). The 'only if' part is true, but the 'if' part is not (though it looks similar to a true statement).*

Not every string we can make using basic propositions and logical connectives makes sense.

Example 3.1.3. *$(p \rightarrow \wedge q) \vee \neg r$ doesn't make sense, whatever meaning we give to p, q and r. It's not true or false, it just doesn't mean anything.*

Well-formed formulas. Intuitively, propositional formulas are well-formed if they are capable of making sense as true or false statements. We have a recursive system for building well-formed formulas. If we can construct a string using this system, then it is a well-formed formula. Otherwise it is not.

- Individual basic proposition symbols are well-formed formulas.
- If ϕ is well-formed then $\neg\phi$ is well-formed.
- If ϕ and ψ are well-formed then $(\phi * \psi)$ is well-formed for all $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

When we write down formulas we often cheat and leave out some of the brackets (we do this in the examples). We also usually write things like $(p \vee q) \vee r$ as $p \vee q \vee r$. This makes some formulas slightly easier for humans to read. In propositional logic we often refer to well-formed formulas as just *formulas*, and sometimes as *sentences*. We define the *length* of a sentence ϕ to be the number of logical connectives that occur in ϕ. E.g. if $\phi = \neg((p \vee q) \wedge q)$ then the length of ϕ is 3.

If ϕ is a formula, then a subformula of ϕ is a substring of ϕ that is also a sentence (i.e. can be obtained by our recursive construction). We consider ϕ to be a subformula of itself.

Truth tables. Every basic proposition must be either true or false, and cannot be both. The same applies to sentences. Whether a sentence is true or false is completely dependent on the true/false values of the basic propositions it is built from. This truth value can be calculated recursively from the truth values of the basic propositions. We use *truth tables* to represent the recursion rules. Let ϕ and ψ be sentences.

	ϕ	$\neg\phi$
$\neg$:	T	F
	F	T

	ϕ	ψ	$\phi \wedge \psi$
$\wedge$:	T	T	T
	T	F	F
	F	T	F
	F	F	F

	ϕ	ψ	$\phi \vee \psi$
$\vee$:	T	T	T
	T	F	T
	F	T	T
	F	F	F

	ϕ	ψ	$\phi \to \psi$
$\to$:	T	T	T
	T	F	F
	F	T	T
	F	F	T

	ϕ	ψ	$\phi \leftrightarrow \psi$
$\leftrightarrow$:	T	T	T
	T	F	F
	F	T	F
	F	F	T

Look carefully at the truth table for $\to$. What this says is that $\phi \to \psi$ is true whenever ϕ is false or ψ is true. For example, if ϕ is "my eyes are closed" and ψ is "I am sleeping", then we want to understand $\phi \to \psi$ as something like "if my eyes are closed, then I am sleeping". When should this be true? Obviously, if my eyes are closed then it will be true if I'm sleeping, and false if I am not. But what if I'm not sleeping? According to the truth table, in this case the statement will be true. It is often not clear to students why this should be the case. The intuition behind this form of implication (*material implication*, as mentioned previously), is that propositions are a snapshot of the present state of a system. So, if ϕ is not true, then $\phi \to \psi$ makes no claim about the system, and therefore must be considered true, whatever ψ may be.

We can contrast this with other forms of implication. For example, a *subjunctive implication* is something like "if I dropped it, then it would break". Intuitively, this statement should be true for something like a chicken egg, and false for something like a tennis ball, irrespective of whether I actually drop the thing or not. But according to material implication, "if I dropped it, then it would break" would be true for a tennis ball so long as I don't drop it!

Another problem is that might want that for $\phi \to \psi$ to be true there should be some relevant connection between ϕ being true and ψ being true. For example, suppose a student who likes yoghurt studies hard and does well in her exams. Then "she studied hard so she did well in her exams" might be considered a true statement, but "she likes yoghurt so she did well in her exams" should probably not be considered true. This kind of conditional where the first part must be relevant to the second part is called an *indicative implication*. Note that in our example here, according to material implication both statements are true, which doesn't seem right.

So material implication is not appropriate for everything, but it is generally thought to be appropriate for mathematics, which mathematicians like to imag-

ine is an unchanging reality of fixed, eternal truths. Also, as mentioned above, it's useful for reasoning about states of systems, so has many applications in computer science.

Example 3.1.4.

p	q	r	$p \wedge q$	$(p \wedge q) \to r$
T	T	T	T	T
T	T	F	T	F
T	F	T	F	T
T	F	F	F	T
F	T	T	F	T
F	T	F	F	T
F	F	T	F	T
F	F	F	F	T

Tautologies and contradictions. When we set every propositional variable to be either true or false then we are making a *truth assignment* (or just *assignment*). If a sentence is true under a particular assignment we say it is *satisfied* by that assignment. A sentence is *satisfiable* if there is some assignment that satisfies it. In other words, if there is a way we can interpret each basic proposition as true or false so that the whole thing becomes true. If Γ is a set of sentences, then we say Γ is satisfiable if there is an assignment that satisfies every sentence in Γ.

A sentence that is satisfied by every assignment is called a *tautology*. In other words, a tautology is something that is always true, whatever truth values the basic propositions take. A basic example for classical propositional logic is $p \vee \neg p$, which we understand as saying that a proposition must be either true or false. Be careful here though, because in some logic systems this is not something we can assume, difficult though that may be to believe (see section 3.2). We sometimes use the symbol $\top$ to denote a tautology.

A sentence that is not satisfiable is called a *contradiction*. A basic example is $p \wedge \neg p$, which we interpret as saying a proposition cannot be both true and false at the same time. We sometimes use the symbol $\bot$ to denote a contradiction. If ϕ is a tautology then $\neg\phi$ is a contradiction, and vice versa.

Logical implication. If ϕ and ψ are sentences then we say that ϕ *logically implies* ψ (or, equivalently, that ψ is a *logical consequence* of ϕ), if whenever an assignment satisfies ϕ, it also satisfies ψ. We write $\phi \models \psi$, and we should observe that this is another way of saying that $\phi \to \psi$ is true. We say that ϕ and ψ are *logically equivalent* if each is a logical consequence of the other. In this case we write $\phi \models\!=\!\models \psi$.

We can also do this with sets of sentences. If Γ is a set of sentences and ψ is a sentence, then ψ is a logical consequence of Γ if, whenever an assignment satisfies ϕ for all $\phi \in \Gamma$, it also satisfies ψ. We write $\Gamma \models \psi$. We sometimes call a set of sentences a *theory*, and then we might say that ψ is a consequence of the theory Γ. The intuition behind this choice of language is that we want to be able to say things like "the fact that the set of primes is infinite is a consequence

of the theory of numbers". A theory can be empty (i.e. have no members). We write $\models \phi$ if ϕ follows from the empty theory, i.e. if ϕ is a tautology.

Example 3.1.5. *Let Γ be the theory $\{p \wedge \neg q, q \vee r\}$. Then $r \wedge p$ is a logical consequence of Γ (i.e. $\Gamma \models p \wedge r$). We can prove this by writing out a truth table:*

p	q	r	$p \wedge \neg q$	$q \vee r$	$r \wedge p$
T	T	T	F	T	T
T	T	F	F	T	F
T	F	T	T	T	T
T	F	F	T	F	F
F	T	T	F	T	F
F	T	F	F	T	F
F	F	T	F	T	F
F	F	F	F	F	F

Looking at this truth table we see there is only one assignment that makes both $p \wedge \neg q$ and $q \vee r$ true, and that is the one that makes p and r true, and makes q false. Looking at the corresponding row in the truth table we see that $r \wedge p$ is also true with this assignment. So, every assignment that makes everything in Γ true must also make $r \wedge p$ true, which means $\Gamma \models r \wedge p$, by definition.

We could also work this out without writing out the whole truth table, e.g. we might notice just by looking at the formulas that $p \wedge \neg q$ being true means p is true and q is false, and then $q \vee r$ can only be true if r is true, which means $r \wedge p$ must be true too.

For complicated formulas, writing out a truth table is quite a lot of effort, so it's usually a good idea to look at the formulas first and see if you can find a quick argument for why one thing logically implies another. But, if you get stuck, the option of working out the truth table is always there.

Sufficiency of connectives. The set $\{\wedge, \vee, \neg, \rightarrow, \leftrightarrow\}$ is bigger than we need. We can use truth tables to check that some of the connectives can be reproduced using combinations of different ones.

Lemma 3.1.6. *If ϕ and ψ are sentences, then $\neg\phi \vee \psi \models \models \phi \rightarrow \psi$.*

Proof.

ϕ	ψ	$\phi \rightarrow \psi$	$\neg\phi \vee \psi$
T	T	T	T
T	F	F	F
F	T	T	T
F	F	T	T

We can see that the last two columns are the same. □

What this means is that whenever $\phi \rightarrow \psi$ appears in a formula, we could replace it with $\neg\phi \vee \psi$ without changing the truth value of the formula. In other words, we don't really need the connective $\rightarrow$, because for every formula

involving $\rightarrow$ there's an equivalent one where it does not appear. This might be intuitively obvious, but we will provide proof now, as the proof method will be very important.

Corollary 3.1.7. *If ϕ is a sentence, then there is a sentence ϕ' where the symbol $\rightarrow$ does not occur, and with $\phi \models\!\!\models \phi'$.*

Proof. We induct on the length of ϕ. In the base case $n = 0$, the only possibility is that ϕ is a basic proposition. In this just we can define ϕ' to be ϕ, and the result is automatic. For the inductive step, suppose the result is true for every formula of length n, and let ϕ have length $n + 1$. There are three cases.

1. $\phi = \neg\psi$ for some ψ. In this case the length of ψ is n, and so the inductive hypothesis applies and gives us ψ' that does not contain '$\rightarrow$' and with $\psi \models\!\!\models \psi'$. We can then define $\phi' = \neg\psi'$ to complete the proof, as, since $\psi \models\!\!\models \psi'$ we must have $\neg\psi \models\!\!\models \neg\psi'$, and $\phi = \neg\psi$.

2. $\phi = \psi_1 * \psi_2$ for some $* \in \{\wedge, \vee, \leftrightarrow\}$. In this case the inductive hypothesis applies to ψ_1 and ψ_2, and so we define $\phi' = \psi_1' * \psi_2'$.

3. $\phi = \psi_1 \rightarrow \psi_2$. In this case we apply the inductive hypothesis to ψ_1 and ψ_2, then lemma 3.1.6 says we can define $\phi' = \neg\psi_1' \vee \psi_2'$.

□

Definition 3.1.8 (Functionally complete set of connectives)**.** *A set of connectives defined by truth tables, S, is* functionally complete *if every formula that can be constructed from $\{\wedge, \vee, \neg, \rightarrow, \leftrightarrow\}$ is logically equivalent to one constructed from S (using a recursive process analogous to the one we defined earlier).*

Proposition 3.1.9. *$\{\wedge, \vee, \neg, \leftrightarrow\}$ is functionally complete.*

Proof. This is what we showed in corollary 3.1.7. □

The proof of corollary 3.1.7 generalizes to other connectives. So if we can show that any sentence containing a particular connective is equivalent to another not containing it, then we know that we will still have a functionally complete set of connectives if we eliminate that connective. We will see in the exercises that $\{\wedge, \neg\}$ is functionally complete, and the same is true for $\{\vee, \neg\}$.

Exercises

Exercise 3.1. *Let ϕ and ψ be sentences. Show that*

$$\phi \leftrightarrow \psi \models\!\!\models (\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$$

Exercise 3.2. *Prove that the set $\{\wedge, \neg\}$ is functionally complete. HINT: think about proposition 3.1.9 and corollary 3.1.7.*

Exercise 3.3. *Define a binary connective | using the following truth table.*

ϕ	ψ	$\phi \mid \psi$
T	T	F
T	F	T
F	T	T
F	F	T

Prove that $\{|\}$ is functionally complete.

Exercise 3.4. *Let p and q be basic propositions. How many possible distinct truth tables are there for formulas involving only the propositions p and q? (We consider two truth tables for formulas involving the same basic propositions to be distinct if there is a truth assignment for the basic propositions such that the evaluation of each formula under this assignment is different in each table. For example, the truth tables of $p \wedge q$ and $p \vee q$ are distinct because the values of these formulas when p is true and q is false are different).*

Exercise 3.5. *A sentence ϕ is in* disjunctive normal form*, (DNF), if it is of the form $\phi_1 \vee \ldots \vee \phi_n$, where each ϕ_i is of the form $l_1 \wedge \ldots \wedge l_k$, and each l_j is either a basic proposition or the negation of a basic proposition. E.g. $(p \wedge q) \vee (\neg p) \vee (p \wedge \neg q \wedge r)$ is in DNF. Show that every sentence is equivalent to a sentence in DNF. HINT: Think about the truth table.*

3.2 Deduction rules for propositional logic

In the previous section we saw how formulas and sets of formulas can imply other formulas according to truth tables. This allows us to make deductions about when a formula must be true assuming that certain other formulas are true. This method of deduction is *semantic*, because it is based on thinking about whether the basic propositions that are components of the various formulas are true or false. In other words, the notion of truth with respect to some 'world' where the basic propositions are interpreted plays a vital role. A fundamentally different approach to logical deduction is to set aside concepts like 'true', 'false' and 'meaning' and just look at the structure of the formulas involved. This is known as *syntax*, and we will develop a syntactical approach to deduction in this section.

Formal proofs in propositional logic. A formal proof begins with a (possibly empty) set of sentences, Γ, (considered to be axioms). Alongside this set of axioms we have a collection of *deduction rules* (also called *inference rules*), which are used to generate new sentences from combinations of ones previously generated. During this process the intended meaning of the sentences are irrelevant. The only important thing is their syntactic form. The set of sentences provable from Γ is the set of sentences that can be obtained from Γ using a finite number of applications of the inference rules.

Natural deduction. There are many ways we can define deduction rules for propositional logic that are equivalent in a technical sense. We use a system

called *natural deduction*. The advantage of this system is that it is relatively easy for humans to follow, and the proofs it constructs resemble natural human reasoning. The disadvantage of the system, from a mathematical point of view, is that the flexibility in the system that allows its proofs to (roughly) follow human thought processes make the format of the proofs, in a sense, less rigid, and therefore more difficult to formally reason about. This is important to mathematical logicians because they want to be able to prove theorems about the deductive power of formal systems, and it's easier to do this when the formal proofs must follow strict patterns. We're not worried about that though.

Inference rules for natural deduction. Our sentences here will use the set of logical connectives $\{\wedge, \vee, \neg, \rightarrow\}$.

Introduction rules. **Elimination rules.**

$$\top_I: \quad \frac{}{\top}$$

$$\bot_E: \quad \frac{\bot}{\phi}$$

$$\wedge_I: \quad \frac{\phi \qquad \psi}{\phi \wedge \psi}$$

$$\wedge_{E_l}: \quad \frac{\phi \wedge \psi}{\phi}$$

$$\vee_{I_l}: \quad \frac{\phi}{\phi \vee \psi}$$

$$\wedge_{E_r}: \quad \frac{\phi \wedge \psi}{\psi}$$

$$\vee_{I_r}: \quad \frac{\psi}{\phi \vee \psi}$$

$$\vee_E: \quad \frac{\phi \vee \psi \qquad \begin{matrix}[\phi] \\ \theta\end{matrix} \qquad \begin{matrix}[\psi] \\ \theta\end{matrix}}{\theta}$$

$$\neg_I: \quad \frac{\begin{matrix}[\phi] \\ \bot\end{matrix}}{\neg\phi}$$

$$\neg_E: \quad \frac{\phi \qquad \neg\phi}{\bot}$$

$$\to_I: \quad \frac{\begin{matrix}[\phi] \\ \psi\end{matrix}}{\phi \to \psi}$$

$$\to_E: \quad \frac{\phi \to \psi \qquad \phi}{\psi}$$

These rules, whose intended meanings we will hopefully become clearer soon, define something called *intuitionistic propositional logic*. This is like classical propositional logic except that here $\neg\neg\phi$ does not imply ϕ (though the converse is still true, see example 3.2.3). To get classical propositional logic we need one extra rule (double negation elimination).

$$\neg\neg_E: \quad \frac{\neg\neg\phi}{\phi}$$

Roughly speaking, introduction rules create new sentences by combining old ones with a logical connective, and elimination rules create new sentences by eliminating logical connectives from old ones, though there are some rules that don't fit this pattern in an obvious way.

Derivations go from top to bottom. We can introduce sentences based on our axioms, then use the inference rules to derive new ones. Derived sentences go below the line. The idea is essentially that the thing above the line is what is known, and the thing below the line is something we can deduce from that. For example, the $\top_I$ rule says that we can always derive a tautology. I.e. something that is always true is always true. The rule $\bot_E$ says that from a contradiction we can derive anything. This is known as the *principle of explosion*. This is not entirely uncontroversial, but the argument for why we should accept it is similar to the argument for why the truth table of $p \to q$ is like it is.

Sentences in square brackets, e.g. $[\phi]$, are *assumptions*. When we make an assumption we have to discharge it (i.e. get rid of it) later using one of the inferences rules $\neg_I$, $\to_I$, or $\vee_E$. We often use a subscript when making an assumption, e.g. $[\phi]_1$, so we can keep track of when we discharge it. We will discharge assumptions using 'last in first out'. So, in a derivation, the last assumption made is the first to be discharged.

Double lines (e.g. in $\vee_E$) represent a subderivation. That is, it stands for some arbitrary derivation beginning with the thing on the top and ending with the thing on the bottom.

To illustrate this, think about the rule $\vee_E$. In words, this rule is intended to represent the fact if θ logically follows from either ϕ or ψ, and if we know that one or both of ϕ or ψ is true, then we also know that θ must be true. In the form of a deduction rule, this says that if we can derive θ from assumption ϕ, and if we can derive θ from assumption ψ, then θ should be a consequence of $\phi \vee \psi$.

If we are doing a complicated derivation with lots of subderivations, then we discharge assumptions only from the same subderivation. For example, suppose we're using the $\vee_E$ rule. Then nothing we do in the subderivation beginning with the assumption $[\phi]$ will ever cause us to discharge an assumption made in the subderivation beginning with the assumption $[\psi]$.

We implicitly assume we can deduce any formula from itself or an assumption of itself:

$$\frac{\phi}{\phi} \qquad\qquad \frac{[\phi]}{\phi}$$

Note that when we make deductions we can usually freely switch the order of sentences. For example, ϕ and $\neg\phi$ could be switched when applying rule $\neg_E$.

The best way to understand derivations is by looking at examples, so here are several.

Example 3.2.1. *We can deduce $\phi \to \phi$ from an empty set of axioms.*

$$\frac{\dfrac{[\phi]_1}{\phi}}{\phi \to \phi}\ (\to_I)_1$$

Example 3.2.2. *If $\phi \vee \psi$ is an axiom then we can deduce $\psi \vee \phi$.*

$$\frac{\phi \vee \psi \qquad \dfrac{\dfrac{[\phi]_1}{\phi}}{\psi \vee \phi}\ (\vee_{I_r}) \qquad \dfrac{\dfrac{[\psi]_1}{\psi}}{\psi \vee \phi}\ (\vee_{I_l})}{\psi \vee \phi}\ (\vee_E)_1$$

Example 3.2.3. *For all sentences ϕ, we can derive $\phi \to \neg\neg\phi$ from an empty set of axioms, without using the rule $\neg\neg_E$.*

$$\dfrac{\dfrac{\dfrac{[\phi]_1}{\phi}\quad\dfrac{[\neg\phi]_2}{\neg\phi}}{\dfrac{\bot}{\neg\neg\phi}\,(\neg_I)_2}\,(\neg_E)}{\phi\to\neg\neg\phi}\,(\to_I)_1$$

Example 3.2.4 (De Morgan's laws)**.**

1. *From $\phi\vee\psi$ we can deduce $\neg(\neg\phi\wedge\neg\psi)$*

$$\dfrac{(\phi\vee\psi)\qquad\dfrac{\dfrac{\dfrac{[\phi]_1}{\phi}\quad\dfrac{[\neg\phi\wedge\neg\psi]_2}{\neg\phi}\,(\wedge_{E_l})}{\bot}\,(\neg_E)}{\neg(\neg\phi\wedge\neg\psi)}\,(\neg_I)_2\qquad\dfrac{\dfrac{\dfrac{[\psi]_1}{\psi}\quad\dfrac{[\neg\phi\wedge\neg\psi]_3}{\neg\psi}\,(\wedge_{E_r})}{\bot}\,(\neg_E)}{\neg(\neg\phi\wedge\neg\psi)}\,(\neg_I)_3}{\neg(\neg\phi\wedge\neg\psi)}\,(\vee_E)_1$$

2. *From $\neg(\neg\phi\wedge\neg\psi)$ we can deduce $\phi\vee\psi$.*

$$\dfrac{\dfrac{\neg(\neg\phi\wedge\neg\psi)\qquad\dfrac{\dfrac{\dfrac{\dfrac{[\phi]_2}{\phi\vee\psi}\,(\vee_{I_l})\quad\dfrac{[\neg(\phi\vee\psi)]_1}{\neg(\phi\vee\psi)}}{\bot}\,(\neg_E)}{\neg\phi}\,(\neg_I)_2\qquad\dfrac{[\psi]_3\qquad[\neg(\phi\vee\psi)]_1}{\neg\psi}}{\neg\phi\wedge\neg\psi}\,(\wedge_I)}{\bot}\,(\neg_E)}{\neg\neg(\phi\vee\psi)}\,(\neg_I)_1}{\phi\vee\psi}\,(\neg\neg_E)$$

Note that in this example we need the extra rule $\neg\neg_E$. The result is not true in intuitionistic propositional logic.

Example 3.2.5. *$\phi\vee\neg\phi$ is a theorem of classical propositional logic (i.e. it can be deduced from an empty set of axioms).*

$$\dfrac{\neg(\neg\phi\vee\phi)\qquad\dfrac{\dfrac{\dfrac{\dfrac{[\neg(\neg\phi\vee\phi)]_1}{\neg(\neg\phi\vee\phi)}\quad\dfrac{\dfrac{[\phi]_2}{\phi}}{\neg\phi\vee\phi}\,(\vee_{I_r})}{\bot}\,(\neg_E)}{\neg\phi}\,(\neg_I)_2}{\neg\phi\vee\phi}\,(\vee_{I_l})}{\bot}\,(\neg_E)}{\neg\neg(\neg\phi\vee\phi)}\,(\neg_I)_1}{\neg\phi\vee\phi}\,(\neg\neg_E)$$

Note that this example also requires $\neg\neg_E$.

Example 3.2.6. *If $\phi \vee \psi$ and $\neg\phi$ are axioms then we can deduce ψ.*

$$\dfrac{\phi \vee \psi \qquad \dfrac{\dfrac{\neg\phi \qquad \dfrac{[\phi]_1}{\phi}}{\dfrac{\bot}{\psi}\,(\bot_E)}\,(\neg_E) \qquad \dfrac{[\psi]_1}{\psi}}{\psi}\,(\vee_E)_1}$$

Exercises

Exercise 3.6. *The following deduction tree proves that $\phi \to \psi$ can be deduced from $\neg\phi \vee \psi$ in intuitionistic propositional logic. Add labels indicating the rules used at each stage.*

$$\dfrac{\neg\phi \vee \psi \qquad \dfrac{\dfrac{[\neg\phi]_1 \qquad \dfrac{[\phi]_2}{\bot}}{\psi}}{\phi \to \psi} \qquad \dfrac{\dfrac{[\psi]_1}{\psi} \qquad \dfrac{[\phi]_3}{\phi \to \psi}}{\phi \to \psi}}{\phi \to \psi}$$

Exercise 3.7. *What is being proved in the following deduction tree? Add labels indicating the rules at each stage.*

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\phi \to \psi \qquad [\phi]_2}{\psi}}{\neg\phi \vee \psi}}{\dfrac{\bot}{\neg\phi}}}{\neg\phi \vee \psi} \qquad \dfrac{[\neg(\neg\phi \vee \psi)]_1}{\neg(\neg\phi \vee \psi)}}{\bot}}{\neg\neg(\neg\phi \vee \psi)}}{\neg\phi \vee \psi}$$

Exercise 3.8. *Show that $(\phi \wedge \psi) \to (\psi \wedge \phi)$ can be deduced from an empty set of axioms.*

Exercise 3.9. *Show that we can deduce $\phi \wedge (\psi \vee \chi)$ if we start with $(\phi \wedge \psi) \vee (\phi \wedge \chi)$.*

Exercise 3.10. *Show that we can deduce $(\phi \wedge \psi) \vee (\phi \wedge \chi)$ if we start with $\phi \wedge (\psi \vee \chi)$.*

3.3 Soundness, completeness and compactness

In section 1 we introduced semantics for propositional logic in the form of truth tables. We wrote $\phi \models \psi$ when ψ is a logical consequence of ϕ, and we determined this just by constructing a truth table and comparing the appropriate columns. In section 2 we defined a formal deduction system (natural deduction). We write $\phi \vdash \psi$ if ψ follows from ϕ by application of the inference rules we defined.

Logical consequence and formal deduction are both supposed to capture the idea of statements being implied by others Intuitively, both systems seem to do this, though they do it in different ways. We want the two systems to be equivalent, in a sense we define below. Note that we interpret $\Gamma \models \bot$ to mean that there is no assignment satisfying Γ.

Definition 3.3.1 (Sound)**.** *A formal deduction system for propositional logic is* sound *if whenever $\Gamma \vdash \phi$, we also have $\Gamma \models \phi$. Intuitively this means that if we make a formal deduction then we know the truth table will show the same result. We sometimes use the slogan "provable implies true".*

Definition 3.3.2 (Complete)**.** *A formal deduction system for propositional logic is* complete *if whenever $\Gamma \models \phi$, we also have $\Gamma \vdash \phi$. Intuitively this means that if we can show a logical implication with a truth table, then we will be able to construct the corresponding proof using our deduction system. We sometimes use the slogan "true implies provable".*

We want our deduction system to be both sound and complete. In other words, we want to be able to say, if we can prove it using deduction rules, then it's true according to truth tables, and if it's true according to truth tables, then we can prove it. A sound and complete deduction system for propositional logic matches up perfectly with the intuitively simple truth tables.

Why have deduction systems? Truth tables are simple to create and check, but formal proofs using deduction rules can be very difficult to find. Since we are mainly interested in deduction systems that are sound and complete, why even bother to define a separate system for formal proofs? Why not just use truth tables? There are two main answers to this. First, occasionally formal deduction rules give an easier way of proving something than setting up a truth table. For example, it follows easily from the natural deduction system that $\phi \to (\phi \to (\phi \to (\phi \to \phi)))$, but proving this by truth table would be quite tedious. More importantly, the difference between syntax and semantics will become much more significant when we start using more powerful logical systems.

In particular, in first-order logic (which we cover in sections 4 and 5), logical consequence is extremely difficult, if not impossible, to test directly, so a sound and complete deduction system becomes extremely important. With this in mind, we could consider learning about deduction systems for propositional logic to be training for the serious work ahead.

Soundness of natural deduction.

Theorem 3.3.3. *The natural deduction system for propositional logic is sound.*

Proof. We need to prove that whenever $\Gamma \vdash \chi$, we also have $\Gamma \models \chi$. I.e. if we can deduce χ from set of assumptions Γ, then whenever an assignment satisfies every sentence in Γ it must also satisfy χ. We use induction on the number of steps in the derivation of χ from Γ. By 'number of steps' we mean, 'number of uses of deduction rules'.

The base case is simple. If the length of the derivation is 1 then there is only a single statement in the proof tree, which must be χ. This is only a valid step in a proof if $\chi \in \Gamma$. In this case any assignment satisfying everything in Γ will obviously satisfy χ.

For the inductive step we assume that the result is true for all derivations with length less than or equal to n, say, and we suppose the length of our derivation is $n+1$. We show that the last move is sound, for all possible choices of last move. We do this systematically by checking each case.

$\top_I$: In this case $\chi = \top$. This case is trivial because every assignment satisfies $\top$, so of course any assignment satisfying Γ must also satisfy χ.

$\bot_E$: This is a subtle case. The last step in the derivation is deriving χ from $\bot$, where $\bot$ has first been derived from Γ. Since we are assuming the derivation of $\bot$ is sound, this means that $\Gamma \models \bot$. In other words, there is no assignment satisfying Γ. So it is vacuously true that $\Gamma \models \chi$, because there are no assignments satisfying Γ to worry about.

$\wedge_I$: Here we deduce $\chi = \phi \wedge \psi$ from ϕ and ψ, with $\Gamma \vdash \phi$ and $\Gamma \vdash \psi$. By the inductive hypothesis, the derivations of ϕ and ψ are both sound, therefore, any assignment that satisfies Γ will satisfy both ϕ and ψ. But then the truth table says it will also satisfy $\phi \wedge \psi$, which is what we want.

$\wedge_{E_l}$: Here $\chi = \phi$, which we deduce from $\phi \wedge \psi$. Again, by the inductive hypothesis we assume the derivation of $\phi \wedge \psi$ from Γ is sound, which means that any assignment that satisfies Γ also satisfies $\phi \wedge \psi$. But then it must also satisfy ϕ, which is what we want.

$\neg_I$: Here $\chi = \neg\phi$, and we assume ϕ and derive a contradiction from $\Gamma \cup \{\phi\}$. By the inductive hypothesis we assume that this derivation is sound, so there is no assignment satisfying $\Gamma \cup \phi$. In other words, any assignment satisfies Γ must also satisfy $\neg\phi$. But this means that the derivation of $\neg\phi$ from Γ is sound, which is what we want.

$\rightarrow_I$: Here $\chi = \phi \rightarrow \psi$, and we derive ψ from $\Gamma \cup \{\phi\}$. Again, by the inductive hypothesis, this derivation is sound, so any assignment that satisfies $\Gamma \cup \{\phi\}$ also satisfies ψ. Suppose an assignment satisfies Γ. If this assignment satisfies ϕ, then we have have just shown it satisfies ψ too, so by the truth table also satisfies $\phi \rightarrow \psi$. Alternatively, if it does not satisfy ϕ, then, again by the truth table, it also satisfies $\phi \rightarrow \psi$. So any assignment satisfying Γ also satisfies $\phi \rightarrow \psi$, which is what we are trying to prove.

$\neg_E$: Here $\chi = \bot$, so in other words we have derived a contradiction from Γ. To apply this rule in a derivation we must first have derived ϕ and $\neg\phi$ from Γ. By the inductive hypothesis, these derivations are sound, so any assignment that satisfies Γ must also satisfy ϕ and $\neg\phi$. This is impossible, so there cannot be an assignment satisfying Γ, and, by definition, this means $\Gamma \models \bot$, which is what we want to prove.

The remaining possibilities are exercise 3.11. □

Completeness of natural deduction.

Theorem 3.3.4. *The natural deduction system for propositional logic is complete.*

To prove this we will need some preliminary results. First note that this result is only true if we include the $\neg\neg_E$ deduction rule, otherwise we cannot formally prove that $\neg\neg\phi \vdash \phi$, or that $\vdash \phi \vee \neg\phi$, and both $\neg\neg\phi \models \phi$ and $\models \phi \vee \neg\phi$ are obviously true.

Lemma 3.3.5. *Let Γ be a set of sentences, then:*

1. $\Gamma \models \neg\phi \iff \Gamma \cup \{\phi\} \models \bot$, *and*
2. $\Gamma \vdash \neg\phi \iff \Gamma \cup \{\phi\} \vdash \bot$.

Proof. If $\Gamma \models \neg\phi$ then every assignment that satisfies Γ must satisfy $\neg\phi$. So there can be no assignment that satisfies $\Gamma \cup \{\phi\}$ (i.e. $\Gamma \cup \{\phi\} \models \bot$). Conversely, if there is no assignment that satisfies $\Gamma \cup \{\phi\}$ then every assignment that satisfies Γ must satisfy $\neg\phi$ (i.e. $\Gamma \models \neg\phi$).

For part 2, suppose first that $\Gamma \vdash \neg\phi$. Then we can derive $\bot$ from $\Gamma \cup \{\phi\}$ using rule $\neg_E$. Conversely, suppose $\Gamma \cup \{\phi\} \vdash \bot$. Then, starting with Γ, we can apply rule $\neg_I$ with assumption ϕ to derive $\neg\phi$ (we copy the derivation of ϕ from Γ), and thus $\Gamma \vdash \neg\phi$. □

Definition 3.3.6 (Consistent)**.** *A set of sentences Γ is* consistent *if $\Gamma \nvdash \bot$. I.e. if we cannot deduce a contradiction from it.*

Lemma 3.3.7. *Completeness of the natural deduction system with $\neg\neg_E$ with is equivalent to the statement:*

Every consistent set of sentences is satisfiable. (†)

Proof. Completeness can be stated as $\Gamma \models \phi \implies \Gamma \vdash \phi$ for all sets of sentences Γ, and (†) translates as $\Gamma \models \bot \implies \Gamma \vdash \bot$ for all sets of sentences Γ.

Now, assuming (†), and using lemma 3.3.5(1), we have

$$\begin{aligned}
\Gamma \models \phi &\iff \Gamma \models \neg\neg\phi \\
&\iff \Gamma \cup \{\neg\phi\} \models \bot \\
&\implies \Gamma \cup \{\neg\phi\} \vdash \bot \\
&\iff \Gamma \vdash \neg\neg\phi \\
&\iff \Gamma \vdash \phi
\end{aligned}$$

So $\Gamma \models \phi \implies \Gamma \vdash \phi$, which is the statement of completeness.

Conversely, assume completeness, and suppose $\Gamma \models \bot$. Then Γ cannot be empty, so let $\phi \in \Gamma$. Then, using lemma 3.3.5(2), we have

$$\begin{aligned}
\Gamma \models \bot &\iff \Gamma \setminus \{\phi\} \cup \{\phi\} \models \bot \\
&\iff \Gamma \setminus \{\phi\} \models \neg\phi \\
&\implies \Gamma \setminus \{\phi\} \vdash \neg\phi \\
&\iff \Gamma \setminus \{\phi\} \cup \{\phi\} \vdash \bot \\
&\iff \Gamma \vdash \bot
\end{aligned}$$

So $\Gamma \vdash \bot \implies \Gamma \models \bot$, which is (†). □

Definition 3.3.8 (maximal consistent)**.** *A consistent set of sentences Γ is maximal consistent if for every sentence ϕ, either $\phi \in \Gamma$ or $\neg\phi \in \Gamma$.*

Note that if Γ is maximal consistent, then a sentence is deducible from Γ if and only if it is actually in Γ. I.e. for all sentences ϕ we have $\Gamma \vdash \phi \iff \phi \in \Gamma$.

Lemma 3.3.9. *For every consistent Γ there is a maximal consistent Γ' with $\Gamma \subseteq \Gamma'$.*

Proof. Let $\phi_0, \phi_1, \phi_2, \ldots$ be an enumeration of all the sentences. It may not be obvious that we can arrange all the sentences in a list like this, but we will prove that it can be done as part of the Counting course, in the class on cardinal numbers. Now we use recursion to define sets Γ_n for $n \in \mathbb{N}$ as follows.

- $\Gamma_0 = \Gamma$.
- $\Gamma_{n+1} = \Gamma_n \cup \{\phi_n\}$, if this is consistent, and $\Gamma_n \cup \{\neg\phi_n\}$ otherwise.

Note that Γ_n is consistent for all n, because Γ_0 is consistent by definition, and, by lemma 3.3.5, if $\Gamma_n \cup \{\phi_n\}$ is not consistent then $\Gamma_n \vdash \neg\phi_n$, and so consistency of $\Gamma_n \cup \{\neg\phi_n\}$ follows from consistency of Γ_n.

We define $\Gamma' = \bigcup_{n\in\mathbb{N}} \Gamma_n$. Then Γ' is consistent, because if $\Gamma' \vdash \bot$, then, as every derivation involves a finite proof tree, there must be $n \in \mathbb{N}$ such that every sentence used in the derivation of $\bot$ from Γ' appears in Γ_n. But then $\Gamma_n \vdash \bot$, which is a contradiction as Γ_n is consistent.

Since Γ' is clearly maximal consistent we are done. □

Proof of theorem 3.3.4. Let Γ be a consistent set of sentences. By lemma 3.3.7, to complete the proof we need to show that Γ is satisfiable, i.e. that there is an assignment that makes every sentence in Γ true. We can suppose without loss of generality that Γ is maximal consistent (using lemma 3.3.9 we know Γ can be extended to a maximal consistent Γ', and an assignment that makes every sentence in Γ' true must make every sentence in Γ true). We construct an assignment v as follows. For all basic propositions p, let $v(p)$ be true if $p \in \Gamma$,

and let $v(p)$ be false otherwise. Then v is well defined, by maximality of Γ. We want to show that, for all sentences ϕ, if $\phi \in \Gamma$ then $v(\phi)$ is true.

We proceed by induction on sentence length, and we assume first that ϕ is constructed using only the connectives $\neg$ and $\vee$. We will show that for such sentences we have $\phi \in \Gamma \iff v(\phi)$ is true. We will use this to prove the result for general sentences. In the base case ϕ is just a basic proposition, so the result holds by definition of v. The inductive step has two cases.

$\neg$: Let $\phi = \neg\psi$. Then $v(\phi)$ is true $\iff$ $v(\psi)$ is false $\iff$ $\psi \notin \Gamma \iff \phi \in \Gamma$.

$\vee$: Let $\phi = \psi \vee \chi$. Then $v(\phi)$ is true $\iff$ ($v(\psi)$ is true and/or $v(\chi)$ is true) $\iff$ ($\psi \in \Gamma$ and/or $\chi \in \Gamma$) $\iff$ $\phi \in \Gamma$.

In the last step in the proof for $\vee$ we are implicitly using the fact that $\psi \in \Gamma$ or $\chi \in \Gamma$ if and only if $\psi \vee \chi \in \Gamma$. To see that this is indeed true note first that if $\Gamma \vdash \psi$ then $\Gamma \vdash \psi \vee \chi$, by the deduction rule $\vee_{I_l}$, and similarly $\Gamma \vdash \psi \implies \Gamma \vdash \psi \vee \chi$. Conversely, if $\psi \vee \chi \in \Gamma$, then, by example 3.2.6, if $\psi \notin \Gamma$ then $\Gamma \vdash \chi$, and similarly if $\chi \notin \Gamma$ then $\Gamma \vdash \psi$.

To complete the proof, let $\phi \in \Gamma$ be constructed using the full set of connectives, and let ϕ' be a sentence using only connectives $\neg$ and $\vee$ such that $\phi \models \dashv\vdash \phi'$ (such a ϕ' exists because $\{\neg, \vee\}$ is functionally complete). Suppose $v(\phi)$ is false. Then $v(\phi')$ is also false. So, by the induction we've just done, we have $\phi' \notin \Gamma$. But then by maximality of Γ we have $\neg\phi' \in \Gamma$. So $\Gamma \vdash \phi \wedge \neg\phi'$, and so $\Gamma \models \phi \wedge \neg\phi'$ by soundness. But this is a contradiction, as $\phi \models \dashv\vdash \phi'$ by choice of ϕ'. Therefore $v(\phi)$ is true, and so v is an assignment satisfying Γ as required.

Compactness. There's another important fundamental result for propositional logic known as the compactness theorem. You will see a precise statement of this in the exercises. Compactness type results occur frequently in mathematics (many of them are even proved as applications of the compactness theorems for propositional or first-order logic). Roughly speaking, the theme of these results is translating statements about infinite structures into statements about finite ones. This is very useful, because it allows us to use our understanding of finite structures to understand infinite ones. When we have a compactness result, we can investigate something infinite by decomposing it into finite pieces in some way. Induction over $\mathbb{N}$ is a bit like this. If we want to prove something for all natural numbers, we don't have to deal with them all at the same time. Using induction, we can get the general result by looking at numbers 'one at a time'. So we can prove results about the infinite set of natural numbers while only ever directly working with finite sets of numbers. Compactness in logic applies this concept to proofs and satisfiability.

Exercises

Exercise 3.11. *Complete the proof of theorem 3.3.3 (don't forget the extra axiom, $\neg\neg_E$).*

Exercise 3.12. *Prove that soundness of a deduction system is equivalent to the statement "every satisfiable set of sentences is consistent". HINT: think about how the proof of lemma 3.3.7 works.*

Exercise 3.13 (Compactness theorem for propositional logic)**.** *Use soundness and completeness to prove the following:*

Theorem. *Let Γ be a set of sentences in propositional logic. Then Γ is satisfiable if and only if every finite subset of Γ is satisfiable.*

3.4 First-order logic

Propositional logic describes how propositions can be combined together to form new ones, and how we can understand the truth values of these composite statements by understanding the truth values of the basic propositions that they are constructed from. We also have a closely related purely syntactic notion of logical deduction that allows us to understand propositions as necessary consequences of other propositions (or sets of propositions). This is good as far as it goes, but it is also quite limited. The main problem is that for us to say anything interesting in propositional logic, the basic propositions have to be pre-defined. The notions of tautology and contradiction allow us to understand how some statements must be true or false based purely on their logical forms, but this is of limited use if we want to describe the state of a complex system.

This brings us to first-order logic, which extends the power of propositional logic by adding the means to create and interpret complex propositions. More explicitly, there are still propositions in first-order logic, and these can be combined and analyzed as they are propositional logic, but rather than relying on basic propositions whose meanings are essentially abstracted away, propositions in first-order logic are statements that can be meaningfully interpreted in appropriate structures. How this works will became clearer soon, but first we review the concept of a relation.

Functions and relations. Intuitively, we understand a function as a rule mapping each element of a set A to an element of another set B (sometimes $B = A$). Similarly, we can think of a relation between two sets A and B as a rule which matches elements of A and B in pairs. For example, "has visited" is a relation between the set of people and the set of cities. We want to make our intuitions a bit more mathematical if we want to reason about functions and relations formally, so we need some precise technical definitions for these familiar concepts.

Definition 3.4.1 (Relation)**.** *An n-ary relation between sets $X_1, \dots X_n$ is a subset of $\prod_{i=1}^{n} X_i$. Given such a relation r, and an n-tuple $(x_1, \dots, x_n) \in \prod_{i=1}^{n} X_i$, we say $r(x_1, \dots, x_n)$ holds if and only if $(x_1, \dots, x_n) \in r$.*

Example 3.4.2.

1. *The order relation $\leq$ is a binary relation on $\mathbb{N}^2$.*

2. *If X is a set, and $Y \subseteq X$, then we can define a unary relation, r_Y, on X by $r_Y(x) \iff x \in Y$.*

3. *We can define a relation, p, on $\mathbb{N}^3$ by $p(x, y, z) \iff x^2 + y^2 = z^2$. This is a 3-ary (ternary) relation.*

4. *We can define a ternary relation, q, on $\mathbb{N} \times \mathbb{N} \times \mathbb{Q}$ by $q(x, y, z) \iff z = \frac{x}{y}$.*

Definition 3.4.3 (Function)**.** *An n-ary function is a well-defined map, f, from $\prod_{i=1}^{n} X_i$ to Y for some sets X_i ($i \in \{1, \ldots n\}$) and Y. In this context, well-defined means that, for every $(x_1, \ldots, x_n) \in \prod_{i=1}^{n} X_i$, the value of $f(x_1, \ldots, x_n)$ exists and is unique.*

We can think of functions as special kinds of relations. I.e. an n-ary function, $f : \prod_{i=1}^{n} X_i \to Y$ is equivalent to an $(n+1)$-ary relation, r_f, on $(\prod_{i=1}^{n} X_i) \times Y$, with the well-definedness property being that, given $(x_1, \ldots, x_n) \in \prod_{i=1}^{n} X_i$, there is always a unique $y \in Y$ such that $r_f(x_1, \ldots, x_n, y)$ holds. The converse is also true, in that any $(n+1)$-ary relation with the well-definedness property can be thought of as an n-ary function.

Example 3.4.4.

1. *Every polynomial $a_0 + a_1 x + \ldots + a_n x^n$ defines a unary function from $\mathbb{N}$ to $\mathbb{N}$ (and from $\mathbb{R}$ to $\mathbb{R}$, or from $\mathbb{N}$ to $\mathbb{R}$ etc.)*

2. *Division can be thought of as a binary function d from $\mathbb{Q} \times (\mathbb{Q} \setminus \{0\})$ to $\mathbb{Q}$ by defining $d(x, y) = \frac{x}{y}$. We can't define a function $\mathbb{Q} \times \mathbb{Q}$ to $\mathbb{Q}$ like this though, as division by zero is not defined.*

First-order languages. Now we have precise definitions for functions and relations, we can begin to construct a formal system powerful enough to handle large parts of mathematics, and consequently, powerful enough to reason about things we can formalize mathematically, such as programming languages, air traffic control systems etc. This will involve some careful setting up, as this is all unavoidably technical.

Definition 3.4.5 ($\mathscr{L}$)**.** *A language, $\mathscr{L}$, for first-order logic consists of the following things.*

1. *Logical symbols.*

   (a) *An infinite set of variables enumerated by natural numbers,*

   $$V = \{x_0, x_1, \ldots\}.$$

   (b) *The equality symbol, $\approx$.*

   (c) *The set of logical connectives, $\{\neg, \vee, \wedge, \to\}$.*

   (d) *The set of quantifier symbols, $\{\forall, \exists\}$.*

(e) *A set of brackets,* $\{(,)\}$.

2. *Non-logical symbols.*

   (a) *A countable (possibly empty) set,* $\mathcal{R}$*, of* predicate *symbols.*

   - *Every predicate symbol has an associated* arity*. Formally, we think of this as a map from* $\mathcal{R}$ *to* $\mathbb{N}$.

   (b) *A countable (possibly empty) set,* $\mathcal{F}$*, of* function *symbols.*

   - *Every function symbol also has an associated* arity*. Formally, we think of this as a map from* $\mathcal{F}$ *to* $\mathbb{N}$.

   (c) *A countable (possibly empty) set,* $\mathcal{C}$*, of* constant *symbols.*

   - *We can think of a constant as a* 0*-ary (nullary) function. I.e. a* 0*-ary function corresponds to a unary relation with the well-definedness property. But this is just a single element.*

The non-logical symbols of $\mathscr{L}$ are known as the *signature* of $\mathscr{L}$. Every language we are interested in will have the same logical symbols, so for us, what distinguishes first-order languages from each other are their signatures. We will specify languages just by giving their signatures, and say things like "Given a first-order signature $\mathscr{L}$". There may be more than one choice of signature that seems appropriate for studying a given mathematical structure or object. The choice of signature is in some ways arbitrary, but it will affect the things that can be done with the language in significant ways, particularly when it comes to the highly formal arguments that studying logical systems often involves.

Note that the choice of logical symbols is also somewhat arbitrary, though less potentially significant. As in propositional logic, we could use a smaller set of logical connectives, and it turns out that only one quantifier is enough, but we use the larger set as it is more intuitive for humans.

Example 3.4.6. *Suppose we want to use first-order logic to talk about arithmetic with natural numbers. What non-logical symbols might we need? We probably want binary functions* $+$ *and* $\times$*, for example, which we hope to give their usual interpretations. We might also want to specify the numbers 0 and 1 using constant symbols. This would give us a language with two binary functions and two constants.*

It could potentially be convenient to have a special unary predicate to tell us when a number is prime, say, so we can add a unary predicate symbol to our collection of non-logical symbols if we like. As discussed above, we are free to choose our signature however we want, but our choice may have consequences later.

First-order formulas Formulas in first-order logic are defined recursively. Unlike in propositional logic, the variables themselves are not supposed to be propositions. I.e. it doesn't make sense for a variable in first-order logic to be true or false.

Definition 3.4.7 (Term)**.** *The set of* terms *of $\mathscr{L}$ is defined recursively.*

- *Every variable x is an $\mathscr{L}$-term.*
- *Every constant c is an $\mathscr{L}$-term.*
- *If f is an n-ary function symbol occurring in $\mathscr{L}$ and $t_1, \ldots, t_n$ are $\mathscr{L}$-terms then $f(t_1, \ldots, t_n)$ is also an $\mathscr{L}$-term.*

Terms, like variables, are not propositions. It does not make sense for them to be true or false.

Definition 3.4.8 (Atomic formula)**.** *The set of* atomic formulas *of $\mathscr{L}$ is defined as follows:*

- *If t_1 and t_2 are $\mathscr{L}$-terms, then $t_1 \approx t_2$ is an atomic $\mathscr{L}$-formula.*
- *If R is an n-ary relation of $\mathscr{L}$, and $t_1, \ldots, t_n$ are $\mathscr{L}$-terms, then $R(t_1, \ldots, t_n)$ is an atomic $\mathscr{L}$-formula.*

Atomic formulas are the simplest formulas that are meant to correspond to propositions. That is, once we have defined our semantics, it will make sense for these to be true or false.

Definition 3.4.9 (Formula)**.** *The set of* formulas *of $\mathscr{L}$ ($\mathscr{L}$-formulas) is defined recursively.*

- *Every atomic $\mathscr{L}$-formula is an $\mathscr{L}$-formula.*
- *If ϕ is an $\mathscr{L}$-formula, then $\neg\phi$ is an $\mathscr{L}$-formula.*
- *If ϕ and ψ are $\mathscr{L}$-formulas, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$ and $(\phi \to \psi)$ are $\mathscr{L}$-formulas.*
- *If ϕ is an $\mathscr{L}$-formula and x is a variable symbol, then $\forall x\phi$ and $\exists x\phi$ are $\mathscr{L}$-formulas.* [2]

As in propositional logic, we are sometimes loose with our use of brackets, adding them or removing them when the result makes the formulas easier for humans to read.

Example 3.4.10. *Let $\mathscr{L}$ have signature $\mathcal{R} = \{R, S\}$, where R is unary and S is binary, $\mathcal{F} = \{f\}$, where f is ternary, and $\mathcal{C} = \{c, d\}$. Let x, y, z be variables.*

[2] This allows formulas like $\exists y \forall y R(y)$. Here the $\exists$ quantifier doesn't do anything, as the only y in $\forall y R(y)$ is already covered by the quantifier $\forall$. In cases like this we say $\exists y$ is a *null quantifier*. The quantifier $\forall$ in $\forall x R(y)$ is also null, because in this case there is no x variable at all. Being able to reuse variables like this can be useful, as logics with only finitely many variable symbols are very important in applications of logic to classes of finite structures (such as in computer science). Logic with finite variable symbols and its applications is not part of this course, but it's something worth knowing about. Since we assume an infinite number of variable symbols in our languages, it's always possible for us to rewrite a formula into an equivalent one (we'll define what *equivalent* here means later!) with no null quantifiers and no reused variables.

1. $f(x, y, f(z, c, d)) \approx c$ *is an atomic* $\mathscr{L}$*-formula.*

2. $\exists z(R(f(x, z, d))) \vee S(f(x, y, x), d)$ *is an* $\mathscr{L}$*-formula.*

3. $f(x, y, z) \wedge c$ *is not an* $\mathscr{L}$*-formula.*

Definition 3.4.11 (Subformula)**.** *If* ϕ *is an* $\mathscr{L}$*-formula, then a* subformula *of* ϕ *is a substring of* ϕ *that is also an* $\mathscr{L}$*-formula.*

Models for first-order languages. As mentioned previously, the basic variables of a first-order language do not correspond to propositions in the sense of propositional logic. In order to give first-order formulas meaning we must interpret them in a structure.

Definition 3.4.12 ($\mathscr{L}$-structure)**.** *Given a first-order signature,* $\mathscr{L}$*, an* $\mathscr{L}$-structure *is a set* X*, plus some additional information giving concrete meaning to the symbols in* $\mathcal{R} \cup \mathcal{F} \cup \mathcal{C}$ *as follows:*

1. *Every* n*-ary relation symbol from* $\mathcal{R}$ *is assigned to an* n*-ary relation on* X^n*.*

2. *Every* n*-ary function symbol from* $\mathcal{F}$ *is assigned to an* n*-ary function from* X^n *to* X*.*

3. *Every constant symbol from* $\mathcal{C}$ *is assigned to a specific element of* X*.*

We will consider a structure to be a pair (X, I)*, where* X *is the underlying set, and* I *is the function that interprets the non-logical symbols of* $\mathscr{L}$ *as relations, functions and constants over* X*. Given, for example, a relation symbol* R*, we will sometimes write* R_I *for the concrete relation corresponding to* R*.*

Definition 3.4.13 (Assignment)**.** *An* assignment *of a first-order signature* $\mathscr{L}$ *to an* $\mathscr{L}$*-structure,* $A = (X, I)$*, is a function* $v : V \to X$*. In other words, an assignment associates every variable with an element of* X*.*

$\mathscr{L}$-formulas are capable of being true or false in $\mathscr{L}$-structures, but we must assign meaning to the terms first. We do this formally in a moment, but first we should understand that the intuition behind this is actually very simple. An $\mathscr{L}$-structure is just a set which we equip with relations, functions and constants corresponding to the symbols from $\mathscr{L}$. An assignment just gives a meaning to the variables of $\mathscr{L}$ as elements of the set.

Once we have assigned meaning to the functions and constants of $\mathscr{L}$, and also to the variables, we also assign meaning to the terms in a natural way, because the terms are just combinations of variables, constants and functions. Since the terms have a meaning, it makes sense for formulas to be true or false, because formulas just assert things like "this relation holds between these elements". This now has a natural meaning, because we can check if the interpretation of this relation holds for the interpretations of the elements.

Example 3.4.14. *Let $\mathscr{L}$ have non-logical symbols $\{\leq, 0\}$, where $\leq$ is a binary relation, and 0 is a constant. We can take $\mathbb{N}$ as a $\mathscr{L}$-structure by giving these symbols their usual meanings.*

1. *Let ϕ be the formula $x \leq y$. Then this is true if our assignment v maps x to 1 and y to 5, for example, but false if v takes x to 465 and y to 7.*
2. *Let ψ be the formula $\forall x(0 \leq x)$. Then ϕ is true whatever v we choose. In fact, although we haven't formally defined this yet, we intuitively see that the truth of this formula shouldn't depend on the assignment v at all, because the only variable is in the scope of a quantifier. I.e. the formula is either true* for all *possible values of x, or not at all.*
3. *Let χ be the formula $\exists x(x \leq y \wedge \neg(x \approx 0))$. Then χ will be true so long as $v(y) \neq 0$.*

Definition 3.4.15 (v^+)**.** *Let $\mathbf{term}(\mathscr{L})$ be the set of terms of $\mathscr{L}$, and let v be an assignment for $\mathscr{L}$ to (X, I). Then define $v^+ : \mathbf{term}(\mathscr{L}) \to X$ recursively as follows:*

- *If x is a variable then $v^+(x) = v(x)$.*
- *If c is a constant then $v^+(c) = c_I$.*
- *If f is an n-ary function, and $t_1, \ldots, t_n$ are terms such that $v^+(t_i)$ has been defined for all $i \in \{1, \ldots, n\}$, then $v^+(f(t_1, \ldots, t_n)) = f_I(v^+(t_1), \ldots, v^+(t_n))$.*

Definition 3.4.16 (Models)**.** *Let $\mathscr{L}$ be a first-order signature, let $A = (X, I)$ be a structure for $\mathscr{L}$, and let v be an assignment of $\mathscr{L}$ to A. Let ϕ be a formula of $\mathscr{L}$. We write $A, v \models \phi$ when A and v provide a model for ϕ, and we define what this means recursively.*

- *Atomic formulas:*
  - $A, v \models t_1 \approx t_2 \iff v^+(t_1) = v^+(t_2)$.
  - $A, v \models R(t_1, \ldots, t_n) \iff R_I(v^+(t_1), \ldots, v^+(t_n))$ *holds.*
- *Suppose ϕ and ψ are formulas such that whether A, u models ϕ and ψ has already been determined, for all assignments $u : V \to X$. Then:*
  - $A, v \models \neg\phi \iff A, v \not\models \phi$.
  - $A, v \models \phi \vee \psi \iff A, v \models \phi$ *or* $A, v \models \psi$.
  - $A, v \models \phi \wedge \psi \iff A, v \models \phi$ *and* $A, v \models \psi$.
  - $A, v \models \phi \to \psi \iff A, v \models \neg\phi$ *or* $A, v \models \psi$.
  - $A, v \models \forall x\phi \iff$ *whenever u is an assignment of $\mathscr{L}$ to A that agrees with v on every variable except, possibly, x, we have $A, u \models \phi$.*
  - $A, v \models \exists x\phi \iff$ *there is an assignment, u, of $\mathscr{L}$ to A that agrees with v on every variable except, possibly, x, and $A, u \models \phi$.*

Free and bound variables. If ϕ is an $\mathscr{L}$-formula, and x is a variable, then we say an occurrence of x is *free* in ϕ if there is no subformula of ϕ containing this occurrence of x that has the form $\forall x\phi'$ or $\exists x\phi'$. If there is a free occurrence of x in ϕ then we say that x is a *free variable* of ϕ. If an occurrence of x is not free in ϕ then we say it is *bound*, and that x occurs *bound* in ϕ. A bound occurrence of a variable is said to be *in the scope of* the corresponding quantifier.

Example 3.4.17. *Let $\mathscr{L}$ have signature $\mathcal{R} = \{R, S\}$, where R is unary and S is binary, $\mathcal{F} = \{f\}$, where f is ternary, and $\mathcal{C} = \{c, d\}$. Let x, y, z be variables.*

1. *$f(x, y, f(z, c, d)) \approx c$ has no bound variables.*
2. *z occurs only bound in $(\exists z(R(f(x, z, d))) \vee S(f(x, y, x), d)$, and x and y occur only free.*
3. *All variables in $\forall x(R(x) \vee S(x, c)) \wedge \exists x(R(f(x, x, x)))$ are bound.*
4. *In $\exists x R(x) \wedge S(x, y)$ the variable x occurs both free and bound. Note that x is still a free variable of this formula, even though it also occurs bound. The variable y occurs only free.*

Definition 3.4.18 (Sentence)**.** *A sentence of $\mathscr{L}$ (an $\mathscr{L}$-sentence) is an $\mathscr{L}$-formula that contains no free variables.*

By exercise 4.4, if a sentence is true for some assignment into a model, then it is true for every assignment into the same model. So in this case we can suppress v and just write, e.g. $A \models \phi$.

Exercises

Exercise 3.14. *Let x, y, z be variables, let R, S be relation symbols, let f, g be function symbols, and let c, d be constant symbols. Assume that the arities of relations and functions are correctly represented by the number of arguments they take in each formula. Which of the following are formulas? In the formulas identify the free and bound variables.*

a) $\forall x(f(c, f(x, d))$

b) $R(x, y, z) \vee S(f(c, d))$

c) $\exists y(R(x) \vee \forall z S(f(x, z), c))$

d) $\exists y(R(x) \vee \forall y S(f(x, y), c))$

e) $R(x) \wedge \exists x S(x)$

Exercise 3.15. *Let $\mathscr{L} = \{0, 1, +, \times\}$ be the language of basic arithmetic. Let $\phi = \forall x(\neg(x \approx 0) \to \exists y(x \times y \approx 1))$. Let $\mathbb{N}$ and $\mathbb{R}$ have their usual meanings, and interpret $\mathscr{L}$ into these languages by giving the non-logical symbols of $\mathscr{L}$ their usual meanings.*

a) Does $\mathbb{N} \models \phi$*?*

b) Does $\mathbb{R} \models \phi$*?*

c) Let $n \in \mathbb{N}$ *with* $n \geq 2$*, and let* $\mathbb{Z}_n$ *be the integers mod* n*. For what values of* n *does* $\mathbb{Z}_n \models \phi$*?*

d) Let $A = (\{a, b\}, I)$*, where* I *interprets* 0 *and* 1 *as* a *and* b *respectively,* $b \times b = b$*, and* $a \times b = a \times a = a$*. Does* $A \models \phi$*?*

e) Let $\psi = \exists x \forall y(\neg(y \approx 0) \to (x \times y \approx 1))$*. Which of the structures in parts a)-d) is a model for* ψ*?*

Exercise 3.16. *We define logical implication for first-order formulas of a language* $\mathscr{L}$ *by saying* $\phi \models \psi$ *if and only if, whenever* A *is an* $\mathscr{L}$*-structure and* v *is an assignment to* A*, we have*

$$A, v \models \phi \implies A, v \models \psi$$

We define two formulas to be logically equivalent if they both logically imply each other (we write e.g. $\phi \models\!\models \psi$*).*

Now, let R *and* S *be unary predicates. Let* $\phi = \forall x R(x) \vee \forall x S(x)$*, and let* $\psi = \forall x \forall y(R(x) \vee S(y))$*. Prove that* $\phi \models\!\models \psi$*.*

HINT: You don't need to worry about assignments here, because ϕ *and* ψ *are both sentences (the next exercise makes this precise). Just think about what the statements are saying. If* ϕ *holds in a structure, why must* ψ *also hold? Conversely, if* ψ *holds why must* ϕ *hold?*

Exercise 3.17. *Let* ϕ *be an* $\mathscr{L}$*-formula, let* A *be an* $\mathscr{L}$*-structure, and let* v *be an assignment for* $\mathscr{L}$ *to* A *with* $A, v \models \phi$*. Prove that* $A, u \models \phi$ *for all assignments* u *such that* $u(x) = v(x)$ *for all variables* x *occurring free in* ϕ*. HINT: You should use induction on the formula construction. First prove that this is true for atomic* $\mathscr{L}$*-formulas, then, assuming it's true for* ϕ *and* ψ *prove it's true for* $\neg\phi$*,* $\phi \vee \psi$*, and* $\forall x \phi$*. This is all we need because of the functional completeness of* $\{\neg, \vee\}$*, and the fact that* $\exists x \phi \models\!\models \neg \forall x \neg \phi$*. This exercise is difficult for people new to formal logic, but it's just a matter of understanding the definitions involved. If you get stuck you need to think carefully about exactly what you are trying to prove.*

It follows easily from this result that if ϕ *is an* $\mathscr{L}$*-sentence, then either* $A, v \models \phi$ *for all* v*, or there is no such* v*.*

3.5 Basic model theory

As discussed in the previous section, $\mathscr{L}$-structures give meaning to $\mathscr{L}$-sentences. So, if we want to understand an $\mathscr{L}$-sentence, or, more usually an $\mathscr{L}$-theory, we can try to understand its models, i.e. the $\mathscr{L}$-structures in which it is true. Conversely, given a mathematical object, we can try to understand it better by interpreting it as an $\mathscr{L}$-structure for some language $\mathscr{L}$, then seeing which

$\mathscr{L}$-sentences it is a model for. For example, we can think of natural number arithmetic as a structure for some suitable language, and we can investigate properties of natural number arithmetic by investigating the formulas of this language. This is not a hypothetical example. Logicians in the mid 20th century used this approach to solve a famous problem about so-called *Diophantine equations* (look up Hilbert's tenth problem). This two-way process of understanding languages through models, and models through languages, is the starting point of the field known as *model theory*.

It is through model theory that mathematical logic finds most of its applications in modern mathematics, and while other areas such as recursion and computability theory are more relevant for research in computer science[3], the basics of model theory are important for anyone who wants to understand applications of logic. This section explores the concept of a model as introduced in the last section. We will see a concept of deduction for first-order logic, based on that for propositional logic, and we will connect the concepts of deductive implication and semantic implication via models using soundness and completeness theorems, as we did for propositional logic. We will also look at the concept of an 'intended model', and see some unavoidable limitations of using first-order logic to describe infinite structures.

Semantics. Generalizing exercise 4.3, if Γ is a set of $\mathscr{L}$-formulas for some first-order signature $\mathscr{L}$, and if ϕ is an $\mathscr{L}$-formula, then we write $\Gamma \models \phi$ if, whenever v is an assignment of the variables of $\mathscr{L}$ into an $\mathscr{L}$-structure A, we have

$$A, v \models \Gamma \implies A, v \models \phi.$$

We say that ϕ is a *logical consequence* of Γ. An important special case is *sentences*, that is, formulas that have no free variables. By exercise 4.4, for sentences the assignment v is irrelevant. In this case we can just write, e.g.

$$A \models \phi.$$

When $A \models \phi$ for a sentence ϕ we say A is a *model* for ϕ. Similarly, if Δ is a set of $\mathscr{L}$-sentences we can write e.g. $A \models \Delta$ when $A \models \phi$ for all $\phi \in \Delta$, and say A is a model for Δ.

Definition 3.5.1. *If ϕ is an $\mathscr{L}$-formula then we say ϕ is:*

- Valid *if $A, v \models \phi$ whenever A is an $\mathscr{L}$-structure and v is an assignment.*
- Satisfiable *if there is an $\mathscr{L}$-structure A and an assignment v with $A, v \models \phi$.*
- *A* contradiction *if it is not satisfiable, i.e. if there is no A, v with $A, v \models \phi$.*

Similarly, if Γ is a set of $\mathscr{L}$-formulas then Γ is:

[3]We should mention here that the proof of the Diophantine problem mentioned above has more to do with computability theory than with modern 'mathematical' model theory, so these subjects are not totally disconnected from pure mathematics.

- Valid *if* $A, v \models \Gamma$ *whenever* A *is an* $\mathscr{L}$*-structure and* v *is an assignment.*
- Satisfiable *if there is an* $\mathscr{L}$*-structure* A *and an assignment* v *with* $A, v \models \Gamma$.
- Contradictory *if it is not satisfiable, i.e. if there is no* A, v *with* $A, v \models \Gamma$. *If* Γ *is not satisfiable we write* $\Gamma \models \bot$.

Example 3.5.2. *Let* $\mathscr{L} = \{0, 1, \times, +\}$ *be the language of arithmetic.*

1. *Let* $\phi = \forall x\big((x \approx 0) \vee \neg(x \approx 0)\big)$. *Then* ϕ *is valid. More generally, if* $\mathscr{L}$ *is a language, and if* $\phi_1, \ldots, \phi_n$ *are* $\mathscr{L}$*-sentences, then any propositional tautology constructed by treating the* ϕ_i *as basic propositions will be valid.*
2. *Let* $\psi = \forall x(\neg(x \approx 0) \to \exists y(x \times y \approx 1))$. *This is true if we take* $\mathbb{R}$ *as our structure, but not if we take* $\mathbb{Z}$. *So* ψ *is satisfiable but not valid.*
3. *If* $\phi_1, \ldots, \phi_n$ *are* $\mathscr{L}$*-sentences, then any propositional contradiction using the* ϕ_i *as basic propositions will be a contradiction.*

Definition 3.5.3 (Theory)**.** *If* $\mathscr{L}$ *is a language, then an* $\mathscr{L}$-theory *is a satisfiable set of* $\mathscr{L}$*-sentences.*

Checking logical consequence, validity etc. is much more complicated for first-order logic than for propositional logic. In propositional logic all we have to do is construct a truth table, which is a deterministic process. It may take a long time but we know that, in the end, we will get an answer. In first-order logic, to check directly if a sentence is valid we have to look at every possible structure and check that it is a model. Since there may be an infinite number of structures this cannot usually be done. We might ask if there is an algorithm that can tell if a sentence is valid, using a trick to avoid having to check every possible model. There is no obvious way to do this, and in fact, no such algorithm can exist (as we will see next semester).

Intended models. When we write down axioms in first-order logic, there is often some particular system whose behaviour we are trying to formalize. For example, we might write down axioms for defining real numbers. The intended model here is $\mathbb{R}$, and we can choose axioms so that $\mathbb{R}$ is indeed a model. But can we choose first-order axioms so that $\mathbb{R}$ is the only model? The answer to this is no. In fact, it is impossible to use first-order logic to define a specific infinite structure, due to the following important theorem (which we state without proof).

Theorem 3.5.4 (Löwenheim-Skolem theorem)**.** *Let* Γ *be a countable* $\mathscr{L}$*-theory. Then, if* Γ *has an infinite model, it has models of every infinite cardinality.*

Theorem 3.5.4 gives us an infinite supply of extra models for any theory that has at least one infinite model. Unintended models need not have different cardinalities though, as the following example illustrates.

Example 3.5.5. *Let $\mathscr{L} = \{0, s\}$, where s is a unary function. Let Γ consist of the following sentences.*

ϕ_1: $\forall x(\neg(x \approx 0) \to \exists y(x = s(y)))$.

ϕ_2: $\forall x(\neg(x \approx s(x)))$.

ϕ_3: $\forall x \forall y((s(x) \approx s(y)) \to (x \approx y))$.

One model of Γ is the natural numbers, where s is interpreted as the 'successor' function. Is $\mathbb{N}$ the only model? No, for example, the disjoint union of $\mathbb{N}$ and $\mathbb{Z}$ is also a model if we interpret 0 as the zero of $\mathbb{N}$, and s as the successor function in both $\mathbb{N}$ and $\mathbb{Z}$.

Substitution. Let ϕ be an $\mathscr{L}$-formula with free variables $x_1, \ldots x_n$. We can express this fact by writing $\phi[x_1, \ldots, x_n]$. Now, let t be an $\mathscr{L}$-term, and let $i \in \{1, \ldots, n\}$. Then we can create a new formula from ϕ by replacing every occurrence of the variable x_i with the term t. We use the notation $\phi[x_1, \ldots, x_{i-1}, t/x_i, x_{i+1}, \ldots, x_n]$ to denote this new formula. Note that this new formula may have different free variables, depending on what variables occur free in t.

Sometimes we will write something like $\phi[t/x]$. This represents substituting some variable x that occurs free in ϕ with a term t. In other words, we sometimes hide some free variables and make explicit only the one we are replacing.

Example 3.5.6. *Let $\mathscr{L} = \{0, s\}$ be the language from example 3.5.5, and let $\phi = s(x) \approx y$. Then we may write $\phi[x, y]$ when we want to explicitly mention the free variables of ϕ. Let $t = s(s(z))$ be a term. Then $\phi[t/x, y] = s(s(s(z))) \approx y$. Alternatively, we could not mention x explicitly and write something like $\phi[t/y]$, which is the $\mathscr{L}$-formula $s(x) \approx s(s(z))$ in this case.*

Syntax. We can extend the natural deduction system for propositional logic to first-order logic. We have all the same deduction rules as before (but with first-order formulas in place of propositional sentences), and also the following extra ones.

Introduction rules.

$$\approx_I: \quad \frac{}{t \approx t}$$

$$\forall_I: \quad \frac{\phi[x'/x]}{\forall x \phi}$$

$$\exists_I: \quad \frac{\phi[t/x]}{\exists x \phi}$$

Elimination rules.

$$\approx_E: \quad \frac{t_1 \approx t_2 \qquad \phi[t_1/z]}{\phi[t_2/z]}$$

$$\forall_E: \quad \frac{\forall x \phi}{\phi[t/x]}$$

$$\exists_E: \quad \frac{\exists x \phi \qquad \dfrac{[\phi[x'/x]]}{\psi}}{\psi}$$

These rules require additional explanation, as the notation hides some details.

$\approx_I$: This is fairly straightforward. It just says that we can deduce the fact that a term is identical to itself from an empty set of assumptions.

$\forall_I$: Here ϕ is a formula where x occurs free. The intuition behind this rule is that, if we can prove ϕ in the case where $x = x'$, for arbitrary x', then ϕ should be true for all possible values of x. To make this rule sound we need to make sure that x' has no special property that specifies it as a member of a strict subset of the domain. Formally, this means that the symbol x' must not occur free in an assumption or axiom anywhere in the proof tree above the application of this rule, or in the formula $\forall x\phi[x]$ itself. Why? Because if x' occurs free in an assumption, axiom or ϕ then we are supposing some fact involving x', and this might constrain x'. So we could not say with certainty that $\phi[x]$ for arbitrary x holds just because $\phi[x']$ holds, as $\phi[x']$ might only hold because of the extra property we are supposing x' has.

$\exists_I$: Here again ϕ is a formula where x occurs free, and t can be any term. The intuition is that, if we can prove ϕ for some value of x, then $\exists x\phi$ must be true.

$\approx_E$: Here ϕ is a formula where z occurs free, and t_1 and t_2 are terms. The intuition is that, if t_1 and t_2 are equal, and if ϕ is true in the case where $z = t_1$, then ϕ should also be true in the case where $z = t_2$.

$\forall_E$: ϕ is a formula where x occurs free and t is a term. The intuition here is that if ϕ is true for all values of x, then, in particular, ϕ should be true when $x = t$.

$\exists_E$: Once again, ϕ is a formula where x occurs free. The idea is that, if we can deduce ψ from ϕ where x is set to any arbitrary value, then, if we know there is some value for x which makes ϕ true (i.e. $\exists x\phi$), then we should be able to conclude that ψ is true. Here again we have to be careful that x' is truly arbitrary, which again means that it must not occur free in ϕ, or in an assumption or axiom previously in the proof tree.

Example 3.5.7. *Let ϕ and ψ be formulas where x occurs free. Then we can deduce $\forall x\psi$ from $\forall x\neg\phi$ and $\forall x(\phi \vee \psi)$.*

$$\dfrac{\dfrac{\forall x\neg\phi}{\neg\phi[x'/x]}(\forall_E) \qquad \dfrac{\forall x(\phi \vee \psi)}{\phi[x'/x] \vee \psi[x'/x]}(\forall_E)}{\text{(propositional deduction, see example 3.2.6)}} \quad \dfrac{\psi[x'/x]}{\forall x\psi}(\forall_I)$$

Example 3.5.8. *Let ϕ and ψ be formulas where x occurs free. Then we can deduce $\exists x\psi$ from $\exists x\neg\phi$ and $\forall x(\phi \vee \psi)$.*

$$\dfrac{\dfrac{\exists x \neg\phi \qquad \dfrac{[\neg\phi[x'/x]]}{\neg\phi[x'/x]}\,(\exists_E)}{\neg\phi[x'/x]} \qquad \dfrac{\forall x(\phi \vee \psi)}{\phi[x'/x] \vee \psi[x'/x]}\,(\forall_E)}{\dfrac{\psi[x'/x]}{\exists x \psi}\,(\exists_I)}\ \text{(propositional deduction)}$$

Note that in this second example, we can't use $\forall_I$ to deduce $\forall x\psi$ at the end. This is because x' occurs in the assumption we used in the deduction of $\neg\phi[x'/x]$ at the start.

Soundness and completeness. As with propositional logic we write $\Gamma \vdash \phi$ if ϕ can be deduced from a set of formulas Γ. We say a set of $\mathscr{L}$-sentences, Γ, is *consistent* if we do not have $\Gamma \vdash \bot$. We sometimes describe a consistent set of $\mathscr{L}$-sentences as an *$\mathscr{L}$-theory*. This is consistent with definition 3.5.3 because, as in propositional logic, there is a strong link between $\vdash$ and $\models$.

Theorem 3.5.9 (Gödel)**.** *Let Γ be a set of $\mathscr{L}$-formulas. Then Γ is consistent if and only if it is satisfiable.*

This theorem is equivalent to the following result (the proof of this is one of the exercises).

Theorem 3.5.10 (Extended soundness and completeness)**.** *Let Γ be a set of $\mathscr{L}$-formulas and let ϕ be an $\mathscr{L}$-formula. Then*

$$\Gamma \vdash \phi \iff \Gamma \models \phi.$$

Proof. This proof is too long for us here, but we provide a sketch of the main ideas involved the argument. Proving soundness is similar to the inductive argument used for the propositional case. The base case again is easy, so the key step is proving for each rule that an argument that has been sound up to a final application of that rule remains sound after this application. Arguments for the rules shared with propositional logic are essentially the same as in the propositional case. For example:

$\wedge_I$: Here we have deduced ϕ and ψ from Γ, and from these have deduction $\phi \wedge \psi$. Assuming that the deductions of ϕ and ψ are both sound, this means that any model, (A, v), of Γ must be a model of both ϕ and ψ, and therefore must also be a model of $\phi \wedge \psi$, by the definition of $\models$.

The new rules are a little technically tricky, but don't require any creative leap. For example:

$\forall_I$: Here we have deduced $\phi[x'/x]$ from Γ for arbitrary choice of x'. Assuming this deduction is sound, this means that any pair (A, v) satisfying Γ will also satisfy $\phi[x'/x]$. We must show that $A, v \models \forall x\phi[x]$ too. We proceed as follows:

- By the rules for $\forall_I$, the variable x' must have not occurred in the deduction tree above $\phi[x'/x]$. So only axioms where x' does not occur free are used.
- Therefore there is a subset Γ' of Γ containing only formulas where x' does not occur free with $\Gamma' \vdash \phi[x'/x]$.
- The inductive hypothesis applied to this deduction gives $\Gamma' \models \phi[x'/x]$.
- Now, let $A, v \models \Gamma$. We want to show that $A, v \models \forall x \phi[x]$.
- For this, we must show that if v' agrees with v about everything except, possibly, x (temporary notation $v' =_x v$), then $A, v' \models \phi[x]$.
- So, let $v' =_x v$.
- Define $v'' =_{x'} v$ by setting $v''(x') = v'(x)$.
- Then $A, v'' \models \Gamma'$, as x' does not occur free in any formula in Γ'.
- So $A, v'' \models \phi[x'/x]$.
- It follows that $A, v' \models \phi[x]$, because x' does not occur free in ϕ (by the rules for $\forall_I$), by definition of v'' we have $v'(x) = v''(x')$, and for $y \notin \{x, x'\}$ we have $v'(y) = v(y) = v''(y)$. In other words, evaluating $\phi[x]$ with v' is exactly the same as evaluating $\phi[x'/x]$ using v''.
- Thus $A, v \models \forall x \phi[x]$ as required.

Completeness is harder, but conceptually similar to the propositional version. Again, proving completeness is equivalent to proving that every consistent set of sentences is satisfiable. The difference here is that, rather than just building a true/false assignment that satisfies a consistent set of propositional sentences, we must find a pair (A, v) satisfying a set of first-order formulas. We omit the lengthy details, but it turns out that it is possible to do this, using the formulas themselves as the base of the structure. □

There's also a compactness theorem for first-order logic, which you will find in the exercises.

Exercises

Exercise 3.18. *Let ϕ be a formula where x occurs free.*

a) Write down a proof tree that shows $\forall x \phi \vdash \neg \exists x \neg \phi$.

b) Write down a proof tree that shows $\exists x \phi \vdash \neg \forall x \neg \phi$.

Exercise 3.19. *Assume that lemma 3.3.5 holds for first-order logic. Prove that theorems 3.5.9 and 3.5.10 are equivalent. I.e., prove the equivalence of the statements*

$$\Gamma \textit{ consistent} \iff \Gamma \textit{ is satisfiable} \tag{†}$$

$$\Gamma \vdash \phi \iff \Gamma \models \phi \tag{‡}$$

Exercise 3.20. *Prove that if* Γ *is an* $\mathscr{L}$*-theory then there is an* $\mathscr{L}$*-theory* Γ' *with* $\Gamma \subseteq \Gamma'$ *such that* Γ' *is* complete *(i.e. if* ϕ *is an* $\mathscr{L}$*-sentence, then either* $\phi \in \Gamma'$ *or* $\neg\phi \in \Gamma'$*). HINT: If* Γ *is consistent then it must have a .....?*

Exercise 3.21. *Let* Γ *be an* $\mathscr{L}$*-theory, and let* ϕ *be an* $\mathscr{L}$*-sentence. Prove that if* $\Gamma \models \phi$ *then* $\Delta \models \phi$ *for some finite* $\Delta \subseteq \Gamma$*.*

Exercise 3.22 (Compactness theorem for first-order logic)**.** *Let* Γ *be a set of* $\mathscr{L}$*-sentences. Prove that* Γ *has a model if and only if every finite subset of* Γ *has a model.*

3.6 Further reading

Some material on logic can be found in [8, chapter 3], though they don't seem to cover deduction. [9] is a short introduction to logic covering most things talked about here, including natural deduction, though unfortunately there are some notation differences.

4 Linear Algebra

4.1 Vector spaces over fields

Linear algebra is an abstract approach to thinking about Euclidean space. In other words, to points existing in a typically two or three dimensional grid defined by axes. The benefit of an abstract approach is that it lets us recognize structures that are not obviously 'space like' as being essentially Euclidean spaces in disguise. This allows us to take techniques and insights from geometric reasoning about Euclidean space and apply them in many diverse situations. In the opposite direction, the powerful machinery of linear algebra can also be used to get easy proofs of geometric facts about Euclidean space.

For example, linear algebra is used in computer graphics to correctly translate three dimensional information into images on a two dimensional screen. In addition, linear algebra underpins several techniques in machine learning, such as artificial neural networks and support vector machines. The Google page rank algorithm has linear algebra at its core. On this short course we will only scratch the surface of this deep subject, but the aim is to lay the foundations for a rigorous understanding of the theory and its applications. We will begin with very general and abstract definitions, and we will end the course by showing how the abstract approach allows us to easily prove some concrete results in Euclidean geometry.

Complex numbers We should all be familiar with the set of real numbers, $\mathbb{R}$. Not all polynomials with real coefficients have real roots. For example, there is no real value of x for which $x^2 + 1 = 0$. We express this fact by saying that $\mathbb{R}$ is not *algebraically closed*, which is just a fancy way of saying that not every polynomial with real coefficients can be factorized into linear factors with real coefficients. For example, we can't express $x^2 + 1$ as $(x + a)(x + b)$ for any $a, b \in \mathbb{R}$.

Often it is convenient to work in an algebraically closed setting, and for this reason we define the complex numbers, $\mathbb{C}$.

Definition 4.1.1 ($\mathbb{C}$)**.** *$\mathbb{C}$ is the set of numbers of form $a+bi$ such that $a, b \in \mathbb{R}$. Addition and multiplication in $\mathbb{C}$ are defined by:*

- $a + bi + c + di = a + c + (b + d)i$.
- $(a + bi) \times (c + di) = ac - bd + (ad + bc)i$.

Another way of putting this is that i is treated as a solution to $x^2 + 1 = 0$. I.e. $i^2 = -1$. Obviously, i is not a real number, and extending $\mathbb{R}$ by i lets us factorize polynomials that we could not factorize before. For example, $x^2 + 1 = (x - i)(x + i)$. In fact, $\mathbb{C}$ is algebraically closed, so every polynomial with complex coefficients can be factorized into linear complex pieces. Since $\mathbb{R} = \{a + bi \in \mathbb{C} : b = 0\}$, this means every polynomial with real coefficients can be factorized in $\mathbb{C}$ too.

This is a deep fact, and not obvious at all, and is often referred to as the *fundamental theorem of algebra*. We are not going to prove it on this course, but hopefully it gives some indication of why complex numbers are useful and interesting.

We don't need to remember the law for multiplying complex numbers, because we can reconstruct it just by remembering that $i^2 = -1$.

Example 4.1.2.

$$\begin{aligned}(2-4i)(1+7i) &= 2+14i-4i-28i^2\\ &= 2+28+10i\\ &= 30+10i\end{aligned}$$

Apart from containing extra roots for polynomials, complex numbers are like real numbers in many ways:

Lemma 4.1.3. *Let α, β and γ be complex numbers. Then:*

1. *$\alpha+\beta = \beta+\alpha$, and $\alpha\beta = \beta\alpha$ (commutativity).*
2. *$\alpha+(\beta+\gamma) = (\alpha+\beta)+\gamma$, and $\alpha(\beta\gamma) = (\alpha\beta)\gamma$ (associativity).*
3. *$0+\alpha = \alpha$, and $1\alpha = \alpha$ (identities).*
4. *There is a unique $-\alpha \in \mathbb{C}$ such that $\alpha+(-\alpha) = 0$ (inverse for addition).*
5. *If $\alpha \neq 0$ there is a unique α^{-1} such that $\alpha\alpha^{-1} = 1$ (inverse for multiplication).*
6. *$\alpha(\beta+\gamma) = \alpha\beta+\alpha\gamma$ (distributivity).*

Proof. We'll prove 1-5, and leave 6 for exercise 4.1.

1) $(a+bi)+(c+di) = (a+c)+(b+d)i = (c+di)+(a+bi)$, so $+$ is commutative. Also, $(a+bi)(c+di) = ac-bd+(ad+bc)i = (c+di)(a+bi)$, so $\times$ is commutative.

2)

$$\begin{aligned}((a+bi)+(c+di))+(e+fi) &= ((a+c)+(b+d)i)+(e+fi)\\ &= (a+c+e)+(b+d+f)i\\ &= (a+bi)+((c+di)+(e+fi)),\end{aligned}$$

so $+$ is associative. Also,

$$\begin{aligned}((a+bi)(c+di))(e+fi) &= ((ac-bd)+(ad+bc)i)(e+fi)\\ &= (ace-bde-adf-bcf)+(ade+bce+acf-bdf)i\\ &= (a+bi)((c+di)(e+fi)),\end{aligned}$$

so $\times$ is also associative.

3) It's obvious that $0 + a + bi = a + bi$, and also that $1(a + bi) = a + bi$.

4) Given $\alpha = a+bi$ define $-\alpha = -a-bi$. Then clearly $\alpha+(-\alpha) = 0$. Moreover, if $a + bi + c + di = 0$ then we must have $c = -a$ and $d = -b$, so $-\alpha$ as we have defined it is unique.

5) given $\alpha = a+bi$, suppose $(a+bi)(c+di) = 1$. Then $ac-bd+(ad+bc)i = 1$, and so

$$ac - bd = 1, \tag{$\dagger$}$$

and

$$ad = -bc. \tag{$\ddagger$}$$

If $b = 0$ then $\alpha^{-1} = \frac{1}{a}$ (we can do this as, since $\alpha \neq 0$, we must have $a \neq 0$), so we assume $b \neq 0$. So we can rewrite ($\ddagger$) as $c = \frac{-ad}{b}$. Substituting this into ($\dagger$) and rearranging gives $d = \frac{-b}{a^2+b^2}$ (we can divide by a^2+b^2 because we are assuming $b \neq 0$). Substituting this value for d into ($\ddagger$) produces $c = \frac{a}{a^2+b^2}$.

So, we define $\alpha^{-1} = \frac{a-bi}{a^2+b^2}$, and since these values of c and d are the only possible choices, this is the unique multiplicative inverse, α^{-1}, for α.

□

Notice that the properties of $\mathbb{C}$ proved in lemma 4.1.3 are the same as the properties of $\mathbb{R}$, and also as those of $\mathbb{Z}_p$ when p is prime. Actually, $\mathbb{R}$, $\mathbb{C}$ and $\mathbb{Z}_p$ are all examples of a mathematical structure known as a *field*. We will not go into the details of the abstract definition of a field, but we will say that most of the results we prove on this course apply to fields in general, and do not rely on any special properties of $\mathbb{R}$ and $\mathbb{C}$, except those that make them fields of course. We will use $\mathbb{F}$ to denote a general (infinite field). For us, this just means that $\mathbb{F}$ can stand for $\mathbb{R}$ or $\mathbb{C}$.

We define subtraction and division in $\mathbb{C}$ using lemma 4.1.3, in particular parts 4 and 5:

Definition 4.1.4. *Let $\alpha, \beta \in \mathbb{C}$, and suppose $\beta \neq 0$. Then:*

- $\alpha - \beta = \alpha + (-\beta)$.
- $\frac{\alpha}{\beta} = \alpha\beta^{-1}$.

Vector spaces We define vector spaces abstractly, but example 4.1.6 below provides some motivating, hopefully familiar examples.

Definition 4.1.5. *Let $\mathbb{F}$ be a field. Then a* vector space over $\mathbb{F}$ *is a set V equipped with a vector addition operation from $V \times V$ to V and a scalar multiplication operation from $\mathbb{F} \times V$ to V that obey the following rules:*

1. *$u + v = v + u$ for all $u, v \in V$.*
2. *$u + (v + w) = (u + v) + w$ for all $u, v, w \in V$.*

3. $(ab)v = a(bv)$ *for all* $a, b \in \mathbb{F}$ *and for all* $v \in V$.

4. *There is a special element* $0 \in V$ *such that* $0 + v = v$ *for all* $v \in V$.

5. *For all* $v \in V$ *there is* $w \in V$ *such that* $v + w = 0$.

6. $1v = v$ *for all* $v \in V$ *(i.e. scalar multiplication by* 1 *does not change* v*).*

7. $a(u + v) = au + av$ *for all* $a \in \mathbb{F}$ *and for all* $u, v \in V$.

8. $(a + b)v = av + bv$ *for all* $a, b \in \mathbb{F}$ *and for all* $v \in V$.

When $\mathbb{F} = \mathbb{R}$ *we say* V *is a* real vector space*. When* $\mathbb{F} = \mathbb{C}$ *we say* V *is a* complex vector space*. We sometimes refer to elements of* V *as* vectors*, or* points*.*

Example 4.1.6.

1. *We can think of any field as a vector space over itself. E.g.* $\mathbb{R}$ *is a real vector space (vector addition and scalar multiplication are just ordinary addition and multiplication in* $\mathbb{R}$*).*

2. $\mathbb{R} \times \mathbb{R}$*, i.e. the Euclidean plane, is a real vector space.*

3. *More generally, for any* $n \in \mathbb{N} \setminus \{0\}$ *we can think of* $\mathbb{F}^n$ *as a vector space over* $\mathbb{F}$ *by defining* $(x_1, \ldots, x_n) + (y_1, \ldots, y_n) = (x_1 + y_1, \ldots, x_n + y_n)$*, and* $a(x_1, \ldots x_n) = (ax_1, \ldots, ax_n)$*.*

4. *Let* $\mathbb{R}[x]$ *be the set of all polynomials with the variable* x*. So*

   $\mathbb{R}[x] = \{a_0 + a_1 x + \ldots + a_n x^n : n \in \mathbb{N}$ *and* $a_i \in \mathbb{R}$ *for all* $i \in \{1, \ldots, n\}\}$.

   Then $\mathbb{R}[x]$ *is a vector space over* $\mathbb{R}$*.*

The following result sums up some important basic properties of vector spaces.

Proposition 4.1.7. *Let* V *be a vector space over* $\mathbb{F}$ *as described in definition 4.1.5. Then:*

1. *The additive identity* 0 *is unique.*

2. *The additive inverse of* v *is unique for all* $v \in V$ *(we call it* $-v$*).*

3. $0v = 0$ *for all* $v \in V$.

4. $-1v = -v$ *for all* $v \in V$.

Proof.

1. Suppose 0 and $0'$ are both additive identities for V. Then $0 = 0 + 0' = 0'$.

2. Suppose $v + u = 0$ and $v + u' = 0$. Then $(v + u) + u' = u'$, and so $(v + u') + u = u'$, which means $u = u'$.

3. $0v = (0 + 0)v = 0v + 0v$, so $0v + (-0v) = (-0v) + 0v + 0v$, and so $0 = 0v$.

4. Exercise 4.3.

□

Subspaces

Definition 4.1.8. *Let V be a vector space over $\mathbb{F}$. Then a subset U of V is a* subspace *of V if it has the following properties:*

1. $0 \in U$.
2. $u + v \in U$ *for all* $u, v \in U$ *(closure under vector addition).*
3. $au \in U$ *for all* $a \in \mathbb{F}$ *and for all* $u \in U$ *(closure under scalar multiplication).*

Lemma 4.1.9. *If V is a vector space over $\mathbb{F}$ then $U \subseteq V$ is a subspace of V if and only if it is also a vector space over $\mathbb{F}$ with the addition and scalar multiplication inherited from V.*

Proof. If U is a vector space with the inherited operations then it must obviously be closed under the inherited operations and contain 0. Conversely, if U satisfies the conditions of definition 4.1.8 then it automatically satisfies all conditions of definition 4.1.5 except (5). To see that (5) also holds in U note that, by proposition 4.1.7(4), given $u \in U$ we have $-u = -1u$, which is in U by definition 4.1.8(3). □

Definition 4.1.10. *Given subspaces $U_1, \ldots, U_n$ of V, the* sum $U_1 + \ldots + U_n$ *is the smallest subspace of V containing $\bigcup_{i=1}^{n} U_i$.*

Lemma 4.1.11. *If $U_1, \ldots, U_n$ are subspaces of V, then*

$$U_1 + \ldots + U_n = \{u_1 + \ldots + u_n : u_i \in U_i \text{ for all } i \in \{1, \ldots, n\}\}.$$

Proof. $\{u_1 + \ldots + u_n : u_i \in U_i \text{ for all } i \in \{1, \ldots, n\}\}$ contains $\bigcup_{i=1}^{n} U_i$ because $u_i = 0 + \ldots + 0 + u_i + 0 + \ldots + 0$ for all $u_i \in U_i$. That it is a subspace follows from the definition of a vector space. It must be the smallest subspace containing $\bigcup_{i=1}^{n} U_i$, because any such subspace must be closed under vector addition. □

Definition 4.1.12. *If $U_1, \ldots, U_n$ are subspaces of V, then the sum $U_1 + \ldots + U_n$ is a* direct sum *if, for all $u \in U_1 + \ldots + U_n$, there is exactly one choice of $\{u_1, \ldots, u_n\}$ such that $u_i \in U_i$ for all i and $u = u_1 + \ldots + u_n$. In this case we write $U_1 \oplus \ldots \oplus U_n$.*

So direct sum is a sum where there is no redundancy. Every element in a direct sum is formed in exactly one way using the subspaces that make up the sum. The next lemma says that to check if a sum is direct all we need to do is check there is no redundancy in the expression of 0.

Lemma 4.1.13. *If $U_1, \ldots, U_n$ are subspaces of V, then $U_1 + \ldots + U_n$ is a direct sum if and only if there is exactly one choice of $\{u_1, \ldots, u_n\}$ such that $u_i \in U_i$ for all i and $0 = u_1 + \ldots + u_n$.*

Proof. If $U = U_1 + \ldots + U_n$ is a direct sum, then by definition there is only one way to express 0 (i.e. $0 = 0 + \ldots + 0$). Conversely, suppose there is only one way to express 0, let $u \in U$, and suppose $u = u_1 + \ldots + u_n = u_1' + \ldots + u_n'$. Then

$$0 = u_1 + \ldots + u_n - (u_1' + \ldots + u_n') = (u_1 - u_1') + \ldots + (u_n - u_n').$$

So $(u_i - u_i') = 0$ for all i, as there is only one way to express 0, and thus $u_i = u_i'$ for all i. □

In the special case of sums of two subspaces we have the following result:

Lemma 4.1.14. *Let U and W be subspaces of V. Then $U + W$ is a direct sum if and only if $U \cap W = \{0\}$.*

Proof. If there is $v \in U \cap W$ then $v = 0 + v$ and $v = v + 0$, so $U + W$ is not a direct sum, as v is not uniquely expressible. Conversely, suppose $U \cap W = \{0\}$ and that $v = u + w$ and $v = u' + w'$. Then $u - u' = w' - w$, and so $u - u'$ and $w' - w$ are both in $U \cap W$, and thus are both 0. This implies $u = u'$ and $w = w'$, and so $U + W$ is a direct sum. □

Example 4.1.15. *Let $V = \mathbb{R}^3$, let $U_1 = \{(2x, 0, z) : x, z \in \mathbb{R}\}$, let $U_2 = \{(0, y, 0) : y \in \mathbb{R}\}$, and let $U_3 = \{(0, z, z) : z \in \mathbb{R}\}$. Then $\mathbb{R}^3 = U_1 + U_2 + U_3$, because given $(a, b, c) \in \mathbb{R}^3$ we have*

$$(a, b, c) = (2(\frac{a}{2}), 0, 0) + (0, b - c, 0) + (0, c, c).$$

However, $U_1 + U_2 + U_3$ is not a direct sum as

$$(0, 0, 0) = (0, 0, 1) + (0, 1, 0) + (0, -1, -1).$$

I.e., 0 is not uniquely expressible.

However, $U_i \cap U_j = \{0\}$ for all $i \neq j$, which indicates that lemma 4.1.14 is a special result for binary sums, and does not hold in general for sums involving more than two subspaces.

Linear independence and span

Definition 4.1.16. *Given a vector space V over $\mathbb{F}$, and vectors $v_1, \ldots v_n \in V$, we say the* span *of $(v_1, \ldots, v_n)$ is the smallest subspace of V containing $\{v_1, \ldots, v_n\}$. By convention we define* span$() = \{0\}$. *If* span$(v_1, \ldots, v_n) = V$ *we say $(v_1, \ldots, v_n)$* spans *V.*

Lemma 4.1.17. *If V is vector space over $\mathbb{F}$, and $v_1, \ldots v_n \in V$, then*

$$\text{span}(v_1, \ldots v_n) = \{a_1 v_1 + \ldots + a_n v_n : a_i \in \mathbb{F} \text{ for all } i\}.$$

Proof. Let $U = \{a_1v_1 + \ldots + a_nv_n : a_i \in \mathbb{F} \text{ for all } i\}$. Then clearly $U \subseteq \text{span}(v_1, \ldots v_n)$, as $\text{span}(v_1, \ldots v_n)$ is closed under vector addition and scalar multiplication. Moreover, U is closed under vector addition and scalar multiplication, so U is a subspace of V. Since $\{v_1, \ldots, v_n\} \subseteq U$, it follows from the definition of $\text{span}(v_1, \ldots v_n)$ as the smallest such subspace that $\text{span}(v_1, \ldots v_n) \subseteq U$. Thus $U = \text{span}(v_1, \ldots v_n)$ as required. □

Definition 4.1.18. *Let V be vector space over $\mathbb{F}$, and let $v_1, \ldots v_n \in V$. Then $(v_1, \ldots, v_n)$ is* linearly independent *if whenever $a_1v_1 + \ldots + a_nv_n = 0$ we have $a_1 = \ldots = a_n = 0$. If $(v_1, \ldots, v_n)$ is not linearly independent then we say it is* linearly dependent*.*

Example 4.1.19.

1. *The vectors $(1, 0, 0)$, $(0, 1, 0)$ and $(0, 0, 1)$ are linearly independent and span $\mathbb{R}^3$ and $\mathbb{C}^3$.*
2. *The span of a single vector v is $\{av : a \in \mathbb{F}\}$. Single vectors are always linearly independent.*
3. *The vectors $(2, 3, 1)$, $(1, -1, 2)$ and $(7, 3, c)$ are linearly independent so long as $c \neq 8$.*
4. *Every list of vectors containing 0 is linearly dependent, by convention.*

Exercises

Exercise 4.1. *Prove lemma 4.1.3(6).*

Exercise 4.2. *Consider the following 'proof'. What is wrong with it?*

$$-1 = i^2 = i.i = \sqrt{-1}\sqrt{-1} = \sqrt{(-1)(-1)} = \sqrt{1} = 1.$$

Exercise 4.3. *Complete the proof of proposition 4.1.7.*

Exercise 4.4. *Given $v \in V$, prove that $-(-v) = v$.*

Exercise 4.5. *Given $a \in \mathbb{F}$ and $v \in V$ prove that $av = 0$ if and only if either $a = 0$ or $v = 0$.*

Exercise 4.6. *Let U and W be subspaces of V. Prove that $U \cap W$ is a subspace of V.*

Exercise 4.7. *(Optional) Let U and W be subspaces of V. Prove that if $U \cup W$ is a subspace of V, then either $U \subseteq W$ or $W \subseteq U$.*

Exercise 4.8. *(Optional) Let V be vector space over $\mathbb{F}$, and let $v_1, \ldots v_n \in V$ such that $(v_1, \ldots, v_n)$ is linearly independent. Let $w \in V$. Prove that $(v_1, \ldots, v_n, w)$ is linearly independent if and only if $w \notin \text{span}(v_1, \ldots, v_n)$.*

4.2 Dimension

Bases Our mental model for vector spaces should be something like $\mathbb{R}^2$, the Euclidean plane. In $\mathbb{R}^2$, every vector is defined by coordinates (x, y). In other words, every vector in $\mathbb{R}^2$ can be written as a sum $x(1,0)+y(0,1)$ of the vectors $(1,0)$ and $(0,1)$. These vectors $(1,0)$ and $(0,1)$ are very special, as their linear combinations generate the whole of $\mathbb{R}^2$, and any set of vectors with this property must have at least size two. The following definition generalizes this idea to abstract vector spaces.

Definition 4.2.1. *If V is a vector space, then a* basis *for V is a linearly independent set that spans V.*

Lemma 4.2.2. *Let V be a vector space over $\mathbb{F}$. Let $v_1, \ldots, v_n \in V$. Then $(v_1, \ldots, v_n)$ is a basis for V if and only if every element u can be expressed as $a_1v_1 + \ldots + a_nv_n$, for some unique $\{a_1, \ldots, a_n\} \subseteq \mathbb{F}$.*

Proof. Suppose $(v_1, \ldots, v_n)$ is a basis for V. Then, given $u \in V$, that $u = a_1v_1 + \ldots + a_nv_n$ for some $\{a_1, \ldots, a_n\} \subseteq \mathbb{F}$ follows directly from the fact that $(v_1, \ldots, v_n)$ spans V. Moreover, if $u = a_1v_1 + \ldots + a_nv_n = b_1v_1 + \ldots + b_nv_n$, then $0 = (a_1 - b_1)v_1 + \ldots + (a_n - b_n)v_n$, so $a_i = b_i$ for all $i \in \{1, \ldots, n\}$, as $(v_1, \ldots, v_n)$ is linearly independent.

Conversely, if $(v_1, \ldots, v_n)$ satisfies the two stated properties then it is a linearly independent spanning set. To see this note that it obviously spans V, and it is linearly independent as the only way to express 0 as $a_1v_1 + \ldots + a_nv_n$ is with $a_1 = \ldots = a_n$. This proves the result. □

Bases are extremely important in the study of vector spaces because, like the prime numbers generate the integers, a vector space is generated by a basis. In other words, if you have a basis, then you know the space. There are natural questions we can ask about bases. Does every vector space have one? Can a space have more than one? If a space has two (or more) bases, are they essentially equivalent? In other words, does it matter what basis we choose when working with a vector space? We will see answers to these questions soon, but first we need the following useful technical lemma.

Lemma 4.2.3. *Let V be a vector space over $\mathbb{F}$, and let $v_1, \ldots v_n \in V$. Suppose that $(v_1, \ldots, v_n)$ is linearly dependent. Then there is $j \in \{1, \ldots, n\}$ such that:*

1. $v_j \in \text{span}(v_1, \ldots, v_{j-1})$.
2. $\text{span}(v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_n) = \text{span}(v_1, \ldots, v_n)$.

Proof. Since $(v_1, \ldots, v_n)$ is linearly dependent, by the definition of linear dependence there are $a_1, \ldots, a_n \in \mathbb{F}$ such that $a_1v_1 + \ldots + a_nv_n = 0$ and such that at least some a_i is not equal to zero. Let j be the largest value such that $a_j \neq 0$. Then $a_1v_1 + \ldots + a_jv_j = 0$, and, since $a_j \neq 0$ we can rewrite this as $v_j = -\frac{a_1}{a_j}v_1 - \ldots - \frac{a_{j-1}}{a_j}v_{j-1}$. This proves (1), and (2) follows easily from (1), so we are done. □

Proposition 4.2.4. *Let V be a vector space over $\mathbb{F}$, let $(u_1, \ldots, u_k)$ be linearly independent, and let $(v_1, \ldots, v_n)$ span V. Then $k \leq n$. In other words, linearly independent lists of vectors cannot be bigger than spanning lists of vectors.*

Proof. We will use lemma 4.2.3 multiple times.

The idea behind the proof is to replace elements of $(v_1, \ldots, v_n)$ with different elements of $(u_1, \ldots, u_k)$, till we have used all the elements of $(u_1, \ldots, u_k)$. Being able to do this implies that $k \leq n$.

To do this, for each $i \in \{1, \ldots, k\}$, we will define a list s_i such that the following properties are satisfied:

1. For all $j \in \{1, \ldots, k-1\}$, if $s_j = (u_j, u_{j-1}, \ldots, u_1, v'_1, \ldots, v'_{n-j})$, then the list s_{j+1} has form

$$(u_{j+1}, u_j, u_{j-1}, \ldots, u_1, v''_1, \ldots, v''_{n-(j+1)}),$$

   where $\{v''_1, \ldots, v''_{n-(j+1)}\} \subset \{v'_1, \ldots, v'_{n-j}\}$.

2. $\text{span}(s_j) = V$ for all $j \in \{1, \ldots, k\}$.

Consider the list $(u_1, v_1, \ldots, v_n)$. By lemma 4.2.3 there is an element w of $(u_1, v_1, \ldots, v_n)$ such that w is in the span of the part of the list $(u_1, v_1, \ldots, v_n)$ that precedes it. Obviously we can't have $w = u_1$, as then u_1 would have to be 0, as this is the only thing in the span of the empty list. So w is in $(v_1, \ldots, v_n)$, and we define s_1 by removing w from $(u_1, v_1, \ldots, v_n)$. We proceed using a recursive process. Suppose we have constructed the lists

$$s_1, \ldots, s_i = (u_i, u_{i-1}, \ldots, u_1, v'_1, \ldots, v'_{n-i})$$

with the required properties. Then, as s_i spans V, the list

$$(u_{i+1}, u_i, u_{i-1}, \ldots, u_1, v'_1, \ldots, v'_{n-i})$$

must be linearly dependent. So, again by lemma 4.2.3, there is an element w of $(u_{i+1}, u_i, u_{i-1}, \ldots, u_1, v'_1, \ldots, v'_{n-i})$ such that w is in the span of the part of the list $(u_{i+1}, u_i, u_{i-1}, \ldots, u_1, v'_1, \ldots, v'_{n-i})$ that precedes it, and the span of the list obtained by removing w is still V.

Since $(u_1, \ldots, u_k)$ is linearly independent, w cannot be a member of $\{u_{i+1}, \ldots, u_1\}$, so the list we get by removing w has form

$$(u_{i+1}, u_i, u_{i-1}, \ldots, u_1, v''_1, \ldots, v''_{n-(i+1)}),$$

where $\{v''_1, \ldots, v''_{n-(i+1)}\} \subset \{v'_1, \ldots, v'_{n-i}\}$. We define s_{i+1} to be this new list, noting that it satisfies the required properties.

This construction works until we hit the limit $i = k$. This proves the result, because every time we remove an element it must be a new element from the original list $(v_1, \ldots, v_n)$. Since we remove k elements in total, this tells us that n cannot be smaller than k. I.e. $k \leq n$ as required. □

Defining dimension

Definition 4.2.5. *A vector space V is* finite dimensional *if it contains a finite spanning list $(v_1, \ldots, v_n)$. If V is not finite dimensional then it is* infinite dimensional.

Theorem 4.2.6. *Let V be a vector space over $\mathbb{R}$. Then:*

1. *If $s = (v_1, \ldots, v_n)$ spans V, then s can be reduced to a basis for V.*
2. *If V is finite dimensional, and if $t = (u_1, \ldots, u_k)$ is linearly independent in V, then t can be extended to a basis for V.*

Proof. For (1), we define a new list s' as follows. First, if $v_1 = 0$, then we remove v_1. Then, for every $i \in \{2, \ldots, n\}$, if $v_i \in \text{span}(v_1, \ldots, v_{i-1})$, we remove v_i. Now, s' still spans V, because we only removed elements in the span of the preceding elements in the list. Also, s' is linearly independent, because if it were not it would be possible to remove an element in the span of preceding elements (by lemma 4.2.3). Since we removed all these elements, this is not possible, so s' must be linearly independent. Thus s' is a basis for V.

For (2), since V is finite dimensional it has a spanning list $(w_1, \ldots, w_m)$. Now, the list $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ also spans V, and so, by (1), reduces to a basis for V. The process defined in the proof of (1) does not remove any elements of t, as t is linearly independent, so the resulting list extends t as required. □

Corollary 4.2.7. *Every finite dimensional vector space has a basis.*

Proof. Just reduce the finite spanning list to a basis. □

It is also true (in classical mathematics), that every *infinite* dimensional vector space also has a basis, but this proof is more difficult, and involves using an infinite choice principle.

Proposition 4.2.8. *If V is a finite vector space then every basis for V has the same length.*

Proof. Let s and t be bases for V. Then, as s is linearly independent and t spans V, by proposition 4.2.4, we must have $|s| \leq |t|$. But t is also linearly independent, and s also spans V, so by the same proposition we also have $|t| \leq |s|$. So $|s| = |t|$ as claimed. □

In view of corollary 4.2.7 and proposition 4.2.8, we can define the basis of a finite dimensional vector space in terms of the size of it's possible bases.

Definition 4.2.9. *If V is a finite dimensional vector space, then we define the* dimension *of V to be the size of its bases. We use* $\dim(V)$ *to denote the dimension of V.*

Example 4.2.10.

1. *The vectors* $(1,0,0)$, $(0,1,0)$ *and* $(0,0,1)$ *provide a basis for* $\mathbb{R}^3$. *So* $\dim(\mathbb{R}^3) = 3$.

2. *The vectors* $(2,0,1)$, $(2,3,0)$ *and* $(0,6,-1)$ *also provide a basis for* $\mathbb{R}^3$.

3. *The vectors* $(1,2,3)$, $(-1,-1,0)$, $(1,1,1)$ *and* $(3,-2,0)$ *must be linearly dependent in* $\mathbb{R}^3$, *because* $\dim(\mathbb{R}^3) = 3$, *and so every linearly independent list must have length at most 3 (by proposition 4.2.4).*

4. *The vectors,* $1, x, x^2, x^4, \ldots$ *provide a basis for* $\mathbb{R}[x]$. *As no finite list spans* $\mathbb{R}[x]$, *it follows that* $\mathbb{R}[x]$ *is infinite dimensional.*

It turns out that in a finite dimensional vector space, lists that are spanning/linearly independent must be bases if they are the right size.

Theorem 4.2.11. *Let* V *be a finite dimensional vector space. Then:*

1. *If* s *is a spanning list for* V *and* $|s| = \dim(V)$ *then* s *is a basis for* V.

2. *If* t *is a linearly independent list in* V *and* $|t| = \dim(V)$ *then* t *is a basis for* V.

Proof. For (1), if s spans V then, by theorem 4.2.6, s can be reduced to a basis, s', for V. By proposition 4.2.8 we must have $|s'| = \dim(V) = |s|$, so s' must be equal to s, and thus s is a basis for V as required.

For (2), if t is linearly independent, then, again by theorem 4.2.6, t can be extended to a basis, t'. for V. As in part (1) we have $|t'| = \dim(V) = |t|$, so t is indeed a basis for V. □

Dimension and subspaces As we would expect, subspaces of finite dimensional vector spaces are themselves finite dimensional. In fact, their dimension can be at most as large as the dimension of the original space. Which is also what we would expect.

Proposition 4.2.12. *Every subspace of a finite dimensional vector space is finite dimensional.*

Proof. Let V be a finite dimensional vector space and let U be a subspace of V. If $U = \{0\}$ then the empty list spans U, so there is nothing to do. If $U \neq \{0\}$ then we construct a basis for U by recursion as follows:

- Since $U \neq \{0\}$ we can choose $v_1 \in U \setminus \{0\}$. Define $s_1 = (v_1)$.
- Given linearly independent $s_i = (v_1, \ldots, v_i)$ in U, if s_i does not span U then there is $v_{i+1} \in U \backslash \text{span}(s_i)$. In this case define $s_{i+1} = (v_1, \ldots, v_i, v_{i+1})$.

Clearly s_i is linearly independent for all i. Moreover, s_i can be no longer than the dimension of U (by proposition 4.2.4), so at some point the process must terminate. That is, there is k such that $U = \text{span}(S_k)$, and then s_k is a finite basis for U as required. □

Corollary 4.2.13. *If V is a finite dimensional vector space and U is a subspace of V, then* $\dim(U) \leq \dim(V)$.

Proof. Let $t = (v_1, \ldots, v_n)$ be a basis for V, and, using proposition 4.2.12, let $s = (u_1, \ldots, u_k)$ be a basis for U. Then s is linearly independent in V, and t spans V, so $|s| \leq |t|$ by proposition 4.2.4. Thus $\dim(U) \leq \dim(V)$ as claimed. □

Every non-trivial subspace of a vector space can be 'extended' via a direct sum to the whole space.

Proposition 4.2.14. *Let V be a finite dimensional vector space, and let U be a subspace of V. Then there is a subspace W of V such that $V = U \oplus W$.*

Proof. Let $s = (u_1, \ldots, u_k)$ be a basis for U. Then s is linearly independent in V, so, by theorem 4.2.6, s can be extended to a basis $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ for V. We define W to be $\text{span}(w_1, \ldots, w_m)$.

To see that $V = U \oplus W$ we just have to check that $V = U + W$, and $U \cap W = \{0\}$ (using lemma 4.1.14). Now, that $V = U + W$ follows immediately from the fact that $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ spans V. That $U \cap W = \{0\}$ follows from the fact that $(u_1, \ldots, u_k)$ is basis for U, $(w_1, \ldots, w_m)$ is a basis for W, and $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ is linearly independent. To see this, suppose $v \in U \cap W$. Then $v = a_1u_1 + \ldots + a_ku_k$ and $v = b_1w_1 + \ldots + b_mw_m$. So $0 = a_1u_1 + \ldots + a_ku_k - b_1w_1 - \ldots - b_mw_m$, and so $a_1 = \ldots = a_k = b_1 = \ldots = b_m = 0$, by linear independence of $(u_1, \ldots, u_k, w_1, \ldots, w_m)$. Thus $v = 0$. □

The next result is a bit like the inclusion-exclusion principal for counting the size of the union of two finite sets.

Proposition 4.2.15. *Let V be a finite dimensional vector space, and let U and W be subspaces of V. Then* $\dim(U + W) = \dim(U) + \dim(W) - \dim(U \cap W)$.

Proof. Let $(v_1, \ldots, v_n)$ be a basis for $U \cap W$ (which is a subspace of V by exercise 1.6). By theorem 4.2.6, we can extend $(v_1, \ldots, v_n)$ to a basis $(u_1, \ldots, u_k, v_1, \ldots, v_n)$ for U, and a basis $(v_1, \ldots, v_n, w_1, \ldots, w_m)$ for W. We claim that

$$s = (u_1, \ldots, u_k, v_1, \ldots, v_n, w_1, \ldots, w_m)$$

is a basis for $U + W$. To see that this is true, note first that s clearly spans $U + W$, so it remains only to show that it is linearly independent.

So, suppose that

$$a_1u_1 + \ldots + a_ku_k + b_1v_1 + \ldots + b_nv_n + c_1w_1 + \ldots + c_mw_m = 0.$$

Then

$$c_1w_1 + \ldots + c_mw_m = -a_1u_1 - \ldots - a_ku_k - b_1v_1 - \ldots - b_nv_n,$$

and it follows that $c_1w_1 + \ldots + c_mw_m \in U \cap W$, so there are $b'_1, \ldots b'_n \in \mathbb{F}$ such that $c_1w_1 + \ldots + c_mw_m = b'_1v_1 + \ldots + b'_nv_n$. In other words,

$$c_1w_1 + \ldots + c_mw_m - b'_1v_1 - \ldots - b'_nv_n = 0.$$

But $(v_1, \ldots, v_n, w_1, \ldots, w_m)$ is a basis for W, and so is linearly independent, and so it follows that $c_i = 0$ for all $i \in \{1, \ldots, m\}$. But then we have

$$a_1 u_1 + \ldots + a_k u_k + b_1 v_1 + \ldots + b_n v_n = 0,$$

and so it follows from the fact that $(u_1, \ldots, u_k, v_1, \ldots, v_n)$ is a basis for U that all the coefficients in this expression are also 0. So s is linearly independent as required. □

Exercises

Exercise 4.9. *For $n \in \mathbb{N}$, define $\mathbb{R}_n[x]$ to be the set of all polynomials of degree at most n. Write down a basis for $\mathbb{R}_6[x]$. Is it possible for a list of 8 polynomials over $\mathbb{R}$ of degree at most 6 to be linearly independent?*

Exercise 4.10. *Let (v_1, v_2, v_3, v_4) be a basis for V. Prove that*

$$(v_1 + v_2, v_2 + v_3, v_3 + v_4, v_4)$$

is also a basis for V.

Exercise 4.11. *Let U and W be subspaces of V and suppose that $V = U \oplus W$. Let $(u_1, \ldots, u_k)$ be a basis for U, and let $(w_1, \ldots, w_m)$ be a basis for W. Prove that $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ is a basis for V.*

Exercise 4.12. *Let U and W be subspaces of $\mathbb{R}^8$, and suppose $\dim(U) = 5$, $\dim(W) = 3$, and $U \cap W = \{0\}$. Prove that $\mathbb{R}^8 = U \oplus W$.*

Exercise 4.13. *Let V be a finite dimensional vector space and $\dim(V) = n > 0$. Show that $V = U_1 \oplus \ldots \oplus U_n$, for some set $\{U_1, \ldots, U_n\}$ of one-dimensional subspaces.*

4.3 Linear maps and matrices

What is a linear map?

Definition 4.3.1. *Let V and W be vector spaces over the same field $\mathbb{F}$. A function $T : V \to W$ is a* linear map *if the following linearity conditions are satisfied:*

1. *$T(u + v) = T(u) + T(v)$ for all $u, v \in V$.*
2. *$T(\lambda v) = \lambda T(v)$ for all $v \in V$ and for all $\lambda \in \mathbb{F}$.*

The definition above is abstract, but hopefully it should be understandable why linear maps are called *linear*. Think of an equation $y = ax$ defining a straight line through the origin in the Euclidean plane (straight lines are, of course, the archetypal linear object). Think of two real numbers x_1 and x_2. Then, at the point $x_1 + x_2$, the value of y is given by $a(x_1 + x_2)$, which is just $ax_1 + ax_2$. Similarly, if b is another real number then the value of y at the

point bx_1 is given by $a(bx_1)$, which is equal to $ba(x_1)$. Linear maps 'behave like' straight lines, in the sense that the value a linear map takes at the sum of two vectors is the same as the sum of the values it takes at each individual vector, and the value it takes at the scalar multiple of a vector is the same as the scalar multiple of the value it takes at that vector.

Definition 4.3.2 ($\mathcal{L}(V,W)$)**.** *If V and W are vector spaces over the same field, then we denote the set of all linear maps from V to W by $\mathcal{L}(V,W)$.*

Example 4.3.3.

1. *We've already seen the example of a straight line of form $y = ax$. This can be thought of as a linear map from $\mathbb{R}$ (considered as a vector space over itself) to itself.*

2. *More generally, we could replace $\mathbb{R}$ with $\mathbb{F}$ and the previous example would still hold.*

3. *For any vectors spaces V and W over the same field, there is a special linear map, the zero map $0 : V \to W$, given by $0(v) = 0$ for all $v \in V$.*

4. *For any vector space V there is a special linear map, the identity map $I : V \to V$ for V, given by $I(v) = v$ for all $v \in V$.*

5. *Remember that $\mathbb{R}[x]$ stands for the vector space of polynomials over $\mathbb{R}$ with the variable x. The map $D : \mathbb{R}(x) \to \mathbb{R}(x)$ defined by taking first derivatives is a linear map. The same is true if we restrict to $\mathbb{R}_n(x)$ (recall exercise 2.1 for this notation).*

Lemma 4.3.4. *If $T : V \to W$ is a linear map, then $T(0) = 0$.*

Proof. $0 = 0 + 0$, so, by linearity,

$$T(0) = T(0 + 0) = T(0) + T(0),$$

and so by subtracting $T(0)$ from both sides we see $T(0) = 0$ as required. □

Example 4.3.3 has some important linear maps, but given arbitrary vector spaces V and W (over the same field), how can we define a linear map T between them? Do we have to specify the value $T(v)$ for every vector $v \in V$? Fortunately the answer to this is no, at least, it is no so long as we know a basis for V. The next result makes this more precise.

Proposition 4.3.5. *Let V be a finite dimensional vector space over $\mathbb{F}$, and let $(v_1, \ldots, v_n)$ be a basis for V. Let W be another vector space over $\mathbb{F}$. Then for any $w_1, \ldots, w_n \in W$, there is a unique linear map $T : V \to W$ such that $T(v_i) = w_i$ for all $i \in \{1, \ldots, n\}$.*

Proof. By the definition of a basis, given an element $v \in V$ we have $v = a_1 v_1 + \ldots + a_n v_n$ for some $a_1, \ldots, a_n \in \mathbb{F}$. Then the requirement that T is linear tells us what value T must take at on v. I.e.

$$T(v) = a_1 T(v_1) + \ldots + a_n T(v_n) = a_1 w_1 + \ldots + a_n w_n.$$

It's straightforward to show that T defined in this way is a linear map (we just have to check the two conditions from definition 4.3.1). □

Proposition 4.3.5 tells us that a linear map from a finite dimensional vector space V is completely determined by what it does to the basis vectors of V (this is also true for infinite dimensional vector spaces, and the proof is essentially the same). More than that, it tells us that a linear map can take any values on the basis vectors of V. We could sum this up by saying that, if $\{v_1, \ldots, v_n\}$ provides a basis for V, then the set of linear maps $\mathcal{L}(V, W)$ is in bijection with the set of functions from $\{v_1, \ldots, v_n\}$ to W.

Definition 4.3.6. *If $S \in \mathcal{L}(U, V)$, and $T \in \mathcal{L}(V, W)$, then it's easy to check that the composition $TS \in \mathcal{L}(U, W)$, where TS is defined by $TS(u) = T(S(u))$ for all $u \in U$.*

Null spaces

Definition 4.3.7. *Given $T \in \mathcal{L}(V, W)$, the* null space *of T, denoted* $\operatorname{null} T$*, is defined by*

$$\operatorname{null} T = \{v \in V : T(v) = 0\}.$$

Lemma 4.3.8. $\operatorname{null} T$ *is a subspace of V.*

Proof. This is exercise 4.16. □

Lemma 4.3.9. *Let $T \in \mathcal{L}(V, W)$. Then T is injective if and only if* $\operatorname{null} T = \{0\}$.

Proof. Clearly if T is injective then only 0 can be mapped to 0 by T, so we have the forward implication. For the converse, suppose T is not injective. Then there are $u, v \in V$ with $u \neq v$ and $T(u) = T(v)$. But then by linearity of T we have $T(u - v) = T(u) - T(v) = 0$, and so $u - v \in \operatorname{null} T$. □

The range of a linear map

Definition 4.3.10. *Given $T \in \mathcal{L}(V, W)$, the* range *of T is defined by*

$$\operatorname{ran} T = \{T(v) : v \in V\}.$$

So $\operatorname{ran} T$ is just the range of T, just like we can define the range for any function. What makes $\operatorname{ran} T$ special in the context of vector spaces, is the following result.

Lemma 4.3.11. *If $T \in \mathcal{L}(V, W)$ then* $\operatorname{ran} T$ *is a subspace of W.*

Proof. We just need to check the conditions of definition 4.1.8 are satisfied:

1. Since $0 = T(0)$ we have $0 \in \operatorname{ran} T$.
2. $T(u) + T(v) = T(u + v)$, so $\operatorname{ran} T$ is closed under vector addition.
3. $\lambda T(v) = T(\lambda v)$, so $\operatorname{ran} T$ is closed under scalar multiplication.

□

The Rank-Nullity Theorem The Rank-Nullity Theorem is the first big result in linear algebra. In [1] it is referred to as the *fundamental theorem of linear maps*.

Theorem 4.3.12 (Rank-Nullity)**.** *Let V be finite dimensional, and let $T \in \mathcal{L}(V, W)$. Then* $\operatorname{ran} T$ *is finite dimensional, and*

$$\dim V = \dim \operatorname{ran} T + \dim \operatorname{null} T.$$

In other words, the dimension of V is equal to the rank of T plus the nullity of T.

Proof. Let $(u_1, \ldots, u_m)$ be a basis for $\operatorname{null} T$. We know such a basis exists as V is finite dimensional and $\operatorname{null} T$ is a subspace of V. By theorem 4.2.6 we can extend $(u_1, \ldots, u_m)$ to a basis $(u_1, \ldots, u_m, v_1, \ldots, v_n)$ for V. We complete the proof by showing that $(T(v_1), \ldots, T(v_n))$ is a basis for $\operatorname{ran} T$.

First we check that $(T(v_1), \ldots, T(v_n))$ is linearly independent. So suppose $0 = a_1 T(v_1) + \ldots + a_n T(v_n)$. Then $T(a_1 v_1 + \ldots + a_n v_n) = 0$, by the linearity of T. But this means $a_1 v_1 + \ldots + a_n v_n \in \operatorname{null} T$, and so there are $b_1, \ldots, b_m$ with $a_1 v_1 + \ldots + a_n v_n = b_1 u_1 + \ldots + b_m u_m$. Rearranging this we have

$$a_1 v_1 + \ldots + a_n v_n - b_1 u_1 - \ldots - b_m u_m = 0,$$

and as $(u_1, \ldots, u_m, v_1, \ldots, v_n)$ is a basis for V (and so is linearly independent), the only way this can happen is if

$$a_1 = \ldots = a_n = b_1 = \ldots = b_m = 0.$$

In particular we have $a_1 = \ldots = a_n = 0$, and so $(T(v_1), \ldots, T(v_n))$ is indeed linearly independent.

To complete the proof we must show that $(T(v_1), \ldots, T(v_n))$ spans $\operatorname{ran} T$. Now, if $w \in \operatorname{ran} T$ then, by definition of range, there must be $v \in V$ with $w = T(v)$. Since $(u_1, \ldots, u_m, v_1, \ldots, v_n)$ is a basis for V, it follows that there must be $a_1, \ldots, a_m, b_1, \ldots, b_n$ with

$$v = a_1 u_1 + \ldots + a_m u_m + b_1 v_1 + \ldots + b_n v_n.$$

Since $u_i \in \operatorname{null} T$ for all $i \in \{1, \ldots, m\}$, and so $T(u_i) = 0$, we have

$$w = T(v) = b_1 T(v_1) + \ldots + b_n T(v_n).$$

So $w \in \operatorname{span}(T(v_1), \ldots, T(v_n))$ as required. □

The Rank-Nullity theorem gets its name from the following old definitions:

Definition 4.3.13. *Given $T \in \mathcal{L}(V,W)$, the* rank *of T is the dimension of* $\operatorname{ran} T$*, and the* nullity *of T is the dimension of* $\operatorname{null} T$*.*

A review of matrix algebra An $m \times n$ matrix over a field $\mathbb{F}$ is an array of elements of $\mathbb{F}$. We can express matrices explicitly, using the following form:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix}$$

Here the element a_{ij} is the one in the ith row and the jth column. We sometimes use the shorthand (a_{ij}) to express a matrix of the form above.

Given an $m \times n$ matrix A over $\mathbb{F}$, and $\lambda \in \mathbb{F}$, we define the scalar product λA to be

$$\begin{bmatrix} \lambda a_{11} & \dots & \lambda a_{1n} \\ \lambda a_{21} & \dots & \lambda a_{2n} \\ \vdots & \ddots & \vdots \\ \lambda a_{m1} & \dots & \lambda a_{mn} \end{bmatrix}$$

Given $m \times n$ matrices $A = (a_{ij})$ and $B = (b_{ij})$ over the same field $\mathbb{F}$, we define the sum $A + B$ to be

$$\begin{bmatrix} a_{11} + b_{11} & \dots & a_{1n} + b_{1n} \\ a_{21} + b_{21} & \dots & a_{2n} + b_{2n} \\ \vdots & \ddots & \vdots \\ a_{m1} + b_{m2} & \dots & a_{mn} + b_{mn} \end{bmatrix}$$

Given an $m \times n$ matrix $A = (a_{ij})$ and an $n \times p$ matrix $B = (b_{jk})$, both over $\mathbb{F}$, we define the matrix product AB to be the $m \times p$ matrix (c_{ik}), where for each $i \in \{1, \dots m\}$ and $k \in \{1, \dots, p\}$, the entry c_{ik} is defined by

$$c_{ik} = \sum_{j=1}^{n} a_{ij} b_{jk}.$$

In other words, the element c_{ik} is defined using the ith row of A, and the kth column of B.

In particular, if A is an $m \times n$ matrix, and v is a $n \times 1$ matrix (so v a column vector in $\mathbb{F}^n$), then the product Av is calculated using

$$Av = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} = \begin{bmatrix} b_1 a_{11} + b_2 a_{12} + \ldots + b_n a_{1n} \\ b_1 a_{21} + b_2 a_{22} + \ldots + b_n a_{2n} \\ \vdots \\ b_1 a_{m1} + b_2 a_{m2} + \ldots + b_n a_{mn} \end{bmatrix}$$

Matrix multiplication as defined here is quite mysterious, and seemingly arbitrary. However, as we shall soon see, this definition is actually extremely natural.

Matrices and linear maps From the definitions of matrix addition and scalar multiplication above, we see that every $m \times n$ matrix over $\mathbb{F}$ defines a linear map from $\mathbb{F}^n$ to $\mathbb{F}^m$. That is, an $m \times n$ matrix over $\mathbb{F}$ takes a vector from $\mathbb{F}^n$ and transforms it into a vector from $\mathbb{F}^m$. Moreover, this transformation is linear. The correspondence between linear maps and matrices actually goes both ways, in that every linear map between finite dimensional vector spaces can be represented by a matrix, as we now explain.

Let $T \in \mathcal{L}(V, W)$, and suppose V and W are both finite dimensional. Let $(v_1, \ldots, v_n)$ be a basis for V, and let $(w_1, \ldots, w_m)$ be a basis for W. By proposition 4.3.5, the map T is defined by what it does to $v_1, \ldots, v_n$. Moreover, as $(w_1, \ldots, w_m)$ is a basis for W, each $T(v_j)$ can be written as a linear combination of elements of $\{w_1, \ldots, w_m\}$. In other words, for each $j \in \{1, \ldots, n\}$, we have

$$T(v_j) = a_{1j}w_1 + \ldots + a_{mj}w_m. \qquad (\dagger)$$

Let $A = (a_{ij})$ be the matrix defined using the a_{ij} defined in (†), with $i \in \{1, \ldots, m\}$.

$$\begin{bmatrix} a_{11} & \ldots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \ldots & a_{mn} \end{bmatrix}$$

Now, think of the column vector of $\mathbb{F}^n$ that has 1 in it's jth place and 0 everywhere else. What happens when we multiply this vector with A? The definition of matrix multiplication says the result is given by

$$\begin{bmatrix} a_{11} & \ldots & a_{1n} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ a_{m1} & \ldots & a_{mn} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix}$$

Now, we can interpret the vector

$$\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

as the element v_j of V, and we can interpret the vector

$$\begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix}$$

as the element of W defined by $a_{1j}w_1 + \ldots + a_{mj}w_m$, which is $T(v_j)$, because of how we defined the a_{ij} values. According to this translation, the matrix multiplication we have just performed says that when v_j is transformed by A the result equals $T(v_j)$. This is true for all $j \in \{1, \ldots, n\}$, so the matrix A corresponds to the action of T on every basis vector v_i. From this we see that A corresponds to the action of T on every element of V, because matrix multiplication has a distributivity property and T is linear, and so

$$A(au + bv) = aAu + bAv = aTu + bTv = T(au + bv)$$

for any vectors $u, v \in V$ and scalars $a, b \in \mathbb{F}$. In particular it's true for basis vectors of V, and all vectors in V are linear combinations of basis vectors.

In other words, the matrix A represents T with respect to the translation given by the choice of bases for V and W. Note that if we chose a different basis for V or W then we would usually get a different matrix corresponding to T, because the a_{ij} values depend on the basis we are using.

This explains why matrix multiplication is defined the way it is. Matrices are motivated by a desire to represent linear maps. This means that they must multiply vectors of form

$$\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

in a very specific way. Moreover, because every column vector can be thought of as a linear combination of vectors in this form, the linearity of the transformation determines how A must act when multiplying column vectors. We can think of the product AB of two matrices as being the list of vectors we get from applying the transformation defined by A to each of the columns of B. From this perspective, the jth column of AB is the result of applying A to the jth column of B.

A nice property of correspondence between matrices and linear maps is that it also extends to compositions of linear maps.

Proposition 4.3.14. *Let $S \in \mathcal{L}(U, V)$ and let $T \in \mathcal{L}(V, W)$. Let $(u_1, \ldots, u_n)$, $(v_1, \ldots, v_m)$ and $(w_1, \ldots, w_p)$ be bases for U, V and W respectively. Suppose that B is the matrix of T with respect to $(v_1, \ldots, v_m)$ and $(w_1, \ldots, w_p)$, and that A is the matrix of S with respect to $(u_1, \ldots, u_n)$ and $(v_1, \ldots, v_m)$. Then BA is the matrix of TS with respect to $(u_1, \ldots, u_n)$ and $(w_1, \ldots, w_p)$.*

Proof. Exercise 4.17. □

What is a linear map, really? We should understand linear maps as linear transformations of space. In other words, transformations of space that keep straight lines straight. The connection between linear maps and matrices is helpful for this. Think of Euclidean space $\mathbb{R}^3$. The vectors $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ define a cube of volume one with one vertex at the origin (the point $(0,0,0)$) in $\mathbb{R}^3$. This cube is known as a *unit cube*.

If $A = (a_{ij})$ is a 3×3 matrix, then the action of A on the vectors $(1,0,0)$, $(0,1,0)$ and $(0,0,1)$ produces three vectors (a_{11}, a_{21}, a_{31}), (a_{12}, a_{22}, a_{32}) and (a_{13}, a_{23}, a_{33}). These vectors also define a shape in Euclidean space. This shape is the result of transforming the unit cube by the transformation defined by A. Now, the linearity of A means it can stretch vectors, and change their directions, but it can't bend them.

Example 4.3.15. *Let A be the real valued matrix*

$$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Think of A as a linear transformation of the Euclidean plane. What does A do to $(1,0)$? Well, according to the definition of matrix multiplication, A takes $(1,0)$ to $(0,1)$, and A takes $(0,1)$ to $(-1,0)$. If $(1,0)$ and $(0,1)$ have their usual meanings as vectors in $\mathbb{R}^2$, then this corresponds to an anticlockwise rotation by $\frac{\pi}{2}$ radians ($90°$).

A comment on determinants Here we assume the reader has already seen a definition for a determinant, but would like to understand how it fits into the framework of linear maps between vector spaces.

Definition 4.3.16. *A linear map $T \in \mathcal{L}(V,W)$ is* invertible *if there is a map $T^{-1} \in \mathcal{L}(W,V)$ such that the map $T^{-1}T$ is the identity map on V, and the map TT^{-1} is the identity map on W.*

Appealing to the correspondence between linear maps and matrices, a linear map T is invertible if and only if the corresponding matrix is invertible. Thinking about a linear map as a transformation of space, such a map $T : V \to W$ should be invertible so long as no information is 'lost' during the transformation. With vector spaces, this 'information loss' happens when $\dim \operatorname{ran} T < \dim V$. In other words, when the effect of T is to compress V into a space with lower dimension.

This brings us to the determinant. One method that students are often taught for checking whether a matrix is invertible or not is to calculate its determinant and see whether it is 0. We are not going to go into detail about how determinants are calculated here, but we will try to build some intuition about what the determinant represents.

In the previous section, we thought about a linear map as a transformation of space, and we tried to understand this by imagining the effect of such a map on

the 'unit cube' in $\mathbb{R}^3$. Building on this idea, the determinant of a matrix representing a transformation of $\mathbb{R}^3$ corresponds to the volume of the unit cube after being transformed. So the determinant of a 3×3 matrix being zero corresponds to the associated linear map 'compressing' $\mathbb{R}^3$ into a lower dimensional space, thus losing information. Now, the determinant can be positive or negative, so it actually gives us more information than just the volume of the transformed unit cube (what is known as a *signed volume*), but the absolute value of the determinant is always equal to this volume.

This also applies to $n \times n$ matrices for all $n \geq 1$. We just have to be comfortable generalizing the idea of 'space' and 'volume' into higher dimensions.

Exercises

Exercise 4.14. *Let $b, c \in \mathbb{R}$. Define $T : \mathbb{R}^3 \to \mathbb{R}^2$ by*

$$T(x, y, z) = (2x - 4y + 3z + b, 6x + cxyz).$$

Prove that T is linear if and only if $b = c = 0$.

Exercise 4.15. *Let $T \in \mathcal{L}(V, W)$. Let $v_1, \ldots, v_n \in V$ and suppose that*

$$(T(v_1), \ldots, T(v_n))$$

is linearly independent in W. Prove that $(v_1, \ldots, v_n)$ is linearly independent in V.

Exercise 4.16. *Prove lemma 4.3.8.*

Exercise 4.17. *Prove proposition 4.3.14.*

Exercise 4.18. *Let $T \in \mathcal{L}(V, W)$, and suppose both V and W are finite dimensional. Prove that, whatever the choice of bases for V and W, the matrix of T with respect to these bases must have at least $\dim \operatorname{ran} T$ entries that are not equal to 0.*

4.4 Inner products on real vector spaces

Here we will work with vector spaces over $\mathbb{R}$. Everything we do here can be adapted for $\mathbb{C}$, but at the cost of slightly more complicated definitions.

What is an inner product? An inner product is a generalization of the idea of a dot product. For example, in $\mathbb{R}^3$, we have $(a, b, c) \cdot (d, e, f) = ad + be + cf$. So an inner product is a function that takes pairs of vectors to a value in the underlying field (e.g. a real number in the case of a vector space over $\mathbb{R}$). This turns out to be useful, because many geometric ideas for Euclidean spaces can be described using dot products, and so inner products provide a way to 'do' geometry in more general vector spaces. In other words, if a vector space has an inner product, then our geometric intuitions apply to it in some sense. This section aims to clarify this statement. First, the main definition:

Definition 4.4.1. *Let V be a vector space over $\mathbb{R}$. An* inner product *for V is a function that takes a pair $(u, v) \in V^2$ to a value $\langle u, v\rangle \in \mathbb{R}$, satisfying the following properties:*

1. $\langle v, v\rangle \geq 0$ *for all $v \in V$ (positivity).*
2. $\langle v, v\rangle = 0 \iff v = 0$ *(definiteness).*
3. $\langle u + v, w\rangle = \langle u, w\rangle + \langle v, w\rangle$ *for all $u, v, w \in V$ (additivity in first slot).*
4. $\langle \lambda u, v\rangle = \lambda\langle u, v\rangle$ *for all $\lambda \in \mathbb{R}$ and for all $u, v \in V$ (homogeneity in first slot).*
5. $\langle u, v\rangle = \langle v, u\rangle$ *for all $u, v \in V$ (symmetry).*

Example 4.4.2.

1. *It's easy to check that the dot product as it is usually defined is indeed an inner product.*
2. *It can be shown that the set of continuous real valued functions on the interval $[-1, 1]$ is a vector space over $\mathbb{R}$. We can define an inner product on this space using $\langle f, g\rangle = \int_{-1}^{1} f(x)g(x)dx$.*

Definition 4.4.3. *A vector space with an inner product is called an* inner product space.

Proposition 4.4.4. *The following properties hold in all real inner product spaces:*

1. *Given $v \in V$, we can define a linear map $\langle -, v\rangle : V \to \mathbb{R}$ by defining $\langle -, v\rangle(u) = \langle u, v\rangle$ for all $u \in V$.*
2. *$\langle v, 0\rangle = \langle 0, v\rangle = 0$ for all $v \in V$.*
3. *$\langle u, v + w\rangle = \langle u, v\rangle + \langle u, w\rangle$ for all $u, v, w \in V$.*
4. *$\langle u, \lambda v\rangle = \lambda\langle u, v\rangle$ for all $\lambda \in \mathbb{R}$ and for all $u, v \in V$*

Proof.

1. Given $u_1, u_2 \in V$ we have $\langle u_1 + u_2, v\rangle = \langle u_1, v\rangle + \langle u_2, v\rangle$, by additivity in the first slot. We also have $\langle \lambda u, v\rangle = \lambda\langle u, v\rangle$ by homogeneity in the first slot.
2. That $\langle 0, v\rangle = 0$ follows from part (1) and the fact that $T(0) = 0$ for all linear maps. We then have $\langle v, 0\rangle = 0$ by symmetry.
3. $\langle u, v + w\rangle = \langle v + w, u\rangle$ by symmetry, and then the result follows from additivity and symmetry again.
4. Symmetry and homogeneity in the first slot.

□

Norms In every real inner product space V we can calculate the value of $\langle v, v\rangle$, which by the definition of 'inner product' must be non-negative. This inspires the following definition.

Definition 4.4.5. *If V is an inner product space, then given $v \in V$, the* norm *of v, $||v||$, is defined by*

$$\|v\| = \sqrt{\langle v, v\rangle}.$$

Example 4.4.6. *In $\mathbb{R}^2$ with the usual dot product, the norm of a vector (a, b) is $\sqrt{(a^2 + b^2)}$. I.e., it is the Euclidean distance of the point (a, b) from the origin.*

Proposition 4.4.7. *The following hold for all real inner product spaces V, and for all $v \in V$:*

1. $\|v\| = 0 \iff v = 0$.
2. $\|\lambda v\| = |\lambda|\|v\|$ *for all* $\lambda \in \mathbb{R}$.

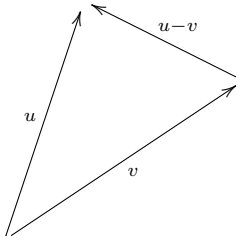
Proof. (1) follows immediately from definiteness of the inner product. (2) follows from homogeneity in the first slot and proposition 4.4.4(4). □

Proposition 4.4.8. *Given $u, v \in \mathbb{R}^2 \setminus \{0\}$, we have*

$$\langle u, v\rangle = \|u\|\|v\| \cos\theta,$$

where θ is the angle between u and v when these are thought of as arrows beginning at the origin.

Proof. Remember that in $\mathbb{R}^2$ the norm of a vector is its length. Consider the picture below.



According to the law of cosines we have

$$\|u - v\|^2 = \|u\|^2 + \|v\|^2 - 2\|u\|\|v\| \cos\theta.$$

Now, $\|u - v\|^2 = \langle u - v, u - v\rangle$, by definition, and

$$\begin{aligned}\langle u - v, u - v\rangle &= \langle u, u - v\rangle - \langle v, u - v\rangle \\ &= \langle u, u\rangle - \langle u, v\rangle - \langle v, u\rangle + \langle v, v\rangle \\ &= \langle u, u\rangle + \langle v, v\rangle - 2\langle u, v\rangle \\ &= \|u\|^2 + \|v\|^2 - 2\langle u, v\rangle.\end{aligned}$$

Putting this together we get $\langle u, v\rangle = \|u\|\|v\| \cos\theta$, which is what we are trying to prove. □

Definition 4.4.9. *If u and v are vectors in an inner product space, then we say u and v are* orthogonal *if $\langle u, v\rangle = 0$.*

Proposition 4.4.8 tells us that two non-zero vectors in $\mathbb{R}^2$ are orthogonal if and only if the cosine of the angle between them is 0. In other words, if and only if they are perpendicular. You can think of 'being orthogonal' as a generalization of the concept of 'being perpendicular'.

Lemma 4.4.10.

1. *0 is orthogonal to everything.*
2. *0 is the only thing that is orthogonal to itself.*

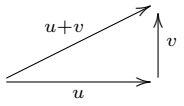
Proof. These follows from proposition 4.4.4(2) and the definiteness of inner products, respectively. □

Geometry in inner product spaces Since inner product spaces generalize the familiar dot product on $\mathbb{R}^n$, we should expect to be able to find generalized versions of familiar results from plane geometry. This is indeed the case, as we demonstrate in this section.

Proposition 4.4.11 (Pythagoras for inner product spaces)**.** *If u and v are vectors in a real inner product space, then*

$$\|u\|^2 + \|v\|^2 = \|u+v\|^2 \iff u \text{ and } v \text{ are orthogonal.}$$

Proof.

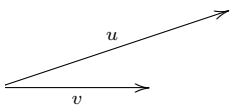


We have

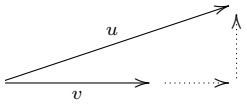
$$\begin{aligned}\|u+v\|^2 &= \langle u+v, u+v\rangle \\ &= \langle u,u\rangle + \langle u,v\rangle + \langle v,u\rangle + \langle v,v\rangle \\ &= \|u\|^2 + \|v\|^2 + 2\langle u,v\rangle.\end{aligned}$$

So $\|u\|^2 + \|v\|^2 = \|u+v\|^2$ if and only if $\langle u, v\rangle = 0$. I.e. if and only if u and v are orthogonal. □

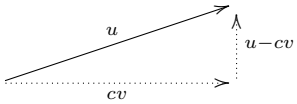
We can think of vectors in a vector space as arrows with a length and direction. For example:



Our geometric intuition says we should be able to turn this into a right angled triangle by drawing some lines. I.e:



In the picture above we have essentially extended v as far as we need, then added a third line. If we use the language of vector spaces, then extending v corresponds to multiplying v by some scalar, c say, to get cv. The associated vector equation is $u = cv + (u - cv)$, as indicated in the diagram below.



In an inner product space, the triangle being 'right angled' corresponds to the vectors v and $(u-cv)$ being orthogonal (i.e. $\langle v, u-cv\rangle = 0$). If our geometric intuition is correct, we should always be able to find a scalar value c such that this is true (so long as u and v are non-zero).

Now, from the properties of the inner product we have

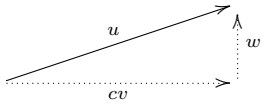
$$\langle v, u - cv\rangle = 0 \iff \langle v, u\rangle - c\|v\|^2 = 0,$$

so we can take

$$c = \frac{\langle v, u\rangle}{\|v\|^2}.$$

We summarize this discussion as the following lemma.

Lemma 4.4.12. *Let V be a real inner product space, let $u, v \in V$ and suppose $v \neq 0$. Then there is $w \in V$ such that $\langle v, w\rangle = 0$, and $u = cv + w$ for some $c \in \mathbb{R}$.*



Proof. Set $c = \frac{\langle v,u\rangle}{\|v\|^2}$ and $w = u - cv$. □

The next result is known as the Cauchy-Schwarz inequality. It is very famous, and useful too. We will go through some applications later, and there are more in the exercises.

Theorem 4.4.13 (Cauchy-Schwarz)**.** *Let V be an inner product space, and let $u, v \in V$. Then*

$$|\langle u, v\rangle| \leq \|u\|\|v\|.$$

Moreover, we have equality if and only if u is a scalar multiple of v or vice versa.

Proof. If v is zero, then everything is zero, and there is nothing to do. So suppose now that $v \neq 0$. Using lemma 4.4.12 write $u = cv + w$. Since w is orthogonal to cv we can appeal to proposition 4.4.11 to get

$$\|u\|^2 = c^2\|v\|^2 + \|w\|^2.$$

The discussion above tells us that $c = \frac{\langle v,u\rangle}{\|v\|^2}$, so this gives us

$$\|u\|^2 = \frac{\langle v,u\rangle^2}{\|v\|^4}\|v\|^2 + \|w\|^2.$$

As $\|w\|^2 \geq 0$ this implies

$$\|u\|^2 \geq \frac{\langle v,u\rangle^2}{\|v\|^4}\|v\|^2,$$

and so

$$\|u\|\|v\| \geq |\langle u,v\rangle|$$

as required.

Now, examining the argument we have just made we see that $|\langle u,v\rangle| = \|u\|\|v\|$ if and only if $\|w\| = 0$, which happens if and only if $w = 0$. I.e. if $u = cv$. □

Example 4.4.14. *Let $x_1, \ldots, x_n, y_1, \ldots, y_n \in \mathbb{R}$. Then, using Cauchy-Schwarz we have*

$$|x_1y_1 + \ldots + x_ny_n|^2 \leq (x_1^2 + \ldots + x_n^2)(y_1^2 + \ldots + y_n^2).$$

It is a basic fact of Euclidean geometry that the length of a side of a triangle is less than the sum of the lengths of the other two sides. Again, we expect this geometric fact to generalize to inner product spaces, and once again it does.

Proposition 4.4.15 (Triangle inequality)**.** *Let V be a real inner product space, and let $u, v \in V$. Then*

$$\|u + v\| \leq \|u\| + \|v\|.$$

Moreover, we have equality if and only if u is a scalar multiple of v or vice versa.

Proof. Appealing to Cauchy-Schwarz for the inequality marked $*$ we have

$$\begin{aligned}
\|u+v\|^2 &= \langle u+v, u+v\rangle \\
&= \langle u,u\rangle + \langle v,u\rangle + \langle u,v\rangle + \langle v,v\rangle \\
&= \|u\|^2 + \|v\|^2 + 2\langle u,v\rangle \\
&\overset{*}{\leq} \|u\|^2 + \|v\|^2 + 2\|u\|\|v\| \\
&= (\|u\| + \|v\|)^2,
\end{aligned}$$

so $\|u + v\| \leq \|u\| + \|v\|$ as claimed.

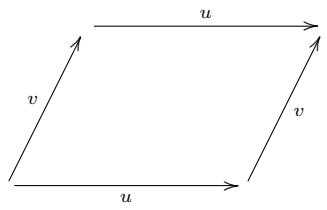
Now, examining the argument we have just made, we see have equality if and only if $\|u\|\|v\| = \langle u,v\rangle$, and from Cauchy-Schwarz we know this happens if and only if one of u or v is a scalar multiple of the other. □

Note that our proof of proposition 4.4.15 assumes that V is a real inner product space, but the result is also true for complex inner product spaces, by a similar argument.

Now lets use what we have proved about inner product spaces to prove a less obvious fact about plain geometry.

Proposition 4.4.16. *In a parallelogram, the sum of the squares of the lengths of the diagonals equals the sum of the squares of the sides.*

Proof. Expressed in terms of vectors, a parallelogram has form



and the diagonals are given by $u - v$ and $u + v$. Now

$$\begin{aligned}\|u+v\|^2 + \|u-v\|^2 &= \langle u+v, u+v\rangle + \langle u-v, u-v\rangle \\ &= \|u\|^2 + \|v\|^2 + 2\langle u, v\rangle + \|u\|^2 + \|v\|^2 - 2\langle u, v\rangle \\ &= 2(\|u\|^2 + \|v\|^2),\end{aligned}$$

which is what we want. □

The identity

$$\|u+v\|^2 + \|u-v\|^2 = 2(\|u\|^2 + \|v\|^2)$$

is called the *parallelogram equality.*

Exercises

Exercise 4.19. *Let a, b, c, d be positive real numbers. Use Cauchy-Schwarz (theorem 4.4.13) to prove that*

$$16 \leq (a+b+c+d)(\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}).$$

Exercise 4.20. *Let $x_1, \ldots, x_n \in \mathbb{R}$. Prove that $(x_1 + \ldots + x_n)^2 \leq n(x_1^2 + \ldots + x_n^2)$.*

Exercise 4.21. *Is there an inner product on $\mathbb{R}^2$ such that the associated norm is given by $\|(x, y)\| = \max\{x, y\}$? Provide a proof for your answer.*

Exercise 4.22. *Let V be a real inner product space.*

(a) Prove that $\langle u+v, u-v\rangle = \|u\|^2 - \|v\|^2$ for all $u, v \in V$.

(b) A rhombus is a parallelogram whose four sides all have equal length. Prove that the diagonals of a rhombus are orthogonal to each other.

(c) Prove that

$$\langle u, v\rangle = \frac{\|u+v\|^2 - \|u-v\|^2}{4}$$

for all $u, v \in V$.

4.5 Further reading

These notes are heavily influenced by [1]. For a completely different approach to the subject, the videos by 3Blue1Brown are highly regarded, though I haven't watched many of them. You can find them here. Another source is [10], which is freely available from the author's website.

5 Counting

5.1 Cardinal numbers

Set theory. In mathematics, we often want to group objects together to form a collection known as a *set*. For example, the set of natural numbers, the set of propositional variables, a set of axioms, and so on. It seems intuitively obvious what a set is, and it's hard to define it in English without using a word that is essentially equivalent (such as *collection*).

Most programming languages implement a *set* data structure. These are collections that are unordered, contain no duplicates, and are defined completely by the things they contain. In other words, two sets are equal if they contain exactly the same things. There is a special set called *the empty set* that contains nothing. This is denoted by the symbol $\emptyset$. Mathematicians think of sets like this too, and for finite sets this is all we really need to know. For infinite sets, we need to be a bit more careful, as we shall see.

The modern subject of set theory emerged in the 19th century from the work of Georg Cantor. As part of his work on trigonometric series (roughly speaking, infinite sums of sin and cos terms), Cantor found it necessary to take seriously the sizes of different infinite sets. Up till this point, mathematicians had assumed that all infinite sets were, essentially, 'the same size'. Of course, the concept of size as associating a set with a natural number telling us how many things are in it doesn't make sense for infinite sets, so when mathematicians 'assumed infinite sets were the same size' we should take that to mean that they didn't see how the notion of size could be extended beyond 'not finite' to meaningfully distinguish between infinite sets. Cantor realized that this was not true, and defined a concept of 'size' for sets which makes intuitive sense, agrees with the obvious concept of size for finite sets, and, crucially, applies just as well to infinite sets. Starting from this new definition, which we will see soon, Cantor was able to prove many surprising results about the sizes of many familiar infinite sets. Despite his revolutionary work on what he came to call *transfinite numbers*, Cantor left the basic notion of a set essentially undefined. This is what we know today as *naive set theory*. This naive treatment of set theory is perhaps understandable, given what we have said about the intuitive nature of the concept of a collection, but this intuitive nature hides some deep and troubling paradoxes. These paradoxes started with the intuitive concept of a set and showed in various ways that if you allow anything you want to form a set, then you will end up proving something impossible, a contradiction. We will illustrate these paradoxes with a single, famous, example.

In the late 19th century, some mathematicians, particularly Gottlob Frege and Bertrand Russell, tried to use the naive set concept to formalize mathematical reasoning. In this theory, a set is just the collection formed by taking every object that satisfies some property. So, for example, we can form the set of every collection with exactly three members. That is,

$$\{X : |X| = 3\}.$$

From here, the idea was that the number three could be *defined* as the set of every collection with exactly three members. The underlying assumption, which Frege and Russell took to be a fact of logic, was that every abstract property could be extended to a set, by taking all the things that satisfy the property. The problem with this assumption is that it leads to a contradiction.

Example 5.1.1 (Russell's paradox)**.** *In naive set theory, every property can be extended to a set, so it is possible for sets to be members of themselves. For example, according to naive set theory, the set of all sets is a set, and so is a member of itself. So, let X be the set of all sets that* are not *members of themselves. Is X a member of itself? If X is a member of itself, then by its own definition it must* not *be a member of itself. Conversely, if X is not a member of itself then it must be a member of itself. This is a contradiction, and illustrates a deep problem with naive set theory.*

To deal with problems like Russell's paradox, mathematicians are very careful what they define sets to be. The most common system used today is ZFC set theory. This is named after two mathematicians involved with its creation (Zermelo and Fraenkel), and the C stands for the axiom of choice. We will not worry about the details, but we note that ZFC is designed to be powerful enough to define lots of the set constructions mathematicians are interested in (e.g. powersets, unions etc.), but not powerful enough that it can construct a paradoxical set, such as the one in Russell's paradox.

It cannot be proved that there is no paradox hiding somewhere in ZFC, but so far none has been found, and most mathematicians are reasonably confident that this is because ZFC is consistent, that is, it cannot be used to prove a contradiction. The results here assume we are using something equivalent to ZFC as our base set theory. We don't worry about the details because we're not going to be using the complicated set constructions mathematicians need for their research. What we will do, however, is introduce the theory of sets as developed by Cantor, and see how his concept of *cardinality* applies to some important sets and useful set constructions.

Cardinal numbers. First we review some basic concepts. Then we can introduce Cantor's concept of 'bigger' and 'smaller' for sets.

Definition 5.1.2 (functions)**.** *If X and Y are sets, then a* function $f : X \to Y$ *is a rule assigning to each element of X a single element of Y. Given $x \in X$ and $y \in Y$, we write $f : x \mapsto y$ to denote that $f(x) = y$.*

- *f is* $1-1$ *(or* injective*) if $f(x_1) = f(x_2) \implies x_1 = x_2$.*
- *f is* onto *(or* surjective*) if for all $y \in Y$ there is an $x \in X$ such that $f(x) = y$.*
- *f is* bijective *if it is 1-1 and onto.*

If X and Y are sets we say $|X| \leq |Y|$ if there is a 1-1 function from X to Y. In words, we say *the cardinality of X is less than or equal to the cardinality*

of Y, or, informally, Y *is at least as big as* X. If $|X| \leq |Y|$ and $|Y| \leq |X|$ then we say $|X| = |Y|$. Note that this defines an equivalence relation between sets, where X and Y are equivalent iff $|X| = |Y|$. Actually, technically this isn't an equivalence relation, because in ZFC the collection of all sets is not a set, but something called a *proper class*, which, informally, is a collection *too big* to be a set. It is, however, essentially the same as an equivalence relation, so we gloss over the issue.

Fact 5.1.3.

1. $|X| \leq |Y| \iff$ *there is an onto (surjective) function from* Y *to* X.
2. *(Cantor-Bernstein theorem).* $|X| = |Y| \iff$ *there is a bijection between* X *and* Y.
3. *Given two sets* X *and* Y*, either* $|X| \leq |Y|$ *or* $|Y| \leq |X|$*, or both.*

Definition 5.1.4 (cardinality)**.** *We define the* cardinality *of* X *to be the equivalence class defined by* $|X|$.

Definition 5.1.4 essentially defines a cardinal number to be the class of all sets of a certain 'size'. This is a refinement of the original idea to define numbers in terms of sets discussed earlier. The reason this is ok while the previous idea failed is that in this version we are careful about exactly what is a 'set' and what is not, but in the original version we let everything be a set, which led to a contradiction.

Definition 5.1.5 (cardinal number)**.** *We define the* cardinal numbers *to be the distinct cardinalities of sets.*

Cardinalities of familiar sets.

Theorem 5.1.6. $|\mathbb{N}| = |\mathbb{Z}|$.

Proof. We define a bijection $f : \mathbb{Z} \to \mathbb{N}$ as follows.

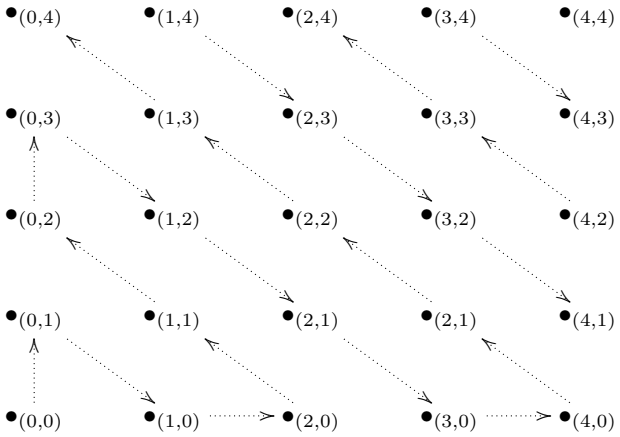
$$f(z) = \begin{cases} 2z \text{ when } z \geq 0 \\ 2|z| - 1 \text{ when } z < 0 \end{cases}$$

□

Theorem 5.1.7. $|\mathbb{N}| = |\mathbb{N} \times \mathbb{N}|$.

Proof. The function $f : \mathbb{N} \to \mathbb{N} \times \mathbb{N}$ defined by $f(n) = (n, n)$ is clearly 1-1. We complete the proof by defining a 1-1 function $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$, which we illustrate

in the diagram below, and then appealing to fact 5.1.3(2).



The meaning of this diagram is that $g(0,0) = 0$, $g(0,1) = 1$, $g(1,0) = 2$, $g(2,0) = 3$ etc. I.e. The value of $g(x,y)$ is determined by where (x,y) comes in the list produced by traveling along the path represented by the arrows in the diagram. □

Corollary 5.1.8. $|\mathbb{N}| = |\mathbb{Q}|$.

Proof. Since $\mathbb{N} \subset \mathbb{Q}$ the inclusion function is an injection from $\mathbb{N}$ to $\mathbb{Q}$. We note that the function $h : \mathbb{Q} \to \mathbb{Z} \times \mathbb{Z}$ defined by

$$h(q) = \begin{cases} (0,0) \text{ when } q = 0 \\ (a,b) \text{ when } \frac{a}{b} \text{ is the most reduced form of } q \end{cases}$$

is 1-1. Composing this with the injection from $g : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ from theorem 5.1.7 and copies f_1 and f_2 of the function f from theorem 5.1.6 gives an injection from $\mathbb{Q}$ to $\mathbb{N}$ as required.

$$\mathbb{Q} \xrightarrow{h} \mathbb{Z} \times \mathbb{Z} \xrightarrow{(f_1, f_2)} \mathbb{N} \times \mathbb{N} \xrightarrow{g} \mathbb{N}$$

Here we are using the easily proved fact that if $g : A \to B$ and $h : B \to C$ are 1-1 functions, then the composition $g \circ h : A \to C$ is also 1-1.

□

Theorem 5.1.9. $|\mathbb{N}| < |\mathbb{R}|$.

Proof. Since $\mathbb{N} \subset \mathbb{R}$ we know that $|\mathbb{N}| \leq |\mathbb{R}|$ as the inclusion function is 1-1. We show that $|\mathbb{N}| \neq |\mathbb{R}|$ by proving that there is no onto function from $\mathbb{N}$ to $\mathbb{R}$. We will show that, if f is a function from $\mathbb{N}$ to the interval $(0,1) \subset \mathbb{R}$, there is an $x \in (0,1)$ such that $f(n) \neq x$ for all $n \in \mathbb{N}$. In other words, $f : \mathbb{N} \to (0,1)$ cannot be onto. If there is no onto function from $\mathbb{N}$ to $(0,1)$ then there is certainly no onto function from $\mathbb{N}$ to $\mathbb{R}$, and so this will prove the claim. This

proof technique is known as *Cantor's diagonal argument*, or just *the diagonal argument*.

We proceed as follows. Every number in $(0, 1)$ can be expressed as an infinite decimal expansion, e.g. $0.x_1x_2x_3\ldots$, where x_n is the nth digit. Define $y = 0.y_1y_2y_3\ldots$ by defining the digits as follows.

$$y_n = \begin{cases} 7 \text{ if the } n\text{th digit of } f(n) \text{ is not } 7 \\ 3 \text{ if the } n\text{th digit of } f(n) \text{ is } 7 \end{cases}$$

Then, by definition, the nth digit of y is different from the nth digit of $f(n)$ for all n, and so $y \neq f(n)$ for all $n \in \mathbb{N}$, which is what we wanted to show. □

Definition 5.1.10 (countable)**.** *A set X is* countable *if $|X| \leq |\mathbb{N}|$. Otherwise it is* uncountable.

Cardinal arithmetic. Given disjoint sets X and Y, we extend the familiar arithmetic operations as follows:

- $|X| + |Y| = |X \cup Y|$.
- $|X| \times |Y| = |X \times Y|$.
- $|X|^{|Y|} = |X^Y|$ (here X^Y stands for the set of functions from Y to X).

Proposition 5.1.11. *If X is a set, then $|\wp(X)| = |2^X|$, where 2 is the two element set $\{0, 1\}$.*

Proof. We define a bijection g from $\wp(X)$ to 2^X by $g(S) = f_S$, where $f_S : X \to \{0, 1\}$ is defined by setting

$$f_S(x) = \begin{cases} 1 \text{ when } x \in S \\ 0 \text{ otherwise.} \end{cases}$$

This f_S is sometimes known as the *characteristic function* of S. Note that g is well defined because every set $S \subseteq X$ defines a unique f_S. Moreover, it is clearly 1-1, and it is onto because given $f : X \to 2$ we can define $S_f = \{x \in X : f(x) = 1\}$, and then $g(S_f) = f$. □

The Continuum Hypothesis.

Fact 5.1.12. $|\mathbb{R}| = |2^{\mathbb{N}}|$.

We know that $|\mathbb{N}| < |\mathbb{R}|$. A question that early set theorists asked was "is there a set Y such that $|\mathbb{N}| < |Y| < |\mathbb{R}|$?". Cantor, the founder of set theory, believed the answer was 'no'. This idea that there is no such Y is the *continuum hypothesis*. Cantor devoted a lot of time trying to prove it from established principles of set theory. However, it turned out that the continuum hypothesis can neither be proved nor disproved using the ZFC axioms. Gödel

showed that it can not be disproved in 1940, and, in 1963, Cohen showed that it can not be proved either. The details of these proofs are well beyond the scope of this course, but the basic idea comes down to simple model theory. Gödel's result showed that there is a model of ZFC where the continuum hypothesis holds, and Cohen showed that there is also a model of ZFC where it does not.

Exercises

Exercise 5.1. *Prove that if X and Y are disjoint finite sets, then the cardinal arithmetic operations agree with the usual arithmetic operations on $|X|$ and $|Y|$. In other words, prove that $|X \cup Y|$ is equal to the result of adding $|X|$ and $|Y|$ as normal, and do similar for the other two arithmetic operations we have defined.*

Exercise 5.2. *Let X_i be countable for all $i \in \mathbb{N}$, and suppose $X_i \cap X_j = \emptyset$ for all $i \neq j \in \mathbb{N}$. Prove that $\bigcup_{i \in \mathbb{N}} X_i$ is countable. HINT: We don't need the condition that $X_i \cap X_j = \emptyset$, but it makes the notation slightly simpler. Think about the proof that $|\mathbb{N}| = |\mathbb{N} \times \mathbb{N}|$.*

Exercise 5.3. *Let X be a countable set. Prove that the set of all finite subsets of X is countable.*

Exercise 5.4. *Let X be a set, let $\wp(X)$ be the powerset of X.*

a) Define a simple injective function from X to $\wp(X)$.

b) Prove that there is no surjective function from X to $\wp(X)$. HINT: Suppose such a function exists. Can you derive a contradiction? The argument used here is similar to the one used in Russell's paradox.

c) What does this tell us about the relationship between $|X|$ and $|\wp(X)|$?

5.2 Enumerative combinatorics

What is enumerative combinatorics? Enumerative combinatorics is the art of counting in finite sets. For example, counting the number of ways 3 balls can be chosen from a bag of 20 balls (an easy question), or how many $n \times n$ matrices there are whose entries are 0 or 1 and such that every row and every column contains exactly 3 ones (a very hard question for most values of n).

In this section we first cover some important basic results producing formulas we can use to easily count things like combinations and permutations (e.g. to answer the 'balls from a bag' question above). Then we'll introduce the simple but deceptively powerful 'pigeon hole principle'. This is an essentially obvious statement that nevertheless is the key to answering all kinds of difficult combinatorial questions. Most of this section will be devoted to examples of applications of this idea. We will end by briefly introducing the subject of 'Ramsey numbers', and finally answering a more difficult version of the 'balls from a bag' question.

Very basics.

Proposition 5.2.1 (Inclusion-exclusion)**.** *If A and B are finite sets, then*

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

More generally, if $A_1, \ldots, A_n$ are finite sets, then

$$\begin{aligned}
|A_1 \cup \ldots \cup A_n| = &\sum_{i=1}^{n} |A_i| \\
&- \sum_{i_1 \neq i_2} |A_{i_1} \cap A_{i_2}| \\
&+ \sum_{i_1 \neq i_2 \neq i_3} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| \\
&\cdot \\
&\cdot \\
&+ (-1)^{k-1} \sum_{i_1 \neq \ldots \neq i_k} |A_{i_1} \cap \ldots \cap A_{i_k}| \\
&\cdot \\
&\cdot \\
&+ (-1)^{n-1} |A_1 \cap \ldots \cap A_n|.
\end{aligned}$$

Proof. The basic case is obvious. You count the number of elements in A and B separately, then correct for double counting by subtracting the number of elements that are in both A and B.

The general version can be proved by induction on n, using the basic version as the base case. For the inductive step we start by noticing that

$$\begin{aligned}
|A_1 \cup \ldots \cup A_n| &= |(A_1 \cup \ldots \cup A_{n-1}) \cup A_n| \\
&= |A_1 \cup \ldots \cup A_{n-1}| + |A_n| - |(A_1 \cup \ldots \cup A_{n-1}) \cap A_n| \\
&= |A_1 \cup \ldots \cup A_{n-1}| + |A_n| - |(A_1 \cap A_n) \cup \ldots \cup (A_{n-1} \cap A_n)|.
\end{aligned}$$

By the inductive hypothesis the claimed formula works for $|A_1 \cup \ldots \cup A_{n-1}|$ and $|(A_1 \cap A_n) \cup \ldots \cup (A_{n-1} \cap A_n)|$, and the proof is completed by writing it all out and matching up expressions so that the claimed formula is obtained for $|A_1 \cup \ldots \cup A_n|$. The details are purely an exercise in working through ugly notation, and we omit them. □

Proposition 5.2.2 (Permutations and combinations)**.** *Let $k \leq n \in \mathbb{N}$. Then:*

1. *The number of ways we can select k objects from a set of n objects, where the order of selection is important, is given by the formula*

$$P(n, k) = \frac{n!}{(n-k)!}.$$

2. *The number of ways we can select k objects from a set of n objects, where the order of selection is* not *important, is given by the formula*

$$C(n,k) = \binom{n}{k} = \frac{n!}{(n-k)!k!}.$$

Proof. The formula $\frac{n!}{(n-k)!}$ reflects the fact that we have n possibilities for the first selection, $n-1$ for the second, and so on down to the kth selection when we have $(n-(k-1))$ possibilities. Thus we have $n\times(n-1)\times\ldots\times(n-(k-1)) = \frac{n!}{(n-k)!}$ total possibilities.

The formula $\frac{n!}{(n-k)!k!}$ reflects the fact that if we don't care about the order, an ordered selection of k objects is equivalent to all the other selections of the same objects but in a different order. There are $k!$ different ways to order a collection of k elements, so we get the formula for $C(n,k)$ by dividing the formula for $P(n,k)$ by $k!$. □

Pigeon hole principle.

Lemma 5.2.3 (Pigeon hole principle)**.** *If $k < n$ and you have n balls in k bags, there must be at least one bag containing at least two balls. More precisely, there must be at least one bag containing at least $\lceil \frac{n}{k} \rceil$ balls.*

This lemma gets its name from the fact that it is often stated in terms of pigeons and pigeon holes, rather than balls and bags. The following is a restatement of the pigeon hole principle that can be more useful in some situations.

Lemma 5.2.4. *In any finite collection of natural numbers, the maximum must be at least as large as the mean, and the minimum must be at most as large as the mean.*

Example 5.2.5. *If you choose five distinct numbers between 1 and 8, then two of those numbers must sum to 9.*

Proof. The four sets $\{1,8\}$, $\{2,7\}$, $\{3,6\}$, $\{4,5\}$ partition $\{1,\ldots,8\}$. Each one of our five numbers must be in one of these sets, so there must be one set containing two, and thus two elements that sum to 9. □

Example 5.2.6. *In a city of 200,000 people, at least 547 people will have the same birthday.*

Proof. There are 366 possible birthdays (including leap years). Since there are 200,000 people, the average number of people born on a day will be $\frac{200{,}000}{366} = 546.45$. By lemma 5.2.4, the day that has the most birthdays must have a larger number of birthdays than this, so at least 547. □

Example 5.2.7. *For every integer n there is a multiple of n that has only* 0*s and* 1*s in its decimal expansion.*

Proof. Consider the numbers $x_1, x_2, \ldots, x_n$, where $x_1 = 1$, $x_2 = 11$, and x_k is 1 repeated k times. There are $n-1$ non-zero values in $\mathbb{Z}_n$, so either $n|x_k$ for some k (in which case we are done), or there are $i < j \leq n$ such that the value of $x_i \mod n$ is the same as the value of $x_j \mod n$. But then $n|(x_j - x_i)$, and $x_j - x_i$ has the required form, so the proof is complete. □

Example 5.2.8. *A baseball team plays every day for 30 days. They can play more than once each day, but they play at most 45 games in total. There is some period of consecutive days where they play exactly 14 games.*

Proof. Let a_j be the number of games played up to and including the jth day. Then $a_1, a_2, \ldots, a_{30}$ is a strictly increasing sequence bounded by 45. Moreover, $a_1 + 14, a_2 + 14, \ldots, a_{30} + 14$ is a strictly increasing sequence bounded by 59. Combining these two sequences gives us 60 elements, each with values between 1 and 59. So, by the pigeon hole principle, there must be two terms in the long sequence with the same value. Since the team plays everyday, the two terms must be from different halves. In other words, we can't have $a_i = a_j$ or $a_i + 14 = a_j + 14$ unless $i = j$. So there are $i < j$ with $a_j = a_i + 14$. But this just means that exactly 14 games are played between the ith day and the jth day, which is what we want to prove. □

Example 5.2.9. *If we have $n+1$ positive integers, each less than or equal to $2n$, there must be one number that divides another one.*

Proof. Every positive integer can be written as $q2^k$, where q is an odd number and k is some natural number. We prove this subclaim by induction. It is obviously true when $n = 1$, so let $n > 1$. If n is odd there is nothing to prove, so suppose $n = 2l$ for some l. Then $l = q2^k$ by the inductive hypothesis, and so $n = q2^{k+1}$.

Now, there are only n odd numbers less than or equal to $2n$, so, given a list of $n+1$ numbers there must be numbers $a \neq b$ in the list with $a = q2^{k_1}$, and $b = q2^{k_2}$ for the same q. If $k_1 < k_2$ then $a|b$, otherwise $b|a$. □

Example 5.2.10. *In any group of more than 2 people, at least two people must have the same number of friends (assuming friendship is symmetric).*

Proof. Suppose there are n people, and $n \geq 2$. Then each person can have between 0 and $n-1$ friends. There are two cases.

1. Everyone has at least one friend. In this case each person has between 1 and $n-1$ friends, so there are n people and $n-1$ possibilities, so at least two people must have the same number of friends.

2. Someone has no friends. In this case each person has between 0 and $n-2$ friends, so there are again n people and $n-1$ possibilities.

□

Example 5.2.11. *In any sequence of n^2+1 distinct real numbers, there must either be a strictly increasing subsequence of size $n+1$, or a strictly decreasing subsequence of size $n+1$.*

Proof. Suppose our set of numbers is $(a_0, a_1, \ldots, a_{n^2})$. For each $k \in \{0, \ldots, n^2\}$ define the pair (i_k, d_k), where i_k is the length of the longest strictly increasing subsequence starting at a_k, and d_k is the length of the longest strictly decreasing subsequence starting at a_k. Suppose there are no strictly increasing or decreasing subsequences of size $n+1$. Then i_k and d_k are both less than or equal to n for all k. Since the minimum possible value for i_k and d_k is 1, this means there are n^2 possible distinct values for (i_k, d_k). But there are n^2+1 terms in the sequence, so there must be $l < k \in \{0, \ldots, n^2\}$ with $(i_l, d_l) = (i_k, d_k)$. But this is impossible, because if $a_l < a_k$ we must have $i_l > i_k$, and if $a_l > a_k$ we must have $d_l > d_k$. □

Ramsey numbers.

Proposition 5.2.12. *Suppose two people can either be friends or enemies. In any group of 6 people, either there are three mutual friends, or three mutual enemies.*

Proof. Choose an arbitrary member of the group, and call this person x. Out of the five remaining people, there must either be three who are friends with x, or three who are not. Suppose there are three people who are friends with x. If any two of them are friends with each other then this provides a group of three mutual friends. If no two of them are friends then they are a group of three mutual enemies. In either case, we are done. The case where there are three enemies of x is the same by symmetry. □

Definition 5.2.13 (Ramsey numbers)**.** *Let m and n be natural numbers greater than or equal to 2. We define the* Ramsey number $R(m, n)$ *to be the minimum number of people at a party so that there are either m mutual friends, or n mutual enemies.*

It's obvious that $R(m, n) = R(n, m)$, for all m and n. By proposition 5.2.12, we know $R(3, 3) = 6$ (as we can find an example of a group of 5 where there are neither three mutual friends, nor three mutual enemies - see exercise 2.1. In general, it is very difficult to find Ramsey numbers, and surprisingly few are known. For example $R(4, 4) = 18$, but $R(5, 5)$ is only known to lie somewhere in the range 43-48, and $R(10, 10)$ is only known to be between 798 and 23556. Calculating Ramsey numbers exactly is a far away goal for combinatorics researchers, and merely making the possible range smaller is a major breakthrough. This is not because Ramsey numbers are themselves particularly important, but because the problem is so difficult that progress requires significantly new ideas.

Combinations with repetition.

Theorem 5.2.14. *Suppose we have an infinite supply of balls in n different colours. Suppose we choose k balls, and the only distinguishing feature of the balls is their colour. Then there are $\binom{n+k-1}{k}$ different possible outcomes if we don't care about the order the balls are chosen.*

Proof. We use a trick. Choosing k balls in n different colours is like putting k different balls into n different boxes. We will represent this graphically using $*$ to represent balls, and $|$ to represent the boundaries of the boxes. For example

$$**|*|***||*$$

would represent a choice of 7 balls in 5 different colours, with details in the table below.

Colour no.	No. balls with colour
1	2
2	1
3	3
4	0
5	1

Given n colours and k balls, every string of k stars and $n-1$ vertical lines represents a possible choice, and every choice can be represented using this system. So there are the same number of choices as there are strings with k stars and $n-1$ lines. We can think of this as starting with $n+k-1$ vertical lines, then choosing k of them to change to stars. But this is just $\binom{n+k-1}{k}$, which is what we aimed to prove. □

Exercises

Exercise 5.5. *Show that there is a group of five people containing neither three mutual friends, nor three mutual enemies.*

Exercise 5.6. *Show that $R(2,n) = n$ for all $n \in \mathbb{N}$ with $n \geq 2$.*

Exercise 5.7. *Suppose there are 5 points in a $1cm \times 1cm$ square. Prove that there must be two points at most $\frac{\sqrt{2}}{2}cm$ apart.*

Exercise 5.8. *Suppose we have a chessboard with two diagonally opposite squares removed. Is it possible to tile the board using domino pieces?*

Exercise 5.9 (Pascal's identity)**.** *Prove that $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$.*

Exercise 5.10. *Use theorem 5.2.14 to find the number of solutions to the equation $a+b+c+d = 17$, where a, b, c, d must all be natural numbers (HINT: think about distributing 17 pebbles into boxes marked a, b, c, d). How many solutions are there if we demand that a, b, c, d are all greater than or equal to 2?*

Exercise 5.11. *Suppose we have 5 points in the xy plane, all with integer coordinates. Show that the midpoint of one of the straight lines connecting pairs of these points also has integer coordinates.*

5.3 Further reading

A lot of the material covered here is touched on in [8], though the relevant sections are scattered throughout the book. A huge amount of material on combinatorics can be found in [4]. I don't know if this book is formally in the public domain, but if you Google it there are several links to PDFs hosted by universities and which shouldn't give you viruses. For more set theory, [5] is supposed to be good but I haven't read it. This will cover ordinals and cardinals in more detail than here.

6 Appendix - Solutions to exercises

Number Theory

Exercise 2.1. *Consider the following (false) theorem:*

Theorem. *If $a, b \in \mathbb{N}$ and $a = b$ then $a = 0$.*

Proof.

$$\begin{aligned} a &= b \\ a^2 &= ab \\ a^2 - b^2 &= ab - b^2 \\ (a-b)(a+b) &= (a-b)b \\ a+b &= b \\ a &= 0 \end{aligned}$$

□

What is wrong with this proof?

Solution: Since $a = b$ we know that $a - b = 0$. We are not allowed to divide by zero, as the result of this is not usually defined. In general, we cannot divide by zero, or even by something that could be zero, and expect the argument to be valid. This means that when we divide by $(a - b)$ in this argument, we are doing something wrong, and we can't trust the proof after that point.

Exercise 2.2. *Use the well-ordering principle to show that*

$$2 + 4 + 6 + \ldots + 2n = n(n + 1).$$

HINT: If there is a value of n there must be a smallest such value. Show that the existence of this smallest value leads to a contradiction.

Solution: Suppose there is some positive integer such that the identity given in the question is not valid. Then, by the well-ordering principle, there must be a smallest such number. Call this number k. Now, k cannot be 1, as we can easily check the identity is true for $n = 1$. Since k is the smallest number for which the identity above does not hold, it must be true for $k-1$. So we have

$$2 + \ldots + 2(k-1) = (k-1)(k),$$

and so, adding $2k$ to both sides we have

$$2 + \ldots + 2(k-1) + 2k = (k-1)(k) + 2k.$$

But

$$(k-1)(k) + 2k = (k)(k+1),$$

and so

$$2 + \ldots + 2k = (k)(k+1).$$

This is a contradiction because k is supposed to invalidate the identity. We conclude that there can be no positive integer for which the identity is false.

Exercise 2.3. *Let $n \in \mathbb{N}$. If n^2 is even must n also be even? Give a proof or a counterexample. HINT: think about the fundamental theorem of arithmetic, specifically the existence of a prime factorization, and also lemma 2.1.9.*

Solution: By the fundamental theorem of arithmetic, n^2 can be written in exactly one way as a product of primes (if we don't care about the order they're written down in), and the same is true for n. Suppose n can be written as the product of primes $p_1 \ldots p_k$. Then obviously n^2 can be written as the product $p_1 \ldots p_k p_1 \ldots p_k$. If n^2 is even then $2|n^2$. Now, by lemma 2.1.9, and the fact that $n^2 = p_1 \ldots p_k p_1 \ldots p_k$, we must have $2|p_i$ for some $i \in \{1, \ldots, k\}$, but this means $2|n$, and so n is even too.

Exercise 2.4. *Let $n \in \mathbb{N} \setminus \{0\}$. Then using theorem 1.2 prove that $\log_5(n)$ is either a natural number or irrational. HINT: Suppose $5^{\frac{a}{b}} = n$. What does this tell us about the ratio $\frac{a}{b}$? The fact that 5 is prime is important. HINT: Suppose $5^{\frac{a}{b}} = n$. What does this tell us about the ratio $\frac{a}{b}$? The fact that 5 is prime is important.*

Solution: If $5^{\frac{a}{b}} = n$ then $5^a = n^b$. Using the fundamental theorem of arithmetic, n^b can be uniquely factorized into primes. Since $n^b = 5^a$ we know this factorization must just be $55 \ldots 5$ (a list of a fives). Again by the fundamental theorem of arithmetic, n must also have a unique factorization into primes, and, as $n^b = 55 \ldots 5$, this factorization of n must just be a list of fives (i.e. $n = 5^k$ for some k). But if we take a product of b copies of this list of fives we get n^b, which is 5^a (i.e. $5^a = (5^k)^b$, so $a = kb$). This means that b must divide a. In other words, $\frac{a}{b}$ must be a natural number.

Exercise 2.5. *Is the result from exercise 2.4 still true if we replace 5 with 4? Provide a proof or a counterexample.*

Solution: It's not true. For example, $\log_4 2 = \frac{1}{2}$.

Exercise 2.6. *Suppose $x \equiv_n y$, and suppose $m|n$. Show that $x \equiv_m y$.*

Solution: Suppose $x - y = kn$, and $n = am$. Then $x - y = (ka)m$.

Exercise 2.7. *Complete the proof of proposition 2.2.8. HINT: Use the fact that $xy - x'y' = xy - xy' + xy' - x'y'$.*

Solution: Suppose $(x - x') = kn$, and $(y - y') = ln$. Note that

$$\begin{aligned} xy - x'y' &= xy - xy' + xy' - x'y' \\ &= x(y - y') - y'(x - x') \\ &= xln - y'kn \\ &= (xl - y'k)n. \end{aligned}$$

Exercise 2.8. *Calculate* $2^{2^{13543}} \mod 3$.

Solution: $2 \equiv_3 -1$, and 2^{13543} is an even number. Since -1 to the power of any even number is 1, we must have $2^{2^{13543}} \equiv_3 1$. We're using proposition 2.8 in the background here. This is what tells us that $2^{2^{13543}} \equiv_3 (-1)^{2^{13543}}$. More generally, if $x \equiv_n y$ then $x^k \equiv_n y^k$ for all n and k.

Exercise 2.9. *Let p and q be distinct primes, and let $x \in \mathbb{Z}$. Prove that if $p|x$ and $q|x$, then $pq|x$.*

Solution: We know that $x = (\pm 1)p_1 \dots p_n$ for some sequence of primes $p_1, \dots, p_n$. This is the easy half of the Fundamental Theorem of Arithmetic. Also, by lemma 1.9, since $p|p_1 \dots p_n$ we must have $p = p_i$ for some $i \in \{1, \dots, n\}$. We also have $q|x$, and so $q = p_j$ for some $j \in \{1, \dots, n\}$, by the same argument. Since p and q are distinct, we can't have $i = j$. Assume without loss of generality that $i = 1$ and $j = 2$. Then $x = (\pm 1)(pq)p_3 \dots p_n$, and so $pq|x$.

Exercise 2.10.

a) Prove that $4 = 9 = -1 \mod 5$.

b) Prove that $4^{1536} \equiv_7 9^{4824}$ *(HINT:* $9 \equiv_7 2$ *and* $8 \equiv_7 1$*).*

c) Using exercise 2.9, and your answers to a) and b), prove that $4^{1536} \equiv_{35} 9^{4824}$.

Solution:

a) $4 - (-1) = 5$, and $9 - (-1) = 2(5)$.

b) First, we have

$$\begin{aligned} 4^{1536} &= 2^{2(1536)} \\ &= 2^{3072} \\ &= 2^{3(1024)} \\ &= 8^{1024} \\ &\equiv_7 1^{1024} \\ &\equiv_7 1. \end{aligned}$$

Second, we have

$$\begin{aligned} 9^{4824} &\equiv_7 2^{4824} \\ &= 2^{3(1608)} \\ &= 8^{1608} \\ &\equiv_7 1^{1608} \\ &\equiv_7 1. \end{aligned}$$

This proves the claim.

c) It follows from part a) that

$$4^{1536} \equiv_5 (-1)^{1536} \equiv_5 1,$$

and also

$$9^{4824} \equiv_5 (-1)^{4824} \equiv_5 1.$$

This means

$$4^{1536} \equiv_5 9^{4824},$$

and so

$$5|(4^{1536} - 9^{4824}).$$

In part b) we proved that

$$4^{1536} \equiv_7 9^{4824},$$

and so

$$7|(4^{1536} - 9^{4824}).$$

By exercise 2.9 this means

$$35|(4^{1536} - 9^{4824}),$$

which is another way of saying that

$$4^{1536} \equiv_{35} 9^{4824}.$$

Exercise 2.11. *Let X be a set and let $\{Y_i : i \in I\}$ be a partition of X (here I is an* indexing set*, i.e. a non-empty set we use to label something, in this case elements of the partition). Prove that the binary relation R, defined by $R(x, y) \iff x$ and y are in Y_i for some $i \in I$, is an equivalence relation.*

Solution: We must show that this relation R satisfies the three conditions of equivalence relations.

1. First we need to show that R is reflexive. So, given $x \in X$, is it true that $R(x, x)$? Yes, because obviously x is in the same part of the partition as itself.

2. Now we need to show that R is symmetric. So, suppose $R(x, y)$. Then x and y are in the same part of the partition. But then $R(y, x)$ by definition.

3. Finally, we show R is transitive. Suppose $R(x, y)$ and $R(y, z)$. Then x is in the same part of the partition as y, and y is in the same part of the partition as z. But this means x is in the same part of the partition as z, and so $R(x, z)$, which is what we want.

Exercise 2.12. *(Optional)*

a) Given an equivalence relation R on a set X, define P_R to be the partition obtained from R in proposition 2.5. Let R_{P_R} be the equivalence relation obtained from P_R as in exercise 2.11. Prove that $R(x, y) \iff R_{P_R}(x, y)$ for all $x, y \in X$.

b) State and prove a similar conjecture on converting from partitions to equivalence relations and back to partitions.

Solution:

a) Suppose first that $R(x, y)$. Then $y \in [x]$, which is a way of saying that y and x are in the same part of the partition P_R. But this means $R_{P_R}(x, y)$. Conversely, if $R_{P_R}(x, y)$, then $y \in [x]$, which means $R(x, y)$. This shows $R = R_{P_R}$.

b) The sensible conjecture is that $P_{R_P} = P$. To prove this, let $P = \{X_i : i \in I\}$. We want to show that $\{X_i : i \in I\} = \{[x]_{R_P} : x \in X\}$. First, given any $x \in X$ we must have $x \in X_i$ for some i, as P is a partition. We must prove that $[x]_{R_P} = X_i$. Now,

$$\begin{aligned} y \in [x]_{R_P} &\iff R_P(x, y) \\ &\iff y \in X_i. \end{aligned}$$

This proves the claim because, because every $[x]_{R_P}$ is equal to X_i where $x \in X_i$, and every X_i is equal to $[x]_{R_P}$ for $x \in X_i$.

Exercise 2.13. *Prove lemma 2.3.4.*

Solution: Suppose $a|bc$ and a and b are coprime. Since $a|bc$ we know there is k with $ak = bc$. Also, as a and b are coprime, $\mathbf{HCF}(a, b) = 1$, so by corollary 1.10 (Bézout's identity) there are x and y with $xa + by = 1$. This means $xac + byc = c$, which means $xac + yak = c$. Rearranging gives $a(xc + yk) = c$, and so $a|c$.

Another way to prove this is to notice that no prime factor of a can divide b (as a and b are coprime). Also, by lemma 1.11 every prime factor of a must divide either b or c. Putting this together means every prime factor of a divides c, and so $a|c$.

Exercise 2.14. *Find all solutions to* $x^2 - 1 \equiv_8 0$. *What does this tell us about Lagrange's theorem in the case where p is not prime?*

Solution: The solutions are $1, 3, 5, 7$. This tells us that Lagrange's theorem is false when p is not prime.

Exercise 2.15. *Calculate* $5^{30,000} - 6^{123,456} \mod 31$. *HINT: Fermat's little theorem is useful here.*

Solution: We have $5^{30000} = 5^{30(1000)}$, and $5^{30} \equiv_{31} 1$ by Fermat's little

theorem. So $5^{30000} \equiv_{31} 1^{1000} \equiv_{31} 1$. Also, $123456 = 30(4115) + 6$, so

$$\begin{aligned} 6^{123456} &= 6^{30(4115)+6} \\ &= 6^{30(4115)}.6^6 \\ &\equiv_{31} 1^{4115}.6^6 \text{ using Fermat's little theorem} \\ &\equiv_{31} 6^2.6^2.6^2 \\ &\equiv_{31} 5.5.5 \\ &\equiv_{31} 125 \\ &\equiv_{31} 1. \end{aligned}$$

So $5^{30,000} - 6^{123,456} = 0 \mod 31$.

Exercise 2.16 (Wilson's theorem)**.**

a) Prove that when $n = 2$ we have $(n-1)! \equiv_n -1$.

b) Let p be an odd prime. Define $g(x) = (x-1)(x-2)\dots(x-(p-1))$.

i) What are the roots of g modulo p?

ii) What is the degree of g?

iii) What is the leading term of g? (The leading term is the one with the highest power of x).

c) Define $h(x) = x^{p-1} - 1$. What are the roots of h modulo p? HINT: Fermat's little theorem.

d) Define $f(x) = g(x) - h(x)$. Prove that f_p must be the constant function $f(x) \equiv_p 0$ for all x. HINT: Lagrange's theorem.

e) Using the conclusion to part d), prove that n is prime if and only if

$$(n-1)! \equiv_n -1$$

(this is known as Wilson's theorem).

Solution:

a) $(2-1)! = 1 \equiv_2 -1$.

b) i) The roots are $1, 2, 3, \dots, p-1$.

ii) The degree of g is $p-1$.

iii) The leading term is x^{p-1}.

c) The roots of h are $1, 2, 3 \dots, p-1$ (all numbers between 1 and $p-1$ are coprime with p so we apply Fermat's little theorem).

d) g and h both have degree $p-1$. Also, the leading term of both is x^{p-1}. This means that the degree of f_p is at most $p-2$. However, f_p has at least $p-1$ roots, as g and h have $p-1$ roots in common. So Lagrange's theorem tells us that f_p must be the constant zero function modulo p, as it has more roots than its degree.

e) Suppose $n = p$ for a prime number p. If $p = 2$ we already proved the claim in part a), so suppose $p > 2$. Then we have proved that $g(x) - h(x) \equiv_p 0$. So in particular we have $g(p) - h(p) \equiv_p 0$. So $(p-1)! + 1 \equiv_p 0$. I.e. $(p-1)! \equiv_p -1$.

Conversely, if $(p-1)! \equiv_p -1$ then $p|((p-1)!+1)$. So, if q is a prime factor of p with $q < p$ then $q|((p-1)!+1)$, and obviously $q|(p-1)!$. So from lemma 1.7 we get $q|1$, but this is impossible. We conclude that p has no prime factors other than itself. I.e. that p is prime.

Exercise 2.17. *Let $p = 11$ and $q = 13$. Choose suitable e and d for use in RSA encryption.*

Solution: We have $(p-1)(q-1) = 10 \times 12 = 120$. The smallest prime that doesn't divide 120 is 7, so set $e = 7$. We need to find the inverse of 7 mod 120. Now, $120 = 17(7) + 1$, so $-17(7) - 1 = -120$. This means the inverse of 7 is -17 mod 120, and $-17 \equiv 103_{120}$. So take $d = 103$.

Exercise 2.18. *Prove that if $a \equiv_n b$ then $a^k \equiv_n b^k$ for all $k \in \mathbb{N}$.*

Solution: Induct on k. If $k = 0$ then its obviously true as $1 = 1$. Suppose it's true for $k-1$. Then $a^k = a.a^{k-1}$ and $b^k = b.b^{k-1}$, and by assumption we have $a \equiv_n b$, and by the inductive hypothesis we have $a^{k-1} \equiv_n b^{k-1}$. Proposition 2.2.8(2) applies and tells us that $a^k \equiv_n b^k$ too.

Exercise 2.19. *Let a and b be coprime. Prove that if $a|c$ and $b|c$ then $ab|c$.*

Solution: By Bézout's identity there are x and y with $xa + yb = 1$. So $cxa + cyb = c$. Also, as $a|c$ there is k with $ak = c$, and as $b|c$ there is l with $bl = c$. So, we have

$$(bl)xa + (ak)yb = c,$$

and rearranging this gives

$$(ab)(xl + yb) = c.$$

This means $ab|c$ as claimed.

For an alternative proof, use prime factorizations $a = p_1 \dots p_m$, and $b = q_1 \dots q_n$. Since a and b are coprime we mist have $p_i \neq q_j$ for all i and j. Since $a|c$ and $b|c$, this means $c = p_1 \dots p_m q_1 \dots q_n k$ for some k. But this means $ab|c$.

Exercise 2.20 (Chinese remainder theorem)**.** *Let $n_1, \dots, n_k \in \mathbb{N}$ all be greater than 1 and such that n_i and n_j are coprime for all $i \neq j$. Define $N = \prod_{i=1}^{k} n_i$. For each $i \in \{1, \dots, k\}$ let $a_i \in \{0, 1, 2, \dots, n_i - 1\}$.*

a) Let x and y be integers with $x \equiv_{n_i} a_i$ and $y \equiv_{n_i} a_i$ for all i. Prove that $x \equiv_N y$.

b) Find $z \in \mathbb{Z}$ with $z \equiv_{n_1} a_1$ and $z \equiv_{n_2} a_2$. HINT: Bézout.

c) Extend part b) to prove that there is z with $z \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k\}$. HINT: Induction.

Combining parts a) and c) we get that there is a number z such that $z_i \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k\}$, and that this z is unique mod N. This result is known as the Chinese remainder theorem. It is attributed to the 3rd century Chinese mathematician Sunzi, though his version was presented very differently.

Solution:

a) we have $x \equiv_{n_i} = y$ for all i, so $n_i | (x - y)$ for all i. Since n_i and n_j are coprime for all $i \neq j$, it follows from exercise 2.19 that $N = \prod_{i=1}^{k} n_i | (x - y)$, and so $x \equiv_N y$.

b) Bézout's identity produces x and y with $xn_1 + yn_2 = 1$ (as n_1 and n_2 are coprime). Set $z = xn_1a_2 + yn_2a_1$. Then

$$\begin{aligned} z &\equiv_{n_1} yn_2a_1 \\ &\equiv_{n_1} a_1(1 - xn_1) \\ &\equiv_{n_1} a_1. \end{aligned}$$

Similarly, we have $z \equiv_{n_2} a_2$.

c) Induct on k. It's obviously true for $k = 1$ (we can just use $z = a_1$). Suppose now that it's true for $k - 1$. Then we can find z' such that $z' \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k-1\}$. Define $N' = \prod_{i=1}^{k-1} n_i$. We can assume that $0 \leq z' < N'$, as any number equal to z' mod N' will also be equal to z' mod n_i for all $i \in \{1, \ldots, k-1\}$. Now, N' and n_k must be coprime, because by lemma 1.11 if $p | N'$ then $p | n_i$ for some $i \in \{1, \ldots, k-1\}$, and therefore $p \nmid n_k$ as n_i and n_k are coprime. So, part b) tells us there is z with $z \equiv_{N'} z'$ and $z \equiv_{n_k} a_k$. This is what we want, because we must have $z \equiv_{n_i} z' \equiv_{n_i} a_i$ for all $i \in \{1, \ldots, k-1\}$ too. We can assume that $0 \leq z < N$, as we can just take the value of z mod N.

Logic

Exercise 3.1. *Let ϕ and ψ be sentences. Show that*

$$\phi \leftrightarrow \psi \models\models (\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$$

Solution: The truth tables are the same:

ϕ	ψ	$\phi \leftrightarrow \psi$	$(\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$
T	T	T	T
T	F	F	F
F	T	F	F
F	F	T	T

Exercise 3.2. *Prove that the set $\{\wedge, \neg\}$ is functionally complete. HINT: think about proposition 3.1.9 and corollary 3.1.7.*

Solution: To prove this we need to show that for every sentence ϕ there is a sentence ϕ' using only $\wedge$ and $\neg$ such that $\phi \models =\!|\phi'$. In proposition 3.1.9 we already proved that $\{\wedge, \vee, \neg, \leftrightarrow\}$ is functionally complete, so we can assume without loss of generality that ϕ only contains connectives from $\{\wedge, \vee, \neg, \leftrightarrow\}$.

As in the proof of corollary 3.1.7, we use induction on formula construction. In the base case $\phi = p$ for some proposition symbol p. In this case we just set $\phi' = p$.

For the inductive step, suppose first that $\phi = \psi_1 \vee \psi_2$, and we have already found ψ_1' and ψ_2'. Then define $\phi' = \neg(\neg\psi_1' \wedge \neg\psi_2')$, and we can use truth tables to show $\phi \models =\!|\phi'$.

Similarly, suppose $\phi = \psi_1 \leftrightarrow \psi_2$ and we have already found ψ_1' and ψ_2'. Notice that $(\psi_1' \wedge \psi_2') \vee (\neg\psi_1' \wedge \neg\psi_2') \models =\!|\psi_1 \leftrightarrow \psi_2$. So, using the first case we can define $\phi' = \neg\big(\neg(\psi_1' \wedge \psi_2') \wedge \neg(\neg\psi_1' \wedge \neg\psi_2')\big)$.

Finally, the cases where $\phi = \neg\psi$ and $\phi = \psi_1 \wedge \psi_2$ are easy, because we can just set $\phi' = \neg\psi'$ or $\phi' = \psi_1' \wedge \psi_2'$.

Exercise 3.3. *Define a binary connective $|$ using the following truth table.*

ϕ	ψ	$\phi\|\psi$
T	T	F
T	F	T
F	T	T
F	F	T

Prove that $\{|\}$ is functionally complete.

Solution: We will do this by showing that if ϕ is a sentence involving only symbols from $\{\wedge, \neg\}$, there is a sentence ϕ' using only $|$ such that $\phi \models =\!|\phi'$. This is enough because we already proved that $\{\wedge, \neg\}$ is functionally complete. We use the same inductive method as before, and the base case is again trivial.

So suppose $\phi = \neg\psi$ and we already have ψ'. Observe that

ψ	$\neg\psi$	$\psi\|\psi$
T	F	F
F	T	T

So we can define $\phi' = \psi'|\psi'$.

Suppose now that $\phi = \psi_1 \wedge \psi_2$ and we already have ψ_1' and ψ_2'. Observe that

ψ_1	ψ_2	$\psi_1 \wedge \psi_2$	$\psi_1\|\psi_2$	$(\psi_1\|\psi_2)\|(\psi_1\|\psi_2)$
T	T	T	F	T
T	F	F	T	F
F	T	F	T	F
F	F	F	T	F

So we can define $\phi' = (\psi_1'|\psi_2')|(\psi_1'|\psi_2')$.

Exercise 3.4. *Let p and q be basic propositions. How many possible distinct truth tables are there for formulas involving only the propositions p and q? (We*

consider two truth tables for formulas involving the same basic propositions to be distinct if there is a truth assignment for the basic propositions such that the evaluation of each formula under this assignment is different in each table. For example, the truth tables of $p \wedge q$ and $p \vee q$ are distinct because the values of these formulas when p is true and q is false are different).

Solution: There are 4 rows in each truth table for p and q.

p	q	ϕ
T	T	?
T	F	?
F	T	?
F	F	?

Here each ? can be true or false. This gives $2^4 = 16$ distinct possibilities.

Exercise 3.5. *A sentence ϕ is in* disjunctive normal form*, (DNF), if it is of the form $\phi_1 \vee \ldots \vee \phi_n$, where each ϕ_i is of the form $l_1 \wedge \ldots \wedge l_k$, and each l_j is either a basic proposition or the negation of a basic proposition. E.g. $(p \wedge q) \vee (\neg p) \vee (p \wedge \neg q \wedge r)$ is in DNF. Show that every sentence is equivalent to a sentence in DNF. HINT: Think about the truth table.*

Solution: Consider this example. Suppose ϕ contains only the proposition symbols p, q, r, and that its truth table is as follows:

p	q	r	ϕ
T	T	T	T
T	T	F	F
T	F	T	F
T	F	F	F
F	T	T	T
F	T	F	T
F	F	T	F
F	F	F	F

Then ϕ is obviously logically equivalent to $(p \wedge q \wedge r) \vee (\neg p \wedge q \wedge r) \vee (\neg p \wedge q \wedge \neg r)$, which is a DNF sentence. This method obviously generalizes. If ϕ is a contradiction, i.e. if all rows in the truth table are F, then ϕ is equivalent to e.g. $p \wedge \neg p$.

Exercise 3.6. *The following deduction tree proves that $\phi \to \psi$ can be deduced from $\neg\phi \vee \psi$ in intuitionistic propositional logic. Add labels indicating the rules used at each stage.*

$$\dfrac{\neg\phi \vee \psi \qquad \dfrac{\dfrac{[\neg\phi]_1 \qquad [\phi]_2}{\bot}}{\psi}}{\phi \to \psi} \qquad \dfrac{\dfrac{[\psi]_1}{\psi} \qquad [\phi]_3}{\phi \to \psi}$$

Solution:

$$\dfrac{\neg\phi \vee \psi \qquad \dfrac{\dfrac{[\neg\phi]_1 \qquad [\phi]_2}{\bot}(\neg_E)}{\psi}(\bot_E) \qquad \dfrac{\dfrac{[\psi]_1}{\psi} \qquad [\phi]_3}{\phi \to \psi}(\to_I)}{\phi \to \psi}(\to_I) \qquad \dfrac{}{}(\vee_E)$$

Exercise 3.7. *What is being proved in the following deduction tree? Add labels indicating the rules at each stage.*

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\phi \to \psi \qquad [\phi]_2}{\psi}}{\neg\phi \vee \psi} \qquad [\neg(\neg\phi \vee \psi)]_1}{\bot}}{\neg\phi}}{\neg\phi \vee \psi} \qquad \dfrac{[\neg(\neg\phi \vee \psi)]_1}{\neg(\neg\phi \vee \psi)}}{\bot}}{\neg\neg(\neg\phi \vee \psi)}}{\neg\phi \vee \psi}$$

Solution: This tree proves that $\neg\phi \vee \psi$ can be deduced from $\phi \to \psi$ in *classical* propositional logic.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\phi \to \psi \qquad [\phi]_2}{\psi}(\to_E)}{\neg\phi \vee \psi}(\vee_{I_r}) \qquad [\neg(\neg\phi \vee \psi)]_1}{\bot}(\neg_I)}{\neg\phi}(\vee_{I_l})}{\neg\phi \vee \psi} \qquad \dfrac{[\neg(\neg\phi \vee \psi)]_1}{\neg(\neg\phi \vee \psi)}}{\bot}(\neg_E)}{\neg\neg(\neg\phi \vee \psi)}(\neg_I)}{\neg\phi \vee \psi}(\neg\neg_E)$$

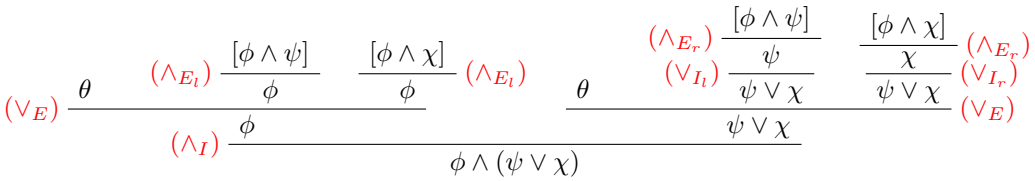
Exercise 3.8. *Show that $(\phi \wedge \psi) \to (\psi \wedge \phi)$ can be deduced from an empty set of axioms.*

Solution:

$$\dfrac{\dfrac{\dfrac{[\phi \wedge \psi]_1}{\phi \wedge \psi}(\wedge_{E_r}) \qquad \dfrac{[\phi \wedge \psi]_1}{\phi \wedge \psi}(\wedge_{E_l})}{\psi \qquad \phi}(\wedge_I)}{\psi \wedge \phi}(\to_I)}{(\phi \wedge \psi) \to (\psi \wedge \phi)}$$

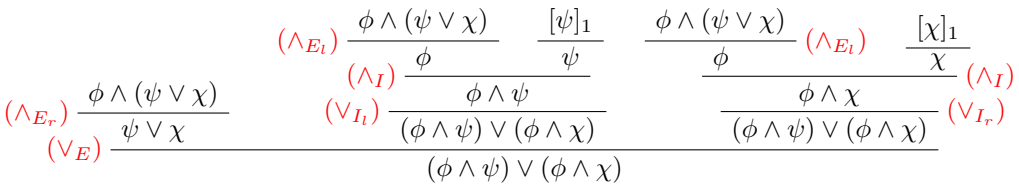
Exercise 3.9. *Show that we can deduce $\phi \wedge (\psi \vee \chi)$ if we start with $(\phi \wedge \psi) \vee (\phi \wedge \chi)$.*

Solution: To save space let $\theta = (\phi \wedge \psi) \vee (\phi \wedge \chi)$.



Exercise 3.10. *Show that we can deduce $(\phi \wedge \psi) \vee (\phi \wedge \chi)$ if we start with $\phi \wedge (\psi \vee \chi)$.*

Solution:



Exercise 3.11. *Complete the proof of theorem 3.3.3 (don't forget the extra axiom, $\neg\neg_E$).*

Solution: There are four rules left to check: $\vee_I$, $\vee_E$, $\rightarrow_E$, and $\neg\neg_E$. Technically $\vee_I$ is two rules, but we'll just do $\vee_{I_l}$ as $\vee_{I_r}$ is essentially the same.

$\vee_I$: Suppose the last step in the proof is an application of $\vee_{I_l}$. Then we have derived ϕ from Γ, and from this we get $\phi \vee \psi$. By the inductive hypothesis we have $\Gamma \models \phi$, and so we obviously have $\Gamma \models \phi \vee \psi$, by properties of truth tables.

$\vee_E$: Suppose the last step is an application of $\vee_E$. Then we have derived $\phi \vee \psi$ from Γ, and we have derived θ from the assumption of $\{\phi\} \cup \Gamma$, and also from the assumption of $\{\psi\} \cup \Gamma$. Since these internal derivations must be shorter than the main derivation, the inductive hypothesis applies. So we have $\Gamma \models \phi \vee \psi$, $\{\phi\} \cup \Gamma \models \theta$, and $\{\psi\} \cup \Gamma \models \theta$. Since $\Gamma \models \phi \vee \psi$ if and only if either $\Gamma \models \phi$ or $\Gamma \models \psi$, any assignment satisfying Γ must also satisfy either $\{\phi\} \cup \Gamma$ or $\{\psi\} \cup \Gamma$. In either case it must satisfy θ too, so we are done.

$\rightarrow_E$: Here we have derived $\phi \rightarrow \psi$ and ϕ from Γ. So, by the inductive hypothesis, any assignment satisfying Γ must also satisfy $\phi \rightarrow \psi$ and ϕ. So any such assignment must also satisfy ψ, and we are done.

$\neg\neg_E$: Here we have $\Gamma \models \neg\neg\phi$, by the inductive hypothesis. Clearly $\Gamma \models \phi$, as an assignment satisfies $\neg\neg\phi$ if and only if it satisfies ϕ.

Exercise 3.12. *Prove that soundness of a deduction system is equivalent to the statement "every satisfiable set of sentences is consistent". HINT: think about how the proof of lemma 3.3.7 works.*

Solution: This argument is essentially the same as the proof of lemma 3.3.7 using lemma 3.3.5. First, soundness is the statement

$$\Gamma \vdash \phi \implies \Gamma \models \phi, \tag{$\dagger$}$$

and "every satisfiable set of sentences is consistent" is $\Gamma \not\models \bot \implies \Gamma \nvdash \bot$, which is equivalent to

$$\Gamma \vdash \bot \implies \Gamma \models \bot. \tag{$\ddagger$}$$

First we show that ($\dagger$) $\implies$ ($\ddagger$). To do this we must, assuming ($\dagger$) and $\Gamma \vdash \bot$, prove that $\Gamma \models \bot$. First, choose $\psi \in \Gamma$. We must be able to do this as if Γ is empty then we would not have $\Gamma \vdash \bot$. Let $\Gamma' = \Gamma \setminus \{\psi\}$. We proceed as follows:

$$\begin{aligned}
\Gamma \vdash \bot &\iff \Gamma' \cup \{\psi\} \vdash \bot \\
&\iff \Gamma' \vdash \neg\psi && \text{by lemma 3.3.5(2)} \\
&\implies \Gamma' \models \neg\psi && \text{by } (\dagger) \\
&\iff \Gamma \models \bot && \text{by lemma 3.3.5(1).}
\end{aligned}$$

This is what we want, so ($\dagger$) $\implies$ ($\ddagger$). Now we must prove that ($\ddagger$) $\implies$ ($\dagger$). To do this we must show that, assuming ($\ddagger$), if $\Gamma \vdash \phi$ then $\Gamma \models \phi$, for any sentence ϕ. We proceed as follows:

$$\begin{aligned}
\Gamma \vdash \phi &\iff \Gamma \vdash \neg\neg\phi && \text{by classical logic} \\
&\iff \Gamma \cup \{\neg\phi\} \vdash \bot && \text{by lemma 3.3.5(2)} \\
&\implies \Gamma \cup \{\neg\phi\} \models \bot && \text{by } (\ddagger) \\
&\iff \Gamma \models \neg\neg\phi && \text{by lemma 3.3.5(1)} \\
&\iff \Gamma \models \phi && \text{because this is true for truth tables.}
\end{aligned}$$

Exercise 3.13 (Compactness theorem for propositional logic)**.** *Use soundness and completeness to prove the following:*

Theorem. *Let Γ be a set of sentences in propositional logic. Then Γ is satisfiable if and only if every finite subset of Γ is satisfiable.*

Solution: Clearly one direction of this is trivial. If Γ is satisfiable then every subset of Γ must be satisfiable, finite or not. Conversely, suppose Γ is *not* satisfiable. Then $\Gamma \models \bot$. By completeness this means $\Gamma \vdash \bot$, so there is a proof of $\bot$ from Γ. But proofs are finite, and so only a finite number of sentences from Γ will be used in this proof. Let Γ' be the finite set of all sentences from Γ that appear in this proof. Then $\Gamma' \vdash \bot$, and so, by soundness, we also have $\Gamma' \models \bot$. But this means Γ' is not satisfiable. So we have shown that Γ not being satisfiable means there is a finite subset of Γ which is not satisfiable, and this is equivalent to saying that if every finite subset of Γ is satisfiable then Γ is also satisfiable, which is what we are trying to prove.

Exercise 3.14. *Let x, y, z be variables, let R, S be relation symbols, let f, g be function symbols, and let c, d be constant symbols. Assume that the arities of relations and functions are correctly represented by the number of arguments they take in each formula. Which of the following are formulas? In the formulas identify the free and bound variables.*

a) $\forall x(f(c, f(x, d))$

b) $R(x, y, z) \vee S(f(c, d))$

c) $\exists y(R(x) \vee \forall z S(f(x, z), c))$

d) $\exists y(R(x) \vee \forall y S(f(x, y), c))$

e) $R(x) \wedge \exists x S(x)$

Solution:

a) This is not a formula.

b) This is a formula, and all its variables are free.

c) This is a formula. Here, x occurs free, and z occurs bound.

d) This is a formula. Here the first $\exists y$ doesn't do anything, as y does not occur free in $R(x) \vee \forall y S(f(x, y), c)$ (it's a *null quantifier*). Here y occurs bound, and x occurs free.

e) This is a formula, and x occurs both free and bound.

Exercise 3.15. *Let $\mathscr{L} = \{0, 1, +, \times\}$ be the language of basic arithmetic. Let $\phi = \forall x(\neg(x \approx 0) \to \exists y(x \times y \approx 1))$. Let $\mathbb{N}$ and $\mathbb{R}$ have their usual meanings, and interpret $\mathscr{L}$ into these languages by giving the non-logical symbols of $\mathscr{L}$ their usual meanings.*

a) Does $\mathbb{N} \models \phi$?

b) Does $\mathbb{R} \models \phi$?

c) Let $n \in \mathbb{N}$ with $n \geq 2$, and let $\mathbb{Z}_n$ be the integers mod n. For what values of n does $\mathbb{Z}_n \models \phi$?

d) Let $A = (\{a, b\}, I)$, where I interprets 0 and 1 as a and b respectively, $b \times b = b$, and $a \times b = a \times a = a$. Does $A \models \phi$?

e) Let $\psi = \exists x \forall y(\neg(y \approx 0) \to (x \times y \approx 1))$. Which of the structures in parts a)-d) is a model for ψ?

Solution:

a) $\mathbb{N}$ does not satisfy ϕ. For example, there is no natural number n such that $2 \times n \approx 1$.

b) $\mathbb{R}$ satisfies ϕ, because every non-zero real number x has a multiplicative inverse $\frac{1}{x}$.

c) Proposition 2.3.5 from the number theory notes gives the answer to this. According to this, a has a multiplicative inverse mod n if and only if a and n are coprime. In order for every non-zero member of $\mathbb{Z}_n$ to have an multiplicative inverse it is necessary and sufficient for n to be prime.

d) In A there is only one non-zero element, and that is b, which corresponds to 1. Since we are told that $b \times b = b$, it follows that $A \models \phi$.

e) ψ says that there is an element x such that whenever y is a non-zero element we have $x \times y = 1$. This is obviously not true in $\mathbb{N}$ or $\mathbb{R}$. It is true for $\mathbb{Z}_2$, and also for A.

Exercise 3.16. *We define logical implication for first-order formulas of a language $\mathscr{L}$ by saying $\phi \models \psi$ if and only if, whenever A is an $\mathscr{L}$-structure and v is an assignment to A, we have*

$$A, v \models \phi \implies A, v \models \psi$$

We define two formulas to be logically equivalent if they both logically imply each other (we write e.g. $\phi \models\!\models \psi$).

Now, let R and S be unary predicates. Let $\phi = \forall x R(x) \vee \forall x S(x)$, and let $\psi = \forall x \forall y (R(x) \vee S(y))$. Prove that $\phi \models\!\models \psi$.

HINT: You don't need to worry about assignments here, because ϕ and ψ are both sentences (the next exercise makes this precise). Just think about what the statements are saying. If ϕ holds in a structure, why must ψ also hold? Conversely, if ψ holds why must ϕ hold?

Solution: Let A be a structure of the appropriate kind. Suppose first that $A \models \phi$. Then either every element of A satisfies R, or every element of A satisfies S. In the former case, for every pair of elements $a, b \in A$ we must have $R(a) \vee S(b)$, because $R(a)$ must be true. So $A \models \forall x \forall y (R(x) \vee S(y))$. This shows $\phi \models \psi$.

Conversely, suppose $A \models \psi$. Suppose that A does not satisfy $\forall x R(x)$. Then there is $a \in A$ with $A \not\models R(a)$. In other words, $R(a)$ is not true. Since $A \models \forall x \forall y (R(x) \vee S(y))$, we must have $A \models \forall y (R(a) \vee S(y))$. As $R(a)$ is not true in A, it follows that $A \models \forall y S(y)$. By changing the variable name we have $A \models \forall x S(x)$, and so $A \models \forall x R(x) \vee \forall x S(x)$. This shows $\psi \models \phi$, and we are done.

Exercise 3.17. *Let ϕ be an $\mathscr{L}$-formula, let A be an $\mathscr{L}$-structure, and let v be an assignment for $\mathscr{L}$ to A with $A, v \models \phi$. Prove that $A, u \models \phi$ for all assignments u such that $u(x) = v(x)$ for all variables x occurring free in ϕ. HINT: You should use induction on the formula construction. First prove that this is true for atomic $\mathscr{L}$-formulas, then, assuming it's true for ϕ and ψ prove it's true for $\neg\phi$, $\phi \vee \psi$, and $\forall x \phi$. This is all we need because of the functional*

completeness of $\{\neg, \vee\}$*, and the fact that* $\exists x\phi \models \neq\!|\neg\forall x\neg\phi$*. This exercise is difficult for people new to formal logic, but it's just a matter of understanding the definitions involved. If you get stuck you need to think carefully about exactly what you are trying to prove.*

It follows easily from this result that if ϕ *is an* $\mathscr{L}$*-sentence, then either* $A, v \models \phi$ *for all* v*, or there is no such* v*.*

Solution: We induct on formula construction. It's obviously true for atomic formulas, because these have no bound variables. Suppose now that it's true for formulas ϕ and ψ.

$\neg\phi$: Suppose $A, v \models \neg\phi$. Then $A, v \not\models \phi$. Suppose that u is an assignment that agrees with v about the free variables of ϕ. Then, if $A, u \models \phi$, by the inductive hypothesis we would have $A, v \models \phi$, which would be a contradiction. So we must have $A, u \models \neg\phi$ as required.

$\phi \vee \psi$ Suppose $A, v \models \phi \vee \psi$. Then, wlog we can assume that $A, v \models \phi$. Let u be an assignment agreeing with v about the free variables of $\phi \vee \psi$. Then it certainly agrees with v about the free variables of ϕ. So $A, u \models \phi$, and thus $A, u \models \phi \vee \psi$ as required.

$\forall x\phi$: Suppose $A, v \models \forall x\phi$, and let u be an assignment agreeing with v about the free variables of $\forall x\phi$. We must show that $A, u \models \forall x\phi$. I.e. that $A, u' \models \phi$ for all u' agreeing with u except possibly at x. Let u' be such an assignment, and let v' be an assignment agreeing with v except possibly at x, where we define $v'(x) = u'(x)$. Then, if F is the set of variables occurring free in ϕ, we have

- v' agrees with v on $F \setminus \{x\}$.
- u agrees with v on $F \setminus \{x\}$.
- u' agrees with u on $F \setminus \{x\}$.
- u' agrees with v' on F.

So,

$$\begin{aligned} A, v \models \forall x\phi &\implies A, v' \models \phi \text{ (by definition of } \models) \\ &\implies A, u' \models \phi \text{ (by the inductive hypothesis),} \end{aligned}$$

and so $A, u \models \forall x\phi$ as required.

$\exists x\phi$: (This is redundant) Suppose $A, v \models \exists x\phi$, and let u be an assignment agreeing with v about the free variables of ϕ. We must show that $A, u \models \exists x\phi$. I.e. that there is u' agreeing with u everywhere except possibly at x such that $A, u' \models \phi$. Now, there is v' agreeing with v except possibly at x and with $A, v' \models \phi$. Define u' so that $u'(x) = v'(x)$, and u' agrees with u everywhere else. Then u' agrees with v' for all free variables of ϕ. Thus $A, u' \models \phi$ by the inductive hypothesis, which is what we want to prove.

Exercise 3.18. *Let ϕ be a formula where x occurs free.*

a) Write down a proof tree that shows $\forall x\phi \vdash \neg\exists x\neg\phi$.

b) Write down a proof tree that shows $\exists x\phi \vdash \neg\forall x\neg\phi$.

Solution:

a)
$$\dfrac{\dfrac{[\exists x\neg\phi]_1 \qquad \dfrac{[\neg\phi[x'/x]]_2 \qquad \dfrac{\forall x\phi}{\phi[x'/x]}\,(\forall_E)}{\bot}\,(\neg_E)}{\bot}\,(\exists_E)}{\neg\exists x\neg\phi}\,(\neg_I)$$

b)
$$\dfrac{\exists x\phi \qquad \dfrac{(\forall_E)\ \dfrac{[\forall x\neg\phi]_1}{\neg\phi[x'/x]} \qquad [\phi[x'/x]]_2}{\bot}\,(\neg_E)}{\bot}\,(\exists_E)}{\neg\forall x\neg\phi}\,(\neg_I)$$

Exercise 3.19. *Assume that lemma 3.3.5 holds for first-order logic. Prove that theorems 3.5.9 and 3.5.10 are equivalent. I.e., prove the equivalence of the statements*

$$\Gamma \textit{ consistent} \iff \Gamma \textit{ is satisfiable} \tag{$\dagger$}$$

$$\Gamma \vdash \phi \iff \Gamma \models \phi \tag{$\ddagger$}$$

Solution: This is similar to exercise 3.12.

$\dagger \implies \ddagger$:

$$\begin{aligned}\Gamma \vdash \phi &\iff \Gamma \cup \{\neg\phi\} \vdash \bot \\ &\iff \Gamma \cup \{\neg\phi\} \models \bot \\ &\iff \Gamma \models \phi\end{aligned}$$

$\ddagger \implies \dagger$:

$$\begin{aligned}\Gamma \vdash \bot &\iff \Gamma \setminus \{\phi\} \cup \{\phi\} \vdash \bot \\ &\iff \Gamma \setminus \{\phi\} \vdash \neg\phi \\ &\iff \Gamma \setminus \{\phi\} \models \neg\phi \\ &\iff \Gamma \models \bot\end{aligned}$$

Exercise 3.20. *Prove that if Γ is an $\mathscr{L}$-theory then there is an $\mathscr{L}$-theory Γ' with $\Gamma \subseteq \Gamma'$ such that Γ' is* complete *(i.e. if ϕ is an $\mathscr{L}$-sentence, then either $\phi \in \Gamma'$ or $\neg\phi \in \Gamma'$). HINT: If Γ is consistent then it must have a .....?*

Solution: An $\mathscr{L}$-theory is a satisfiable set of $\mathscr{L}$-sentences. Since Γ is satisfiable, it must have a model. Let A be a model for Γ, and let

$$\Gamma' = \{\phi : \phi \text{ is an } \mathscr{L}\text{-sentence and } A \models \phi\}.$$

Then $\Gamma \subseteq \Gamma'$, because $A \models \Gamma$, and Γ' is complete because every $\mathscr{L}$-sentence is either true or false in A.

Exercise 3.21. *Let Γ be an $\mathscr{L}$-theory, and let ϕ be an $\mathscr{L}$-sentence. Prove that if $\Gamma \models \phi$ then $\Delta \models \phi$ for some finite $\Delta \subseteq \Gamma$.*

Solution: Suppose $\Gamma \models \phi$. Then, by completeness we have $\Gamma \vdash \phi$. So there is a deduction tree using Γ that proves ϕ. As deduction trees are finite, this tree involves only a finite number of sentences from Γ. Define Δ to be the set of sentences from Γ used in the proof of ϕ. Then $\Delta \vdash \phi$, and so $\Delta \models \phi$ by soundness.

Exercise 3.22 (Compactness theorem for first-order logic)**.** *Let Γ be a set of $\mathscr{L}$-sentences. Prove that Γ has a model if and only if every finite subset of Γ has a model.*

Solution: This is very similar to exercise 3.13. First, if Γ has a model then every subset of Γ obviously has a model too. Conversely, suppose Γ does *not* have a model. Then $\Gamma \models \bot$. So by exercise 4 there is finite $\Delta \subseteq \Gamma$ with $\Delta \models \bot$. I.e. Δ does not have a model.

Linear Algebra

Exercise 4.1. *Prove lemma 4.1.3(6).*

Solution: Let $\alpha = a + bi$, let $\beta = c + di$, and let $\gamma = e + fi$.

6)

$$\begin{aligned}(a+bi)((c+di)+(e+fi)) &= (a+bi)((c+e)+(d+f)i)\\ &= a(c+e) - b(d+f) + (b(c+e) + a(d+f))i\\ &= ac + ae - bd - bf + (bc + be + ad + af)i.\end{aligned}$$

Also,

$$\begin{aligned}(a+bi)(c+di) + (a+bi)(e+fi) &= (ac-bd) + (ad+bc)i + (ae-bf) + (af+be)i\\ &= ac + ae - bd - bf + (ad + bc + af + be)i.\end{aligned}$$

These two things are the same, so we have distributivity.

Exercise 4.2. *Consider the following 'proof'. What is wrong with it?*

$$-1 = i^2 = i.i = \sqrt{-1}\sqrt{-1} = \sqrt{(-1)(-1)} = \sqrt{1} = 1.$$

Solution: The problem is that $(-1 \times -1)^{\frac{1}{2}}$ does not equal $(-1)^{\frac{1}{2}}(-1)^{\frac{1}{2}}$. We think it should because when $a, b, c \in \mathbb{R}$ are all positive we have $(ab)^c = a^c b^c$ (it's not obvious that this is true for irrational values of c, or even exactly what it means to take an irrational power, but it's obviously true for rational c, and it turns out to be true for irrational values too). What the argument above shows it that this is no longer always true when we have negative values for a and b.

Exercise 4.3. *Complete the proof of proposition 4.1.7.*

Solution: We must show $-1v = -v$ for all $v \in V$. By proposition 4.1.7(3) we have $0v = 0$. So $(1-1)v = 0$, and so $v + (-1)v = 0$ by definition 4.1.5(8) and (6). So $(-1)v = -v$ by proposition 4.1.7(2).

Exercise 4.4. *Given $v \in V$, prove that $-(-v) = v$.*

Solution: We know $v + (-v) = 0$, so $-(-v) = v$, as additive inverses are unique, by proposition 4.1.7(2).

Exercise 4.5. *Given $a \in \mathbb{F}$ and $v \in V$ prove that $av = 0$ if and only if either $a = 0$ or $v = 0$.*

Solution: If $a = 0$ then $av = 0$ by proposition 4.1.7(3). Now, let $v = 0$, and suppose $a \neq 0$. Let w be any vector. Then $a0 + w = a0 + aa^{-1}w = a(0 + a^{-1}w) = a(a^{-1}w) = 1w = w$. So $a0 = 0$, as the zero of a vector space is unique (by proposition 4.1.7(1)). Conversely, suppose $av = 0$ and that $a \neq 0$. Then $a^{-1}av = a^{-1}0 = 0$, and so $v = 0$, as $aa^{-1} = 1$.

Exercise 4.6. *Let U and W be subspaces of V. Prove that $U \cap W$ is a subspace of V.*

Solution: We need to check the three conditions of definition 4.1.8. First, 0 is obviously in $U \cap W$, as both U and W are subspaces. Second, if $u, v \in U \cap W$, then $u, v \in U$ and $u, v \in W$, and, as both U and W are closed under addition (as they are subspaces), we have $u + v \in U \cap W$. Finally, if $v \in U \cap W$ then $v \in U$ and $v \in W$, so $\alpha v \in U \cap W$ for all α as both U and W are closed under scalar multiplication.

Exercise 4.7. *(Optional) Let U and W be subspaces of V. Prove that if $U \cup W$ is a subspace of V, then either $U \subseteq W$ or $W \subseteq U$.*

Solution: Suppose $U \cup W$ is a subspace of V, and suppose U is not a subspace of W. Choose $u \in U \setminus W$, and let $w \in W$. Then $u + w \in U \cup W$, as $U \cup W$ is a subspace, and so is closed under $+$. So either $u + w \in U$ or $u + w \in W$. If $u + w \in W$, then $u + w - w = u$ is also in W, but this contradicts the choice of u. So $u + w \in U$, and so $u + w - u = w \in U$. This is true for all $w \in W$, so W is a subspace of U.

Exercise 4.8. *(Optional) Let V be vector space over $\mathbb{F}$, and let $v_1, \ldots v_n \in V$ such that $(v_1, \ldots, v_n)$ is linearly independent. Let $w \in V$. Prove that $(v_1, \ldots, v_n, w)$ is linearly independent if and only if $w \notin \operatorname{span}(v_1, \ldots, v_n)$.*

Solution: If $w \in \text{span}(v_1, \ldots, v_n)$, then $w = a_1v_1 + \ldots + a_nv_n$ for some $a_1, \ldots, a_n$, and so $0 = (-1)w + a_1v_1 + \ldots + a_nv_n$, and so $(v_1, \ldots, v_n, w)$ is not linearly independent. Conversely, if $(v_1, \ldots, v_n, w)$ is not linearly independent then we have $a_0w + a_1v_1 + \ldots + a_nv_n = 0$ for some $a_0, \ldots, a_n$ not all zero, and a_0 cannot be zero, as $(v_1, \ldots, v_n)$ is linearly independent. So $w = \frac{-a_1}{a_0}v_1 + \ldots + \frac{-a_n}{a_0}v_n$, and is therefore in $\text{span}(v_1, \ldots, v_n)$.

Exercise 4.9. *For $n \in \mathbb{N}$, define $\mathbb{R}_n[x]$ to be the set of all polynomials of degree at most n. Write down a basis for $\mathbb{R}_6[x]$. Is it possible for a list of 8 polynomials over $\mathbb{R}$ of degree at most 6 to be linearly independent?*

Solution: We can use $(1, x, x^2, \ldots, x^6)$. It is not possible for a list of 8 polynomials of degree at most 6 to be linearly independent. To prove, this notice that we have shown that there is a spanning set of 7 elements, and by proposition 2.4 a linearly independent list can't be bigger than a spanning list.

Exercise 4.10. *Let (v_1, v_2, v_3, v_4) be a basis for V. Prove that*

$$(v_1 + v_2, v_2 + v_3, v_3 + v_4, v_4)$$

is also a basis for V.

Solution: By theorem 4.2.11, a linearly independent list with the right size is also a basis. Since $(v_1 + v_2, v_2 + v_3, v_3 + v_4, v_4)$ has the right size, we just need to show it is linearly independent (alternatively we could show it spans). Suppose

$$a_1(v_1 + v_2) + a_2(v_2 + v_3) + a_3(v_3 + v_4) + a_4v_4 = 0.$$

Then

$$a_1v_1 + (a_1 + a_2)v_2 + (a_2 + a_3)v_3 + (a_3 + a_4)v_4 = 0,$$

and, as (v_1, v_2, v_3, v_4) is linearly independent, it follows that $a_1 = (a_1 + a_2) = (a_2 + a_3) = (a_3 + a_4) = 0$. From this we easily see that $a_i = 0$ for all $i = 1, 2, 3, 4$. Thus $(v_1 + v_2, v_2 + v_3, v_3 + v_4, v_4)$ is linearly independent as required.

Exercise 4.11. *Let U and W be subspaces of V and suppose that $V = U \oplus W$. Let $(u_1, \ldots, u_k)$ be a basis for U, and let $(w_1, \ldots, w_m)$ be a basis for W. Prove that $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ is a basis for V.*

Solution: We will first show that $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ is linearly independent. Suppose $0 = a_1u_1 + \ldots + a_ku_k + b_1w_1 + \ldots + b_mw_m$. Since $U \oplus W$ is a direct sum, by definition there is a unique $u \in U$ and $w \in W$ with $u + w = 0$, and this must be $u = 0$ and $w = 0$. So $b_1w_1 + \ldots + b_mw_m = 0$ and $a_1u_1 + \ldots + a_ku_k = 0$. But as $(u_1, \ldots, u_k)$ is a basis for U and $(w_1, \ldots, w_m)$ is a basis for W, we must have $a_1 = \ldots = a_k = b_1 = \ldots = b_m = 0$. But this means $(u_1, \ldots, u_k, w_1, \ldots, w_m)$ is linearly independent as claimed.

All we need to do now is show the list spans V. Let $v \in V$. Then by lemma 4.1.14 there is $u \in U$ and $w \in W$ with $v = u + w$. So, as $(u_1, \ldots, u_k)$ and $(w_1, \ldots, w_m)$ span U and W respectively, we have $b_1w_1 + \ldots + b_mw_m = w$ and $a_1u_1 + \ldots + a_ku_k = u$, for some choice of coefficients. But this means $v = a_1u_1 + \ldots + a_ku_k + b_1w_1 + \ldots + b_mw_m$, so the list does span v.

Exercise 4.12. *Let U and W be subspaces of $\mathbb{R}^8$, and suppose $\dim(U) = 5$, $\dim(W) = 3$, and $U \cap W = \{0\}$. Prove that $\mathbb{R}^8 = U \oplus W$.*

Solution: Let $(u_1, \ldots, u_5)$ be a basis for U, and let (w_1, w_2, w_3) be a basis for W. By lemma 4.1.14, $U + W$ is a direct sum, so $(u_1, \ldots, u_5, w_1, w_2, w_3)$ is linearly independent. It also has 8 elements, which is the same as the dimension of $\mathbb{R}^8$. So, by theorem 4.2.11, $(u_1, \ldots, u_5, w_1, w_2, w_3)$ is a basis for $\mathbb{R}^8$, and so $U \oplus W = \mathbb{R}^8$.

Exercise 4.13. *Let V be a finite dimensional vector space and $\dim(V) = n > 0$. Show that $V = U_1 \oplus \ldots \oplus U_n$, for some set $\{U_1, \ldots, U_n\}$ of one-dimensional subspaces.*

Solution: Let $(v_1, \ldots, v_n)$ be a basis for V. For each $i \in \{1 \ldots, n\}$, let $U_i = \text{span}(v_i)$. Then $V = U_1 + \ldots + U_n$, as a basis spans V, by definition. Also, if $0 = a_1 v_n + \ldots + a_n v_n$ then $a_1 = \ldots = a_n = 0$, by linear independence, so the sum is direct, by lemma 4.1.13.

Exercise 4.14. *Let $b, c \in \mathbb{R}$. Define $T : \mathbb{R}^3 \to \mathbb{R}^2$ by*

$$T(x, y, z) = (2x - 4y + 3z + b, 6x + cxyz).$$

Prove that T is linear if and only if $b = c = 0$.

Solution: If $b = c = 0$ then $T(x, y, z) = (2x - 4y + 3z, 6x)$. We check the conditions of definition 4.3.1. First,

$$\begin{aligned} T(x_1 + x_2, y_1 + y_2, z_1 + z_2) &= (2(x_1 + x_2) - 4(y_1 + y_2) + 3(z_1 + z_2), 6(x_1 + x_2)) \\ &= (2x_1 - 4y_1 + 3z_1, 6x_1) + (2x_2 - 4y_2 + 3z_2, 6x_2) \\ &= T(x_1, y_1, z_1) + T(x_2, y_2, z_2). \end{aligned}$$

Second,

$$\begin{aligned} T(\lambda x, \lambda y, \lambda z) &= (2\lambda x - 4\lambda y + 3\lambda z, 6\lambda x) \\ &= \lambda(2x - 4y + 3z, 6x) \\ &= \lambda T(x, y, z). \end{aligned}$$

Conversely, if T is linear then $2T(1, 0, 0) = T(2, 0, 0)$, and so $2(2 + b, 6) = (4+b, 12)$. I.e. $(4+2b, 12) = (4+b, 12)$, and so b must be zero. Also, $T(1, 1, 1) = T(1, 0, 0) + T(0, 1, 1)$. So

$$(2 - 4 + 3, 6 + c) = (2, 6) + (-4 + 3, 0) = (2 - 4 + 3, 6),$$

so $c = 0$.

Exercise 4.15. *Let $T \in \mathcal{L}(V, W)$. Let $v_1, \ldots, v_n \in V$ and suppose that*

$$(T(v_1), \ldots, T(v_n))$$

is linearly independent in W. Prove that $(v_1, \ldots, v_n)$ is linearly independent in V.

Solution: Suppose $a_1v_1 + \ldots + a_nv_n = 0$. Then, as T is linear, we have

$$a_1T(v_1) + \ldots + a_nT(v_n) = T(a_1v_1 + \ldots + a_nv_n) = T(0) = 0.$$

As $T(v_1), \ldots, T(v_n)$ is linearly independent, it follows that $a_1 = \ldots = a_n = 0$. So $(v_1, \ldots, v_n)$ is also linearly independent.

Exercise 4.16. *Prove lemma 4.3.8.*

Solution: We have $0 \in \operatorname{null} T$ by lemma 4.3.4. Also, if $T(v) = 0$ then

$$T(\lambda v) = \lambda T(v) = \lambda.0 = 0,$$

so T is closed under scalar multiplication. Also, if $T(u) = T(v) = 0$, then

$$T(u + v) = T(u) + T(v) = 0 + 0 = 0,$$

so T is closed under addition.

Exercise 4.17. *Prove proposition 4.3.14.*

Solution: What should the transformation TS do to the basis vector u_i of U? As A is the matrix of S, to find Su_i we look at what A does to the column vector that is zeroes except for 1 in the ith place. So the result is $a_{1i}v_1 + \ldots + a_{mi}v_m$. What does T do to a basis vector v_j of V? Now we look at the matrix B, which tells us that $T(v_j) = b_{1j}w_1 + \ldots + b_{pj}w_p$. So,

$$\begin{aligned} TS(u_i) &= T(a_{1i}v_1 + \ldots + a_{mi}v_m) \\ &= a_{1i}T(v_1) + \ldots + a_{mi}T(v_m) \\ &= a_{1i}(b_{11}w_1 + \ldots + b_{p1}w_p) + \ldots + a_{mi}(b_{1m}w_1 + \ldots + b_{pm}w_p). \end{aligned}$$

We can rearrange this as

$$\begin{aligned} &(a_{1i}b_{11} + \ldots + a_{mi}b_{1m})w_1 \\ &+(a_{1i}b_{21} + \ldots + a_{mi}b_{2m})w_2 \\ &+\ldots \\ &+(a_{1i}b_{p1} + \ldots + a_{mi}b_{pm})w_p. \end{aligned}$$

But

$$\begin{bmatrix} a_{1i}b_{11} + \ldots + a_{mi}b_{1m} \\ a_{1i}b_{21} + \ldots + a_{mi}b_{2m} \\ \vdots \\ a_{1i}b_{p1} + \ldots + a_{mi}b_{pm} \end{bmatrix}$$

is the ith column of the matrix BA. Since this is true for every basis vector u_i of U, the transformation TS is given by the matrix BA as claimed.

Exercise 4.18. *Let $T \in \mathcal{L}(V, W)$, and suppose both V and W are finite dimensional. Prove that, whatever the choice of bases for V and W, the matrix of T with respect to these bases must have at least* $\dim \operatorname{ran} T$ *entries that are not equal to* 0.

Solution: Let A be the matrix of T with respect to some pair of bases. If the ith column of A is all zeroes, then this means $T(v_i) = 0$, where v_i is the ith basis vector for V. Since $(T(v_1), \ldots, T(v_n))$ spans $\operatorname{ran} T$, there must be at least $\dim \operatorname{ran} T$ columns of A that are not all zeroes. This requires at least $\dim \operatorname{ran} T$ non-zero entries.

Exercise 4.19. *Let a, b, c, d be positive real numbers. Use Cauchy-Schwarz (theorem 4.4.13) to prove that*

$$16 \leq (a+b+c+d)(\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}).$$

Solution: Let $u = (\sqrt{a}, \sqrt{b}, \sqrt{c}, \sqrt{d})$, and let $v = (\frac{1}{\sqrt{a}}, \frac{1}{\sqrt{b}}\frac{1}{\sqrt{c}}, \frac{1}{\sqrt{d}})$. Then $\langle u, v\rangle^2 = (1+1+1+1)^2 = 16$. Also, $\|u\|^2 = \langle u, u\rangle = a+b+c+d$, and $\|v\|^2 = \langle v, v\rangle = \frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}$. By Cauchy-Schwarz we have $\langle u, v\rangle^2 \leq \|u\|^2\|v\|^2$. I.e. $16 \leq (a+b+c+d)(\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d})$, which is what we want to prove.

Exercise 4.20. *Let $x_1, \ldots, x_n \in \mathbb{R}$. Prove that $(x_1 + \ldots + x_n)^2 \leq n(x_1^2 + \ldots + x_n^2)$.*

Solution: Let $u = (x_1, \ldots, x_n)$, and let $v = (1, \ldots, 1)$. So $\langle u, v\rangle^2 = (x_1 + \ldots + x_n)^2$. Also, $\|u\|^2 = x_1^2 + \ldots + x_n^2$, and $\|v\|^2 = 1 + 1 + \ldots + 1 = n$. So, by Cauchy-Schwarz, we have $(x_1 + \ldots + x_n)^2 \leq n(x_1^2 + \ldots + x_n^2)$ as claimed.

Exercise 4.21. *Is there an inner product on $\mathbb{R}^2$ such that the associated norm is given by $\|(x, y)\| = \max\{x, y\}$? Provide a proof for your answer.*

Solution: No. For example, think about when both x and y are negative. Then $\max\{x, y\}$ is also negative, but norms are never negative.

Exercise 4.22. *Let V be a real inner product space.*

(a) Prove that $\langle u+v, u-v\rangle = \|u\|^2 - \|v\|^2$ for all $u, v \in V$.

(b) A rhombus is a parallelogram whose four sides all have equal length. Prove that the diagonals of a rhombus are orthogonal to each other.

(c) Prove that

$$\langle u, v\rangle = \frac{\|u+v\|^2 - \|u-v\|^2}{4}$$

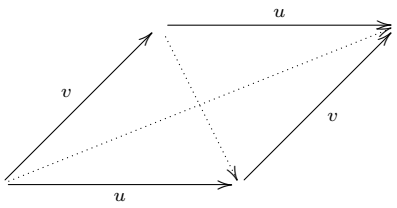
for all $u, v \in V$.

Solution:

(a)

$$\begin{aligned}\langle u+v, u-v\rangle &= \langle u, u-v\rangle + \langle v, u-v\rangle \\ &= \langle u, u\rangle - \langle u, v\rangle + \langle v, u\rangle - \langle v, v\rangle \\ &= \|u\|^2 - \|v\|^2.\end{aligned}$$

(b) Think about this picture:



This is a rhombus. The up-right diagonal is given by $u+v$, and the down-right diagonal is given by $u-v$. By part (a) we have $\langle u+v, u-v\rangle = \|u\|^2 - \|v\|^2$, and by the assumption that u and v are the same length we have $\|u\|^2 = \|v\|^2$. So $\langle u+v, u-v\rangle = 0$, which means the diagonals are orthogonal to each other.

(c) We could work through this calculation directly, but with a little cleverness we can use a trick and save some effort. Set $x = \frac{u+v}{2}$, and set $y = \frac{u-v}{2}$. Then $\langle u, v\rangle = \langle x+y, x-y\rangle$, and by part (a) we have

$$\langle x+y, x-y\rangle = \|x\|^2 - \|y\|^2 = \frac{\|u+v\|^2}{4} - \frac{\|u-v\|^2}{4}.$$

Counting

Exercise 5.1. *Prove that if X and Y are disjoint finite sets, then the cardinal arithmetic operations agree with the usual arithmetic operations on $|X|$ and $|Y|$. In other words, prove that $|X \cup Y|$ is equal to the result of adding $|X|$ and $|Y|$ as normal, and do similar for the other two arithmetic operations we have defined.*

Solution:

$|X|+|Y|$: Since X and Y are disjoint, if $z \in X \cup Y$ then either $z \in X$, or $z \in Y$, but not both. So, if $|X| = m$ and $|Y| = n$, we have $|X \cup Y| = m+n = |X|+|Y|$ as required.

$|X| \times |Y|$: $X \times Y = \{(x, y) : x \in X \text{ and } y \in Y\}$. If $|X| = m$ and $|Y| = n$ then $|X \times Y| = m \times n = |X| \times |Y|$.

$|X|^{|Y|}$: X^Y is the set of all functions from Y to X. How many functions are there? Well, such a function must map each element of Y to exactly one element of X. So, for each $y \in Y$ there are exactly $|X|$ choices. So, if $|X| = m$ and $|Y| = n$, we get m^n different functions. So $|X^Y| = m^n$ as required.

Exercise 5.2. *Let X_i be countable for all $i \in \mathbb{N}$, and suppose $X_i \cap X_j = \emptyset$ for all $i \neq j \in \mathbb{N}$. Prove that $\bigcup_{i\in\mathbb{N}} X_i$ is countable. HINT: We don't need the condition that $X_i \cap X_j = \emptyset$, but it makes the notation slightly simpler. Think about the proof that $|\mathbb{N}| = |\mathbb{N} \times \mathbb{N}|$.*

Solution: We need to find an injective function from $\bigcup_{i\in\mathbb{N}} X_i$ to $\mathbb{N}$. Since $\mathbb{N}\times\mathbb{N}$ is countable, there is an injective $f:\mathbb{N}\times\mathbb{N}\to\mathbb{N}$. Since each X_i is countable, there are injective functions $g_i : X_i \to \mathbb{N}$ for all $i \in \mathbb{N}$. Define a function $g : \bigcup_{i\in\mathbb{N}} X_i \to \mathbb{N}\times\mathbb{N}$ by $g(x) = (i, g_i(x))$, where $x \in X_i$. This is well defined because $X_i \cap X_j = \emptyset$ for all $i \neq j$. Then g is injective, because, given $x_1 \in X_i$ and $x_2 \in X_j$ with $x_1 \neq x_2$, if $i \neq j$ then $(i, g_i(x_1)) \neq (j, g_j(x_2))$, as $i \neq j$, and if $i = j$ then $g_i(x_1) \neq g_i(x_2)$, as g_i is injective. So $f \circ g : \bigcup_{i\in\mathbb{N}} X_i \to \mathbb{N}$ is the composition of two injective functions, and so is injective. This gives us the injective function we need.

Exercise 5.3. *Let X be a countable set. Prove that the set of all finite subsets of X is countable.*

Solution: Let $f : X \to \mathbb{N}$ be injective. Arrange the prime numbers in a list as $p_0, p_1, \ldots$. We know from number theory that the set of primes is infinite, so this is a countably infinite list. Given $S = \{x_1, \ldots, x_n\} \subseteq X$, define $g(S) = p_{f(x_0)} \times p_{f(x_1)} \times \ldots \times p_{f(x_n)}$. Then g is a function from the set of all finite subsets of X to $\mathbb{N}$. Moreover, g is injective, because if $S_1 \neq S_2$ then $g(S_1)$ and $g(S_2)$ will have different prime factorizations, and so we know from the Fundamental Theorem of Arithmetic that this means they must be different numbers.

Exercise 5.4. *Let X be a set, let $\wp(X)$ be the powerset of X.*

a) Define a simple injective function from X to $\wp(X)$.

b) Prove that there is no surjective function from X to $\wp(X)$. HINT: Suppose such a function exists. Can you derive a contradiction? The argument used here is similar to the one used in Russell's paradox.

c) What does this tell us about the relationship between $|X|$ and $|\wp(X)|$?

Solution:

a) Use e.g. $x \mapsto \{x\}$.

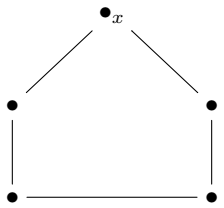
b) Suppose $f : X \to \wp(X)$ is surjective. Let $Z = \{x \in X : x \notin f(x)\}$. Then, as f is surjective, there is $z \in X$ with $f(z) = Z$. Suppose $z \in Z$. Then, by definition of Z and z, we must have $z \notin Z$. On the other hand, if $z \notin Z$, then $z \in f(z) = Z$, which is also a contradiction. We conclude there can be no such surjective function.

c) We see that $|X| < |\wp(X)|$. In other words, X is strictly smaller than $\wp(X)$.

Exercise 5.5. *Show that there is a group of five people containing neither three mutual friends, nor three mutual enemies.*

Solution: Look at the graph in the picture, where edges represent friendship, and consider the vertex x. Then x cannot be part of a group of three mutual friends, because x only has two friends, and they are not friends with

each other. Also, x can't be part of a group of three mutual enemies, because x only has two enemies, and they are friends with each other. From the shape of the graph, this reasoning obviously applies to every vertex, so the graph has the required property.



Exercise 5.6. *Show that $R(2, n) = n$ for all $n \in \mathbb{N}$ with $n \geq 2$.*

Solution: Given a group of n people, if none of the people are friends, then the whole group is mutual enemies. This proves $R(2, n) \leq n$. Also, given a group of $n - 1$ people, if none of them are friends than there are only $n - 1$ mutual enemies. This proves $R(2, n) \geq n$, and so $R(2, n) = n$ as claimed.

Exercise 5.7. *Suppose there are 5 points in a $1cm \times 1cm$ square. Prove that there must be two points at most $\frac{\sqrt{2}}{2}cm$ apart.*

Solution: Divide the square into four quarters. Since there are five points, at least two must be in the same quarter. If two points are in the same quarter then they are both inside a square whose edges are 0.5cm long. The furthest apart they can be is the length of the diagonal of this square, which is $\sqrt{(\frac{1}{2})^2 + (\frac{1}{2})^2}cm$, which is $\frac{\sqrt{2}}{2}cm$.

Exercise 5.8. *Suppose we have a chessboard with two diagonally opposite squares removed. Is it possible to tile the board using domino pieces?*

Solution: No. Every domino must cover one white square and one black square, so we can only ever cover the same number of black and white squares with dominoes. If we remove two diagonally opposite squares then we remove two squares that are the same colour, and there are not the same number of squares for each colour anymore.

Exercise 5.9 (Pascal's identity)**.** *Prove that $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$.*

Solution: There are two basic ways to do this. First, a purely formal proof:

$$\begin{aligned}\binom{n}{k-1}+\binom{n}{k} &= \frac{n!}{(k-1)!(n-(k-1))!}+\frac{n!}{k!(n-k)!}\\ &= \frac{n!}{(k-1)!(n+1-k)!}+\frac{n!}{k!(n-k)!}\\ &= \frac{kn!+(n+1-k)n!}{k!(n+1-k)!}\\ &= \frac{(n+1)n!}{k!(n+1-k)!}\\ &= \frac{(n+1)!}{k!((n+1)-k)!}\\ &= \binom{n+1}{k}.\end{aligned}$$

Alternatively, we can think conceptually. Let X be a set of $n+1$ objects. Then $\binom{n+1}{k}$ is the number of ways we can choose k objects from X when the order doesn't matter. Let x be some element of X. When we choose k elements of X, we can either choose x or not. This gives two cases. First, if we choose x, then we have to choose $k-1$ elements from the remaining n elements of $X \setminus \{x\}$. There are $\binom{n}{k-1}$ ways to do this. Alternatively, if we don't choose x, then we choose all k elements from the remaining n elements of $X \setminus \{x\}$. There are $\binom{n}{k}$ ways to do this. It follows that $\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$ as claimed.

Exercise 5.10. *Use theorem 5.2.14 to find the number of solutions to the equation $a+b+c+d=17$, where a, b, c, d must all be natural numbers (HINT: think about distributing 17 pebbles into boxes marked a, b, c, d). How many solutions are there if we demand that a, b, c, d are all greater than or equal to 2?*

Solution: This is putting 17 balls in 4 boxes. Using theorem 2.14 the answer is $\binom{4+17-1}{17} = \binom{20}{17} = 1140$. Forcing each of a, b, c, d to be at least 2 is equivalent to fixing 8 of the pebbles, which leaves 9. So the answer to the second part is $\binom{4+9-1}{9} = \binom{12}{9} = 220$.

Exercise 5.11. *Suppose we have 5 points in the xy plane, all with integer coordinates. Show that the midpoint of one of the straight lines connecting pairs of these points also has integer coordinates.*

Solution: Given two points, (x_1, y_1) and (x_2, y_2), the midpoint is $(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2})$. So, to have a midpoint with integer coordinates we require $x_1 + x_2$ and $y_1 + y_2$ to both be even. The *parity* of an integer is another word for saying whether it is odd or even. Now, $x_1 + x_2$ is even if and only if the parity of x_1 is the same as the parity of x_2. The same is true for $y_1 + y_2$. So the exact values of the coordinates for each point is not important, just their parities. For a point, there are four possible parity combinations: (odd, odd), (odd, even), (even, odd), (even, even). So, if we have five points, then, by the pigeon hole principle, at least two of the points must have the same parity in each coordinate. This will produce a midpoint with integer coordinates as required.

References

[1] Sheldon Axler. *Linear algebra done right*. Undergraduate Texts in Mathematics. Springer, Cham, third edition, 2015. doi:10.1007/978-3-319-11080-6.

[2] S. Dasgupta, C.H Papadimitriou, and U. Vazirani. *Algorithms*. McGraw-Hill Education, 2006. URL: http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf.

[3] T. Gowers. Proving the fundamental theorem of arithmetic, 2011. URL: https://gowers.wordpress.com/2011/11/18/proving-the-fundamental-theorem-of-arithmetic/.

[4] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete mathematics*. Addison-Wesley Publishing Company, Reading, MA, second edition, 1994. A foundation for computer science.

[5] Paul R. Halmos. *Naive set theory*. Springer-Verlag, New York-Heidelberg, 1974. Reprint of the 1960 edition, Undergraduate Texts in Mathematics.

[6] Joseph Y. Halpern, Robert Harper, Neil Immerman, Phokion G. Kolaitis, Moshe Y. Vardi, and Victor Vianu. On the unusual effectiveness of logic in computer science. *Bull. Symbolic Logic*, 7(2):213–236, 2001. doi:10.2307/2687775.

[7] Imre Lakatos. *Proofs and refutations*. Cambridge University Press, Cambridge-New York-Melbourne, 1976. The logic of mathematical discovery, Edited by John Worrall and Elie Zahar.

[8] E. Lehman, F. Thompson Leighton, and A. Meyer. *Mathematics for computer science*. Samurai Media Limited, 2017. URL: https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-spring-2015/readings/MIT6_042JS15_textbook.pdf.

[9] P. Teller. *A Modern Formal Logic Primer*. Prentice Hall, 1989. URL: http://tellerprimer.ucdavis.edu/.

[10] Sergei Treil. *Linear Algebra Done Wrong*. 2017.