Elements of Discrete Mathematics

E. Dashkov

December 31, 2022

Contents

1 Truth, Falsity, and Propositions

This lecture gives an informal introduction to logic. It is evident from our experience that such an introduction is of much help if the Instructor prefers a formal presentation style later in the Course (as we do). Surely, many details are necessarily omitted from here (most notably, any formal definition of truth). We suggest the Instructor focus on discussing examples and ideas rather than rigorous definitions. It is also recommended to progress gradually from statements about the 'real world' to those of 'mathematical' content.

Our first objective is to see the most important logical constructions in the natural language when applied to both everyday matters and mathematics. We are well aware that the backbone of our everyday speech is formed by *statements* (also known as *declarative sentences*). These do declare or state something unlike interrogative sentences, which pose a question, or imperative ones, those are to persuade you to do something. For example,

The quick brown fox jumps over the lazy dog

is an example of a statement. Another one is

Today is Thursday and $2 + 3$ equals 5.

Most basically, some statements are *true* and all the others are *false.* Grammatically, you can pose a *general question* to a statement (*Does the quick brown fox jump over the lazy dog?*), whose required answer labels the statement either as true (*Yes, it does*) or as false (*No, it doesn't*). The distinction between *true* and *false* is very well known while being one of the hardest to define.

Example 1.1. It is true that $2 + 3$ *equals* 5, as far as the arithmetic holds true and we understand the numbers correctly. It depends on the current date whether *today is Thursday.* It is still not known to the science if *each perfect*[1] *number is even.*

The sentence *this sentence is true* can be either true or false without a contradiction but there hardly is a way to check which possibility takes place. The sentence *this sentence is false* can be neither true nor false, for either assumption leads to a contradiction: if the statement is true, then it must be false as it says, etc.

It is not recommended to discuss any paradoxes at length.

Hence, our 'most basic' analysis is far from being exhaustive. Nevertheless, neither unsolved mathematical problems nor artificial self-referential examples can prevent us from gaining command of logic.

The reason for that is that we are going to study just well-behaved *models* of real-world sentences. We reserve the name 'statements' for such models and reiterate that *every statement is either true or false but not both.*

Some statements are *compound*, which means they are composed of one or more simpler statements. E.g., 2 *equals* 3 *or* 1 *equals* 1; *it is not the case that* 2 *equals* 1; *somebody believes that* 2 *equals* 3.

Let us consider the constructions able to produce compound statements, such as *... or ...*, *... but ...*, *if ... then ...*, *it is not the case that ...*, *somebody believes that ...*, etc. Any such construction is called a *(logical) connective* if the truth of the compound statement depends exclusively on which of the simpler statements are true.

[1] A natural number is *perfect* if it equals the sum of its divisors except itself, like $6 = 1+2+3$ and $28 = 1+2+4+7+14$ do.

Example 1.2. When is it the case that A *or* B is true (where A and B stand for some arbitrary statements)? If and only if at least one of the statements A and B is true. So, . . . *or* . . . is a logical connective. Yet *each student knows that* . . . is not. Indeed, one can consider the statements $0 = 0$ and $\pi < 3.14159265358979323847$. Both are true, whereas *each student knows that* $0 = 0$ is likely true and *each student knows that* $\pi < 3.14159265358979323847$ is likely false. That is, a statement's A being true does not suffice to make the statement *each student knows that* A either true or false.

We call a statement a *(logical) atom*, if it cannot reasonably be split into simpler ones using logical connectives. So, 2 *equals* 3 *or* 1 *equals* 1 is not an atom, yet $2 = 3$ and *somebody believes that* 2 *equals* 3 are.

Now, we are going to fix the traditional meaning of the most important connectives found in mathematics. The meaning of a connective is nothing more than the way the compound statement's truth depends on the truths of its parts. From now on, we will use letters to denote arbitrary statements: e.g., *A and B*; *if A then B*. We will use 1 for *truth* and 0 for *falsity.* So we obtain the following *truth table.*

A	B	not A	A and B	A or B	if A then B	A if and only if B
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Exercise 1.3. Construct a truth table for the compound statement *if not (A or B), then C and A.*

There are plenty of constructions in natural languages which can be seen as logical connectives. In general, this costs them a part of their meaning. (Is it true that $0 = 0$ *although* $1 = 1$?) The following tables show a few roughly equivalent ways to present basic logical connectives in English as well as the logical symbols and common names for the latter.

Connective	*negation*	*conjunction*	*disjunction*
In logic	not A	A and B	A or B
In symbols	$\neg A$, $\bar{A}$	$A \wedge B$, $A \,\&\, B$	$A \vee B$
In English	it is not the case that A; A doesn't hold; A isn't so	both A and B; A but B; A despite B; A while B	either A or B; A and/or B; A or B or both; A unless B

Connective	*implication*	*equivalence*
In logic	if A then B	A if and only if B
In symbols	$A \to B$, $A \Rightarrow B$	$A \leftrightarrow B$, $A \Leftrightarrow B$
In English	B if A; B when A; A only when B; A only if B; from A, it follows that B; A implies B; B provided that A; in case A, B; A is sufficient for B; B is necessary for A	A iff B; A is equivalent to B; if A then B, and vice versa; A exactly if B; A just in case B; A is necessary and sufficient for B

It is rewarding to pay attention to the variety of natural language expressions for connectives. Students have often a hard time trying to comprehend clichés like *only if*, *is necessary for*, etc.

Example 1.4. Clearly, given some statements $A, B, C, \ldots$ one can freely construct more and more involved expressions using logical connectives. Say,

$$\big((A \to \neg(B \to C)) \lor (B \land \neg A)\big) \to \big(((\neg A) \land ((\neg C) \to A)) \land B\big)$$

is such an expression. We will sometimes call expressions of this kind *formulas*.

To make formulas more readable, they usually omit some brackets according to certain rules. One of these rules says that conjunction and disjunction *bind weaker* than negation but *stronger* than implication. So, $\neg X \land Y$ means $(\neg X) \land Y$ (rather than $\neg(X \land Y)$), whereas $X \land Y \to Z \lor W$ stands for $(X \land Y) \to (Z \lor W)$. Another rule says that $X \land Y \land Z$ means $(X \land Y) \land Z$—as we shall see, the *different* formula $X \land (Y \land Z)$ is nevertheless *equivalent* to $(X \land Y) \land Z$, so this choice is arbitrary (but necessary from the formal point of view). A similar rule applies for $\lor$. With these rules, we can rewrite our example as

$$(A \to \neg(B \to C)) \lor (B \land \neg A) \to \neg A \land (\neg C \to A) \land B.$$

Equivalences, tautologies and valid arguments. Suppose we have two statements F and G built from some simpler statements $A_1, \ldots, A_n$ using logical connectives. (We will use the notation like $F(A_1, \ldots, A_n)$ then.) The statements F and G are *(logically) equivalent* if (and only if)[2] they are both true or both false, whatever truth values the statements $A_1, \ldots, A_n$ have. We shall write $F \equiv G$ in this case.

Example 1.5. The statements $A \land B$ and $B \land A$ are equivalent. This is obvious from their truth tables. The statements $\neg\neg A$ and A are equivalent as well, for the former is true iff $\neg A$ is false, that is, A is true.

The statements $A \to B$ and $\neg A \lor B$ are equivalent. Indeed, each is false iff A is true and B is false. Otherwise, both are true. Similarly $\neg(A \to B)$ and $A \land \neg B$ are equivalent: each is true iff A is true and B is false; otherwise, both are false.

The statements $A \to \neg B$ and $B \to \neg A$ are also equivalent (this fact is known as *the law of contraposition* or *of contrapositive*). Indeed, when is the former false? If and only if A is true and $\neg B$ is false. This means that both A and B are true. But the statement $B \to \neg A$ is false iff both B and A are true. Hence those statements are equivalent as it is not possible for one of them to be false the other one being true.

In contrast, the statements $A \to B$ and $B \to A$ are not equivalent for their truth values differ when A is true and B is false.

> We believe it is very important to make a clear distinction between a statement and its truth value. We warn students against their bad yet widespread habit of writing $A = 1$ instead of 'A is true': let A be $2 + 2 = 4$ and B be the statement of Fermat's Last Theorem; both are known true, so from $A = 1$ and $B = 1$ one could naturally derive $A = B$; are these statements really *the same* while differing so much in complexity? In practice we use the notation $[A] = 1$ to say that the value of A is truth (in a given context), which is, of course, quite lax as it does not specify that context (say, the valuation of atoms when A is compound).

The statement F built from $A_1, \ldots, A_n$ is a *tautology* if it is true for any truth values of $A_1, \ldots, A_n$. Tautologies can be understood as laws of logic.

Example 1.6. To check if a statement is a tautology, it suffices to construct its truth table. For example, $A \to A$ is an obvious tautology. This procedure can be very laborious when the statement depends on many atoms. Sometimes you can save your effort the following way.

[2] This part is frequently implied by but omitted from a mathematical definition.

Consider the statement $F = (A \to B) \to ((B \to C) \to (A \to C))$. Assume that F is not a tautology, that is, it is false for some values of A, B, C. What could these values be? Clearly, $A \to B$ is true while $(B \to C) \to (A \to C)$ is false. The latter implies that $B \to C$ is true and $A \to C$ is false. Hence, A is true but C is false. As $A \to B$ holds, B does as well. From $B \to C$, we get that C is true despite our previous conclusion. Thus, we have got a *contradiction.* Since all our arguments are sound, the only weak link is the assumption that F is false. Hence, F cannot be false.

Of course, this version of semantic tableaux can easily come across an explicit branching (say, when $X \to Y$ is true or $X \wedge Y$ is false), which makes its applications much less neat. The Instructor should consider a few of such examples and stress that there is no known 'good' algorithm to test *any* formula for being a tautology.

Arguments. Naturally, when making an argument we make some assumptions firstly, and then draw a conclusion. We hope that the conclusion is *always* true whenever all our assumptions are true (our statements may be true or false depending on situation, whence this 'always' comes). In such a case, the argument is called *valid.* (One can equivalently put it this way: *it is never the case that all the assumptions are true whereas the conclusion is false.*) Consider the following argument:

Today is Thursday. It rains every Thursday. Therefore, $3 = 1 + 2$.

It is not very convincing despite the fact the conclusion is known to be *always* true (while it depends on today's date whether the first assumption is true). Moreover, as the conclusion is *known* to be true, the argument is useless. Let us modify it a little:

Today is Thursday. It rains every Thursday. Therefore, it is raining today.

This is much better, although both the assumptions and the conclusion are quite doubtful and still depend on what day it is today. The conclusion is now *guaranteed* to be true whenever the assumptions hold true. If they do not, all bets are off, indeed. This guarantee holds *always*, *everyday*, for whatever situation. We need know *nothing* about the truth values of those statements in any particular situation to be sure that the conclusion is no less true than the assumptions and new errors are thus impossible.

It is not easy to clarify the implied universal quantifier in our 'definition' of validity. To demonstrate the phenomenon of 'semantic validity' (i. e., due to the logical form only), the Instructor could use statements whose 'real-world' truth is 'uncertain' (see argument examples from the next lecture).

One can render this argument into symbols as follows:

$$(T \wedge (T \to R)) \to R,$$

where T means 'today is Thursday' and R means 'it is raining today'. (Please notice the model: if the assumptions are $A_1, \ldots, A_n$ and the conclusion is C, we construct the formula $(A_1 \wedge \ldots \wedge A_n) \to C$). It is easy to see that the resulting statement is a tautology. In general, an argument is valid if the respective statement is a tautology (but the converse is not necessary).

Example 1.7. Consider the argument:

It is raining today. It rains every Thursday. Therefore, it is Thursday today.

This can be rendered as $(R \wedge (T \to R)) \to T$, where the atoms T and R have kept their meaning. The resulting statement is not a tautology, since it is false when R is true but T is false. The argument is indeed invalid: it fails in a situation when, for example, it rains every day but today is Saturday.

In science, one hopes to draw previously unknown yet necessarily true conclusions from one's assumptions (given they are true themselves). Hence, a scientist should stick to valid arguments whenever possible. In particular, each mathematical proof should be a valid argument.

2 The Language of Mathematics

This lecture continues the line of the previous one. A predicate is 'defined' to be a syntactic function from names to statements, while quantifiers lack general definition (without advanced semantics, one could just identify them with variable binders). We (almost tacitly) generalize logical equivalence to the case of quantified statements and appeal routinely to the natural intuition of 'existence' and 'universality' as one usually does in mathematical proofs.

Now, let us try to discover some inner structure in logical atoms. Let us split the atoms! As every atom is an indicative sentence, a rudimentary linguistic analysis suggests that each such sentence should have its *subject* and *predicate*. Basically, the predicate is *what is being said* and the subject is *what it is being said about*.[3] For example, in the sentence

The quick brown fox jumps over the lazy dog,

'the quick brown fox' can be seen as the subject while 'jumps over the lazy dog' as the predicate. We can change our sentence a bit:

The slow gray cat jumps over the lazy dog.

Here the same predicate is said about another subject. In general, we see that the expression

x jumps over the lazy dog

turns into a statement when one replaces its part x (which is called a *variable* for it varies) with the name of a reasonable thing. Neglecting the grammatical terminology, the expression

x jumps over y,

which turns into a statement when substituting some names for the *variables* x and y, can also be called a predicate. The former predicate contains just one variable and is thus called *unary*, while the latter one is called *binary* for it has two distinct variables. Likewise, we can speak about an *n-ary* predicate with variables $x_1, \dots, x_n$.

In general, a *predicate* (or *property*) is an expression which turns into a statement (either true or false) after one substitutes some reasonable object names for all its variables. Each statement can be looked upon as a 0-ary (*nullary*) predicate.

Example 2.1. The following expressions are typical predicates: x *is even*, x *is greater than* y. They turn into statements (e. g., 4 *is even*, 3 *is greater than* 7) if we substitute numbers for variables. On the other hand, the predicate x *is a son of* y can be turned into a statement when substituting human names.

It is natural to associate a *domain* with a predicate, whence the objects are taken from. For example, the domain of the predicate x *is even* could be natural or integer numbers, while the same domain choice does not make much sense for x *is a son of* y.

The exact nature of domains is out of the discussion scope here, but it might be noteworthy that two arguments of one predicate may have distinct domains in general.

[3] Sometimes, it is quite tricky to identify these parts: say, what is the subject in the sentence *it rains*? What does that 'it' refer to?

One can easily construct new predicates from some given ones applying logical connectives. E. g., *x is even and it is not the case that y is greater than x*. But there is a more interesting way to do so. Namely, it is possible to express some meaning about the whole domain: *there is some x such that x is even*; *all x are even*; *there are many x which are even*; *there exists a unique even x*; *most of x are even*; *there are infinitely many x being even*, etc.

Such constructions are called *quantifiers.* A quantifier *binds* some variable in a predicate so that it is no more available for a substitution (such a variable is called *bound*; otherwise it is still *free*). Indeed, the predicate *either x is even or x is odd* may be turned into the statement *either* 7 *is even or* 7 *is odd*, while the expression *for each* 7*, either* 7 *is even or* 7 *is odd* does not make much sense.

Among many conceivable quantifiers, two are the most important. These are the *existential* quantifier *there exists some x such that...* and the *universal* quantifier *for each x it holds that....* We will sometimes write $A(x)$ for a predicate A with a variable x (like *x is...*, *x does...*, etc.). For example, one can write *even*(x) instead of *x is even*; here A itself can be expressed by words as *to be even.*

Quantifier	*existential*	*universal*
In logic	there exists some x such that A	for each x it holds that A
In symbols	$\exists x\, A(x)$; $\exists x\, A$	$\forall x\, A(x)$; $\forall x\, A$

In English	for some x, $A(x)$; for (a) suitable x, $A(x)$;
	something is A; somebody is A;
	at least one is A; for at least one x, $A(x)$;
	there is an x such that $A(x)$
		for all x, $A(x)$; for every x, $A(x)$;
		for (an) arbitrary x, $A(x)$;
		whatever x is, $A(x)$; $A(x)$ always holds;
		everything is A; everybody is A

> As in the above, it is very useful to have some exercises in translating (standard) natural language expressions with quantifiers into symbols, and vice versa. Even the simplest patterns like *all A's are B's*, *no A is B*, *some A's are B's* may prove difficult for the students.

Example 2.2. Let us translate some English phrases into logical symbolism. *All roses are red.* In order to translate this one, we need predicates of redness and 'roseness', that is, $Rs(x)$ will stand for *x is a rose* and $Rd(x)$ for *x is red.* The domains for these predicates must be chosen reasonably, say as 'all flowers' or even 'all the real things'. Then our phrase is rendered as $\forall x\, (Rs(x) \to Rd(x))$. Please notice the implication here: we are effectively *asserting* that x is red—yet under a condition: namely, that x is a rose. We are saying *nothing* about objects which are not roses, nor we are asserting anything to be a rose.

No rose is red. This one may sound as an equivalent to the negation of the previous one but it is far from that. There are a few (equivalent) ways to put this into symbols: $\forall x\, (Rs(x) \to \neg Rd(x))$ (*everything being a rose is not red*), $\neg\exists x\, (Rs(x) \wedge Rd(x))$ (*it is not the case that there is something being both a rose and red*), $\neg\exists x\, (Rd(x) \wedge Rs(x))$, $\forall x\, (Rd(x) \to \neg Rs(x))$ (*everything red is not a rose*).

Observe that "on a distant planet" where no rose exists, both $\forall x\, (Rs(x) \to Rd(x))$ and $\forall x\, (Rs(x) \to \neg Rd(x))$ would be true since the implication's assumption $Rs(x)$ would fail for every x from that planet, which would make both implications true. So, these two formulas do not negate each other. For more details, see Example 2.6 below.

But how can one translate the negation of the first phrase? $\neg\forall x\, (Rs(x) \to Rd(x))$, of course, or $\exists x\, (Rs(x) \wedge \neg Rd(x))$ equivalently (*there is something being a rose but not red*). These transformations under negation are not at all complicated: they are governed by few simple laws (see below).

Some quantifiers can be expressed via others. E.g., $\exists x\, A(x)$ is usually considered to be equivalent to $\neg\forall x\, \neg A(x)$ ('some x satisfies the property A if and only if it is not the case that the property A fails for every x'), and dually, $\forall x\, A(x)$ can be seen equivalent to $\neg\exists x\, \neg A(x)$. If we have equality at our disposal, we can express *there exists a unique such x that* $A(x)$ as $\exists x\, (A(x) \wedge \forall y\, (A(y) \rightarrow y = x))$.

In general, two predicates are (logically) equivalent if they hold (or do not) 'simultaneously'. It is a too big task for us now to make this 'definition' precise. Thus, we have $\exists x\, A(x) \equiv \neg\forall x\, \neg A(x)$ and $\forall x\, A(x) \equiv \neg\exists x\, \neg A(x)$.

Example 2.3. Continuing the previous example, we have

$$\neg\forall x\, (Rs(x) \rightarrow Rd(x)) \equiv \neg\neg\exists x \neg\, (Rs(x) \rightarrow Rd(x)) \equiv \\ \exists x \neg\, (Rs(x) \rightarrow Rd(x)) \equiv \exists x\, (Rs(x) \wedge \neg Rd(x)),$$

as $\neg(A \rightarrow B) \equiv A \wedge \neg B$.

Remark 2.4. When used in a formula, a quantifier binds its variable's occurrences in some part of that formula. For example, the formula $\forall x\, A(x) \rightarrow \forall x\, B(x)$ is conventionally read as $\forall x\, (A(x)) \rightarrow \forall x\, (B(x))$, that is, the first quantifier's *scope* is limited to the subformula $A(x)$ and the second one's—to $B(x)$. The rule is that a quantifier binds stronger (has higher precedence) than any binary logical connective, so a quantifier's scope ends by default at the first binary connective to the right of the quantifier. In order to include a connective into the scope, parentheses are needed: e. g., in $\forall x\, (A(x) \rightarrow B(x))$, the quantifier binds occurrences of x in both A and B. On the other hand, in the formula $\forall x\, A(x) \rightarrow B(x)$, no occurrence of x in $B(x)$ is bound by the first quantifier. Each such occurrence is free for a substitution (if not bound by any inner quantifier in the formula $B(x)$).

If two occurrences of x are bound by the same quantifier, they are *guaranteed* to refer to one object. Say, in $\exists x\, (A(x) \wedge B(x))$, we mean *one* x with both the properties A and B. If two occurrences of a variable belong to distinct scopes, they may refer to distinct things. E.g., $\exists x\, A(x) \wedge \exists x\, B(x)$ states that something satisfies A and some (other) thing satisfies B—no identity between the two is assumed. In particular, one may safely rename a bound variable, so that $\exists x\, A(x) \wedge \exists x\, B(x) \equiv \exists x\, A(x) \wedge \exists y\, B(y)$.

In programming terms, a bound variable is *local* and retains its value throughout its scope but not beyond it.

When using a quantifier on some variable x, we always have to fix a *domain* for it, whence the possible objects to replace x come from. E.g., the statement $\forall x\, (\mathit{even}(x) \vee \mathit{odd}(x))$ makes sense for numerical integer domains but hardly so if x can denote, say, a mineral. The truth value of a statement built using quantifiers depends very much on the domain chosen.

Example 2.5. Consider the statement $\exists x\, (2x + 3 = 1)$. This means, of course, that the equation $2x + 3 = 1$ has a solution. But in which domain? There is an integer (hence, rational and real) solution $x = -2$ but there are no natural solutions. So, the statement is false for the natural domain but true for the integer one.

Another example. The statement is $\exists x \forall y\, x \leq y$ is false for the domain of integers but true for the domain of naturals (as one can consider $x = 0$).

Example 2.6. There is a degenerate but common situation in mathematics, that is sometimes perceived as paradoxical. Let us consider the statement:

> The present King of France is bald.

If we interpret this as

> For each x, if x is the present King of France, then x is bald,

taking all human beings who ever lived as the domain, we see that the assumption x *is the present King of France* is false for every x (as of the year A. D. 2022), hence the whole implication *is true.* Clearly, the statement 'the present King of France is hairy' is no less true. Such situations (and paradoxically true statements themselves) are called *vacuous truths* and are indeed very common in mathematics.

For example, the typical pattern of proving that *there is no* x *such that* $A(x)$ is as follows: consider an arbitrary x with $A(x)$ and prove that a knowingly false statement *logically* follows from $A(x)$. Therefore, if x with $A(x)$ existed, that falsity would be true, which is impossible. No such x thus exists. Notice that here we reasoned about x which had appeared *nonexistent* by the end of our argument. That is, our *logic* must be valid for such *counterfactual* situations, where we have effectively proved a vacuous truth: "for every x with $A(x)$, a falsity holds".

> The Instructor might add that vacuous truths may sound very logical, like *every round square is round*, or be vacuous for a certain parameter value (still being true for all its values), like *every finite set of integers has an upper bound.* So, they are not mere fruits of an idle mind but a necessity.

Now let us check some arguments employing quantifiers for validity. As in the above, an argument is valid iff its conclusion is true whenever all its assumptions are true.

Example 2.7. Let us begin with the classics. Consider the argument:

All humans are mortal. Socrates is a human. Therefore, Socrates is mortal.

As we know who Socrates was, we are sure he was mortal. But what if we did not know? (Which situation is usual for any practical reasoning.) What if the meaning of the name 'Socrates' may vary?

One can put this in symbols as $\forall x\,(H(x) \to M(x)) \land H(Socrates) \to M(Socrates)$ denoting 'to be a human' by H, 'to be mortal'—by M, and choosing any domain that contains all humans who ever lived. By the assumption, $H(x) \to M(x)$ must hold for every x from the domain. If we *instantiate* x as the name 'Socrates', we obtain a tautology $(H(Socrates) \to M(Socrates)) \land H(Socrates) \to M(Socrates)$. So the argument is valid. Yet notice that it is not a tautology itself since $\forall x\,(H(x) \to M(x))$ is an atom whose inner structure cannot be grasped just in terms of logical connectives.

> Logical proofs in these examples should be presented as guided by the natural intuition rather than any formal inference rule. Much like one does for ordinary theorems of 'mathematical content'.

Example 2.8. Here is a slight variation of the above:

Every (natural) number that is divisible by 4 is even. All numbers are divisible by 4. Therefore, all numbers are even.

Again, here we surely have a false assumption. Yet this assurance requires some knowledge of arithmetic, while logic guarantees our argument to be valid without any such knowledge.

So, we may formalize our argument as follows: $\forall x\,(D(x) \to E(x)) \land \forall x\,D(x) \to \forall x\,E(x)$ over the domain $\mathbb{N} = \{0, 1, 2, \ldots\}$ of natural numbers, where $D(x)$ means that x is divisible by 4 and $E(x)$ stands for x being even. Clearly, our first assumption is true, while the second one is not. However, we need not know anything about numbers in order to accept this argument as valid. Indeed, if our assumptions are (or *were* as the English grammar suggests for our 'unreal' condition) true, we might *prove* the conclusion to be true as well.

Indeed, given the assumptions, we need $\forall x\,E(x)$. Consider an *arbitrary* (=nothing special) number x'. As every number satisfies D, our x' also does. So, $D(x')$ is true. We can likewise instantiate x as

x' in the first assumption. As both $D(x') \to E(x')$ and $D(x')$ are true, $E(x')$ also is. Since our x' was arbitrary, we could get the same conclusion for any other number, so we may *generalize* and obtain $\forall x\, D(x)$ therefrom.

Example 2.9. Another example will show quantifiers' great expressive power. Here it is:

> When I am hungry, I want to go home or to go to a restaurant. Sometimes, I am hungry but do not want to go home. Therefore, sometimes I want to go to a restaurant.

To choose the right domain is the key. This could be the continuity of time (so, every variable refers to a point in time). We thus get

$$\forall x\, (H(x) \to Hm(x) \vee R(x)) \wedge \exists x\, (H(x) \wedge \neg Hm(x)) \to \exists x\, R(x).$$

This argument is also valid. Indeed, let x_0 be some moment when I am hungry but do not want to go home. Instantiate x in the first assumption as x_0 to obtain the tautology

$$(H(x_0) \to Hm(x_0) \vee R(x_0)) \wedge H(x_0) \wedge \neg Hm(x_0) \to R(x_0).$$

Hence, our assumptions imply $R(x_0)$, whence it follows that $\exists x\, R(x)$.

Example 2.10. Now, the final example. Consider the argument:

> Everyone loves himself. Therefore, someone is loved by somebody.

In symbols this looks like $\forall x\, L(x, x) \to \exists y \exists x\, L(x, y)$. Is it valid? At first glance, it is for the assumption clearly implies $\forall y \exists x\, L(x, y)$ (take y for x here and instantiate x as y in the assumption). But does the latter imply $\exists y \exists x\, L(x, y)$? In general, does it follow that $\exists z\, A(z)$ from $\forall z\, A(z)$?

It depends on whether the domain is empty (remember *vacuous truths* about the *absent* 'present King of France'). Usually, domains are required to be non-empty. Then the argument must be valid.

Exercise 2.11. Consider the argument:

> Some key can unlock every door. Therefore, every door may be unlocked with some key.

Put this argument in symbols and check it for validity. Do the same for the 'inverse' argument:

> Every door may be unlocked with some key. Therefore, some key can unlock every door.

The exact rules for checking such arguments for validity are given by *predicate* or *first-order logic* and are out of the scope of our course.

3 A Case Study: Strings

This section introduces an 'inductive type' of strings without much 'foundational' explanation. The main goal is to present the ideas of induction and recursion (which we looked upon as the heart of 'discrete mathematics') without boring traditional examples of summing consecutive naturals, etc. We see this especially useful when the students have some (functional) programming experience. This section has few (if any) dependencies in the Course and may thus be freely omitted.

Our next goal is to taste some *discrete mathematics* while trying to stick to the logical formalism. We are not going to dig too deep for the 'foundations' and will take some notions as familiar and statements as obvious. Our case study deals with *strings* (or *lists*, or *words*), which are one of the most basic programming 'data types'.

Intuitively, a *string* is a finite sequence of *symbols* of arbitrary nature. For example, $[x, y, x]$, $[4, 12, 2, 3]$, and $[\text{Socrates}, \text{Plato}, \text{Aristotle}]$ are strings where x, y, 3, 12, and Plato are symbols. We use $[\ldots]$ and commas here to separate stings and symbols from each other. If we know that, say, x and y are treated as symbols (but not xy), we may safely omit the delimiters and render the first string as xyx. In general, we shall consider the strings over an arbitrary but fixed collection (or *set*, in mathematical parlance) A of symbols, which is called an *alphabet.* E. g., $[4, 12, 2, 3]$ is a string over the set of all natural numbers $\mathbb{N} = \{0, 1, 2, \ldots\}$. Needless to say, an alphabet could be a set of strings over some other alphabet.

As it is usual in mathematics, we are interested in 'borderline' degenerate objects. The empty string $[\,]$ that contains no symbol is just such an object.

Inductive definition. Our account of what a string is seems quite clear for a typical human (but not so much for the Mathematician who could ask what 'finite sequence' means, etc.). But if we try to make this definition clear for a computer, that is, to describe it programmatically from 'scratch', we may discover this task being quite hard. Typically, one has a certain 'array' or 'string' type already implemented in his favorite programming language. But what if you have not one?

One possible approach is based on the following observation: every string s is either empty or is obtained from another string s' by adding a certain symbol x to it, or $s = x : s'$ is symbols. (Assuming that x is added as the leftmost symbol, we obtain $[4, 12, 2] = 4 : [12, 2] = 4 : (12 : [2]) = 4 : (12 : (2 : [\,]))$. Let us call x the *head* of the string s, and call s' the *tail* thereof. Now, we may *forget* about 'finite sequences' and *define* the set $S(A)$ of strings over an alphabet A via the following generating rules:

$[\,]$ is a string over A;
if s' is a string over A and x is a symbol from A, then $x : s'$ is a string over A,

which might be rendered symbolically as

$$[\,] \in S(A); \quad \forall s' \, \forall x \, (\, (s' \in S(A) \wedge x \in A) \;\rightarrow\; x : s' \in S(A) \,)\,.$$

We also tacitly suppose that *every* sting is constructed according to these rules. Here, the symbols $[\,]$ and $:$ *lack any definition.* They are kind of 'axioms' or, as the Programmer would say, *constructors* (of the data type 'string'). In a word, we replace the question *what a string is* with the question *how one can construct a string.* Along these lines, we can easily return to our 'natural' symbolism by defining $[x_1, x_2, \ldots, x_n]$ as $x_1 : (x_2 : (\ldots : (x_n : [\,]) \ldots)))$.

Such 'definitions by construction' are known as *inductive definitions.*

Induction Principle. If we want to prove anything about our redefined strings, we need to fix some basic properties which will be used as 'axioms' in our arguments. Clearly, these properties should be as intuitively plausible as possible.

First, we assume that $[\,] \neq x : s$ for all x and s, as the empty string should differ from any other. Then we postulate that $x : s = y : t \iff (x = y \wedge s = t)$ for all $x, y \in A$ and $s, t \in S(A)$, that is two strings constructed via : are equal iff their respective heads and tails are equal. Let us denote the first axiom by (A1) and the second one denote by (A2).

Example 3.1. The facts the principles (A1) and (A2) can prove are not too astonishing. We can show, say, that $[1, 2] \neq [1, 2, 3]$. We need then to check that the equation $[1, 2] = [1, 2, 3]$ is false. We will yet suppose that it is true, and then prove such a situation impossible. Hence, the equation must be false indeed. Such a technique is know as *proof by contradiction* and is widely used.

So, assume $1 : [2] = 1 : [2, 3]$. From (A2), it follows that $2 : [\,] = [2] = [2, 3] = 2 : [3]$. Yet another application of this principle results in $[\,] = [3] = 3 : [\,]$. On the other hand, $[\,] \neq 3 : [\,]$ by (A1). Thus, the equality $[\,] = 3 : [\,]$ is both true and false, which is not possible. A contradiction. Hence, our assumption does fail and, as a matter of fact, $[1, 2] \neq [1, 2, 3]$.

The Pedantic Reader might have noticed that we use a little more than just principles (A1) and (A2) in our argument. Namely, we have used 'obvious' properties of equality, like *transitivity* (if $x = y$ and $y = z$, then $x = z$), and the logic itself ($\neg P$ and P cannot be both true). In the next exercise, you are also supposed to use the 'obvious' properties of the familiar alphabet $\mathbb{N}$, like $0 \neq 1$.

Exercise 3.2. Prove formally that $[1, 2, 3] \neq [1, 3, 4]$ and $[1 + 2, 1 + 1] = [5 - 2, 2]$ (all strings are over the alphabet $\mathbb{N}$; $1 + 2$ is just another name for the symbol 3, so that $1 + 2 = 3$).

To prove anything interesting, we need another principle. Surprisingly, one is enough. Being so versatile, this principle is necessary abstract.

Let $P(s)$ be a unary predicate with the domain $S(A)$, that is, a property of strings over A. Say, one might have $P =$ "John likes s better than the number n" where n is fixed. Assume that we want to prove that *each string s over A satisfies the property P*, or $\forall s\, P(s)$ symbolically.

With this in view, we could consider an *arbitrary* string t and try to prove $P(t)$. As we know *nothing special* about t, there is not much we can do here. But we know that each string has been constructed according to our inductive definition, that is, t is either $[\,]$ or $x : t'$ for some $x \in A$ and another string t'. For our goal $\forall s\, P(s)$, it is clearly necessary that $[\,]$ satisfies P, or $P([\,])$ in symbols. Assume that it is the case.

If $t = x : t'$, we get just another 'arbitrary' string t'. It is yet clear intuitively that if t' is not $[\,]$, we would have $t' = x' : t''$ etc.,—and get $[\,]$ eventually after a finite number of steps. This gives us the following idea: if $P([\,])$ and P survives each step of string construction, i. e., from $P(t')$ it follows $P(x : t')$ for every t' and x (in symbols, $\forall t' \forall x\ (P(t') \to P(x : t'))$), then we obtain $P(t)$ for an arbitrary t and, finally, $\forall s\, P(s)$.

To sum it up, we want to use the following *Induction Principle* (IP) for strings:

$$\text{for every unary predicate } P \text{ over } S(A),\text{ if } P([\,]) \text{ and } \forall s \forall x\ (P(s) \to P(x : s)),\text{ then } \forall s\, P(s).$$

The assumption $P([\,])$ is called the *base case* of induction, $\forall s \forall x\ (P(s) \to P(x : s))$ is called the *inductive step*, and $P(s)$ is called *inductive hypothesis* (which is usually abbreviated to IH) when used to infer $P(x : s)$ from it.

While we cannot prove (IP) in our current setting and have to take this principle as an axiom, we can support its intuitive validity by the following argument.

Example 3.3. We let $A = \mathbb{N}$ and let both the assumptions $P([\,])$ (ass. 1) and $\forall s \forall x\ (P(s) \to P(x : s))$ (ass. 2) hold. Consider the string $t = [1, 2, 3] = 1 : (2 : (3 : [\,]))$. From (IP), it follows that $\forall s\, P(s)$, whence $P(t)$. But one can prove this particular instance $P([1, 2, 3])$ directly from our assumptions *without* (IP). Indeed, we obtain the following derivation:

(1)	$P([\,])$	by (ass. 1)
(2)	$P([\,]) \to P(3 : [\,])$	by (ass. 2)
(3)	$P(3 : [\,])$	from (1) and (2)
(4)	$P(3 : [\,]) \to P(2 : (3 : [\,]))$	by (ass. 2)
(5)	$P(2 : (3 : [\,]))$	from (3) and (4)
(6)	$P(2 : (3 : [\,])) \to P(1 : (2 : (3 : [\,])))$	by (ass. 2)
(7)	$P(1 : (2 : (3 : [\,])))$	from (5) and (6)

It is clear that our t is nothing special, so we can prove $P(s)$ for each particular s this way, while the proof gets longer with s. Essentially, the Induction Principle 'packs' this infinite multitude of proofs of unboundedly increasing length into one proof based on (IP).

Recursive functions. So far, we have a mighty principle to prove things but not many statements to try it for. In programming practice, one defines a plethora of *functions* over strings, whose properties are natural challenges to prove. As in the case of Induction Principle for proofs, we will take advantage of the the *inductive* definition for strings when defining our functions.

Let us start with the string *length*. This function takes a string s over A and returns a natural number $lh(s)$. Intuitively, we want the following equations to hold: $lh([\,]) = 0$, $lh([8]) = 1$, $lh([3, 11]) = 2$, and so on. We thus *define* lh by these two rules:

$lh([\,]) = 0$;
$lh(x : s) = 1 + lh(s)$ for all x and s.

We assume such a function lh that satisfies the two properties does indeed exist and is unique. They say that lh is defined *by recursion* on strings (since one *recurs* to the 'previous' value $lh(s)$ in order to get $lh(x : s)$). As in the case of (IP), the definition mimics somehow the structure of the inductive definition for strings.

> It is easy to prove the uniqueness by IP application but the existence would likely require some 'set-theoretic' considerations, which we want to avoid here.

Example 3.4. Our definition for lh is *computationally effective*, i. e., it provides a recipe to compute the value $lh(s)$ for every particular string s. For example, $lh([3, 11]) = lh(3 : (11 : [\,])) = 1 + lh(11 : [\,]) = 1 + (1 + lh([\,])) = 1 + (1 + 0) = 2$.

Exercise 3.5. For strings over the English alphabet, prove that $lh(student) = 7$ formally.

Another interesting function is *append* (or *concatenate*) which 'glues' two strings together, so that $app([1, 3, 2], [3, 2]) = [1, 3, 2, 3, 2]$ and $app([5, 8], [\,]) = app([\,], [5, 8]) = [5, 8]$. We can define app recursively the following way:

$app([\,], t) = t$ for each t;
$app(x : s, t) = x : app(s, t)$ for all x, s, and t.

Exercise 3.6. Prove formally that $app([1, 3, 2], [3, 2]) = [1, 3, 2, 3, 2]$.

Now, we have got just enough to state and prove moderately interesting theorems about string functions. From the definition, we know that $app([\,], s) = s$. But what can one say about $app(s, [\,])$? Clearly, $app([1, 2], [\,]) = 1 : app([2], [\,]) = 1 : (2 : [\,]) = [1, 2]$. Unsurprisingly, in general we have

Lemma 3.7. *For every $s \in S(A)$, $app(s, [\,]) = s$.*

Proof. By induction on s. If we let P be just the equation $app(s, [\,]) = s$, it suffices to show that $P([\,])$ and $\forall s \forall x\ (P(s) \to P(x : s))$ hold, that is, $app([\,], [\,]) = [\,]$ and $\forall s \forall x\ (app(s, [\,]) = s \to app(x : s, [\,]) = x : s)$. The first statement holds by the definition of app. Let us prove the second one. Consider arbitrary s and x and assume that $app(s, [\,]) = s$ is true (otherwise, the implication is clearly true). By definition, we have $app(x : s, [\,]) = x : app(s, [\,])$, whence $app(x : s, [\,]) = x : s$ as required. Generalizing s and x, we obtain what we want. □

It appears that $[\,]$ plays the same role for app as 0 does for $+$, where one has $n+0 = n = 0+n$. How far can we extend this analogy? For example, we know that $n + (m + l) = (n + m) + l$. What about app?

Lemma 3.8. *For every $s, t, r \in S(A)$, $app(s, app(t, r)) = app(app(s, t), r)$.*

Proof. As (IP) deals with unary predicates while the equation $app(s, app(t, r)) = app(app(s, t), r)$ has three parameters, we have to fix two of them. The right choice of one to base the induction on is crucial! We consider some arbitrary strings t, r and apply induction on s for those fixed with $P(s) = (app(s, app(t, r)) = app(app(s, t), r))$.

Clearly, $P([\,]) = (app([\,], app(t, r)) = app(app([\,], t), r))$. Both sides of this equation equal $app(t, r)$ by the definition of app, hence the statement $P([\,])$ holds. Now, let us assume $P(s) = (app(s, app(t, r)) = app(app(s, t), r))$ for an arbitrary s and x, then try to get $P(x : s)$. Indeed,

$$app(x : s, app(t, r)) = x : app(s, app(t, r)) = x : app(app(s, t), r)) = \\ app(x : app(s, t), r)) = app(app(x : s, t), r)).$$

Here, we have applied the inductive hypothesis $P(s)$ and the definition of app. As both the base case and the inductive step are proved now, we may conclude $\forall s\, P(s)$ by virtue of (IP). Generalizing our arbitrary t and r, we obtain the required statement. □

Exercise 3.9. Try to prove the statement above by induction on either t or r. What obstacles do you face when doing so? Can you overcome them?

In general, you should apply induction to the argument that is used in recursive definitions of the functions involved. The function app is defined by recursion on its *first* argument, whence our choice of induction on s.

So far so good, but we also have $n + m = m + n$ for any numbers n, m. Does a similar property hold for app?

Exercise 3.10. Prove formally that it does not, i. e., it is not the case that $app(s, t) = app(t, s)$ for every s and t. Notice that you do not need (IP) here but rather (A1) and (A2).

The operations app on strings and $+$ on natural numbers are not only somewhat similar but 'interrelated' via the function lh. In fact, one has $lh([\,]) = 0$ and $lh(app(s, t)) = lh(s) + lh(t)$ for every s, t. Relations of this form are called *homomorphisms* in mathematics.

Exercise 3.11. Prove these statements formally.

For natural numbers, from $n+m=n+l$, it follows that $m=l$. Does a similar property hold for *app*?

Lemma 3.12. *For every $s,t,r \in S(A)$, if $app(s,t)=app(s,r)$, then $t=r$.*

Proof. Let us apply induction on s to the predicate $P(s) = (app(s,t) = app(s,r) \to t = r)$. The base case is to infer $t=r$ from $app([\,],t) = app([\,],r)$, which is clear. For the inductive step, we assume (as the IH) that $app(s,t)=app(s,r)$ implies $t=r$ and try to prove $t=r$ from $app(x:s,t)=app(x:s,r)$. Simplifying the latter equation, we get $x:app(s,t) = x:app(s,t)$, whence $app(s,t)=app(s,r)$ by (A2). Applying the IH, obtain $t=r$. □

Exercise 3.13. Try to prove that from $app(t,s)=app(r,s)$, it follows that $t=r$. Probably, you will need to state and prove a few auxiliary lemmas to complete the task. Please read on for an easier but indirect approach.

Another interesting string function is *rev*, which reverses a string, so that $rev([1,2,3]) = [3,2,1]$. Let us define it formally:

$rev([\,]) = [\,]$;
$rev(x:s) = app(rev(s),[x])$ for all x and s.

The function *rev* relates to *app* much like matrix transposition (or inversion) relates to matrix multiplication, where one has $(AB)^T = B^TA^T$ and $(AB)^{-1} = B^{-1}A^{-1}$ when everything is well-defined. E. g., $rev(app([1,2],[3,4])) = [4,3,2,1] = app(rev[3,4], rev[1,2])$. In fact, the same pattern might be seen in the usual rational number inversion and multiplication, as $\frac{1}{pq} = \frac{1}{q} \cdot \frac{1}{p}$ when $p,q \neq 0$.

Exercise 3.14. Prove that $rev(app(s,t)) = app(rev(t),rev(s))$ for every $s,t \in S(A)$. Probably, you will need Lemmas 3.7 and 3.8 at some stage.

Clearly, $(p^{-1})^{-1}$ for both rationals and matrices, and $(A^T)^T$ holds for the latter as well. A similar statement is true for strings.

Exercise 3.15. Prove that $rev(rev(s)) = s$ for each $s \in S(A)$.

Exercise 3.16. Prove that $app(t,s) = app(r,s)$ implies $t=r$ for every $s,t,r \in S(A)$. Try to find a simple proof using Lemmas 3.15, 3.14, and 3.12, yet no explicit induction.

Many simple proofs employ inductive definitions but not induction itself.

Lemma 3.17. *For every $s,t \in S(A)$, if $app(s,t) = [\,]$, then $s=[\,]$ and $t=[\,]$.*

Proof. According to the definition, each string is either $[\,]$ or of the form $x:r$ for some x,r. Let us consider the two possible cases for the structure of s. (Such a procedure is known as a *proof by cases.*)

If $s=[\,]$, then $[\,] = app(s,t) = app([\,],t) = t$, so $t=[\,]$ as well. Assume now that $s = x:r$ for some x and r. Then $[\,] = app(x:r,t) = x:app(r,t)$, which is forbidden by (A1). Hence, this situation is impossible, we have got a contradiction (i. e., a false statement). Formally, from a falsity, it follows anything. In particular, we may infer $s=[\,]$ and $t=[\,]$.

As each possible case results in $s=[\,]$ and $t=[\,]$, these statements do hold. □

Exercise 3.18. Prove that $lh(s)=0$ is equivalent to $s=[\,]$ for any $s \in S(A)$.

4 Sets

This section (besides its foundational and formalism-related contents) tries to introduce some 'axiomatic thinking'. In this respect, it continues the line of the previous one (being totally independent thereof). In general, the axiomatic approach is painful for many students, since they are used to giving too much credit to their intuition and do not take axioms nor definitions seriously. In the case of sets, their intuition is most likely rigidly fixed to Euler diagrams etc., where one can clearly see two object types: 'elements' and 'sets'. This is, of course, inadequate for any interesting set-theoretic construction, like union or binary relation. As we want our presentation of 'Discrete Mathematics' to be quite rigorous, this is unacceptable for us; therefore have we made this bitter 'axiomatic pill' for our students. In practice, we however give it them *after* more concrete and intuitive chapters on natural number induction and elementary arithmetic.

Technically, we define set equality in terms of $\in$ with the Extensionality Axiom, give four set existence axioms (Pairing, Specification, Powerset, and Union) from ZF, and the Axiom of Foundation (as if we had the Axiom of Choice). We then use these to obtain all the 'high school set theory' without much reference to intuition. In this Course, we neither give an explicit definition for the set $\mathbb{N}$ nor the Axiom of Infinity, nor we state any other axiom of ZFC. Throughout the Course, a few statements depend on the Axiom of Choice but we omit their proofs altogether.

Suppose we have a predicate $\varphi(x)$ with a domain D. It seems quite natural to group all x from D which satisfy φ together to form a new object. For example, if $\varphi(x)$ means x *is even* and D means all natural numbers, such an object would be 'all even natural numbers'. Clearly, this is a 'part' of the domain D carved by the property φ (also known as the *extension* of the predicate φ). Such a 'part', which may consist of many individuals (e. g., numbers) but is to be treated as a unitary object, is an example of what mathematicians call a *set*.

Intuitively, a *set* is an assembly (collection, multitude) of some individuals of arbitrary nature. Clearly, this does not comprise a rigorous definition since *assembly* and *objects* are still to be defined. The main mission of the notion of set is to be *the* uniform object of mathematics, generalizing numbers of various kinds (integer, real, complex etc.), functions, geometric figures and many others.

Apparently, we cannot define such a basic notion via any other. However we can *avoid any definition* but postulate some rules how sets do 'behave'. Thus, *set* is nothing more than a code name for a piece in our game, whose 'meaning' is just its part in the game rules. (Much like a *bishop* in the game of chess retaining the very same 'meaning' if we call it an *elephant* or an *officer*.) This is the essence of the *axiomatic method*.

Yet which *behavior* of sets do we define? Basically, one set A can *be an element* of another set B. Then we also say that A *belongs to* B and write $A \in B$. Intuitively, this means that the 'individual' A is in the 'collection' B. But do not allow this intuition to fool you: our sets are just an abstract *model* (of mathematical or other realities) and are fully entitled to behave counter-intuitively.

For example, in our model *everything is a set*, which is not in a good accord with the 'individual–collection' interpretation because there is no essential difference between the former and the latter.

Hence, any element of any set is a set itself having some sets as elements and so on. We shall see that there exists an *empty* set that has no elements at all. But anyway, we postulate that

there are no infinite chains of the form $x_0 \ni x_1 \ni \ldots \ni x_n \ni x_{n+1} \ni \ldots$.

A statement quite close to this (yet not the same) is called the *Axiom of Foundation*.

Clearly, the standard Axiom of Foundation implies this one. For the other direction, they usually apply the Axiom of (Dependent) Choice.

Example 4.1. The statement $a \in a$ holds for no set a. Otherwise, there would be the chain $a \ni a \ni a \ni \ldots$.

Exercise 4.2. Prove that there are no sets a and b such that both $a \in b$ and $b \in a$.

In many branches of mathematics, we are interested in some equalities. In our model, the equality of sets is definable. Namely, $A = B$ iff (if and only if) for each set x, x belongs to A iff x belongs to B; that is, A and B have the same elements. We can put it in symbols:

$$A = B \iff \forall x \, (x \in A \iff x \in B).$$

But what if $A = B$ and $B \in C$? Can one derive that $A \in C$? In fact, one can't. So we postulate this as the *Axiom of Equality*:

> if $A = B$ and $B \in C$ then $A \in C$.

Combined with the definition of $=$ the Axiom yields that equal sets behave identically on both sides of $\in$. Hence, every notion defined in terms of $\in$ (i. e., every notion in our model) is invariant w. r. t. (with respect to) the equality $=$. If $A = B$ and A does something, then B does just the same. Symbolically,

> for each predicate φ, if $A = B$, then $\varphi(A) \iff \varphi(B)$.

Quite intuitive, isn't it?

> Clearly, one needs some form of induction over 'predicates' (or, better, formulas) φ in order to prove this statement; some metatheory is thus needed. On the other hand, we can easily prove it for every *particular* φ below. So, we have decided to believe it without a proof.

Besides equality, there is a natural notion of comparison for sets. Namely, we say that a set A is *included* into a set B iff for each set x, if x belongs to A then x belongs to B; i. e., any element of A is an element of B. In symbols,

$$A \subseteq B \iff \forall x \, (x \in A \Longrightarrow x \in B).$$

In this case, we also say that A is a *subset* of B while B is a *superset* of A. If $A \subseteq B$ but $A \neq B$, then the set A is said to be a *proper* subset of B.

Lemma 4.3. *For all sets A, B, and C,*

1. *$A \subseteq A$;*
2. *if $A \subseteq B$ and $B \subseteq C$, then $A \subseteq C$;*
3. *$A = B$ iff $A \subseteq B$ and $B \subseteq A$.*

Proof. Let's prove just the second statement. We need to show that $x \in C$ for all $x \in A$. Consider an arbitrary set x and assume that $x \in A$. Then $x \in B$ holds as $A \subseteq B$. Likewise, by $B \subseteq C$, we get $x \in C$. Since for an *arbitrary* x, from $x \in A$ it follows that $x \in C$, this is true for *all* x. □

Corollary 4.4. *For all sets A, B, and C,*

1. *$A = A$;*
2. *if $A = B$ and $B = C$, then $A = C$;*

3. if $A = B$, then $B = A$.

Are there any sets in existence? Obviously, if there weren't any, our model would make no sense. We won't seek a formal reason for this, but rather assume that there exist the well-known sets $\mathbb{N} = \{0, 1, 2, \ldots\}$ of natural, $\mathbb{Z}$ of integer, $\mathbb{Q}$ of rational, and $\mathbb{R}$ of real numbers. Of course, their elements have to be sets themselves. E. g., 1 and 2 are some sets. *We omit the definitions* yet we take for granted some elementary properties, like $1 \neq 2$. In other words, we *avoid* using the internal structure of natural numbers (e. g., whether $1 \in 2$ or not), but assume that $1 \neq 2$, $2 < 3$, $2 + 3 = 5$ etc.

> I usually tell the students that from the standard definition of $\mathbb{N}$ it follows that $0 = \varnothing$ and $n+1 = \{0, 1, \ldots, n\}$ for each n; then I ask them not to use these equations in their proofs since we currently have no means to make this *recursive* 'definition' rigorous.
>
> When solving class problems, the Instructor should warn the students against making their examples dependent on whether $3 \neq \{4\}$ etc. They had better look for *simpler* examples based on sets whose elements are known for sure, like $\varnothing$, $\{\varnothing\}$, etc.

If we have some sets (like $\mathbb{N}$ and $\mathbb{Z}$), can we define any other? Indeed, the most important mission of our model is to provide (hopefully) safe ways to do so. (Here *safe* means logically consistent; the explanation shall follow).

"Out of many, one." Let $a_1, \ldots, a_n$ be some sets. Then there exists a set $\{a_1, \ldots, a_n\}$ such that

$$x \in \{a_1, \ldots, a_n\} \iff x = a_1 \vee x = a_2 \vee \ldots \vee x = a_n$$

for each set x.

> Of course, there is no *logical* necessity in turning the Axiom of Pairing into a schema for various $n \in \mathbb{N}$ as $\{a, b, c\} = \cup\{\{a, b\}, \{c, c\}\}$. Yet this observation uses the union essentially; so we have here preferred a freshman's comfort to logical elegance.

Example 4.5. We can construct sets $\{\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}\}$ and $\{\mathbb{N}\}$. Clearly, $\mathbb{N} \in \{\mathbb{N}\}$. However $\mathbb{N} \not\subseteq \{\mathbb{N}\}$[4] for otherwise, from $0 \in \mathbb{N}$, it follows that $0 \in \{\mathbb{N}\}$, whence $0 = \mathbb{N}$ and $0 \in 0$. The latter is not possible by the Axiom of Foundation. On the other hand, $\mathbb{N} \subseteq \mathbb{N}$ yet $\mathbb{N} \notin \mathbb{N}$ (again, by the Axiom of Foundation).

For any set A, consider the set $\{A\}$, which is called the *singleton* of the set A. By definition, we get

$$x \in \{A\} \iff x = A$$

for each x. That is, A is the only element of $\{A\}$. The difference between A and $\{A\}$ is not readily conceivable for our intuition as we usually do not see the group of one individual differ essentially from that individual itself. (Just think about a point of the plane and the geometric figure formed solely by that point.) However, such a distinction is crucial for the set-theoretic model.

Example 4.6. We have $\{2, 2, 3\} = \{3, 2\}$. Indeed,

$$x \in \{2, 2, 3\} \iff x = 2 \vee x = 2 \vee x = 3 \iff x = 3 \vee x = 2 \iff x \in \{3, 2\}.$$

Thus, neither the order of elements nor the occurrence count thereof are caught by the notion of set. We shall later see ways to express these ideas indirectly.

[4] The notations $A \not\subseteq B$ and $A \notin B$ stand for $\neg(A \subseteq B)$ and $\neg(A \in B)$ respectively.

Subset specification. Let us recall that we write $\varphi(x)$ if x satisfies a property (or predicate) φ. For example, even(2) means that the number 2 satisfies the property to be even, i. e., 2 is even. Let A be a set and φ be some property. Then there exists a set $\{x \in A \mid \varphi(x)\}$ such that

$$y \in \{x \in A \mid \varphi(x)\} \iff y \in A \wedge \varphi(y)$$

for all y. In other words, we pick up and put together all the elements of A satisfying φ. Clearly, $\{x \in A \mid \varphi(x)\} \subseteq A$.

Symbols like $\{x \in A \mid \varphi(x)\}$ are widely known as *set-builder notation.*

Example 4.7. The set $\{x \in \mathbb{N} \mid x \text{ is even}\}$ is the set of all even natural numbers.

But what if φ is a property satisfied by no set? Namely, consider the set

$$\varnothing = \{x \in \mathbb{N} \mid x \neq x\}.$$

By Corollary 4.4, $x = x$ for each x. By definition, $x \neq x$ for each $x \in \varnothing$. Hence, *there is no x such that $x \in \varnothing$.* A set E is called *empty* (or *void*) if $x \notin E$ for all x. Thus, the set $\varnothing$ is empty. Is there any other empty set?

Lemma 4.8.

1. *If E is an empty set, then $E \subseteq A$ for each set A.*

2. *If sets E_1 and E_2 are empty, then $E_1 = E_2$.*

Proof. The second statement follows from the first one by Lemma 4.3. Let us prove the first statement. For each x, we have $x \notin E$. So if we suppose that $x \in E$, we immediately get a falsity, a contradiction. By the logical convention, *from a falsity it follows everything.* Thus, we can safely conclude that $x \in A$ (as well as $1 = 2$, etc.) when $x \in E$. □

Corollary 4.9. *The set $\varnothing$ is the unique empty set.*

Sets and predicates. As we have just seen, the subset specification principle allows to form a set of all those elements of some given set A which satisfy a predicate φ. This is one of the most important ideas behind sets indeed. Often, we will replace predicates with sets in mathematical arguments.

For example, assume that we want to formalize the statement “every even number equals the sum of two odd numbers”. Stating from the formalization

$$\forall x \left(x \text{ is even } \to \exists y \exists z \left(y \text{ is odd} \wedge z \text{ is odd } \wedge x = y + z\right)\right),$$

we may transform it to

$$\forall x \left(x \in A \to \exists y \exists z \left(y \in B \wedge z \in B \wedge x = y + z\right)\right),$$

where A and B stand for the sets of even and odd numbers, respectively. Please notice how “satisfying a predicate” has been replaced by “belonging to a set” here. Usually, they would employ a more concise notation (so called *bounded quantifiers*) in this case:

$$\forall x \in A \, \exists y \in B \, \exists z \in B \, (x = y + z)$$

(to be read as “for each x from A there exist y and z from B such that…”—which sounds very intuitive and natural, does not it?).

Bounded quantifiers come in many varieties ($\forall x \in A$, $\forall x < 10$, $\exists x > 10$, etc.) but it is important to realize that *bounded universal* quantifier means *implication*, while *bounded existential* quantifier means *conjunction*. Formally,

$$\forall x \in A\ \varphi \equiv \forall x\,(x \in A \to \varphi) \qquad \text{and} \qquad \exists x \in A\ \varphi \equiv \exists x\,(x \in A \land \varphi).$$

Exercise 4.10. Prove that $\forall x \in A\ \varphi$ is true whereas $\exists x \in A\ \varphi$ is false for $A = \varnothing$ and whatever statement φ.

Exercise 4.11. Prove that $\neg\forall x \in A\ \varphi \equiv \exists x \in A\ \neg\varphi$ and $\neg\exists x \in A\ \varphi \equiv \forall x \in A\ \neg\varphi$ for every set A and statement φ.

Russell’s Paradox. Clearly, we can only specify subsets of a given set A via $\{x \in A \mid \varphi(x)\}$. But could not we put together *all* x in existence (i. e. not only from A) such that x satisfies φ? Historically, this was the first intuition backing the notion of set: any collection of objects having something in common or satisfying some property.

As we shall see in a moment, this intuition proves to be logically inconsistent.

Lemma 4.12 (“Russell’s Paradox”)**.** *There is no set R such that*

$$\forall x\,(x \in R \iff x \notin x).$$

Proof. Suppose there is such a set R. At first, assume $R \notin R$. By the constraint on R, this implies $R \in R$. By contradiction, $R \notin R$ must be false, so we have proved $R \in R$ (assuming the existence of R). But then $R \notin R$ by the same constraint on R. Thus, the assumption that such R exists is contradictory. Hence, it does not exist. □

So, one cannot define a set $\{x \mid x \notin x\}$ comprising *all* sets x satisfying quite a natural condition $x \notin x$. This very fact was seen as paradoxical and counter-intuitive.

In view of the Axiom of Foundation, R is *the set of all sets*, which does not thus exist. The Instructor might ask the students to prove the latter statement *without* that axiom and might then explain why proofs from *weaker* assumptions (even the “axioms”) are usually worth looking for.

Exercise 4.13. There is no set V such that $\forall x\,(x \in V)$. In other words, there is no set of all sets. Try not to use the Axiom of Foundation.

Power set. Let A be a set. Then there exists a set $\mathcal{P}(A)$ such that

$$x \in \mathcal{P}(A) \iff x \subseteq A$$

for all x. In other words, $\mathcal{P}(A)$ is the set of all *subsets* of the set A.

As $\varnothing \subseteq A$, $\varnothing \in \mathcal{P}(A)$ for any set A. We have $\mathcal{P}(\varnothing) = \{\varnothing\}$, $\mathcal{P}(1) = \{\varnothing, \{1\}\}$, and $\mathcal{P}(\{1,2\}) = \{\varnothing, \{1\}, \{2\}, \{1,2\}\}$. Note that $\{1,2\}$ has 2 elements, while $\mathcal{P}(\{1,2\})$ has $4 = 2^2$ elements, $\mathcal{P}(\varnothing)$ has $1 = 2^0$ elements, and $\mathcal{P}(\{1\})$ has $2 = 2^1$ elements. This analogy between power sets and numeric *powers* of 2 is not a coincidence.

Clearly, every formal proof that the set $\{1,2\}$ has just those subsets as specified above boils down to an exhaustive search argument (for one must at least write all the subsets down). It might be beneficial to draw the tree of all subsets for a small size example, yet we skip such formal proofs in general.

Example 4.14. If $\mathcal{P}(X) = \mathcal{P}(Y)$ then $X = Y$, and vice versa.

Indeed, suppose that $\mathcal{P}(X) = \mathcal{P}(Y)$. We have $X \subseteq X$, that is, $X \in \mathcal{P}(X)$. By the assumption, $X \in \mathcal{P}(Y)$, whence $X \subseteq Y$. By a similar argument, $Y \subseteq X$. Then we get $X = Y$ by Lemma 4.3.

Now suppose that $X = Y$. Consider an arbitrary $a \in \mathcal{P}(X)$. Then $a \subseteq X = Y$, which yields $a \subseteq Y$ by Lemma 4.3, hence $a \in \mathcal{P}(Y)$. So, $\mathcal{P}(X) \subseteq \mathcal{P}(Y)$. The other inclusion is similar.

Union. Let A be a set. Then there exists a set $\cup A$ such that

$$x \in \cup A \iff \exists \alpha\ (x \in \alpha \wedge \alpha \in A)$$

for all x. Thus, $\cup A$ is the set of elements of elements of the set A.

> This concept is most likely unknown for most students; so they may need some time to grasp it. In our practice, drawing (potentially) infinite families of planar figures and highlighting their unions with color prove beneficial.
>
> Moreover, depending on the audience, it might be better to replace this general principle by a weaker version: *For every sets A and B, there exists a set $A \cup B$ such that $x \in A \cup B \iff x \in A \vee x \in B$ for all x.* This version is sufficient for most of this Course's developments. Countable unions pose a noticeable exception; they are yet neither frequent, nor crucial, nor hard to explain as most students will consider them "natural" and "logical".

Example 4.15. We have $\cup\varnothing = \varnothing$. Indeed, if $x \in \cup\varnothing$, then $x \in \alpha \in \varnothing$ for some set α; but $\alpha \in \varnothing$ is impossible, hence a contradiction; thus, no x can be an element of $\cup\varnothing$; therefore, the set $\cup\varnothing$ is empty. By Lemma 4.8, $\varnothing$ is the only empty set.

For any A, $\cup\{A\} = A$. We see that $x \in \cup\{A\} \iff \exists\alpha\ x \in \alpha \in \{A\}$, while $\alpha \in \{A\}$ means that $\alpha = A$. So, $x \in \cup\{A\} \iff \exists\alpha\ x \in \alpha = A \iff x \in A$. (For the last equivalence, one can put $\alpha = A$).

If $A = \{\{1,2,3\},\{1\},\{2,4\}\}$, then for any x,

$$\begin{aligned}
x \in \cup A &\iff \exists\alpha\ (x \in \alpha \wedge \alpha \in A) \\
&\iff \exists\alpha\ (x \in \alpha \wedge (\alpha = \{1,2,3\} \vee \alpha = \{1\} \vee \alpha = \{2,4\})) \\
&\iff \exists\alpha\ ((x \in \alpha \wedge \alpha = \{1,2,3\}) \vee (x \in \alpha \wedge \alpha = \{1\}) \vee (x \in \alpha \wedge \alpha = \{2,4\})) \\
&\iff \exists\alpha\ (x \in \alpha \wedge \alpha = \{1,2,3\}) \vee \exists\alpha\ (x \in \alpha \wedge \alpha = \{1\}) \vee \exists\alpha\ (x \in \alpha \wedge \alpha = \{2,4\}) \\
&\iff x \in \{1,2,3\} \vee x \in \{1\} \vee x \in \{2,4\} \\
&\iff x = 1 \vee x = 2 \vee x = 3 \vee x = 1 \vee x = 2 \vee x = 4 \\
&\iff x = 1 \vee x = 2 \vee x = 3 \vee x = 4 \\
&\iff x \in \{1,2,3,4\}.
\end{aligned}$$

Thus, $\cup A = \{1,2,3,4\}$.

> For the above argument, the Instructor might wish to explain a few logical equivalences: $\exists x\,(A \vee B) \equiv \exists x\,A \vee \exists x\,B$ and $\exists x\,(A(x) \wedge x = t) \equiv A(t)$.

Example 4.16. Let S be the set of all segments of length 1 in the line $\mathbb{R}$. (Let's recall that a *segment* is a set of the form $[a,b] = \{x \in \mathbb{R} \mid a \leq x \leq b\}$ for some numbers $a, b \in \mathbb{R}$. Clearly, $[a,b] = \varnothing$ if $b < a$. Assume that $a \leq b$. Then the number $b - a \geq 0$ is called the *length* of the segment $[a,b]$. Easily, the length of $[a,b]$ equals 1 iff $b = a + 1$.)

Obviously, $S = \{X \in \mathcal{P}(\mathbb{R}) \mid \exists a \in \mathbb{R}\ X = [a, a+1]\}$. Less accurately, the set S can also be denoted by $\{[a, a+1] \mid a \in \mathbb{R}\}$. What is the set $\cup S$? Let us show that $\cup S = \mathbb{R}$.

If $x \in \cup S$, then $x \in X \in S$ for some X, or equivalently, $x \in [a, a+1]$ for some $a \in \mathbb{R}$. By the definition of segment, it should be that $x \in \mathbb{R}$.

For the other direction, suppose $x \in \mathbb{R}$. There exists a segment from S covering (i. e. containing) x—in fact, infinitely many such segments. Indeed, we have $x \leq x \leq x+1$, $x - \frac{1}{2} \leq x \leq x + \frac{1}{2}$, and so on. Thus, $x \in [x, x+1] \in S$, whence $x \in \cup S$.

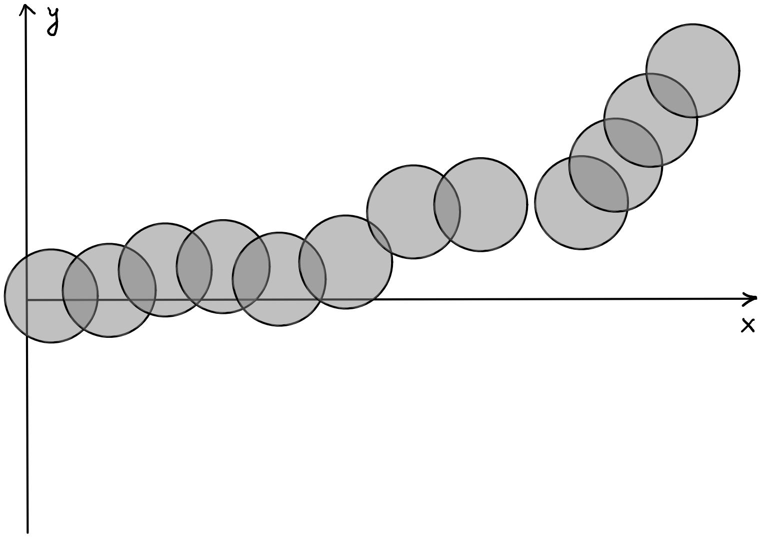

Figure 1: A union in the plane. The highlighted area $\cup C$, which is a set of plane *points*, is the union of a set of *circles* C. In particular, C is finite while $\cup C$ is infinite.

Example 4.17. If $X \subseteq Y$, then $\cup X \subseteq \cup Y$.

Consider an arbitrary $a \in \cup X$. By definition, we get $a \in A$ for some $A \in X$. From our assumption, it follows that $A \in Y$. Thus, $\exists a \; a \in A \in Y$, which means $a \in \cup Y$. Generalizing, we obtain that $a \in \cup X$ implies $a \in \cup Y$ for *any* a, i. e., $\cup X \subseteq \cup Y$.

Algebra of sets. For arbitrary sets A and B, by $A \cup B$ we denote the set $\cup\{A, B\}$, which is called the *union of the sets A and B*. Arguing similarly to Example 4.15, it is easy to prove that

$$x \in A \cup B \iff x \in A \vee x \in B$$

for any x. Indeed,

$$\begin{aligned} x \in A \cup B &\iff x \in \cup\{A, B\} \\ &\iff \exists \alpha \; (x \in \alpha \wedge \alpha \in \{A, B\}) \\ &\iff \exists \alpha \; (x \in \alpha \wedge (\alpha = A \vee \alpha = B)) \\ &\iff \exists \alpha \; ((x \in \alpha \wedge \alpha = A) \vee (x \in \alpha \wedge \alpha = B)) \\ &\iff \exists \alpha \; (x \in \alpha \wedge \alpha = A) \vee \exists \alpha \; (x \in \alpha \wedge \alpha = B) \\ &\iff x \in A \vee x \in B. \end{aligned}$$

Also, we define the *intersection*

$$A \cap B = \{x \in A \mid x \in B\}$$

and the *difference*

$$A \smallsetminus B = \{x \in A \mid x \notin B\}$$

of sets A and B. Clearly,

$$x \in A \cap B \iff x \in A \wedge x \in B$$

for any x. We say that the set A *intersects* B if $A \cap B \neq \varnothing$. Otherwise, these sets are called *disjoint*.

Lemma 4.18. *For any sets A and B, one has $A \cap B \subseteq X \subseteq A \cup B$ if $X \in \{A, B\}$. Also, $A \setminus B \subseteq A$ and $(A \setminus B) \cap B = \varnothing$.*

Lemma 4.19. *For any sets A and B, the following statements are equivalent:*

1. *$A \subseteq B$;*
2. *$A \cap B = A$;*
3. *$A \cup B = B$.*

Proof. It suffices to show that the first one implies the second, the second implies the third, and the third one implies the first statement.

Suppose that $A \subseteq B$. By Lemma 4.18, we get $A \cap B \subseteq A$. Let us check if $A \subseteq A \cap B$. Consider an arbitrary $x \in A$. Then $x \in B$ as $A \subseteq B$. Hence, $x \in A \cap B$. By Lemma 4.3, from $A \cap B \subseteq A$ and $A \subseteq A \cap B$ it follows that $A \cap B = A$.

Now assume that $A \cap B = A$. By Lemma 4.18, $B \subseteq A \cup B$. It remains to prove that $A \cup B \subseteq B$. If $x \in A \cup B$, then $x \in A$ or $x \in B$. In the former case, obtain $x \in A \cap B$ from $A = A \cap B$, whence $x \in B$. In the latter one, $x \in B$ is immediate.

Finally, suppose that $A \cup B = B$. By Lemma 4.18, we have $A \subseteq A \cup B$. By assumption, we get $A \cup B \subseteq B$, hence $A \subseteq B$. □

Sometimes, all the sets we consider are subsets of some fixed set U, which is called *the universe* (for this particular case). If a universe U has been specified (or is clear from the context), for any $A \subseteq U$ we define the set

$$\bar{A} = U \setminus A,$$

which is called the *complement* of A (to U). Obviously, $A \setminus B = A \cap \bar{B}$ for every $A, B \subseteq U$.

> Some students mistake this ‘universe’ for “the set of all sets”. The Instructor should discourage this misconception.

Theorem 4.20 (Set algebra identities)**.** *For all sets U and $A, B, C \subseteq U$,*

1. *$A \cap B = B \cap A$; $A \cup B = B \cup A$;*
2. *$(A \cap B) \cap C = A \cap (B \cap C)$; $(A \cup B) \cup C = A \cup (B \cup C)$;*
3. *$A \cap A = A$; $A \cup A = A$;*
4. *$A \cap (A \cup B) = A$; $A \cup (A \cap B) = A$;*
5. *$\bar{\bar{A}} = A$;*
6. *$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$; $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$;*
7. *$\overline{A \cap B} = \bar{A} \cup \bar{B}$; $\overline{A \cup B} = \bar{A} \cap \bar{B}$;*
8. *$A \cap \varnothing = \varnothing$; $A \cup \varnothing = A$; $A \cap U = A$; $A \cup U = U$; $\bar{\varnothing} = U$; $\bar{U} = \varnothing$;*

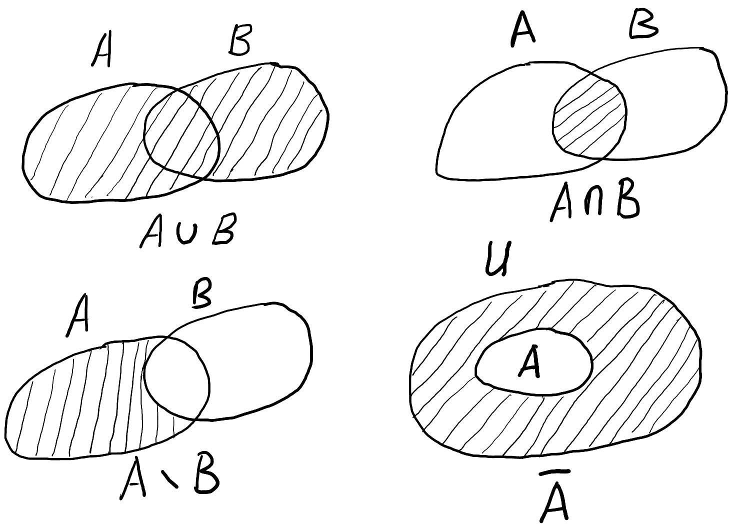

Figure 2: *Euler diagrams* for set algebra operations. Sets are identified with planar figures and their elements—with points of the plane. However useful they are, no such diagram *proves* anything.

9. $A \cap \bar{A} = \varnothing$*;* $A \cup \bar{A} = U$*.*

Proof. The proof for the theorem is just a rephrasing of logical equivalences for conjunction, disjunction, and negation, which follow from the respective truth tables.

Let us check, say, whether $\overline{A \cap B} = \bar{A} \cup \bar{B}$. For every x, one has

$$x \in \overline{A \cap B} \iff x \in U \wedge \neg(x \in A \cap B) \iff x \in U \wedge \neg(x \in A \wedge x \in B) \iff$$
$$x \in U \wedge (\neg x \in A \vee \neg x \in B) \iff (x \in U \wedge \neg x \in A) \vee (x \in U \wedge \neg x \in B) \iff$$
$$x \in \bar{A} \vee x \in \bar{B} \iff x \in \bar{A} \cup \bar{B}.$$

□

Remark 4.21. As $(A \cap B) \cap C = A \cap (B \cap C)$, we can omit parentheses in expressions like $A_1 \cap A_2 \cap \ldots \cap A_n$. Of course, the same applies for the union as well.

When studying set identities, one often uses complement without any explanations. Say, in the example following, one can put $U = \cup\{A, B, C\}$ for *any* choice of the sets A, B, C.

Example 4.22. Let us show that $A \smallsetminus (B \smallsetminus C) = (A \smallsetminus B) \cup (A \cap C)$ using the identities already known. Indeed,

$$A \smallsetminus (B \smallsetminus C) = A \cap \overline{B \cap \bar{C}} = A \cap (\bar{B} \cup \bar{\bar{C}}) =$$
$$A \cap (\bar{B} \cup C) = (A \cap \bar{B}) \cup (A \cap C) = (A \smallsetminus B) \cup (A \cap C).$$

Example 4.23. For every sets A, B, C, if $A \subseteq C$ and $B \subseteq C$, then $A \cup B \subseteq C$.

By the assumption and Lemma 4.19, obtain $C = A \cup C$ and $C = B \cup C$. Now, substitute $B \cup C$ for the second occurrence of C in the first equality. Then $C = A \cup (B \cup C)$. But $A \cup (B \cup C) = (A \cup B) \cup C$ by Theorem 4.20, hence $C = (A \cup B) \cup C$. This yields that $A \cup B \subseteq C$ in view of Lemma 4.19.

Exercise 4.24. Prove that if $C \subseteq A$ and $C \subseteq B$, then $C \subseteq A \cap B$.

Cartesian product. Most branches of mathematics employ ordered "pairs", "triplets" etc. E. g., points of the plain are routinely identified with their coordinates, whose order does matter. Indeed, $(0, 1)$ and $(1, 0)$ are *distinct* points. While sets themselves do not respect any "order of the elements" (as $\{2, 3\} = \{3, 2\}$), ordered pairs can be easily modeled using sets.

For arbitrary sets a and b consider the set

$$(a, b) = \{\{a\}, \{a, b\}\},$$

which is called an *(ordered) pair* of sets a и b. The main property of such a pair is the following

Lemma 4.25. *For all sets a, b, c, d,*

$$(a, b) = (c, d) \iff a = c \wedge b = d.$$

Proof. Assume that $\{\{a\}, \{a, b\}\} = \{\{c\}, \{c, d\}\}$. Then $\{a\} \in \{\{c\}, \{c, d\}\}$, i. e., $\{a\} = \{c\}$ or $\{a\} = \{c, d\}$. In the first case, $a \in \{c\}$, hence $a = c$. In the second case, $c \in \{a\}$ and $c = a$ again. Thus, $a = c$.

From the assumption, it also follows that $\{a, b\} = \{c\}$ or $\{a, b\} = \{c, d\}$.

In the former case, $b \in \{c\}$, whence $b = c = a$. By the assumption, $\{c, d\} = \{a\}$ or $\{c, d\} = \{a, b\}$. Therefore, $d = a$ or $d = b$. Anyway, $b = d$.

Now suppose that $\{a, b\} = \{c, d\}$. If $d = b$, there is nothing to prove. Otherwise, $d = a = c$, i. e., $\{a, b\} = \{d\}$, whence $b = d$ again.

For the other implication, it suffices to recall that equal sets are fully interchangeable; so, from $(a, b) = (a, b)$, $a = c$ and $b = d$, it follows that $(a, b) = (c, d)$. □

Corollary 4.26. *For all sets a and b, one has $(a, b) = (b, a)$ iff $a = b$.*

Remark 4.27. If $a, b \in X$, then $(a, b) \in \mathcal{P}(\mathcal{P}(X))$.

Indeed, both $\{a\}$ and $\{a, b\}$ are subsets of X and are thus elements of $\mathcal{P}(X)$. Hence, $(a, b) = \{\{a\}, \{a, b\}\} \subseteq \mathcal{P}(X)$.

The *Cartesian product* of sets A and B is the set

$$A \times B = \{z \in \mathcal{P}(\mathcal{P}(A \cup B)) \mid \exists a \in A\, \exists b \in B\ z = (a, b)\}.$$

The existence of this set is secured by the subset specification principle. In the above formula, the term $\mathcal{P}(\mathcal{P}(A \cup B))$ (denoting the set where z is taken from) is required by the principle. But we can omit it otherwise: for every set z,

$$z \in A \times B \iff \exists a \in A\, \exists b \in B\ z = (a, b).$$

Indeed, if the right-hand side holds, one has $a, b \in A \cup B$ and $(a, b) \in \mathcal{P}(\mathcal{P}(A \cup B))$ by Remark 4.27. A less accurate but simpler notation for Cartesian product

$$A \times B = \{(a, b) \mid a \in A,\ b \in B\}$$

is inspired by such considerations.

Example 4.28. Clearly, $A \times \varnothing = \varnothing$ for any A. Indeed, if $z \in A \times \varnothing$, then $z = (a, b)$ and $b \in \varnothing$ would be true for some sets a, b. But the latter is impossible. Hence, $A \times \varnothing$ is empty.

Example 4.29. Let us see why $A \times B$ is not necessarily equal to $B \times A$. Indeed, for $A = \{x\}$ and $B = \{y\}$, one has $A \times B = \{(x, y)\}$ and $B \times A = \{(y, x)\}$. Now, consider $x = \varnothing$ and $y = \{\varnothing\}$. As $\varnothing \in y$, y is non-empty and $y \neq x$. Thus, $(x, y) \neq (y, x)$ by Lemma 4.25, hence $A \times B \neq B \times A$.

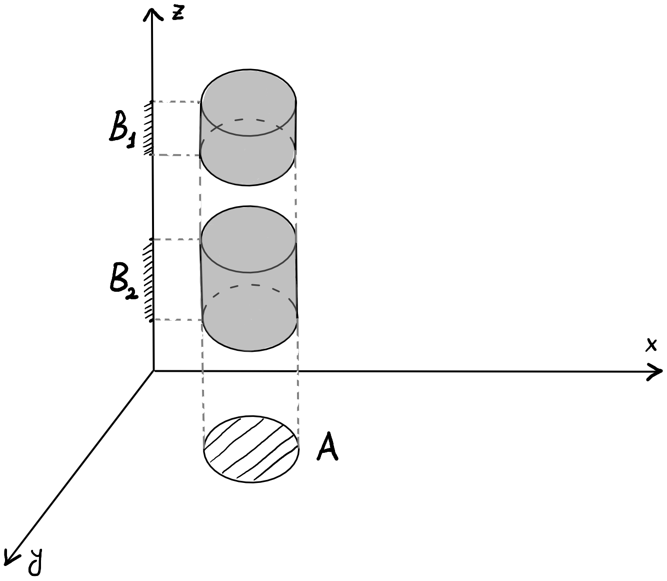

Figure 3: A Cartesian product in the space. The set $A \times (B_1 \cup B_2)$ is highlighted.

Although $A \times (B \times C) \neq (A \times B) \times C$ generally, we can omit the parentheses in products according to the rule:

$$A_1 \times A_2 \times A_3 \times \ldots \times A_{n-1} \times A_n = (\ldots((A_1 \times A_2) \times A_3) \times \ldots \times A_{n-1}) \times A_n.$$

Remark 4.30. If $(x, y) \in A \times B$, then $x \in A$ and $y \in B$ (and vice versa).

Indeed, suppose that $(x, y) \in A \times B$. By the definition of product, there are some $a \in A$ and $b \in B$ such that $(x, y) = (a, b)$. By Lemma 4.25, get $x = a$ and $y = b$, whence $x \in A$ and $y \in B$.

This statement is not such a trifle as it seems to be. Consider a 'sum' of numeric sets to see it. Let $A, B \subseteq \mathbb{N}$ and $A + B = \{a + b \mid a \in A, b \in B\} = \{z \in \mathbb{N} \mid \exists a \in A \, \exists b \in B \;\, z = a + b\}$. Then $2 + 3 = 4 + 1 \in \{1, 4, 0\} + \{5, 1\}$, despite $2 \notin \{1, 4, 0\}$ and $3 \notin \{5, 1\}$.

Example 4.31. For all sets A, B, C, D,

$$(A \times B) \cap (C \times D) = (A \cap C) \times (B \cap D).$$

Let $z \in (A \times B) \cap (C \times D)$. Then $z \in A \times B$ and $z \in C \times D$. Hence, $z = (a, b)$ for some $a \in A$ and $b \in B$. From $(a, b) \in C \times D$, obtain $a \in C$ and $b \in D$ by Remark 4.30. We have $a \in A \cap C$ and $b \in B \cap D$, whence $z = (a, b) \in (A \cap C) \times (B \cap D)$.

For the other inclusion, suppose that $z \in (A \cap C) \times (B \cap D)$. Then there exist $x \in A \cap C$ and $y \in B \cap D$ such that $z = (x, y)$. Since $x \in A$ and $y \in B$, we get $z \in A \times B$. Similarly, $z \in C \times D$.

> We will allow more laxity (like in saying "let $(x, y) \in (A \times B) \cap (C \times D)$") in treating pairs when the students have clearly understood (hopefully) that one owes all such laxity solely to Lemma 4.25.

Exercise 4.32. Prove that $(A \cup B) \times C = (A \times C) \cup (B \times C)$.

Example 4.33. Let a set C be non-empty. Then from $A \times C \subseteq B \times C$, it follows that $A \subseteq B$, and vice versa.

Suppose that $A \times C \subseteq B \times C$ and $a \in A$. As $C \neq \varnothing$, there is some $c_0 \in C$, that satisfies $(a, c_0) \in A \times C$. Hence, $(a, c_0) \in B \times C$ and $a \in B$.

Now, assume $A \subseteq B$. By Lemma 4.19, get $A = A \cap B$, whence $A \times C = A \times (C \cap C) = (A \cap B) \times (C \cap C)$. Applying Example 4.31, obtain

$$A \times C = (A \times C) \cap (B \times C) \subseteq B \times C.$$

Note that it is necessary to assume $C \neq \varnothing$ for the first implication. For otherwise, one has $\mathbb{R} \times \varnothing = \varnothing = \mathbb{N} \times \varnothing$ but $\mathbb{R} \nsubseteq \mathbb{N}$.

> The clear resemblance of this statement to the cancellation laws for numerical multiplication is noteworthy. In this one and many examples that follow, the Instructor should try to persuade the students, that the Cartesian *product* is a "product" indeed.

Cartesian power. Just like for numerical multiplication, we can define powers in the sense of Cartesian product. For an arbitrary set A and for all natural $n \geq 2$, let

$$\begin{array}{rcl} A^0 & = & \{\varnothing\}; \\ A^1 & = & A; \\ A^n & = & \underbrace{A \times A \times \ldots \times A}_{n \text{ copies of } A}. \end{array}$$

> We do not want any explicit recursion here since we have no Replacement axioms to make it rigorous. We prefer this schematic definition for n running over $\mathbb{N}$.

Remark 4.34. Notice that $A^{n+1} = A^n \times A$ for all $n \geq 1$ whereas A^0 has exactly *one* element. (Like $x^{n+1} = x^n \cdot x$ and $x^0 = 1$ for any numbers x and n .)

For $n \geq 2$, the set

$$(a_1, \ldots, a_{n-1}, a_n) = ((\ldots((a_1, a_2), a_3), \ldots, a_{n-1}), a_n)$$

is called the *n-tuple* of sets $a_1, \ldots, a_{n-1}$, a_n. The sets a_i are then called *components* or *coordinates* of the tuple $(a_1, \ldots, a_{n-1}, a_n)$.

Lemma 4.35. *For any number $n \geq 2$ and sets $a_1, \ldots, a_n$, $b_1, \ldots, b_n$,*

$$(a_1, \ldots, a_n) = (b_1, \ldots, b_n) \iff a_i = b_i \text{ for all } i \in \{1, \ldots, n\}.$$

Proof. For every n, it suffices to apply Lemma 4.25 $(n-1)$ times. □

Remark 4.36. Clearly, for every x,

$$x \in A_1 \times A_2 \times \ldots \times A_n \iff \exists a_1 \in A_1 \ldots \exists a_n \in A_n \; x = (a_1, \ldots, a_n),$$

and

$$x \in A^n \iff \exists a_1 \in A \ldots \exists a_n \in A \; x = (a_1, \ldots, a_n)$$

in particular.

5 Induction on Naturals

The main goal of this section is to introduce three popular forms of induction principle for $\mathbb{N}$: the most common one, the 'Strong' (or 'Well-founded' or 'Noetherian') induction, and the Least Number principles. Then we prove that all three principles are equivalent to each other. The Instructor should encourage the students to apply these principles as formally as it is reasonably possible at their first steps.

Let us have a closer look at the set $\mathbb{N} = \{0, 1, 2, \dots\}$ of *natural numbers*. In fact, we have not yet defined this set accurately, and we will refrain from doing it during this course.[5] We have taken the set $\mathbb{N}$ with some 'basic properties', like $0 \neq 1$ or even $2 < 3$ and $3 + 2 = 5$. Now we are to analyze the main property of $\mathbb{N}$—that of *induction*.

In its most well-known form, the *Induction Principle* says that

> For an arbitrary predicate φ, if $\varphi(0)$ holds and for each $n \in \mathbb{N}$, $\varphi(n)$ implies $\varphi(n+1)$, then $\varphi(n)$ holds for all $n \in \mathbb{N}$.

One can easily put this into symbols (here φ is fixed):

$$\big(\varphi(0) \wedge \forall n \in \mathbb{N}\,(\varphi(n) \to \varphi(n+1))\big) \to \forall n \in \mathbb{N}\,\varphi(n).$$

The statement $\varphi(0)$ is called the *base step of induction*, while $\forall n \in \mathbb{N}\,(\varphi(n) \to \varphi(n+1))$ is called the *inductive step*, and the statement $\varphi(n)$ for each n is called the *inductive hypothesis* (which is usually shortened to 'IH').

If the Instructor has skipped the chapter on strings, I suggest he proves 'intuitive validity' of the Induction Principle similarly to Example 3.3.

It is possible to rephrase the principle without mentioning any 'predicates' but sets:

> For an arbitrary set $X \subseteq \mathbb{N}$, if $0 \in X$ and for each $n \in \mathbb{N}$, $n \in X$ implies $n + 1 \in X$, then $X = \mathbb{N}$.

Indeed, for each predicate φ, we can consider the subset $X = \{n \in \mathbb{N} \mid \varphi(n)\}$ of $\mathbb{N}$ and, conversely, we can put the predicate $n \in X$ into correspondence with a set $X \subseteq \mathbb{N}$. Thus, the new statement of the Principle is equivalent to the original one.

In practice, the students usually ask me to elaborate on this point. Sometimes, we abandon the 'set version' of induction (except for the Least Number Principle, of course) altogether.

Example 5.1. Let us show that for each natural $n \geq 3$, there are some numbers $a_1, \dots, a_n \in \mathbb{N}_+$ such that

$$1 = \frac{1}{a_1} + \ldots + \frac{1}{a_n},$$

where $a_i \neq a_j$ whenever $i \neq j$.

We give a (too) detailed proof. Consider the set

$$X = \{n \in \mathbb{N} \mid \ n \geq 3 \to \text{there exist } a_1, \dots, a_n \text{ as required}\}.$$

[5] The standard set-theoretic model for naturals is as follows: $0 = \varnothing$, $1 = \{0\}$, $2 = \{0, 1\}$, $\dots$, $n + 1 = n \cup \{n\} = \{0, 1, \dots, n\}$. Then one can define $n < m$ as $n \in m$ and define $n + m$ recursively by the equations $n + 0 = n$ and $n + (m + 1) = (n + m) + 1$. Yet there are still plenty of things to elaborate on.

The Instructor might prefer a ‘predicate version’ of this inductive proof instead.

Many students justly see that the base case here is that of $n = 3$ essentially. Then they try to change *the inductive principle* accordingly rather than the set or predicate in question. In my view, this practice is caused by their aversion to anything else than a simple equation in the inductive statement (like, say, an implication). This is counterproductive and the students should be discouraged from doing so at least in these easy early stage examples. They should be rather taught to adapt to the *principle as it is* and to flexibly change the statement they want. Of course, these requirements may be finally lifted for harder problems and for students able to solve them.

Let us check if X satisfies the assumptions of the Induction Principle. Clearly, $0 \in X$. For an arbitrary $n \in X$, we are to prove that $n + 1 \in X$. This is immediate if $n + 1 < 3$. When $n + 1 = 3$, we get $n + 1 \in X$ by the identity

$$1 = \frac{1}{2} + \frac{1}{3} + \frac{1}{6}.$$

See that the IH $2 \in X$ is of no use here.[6] Essentially, the statement $3 \in X$ is the base step of this induction, but *formally* we are content with the base of 0.

Finally, suppose that $n+1 > 3$. Then $n \geq 3$. When combined with $n \in X$, this yields some pairwise distinct numbers $a_1, \ldots, a_n$ s. t.

$$1 = \frac{1}{a_1} + \ldots + \frac{1}{a_n}.$$

Multiply the latter equality by $\frac{1}{2}$, then add $\frac{1}{2}$ to both sides thus obtaining

$$1 = \frac{1}{2} + \frac{1}{2} = \frac{1}{2} + \frac{1}{2a_1} + \ldots + \frac{1}{2a_n}.$$

Clearly, we can take $2, 2a_1, \ldots, 2a_n$ as the numbers we are searching for, as soon as we prove them to be pairwise distinct. When $i \neq j$, we have both $a_i \neq a_j$ and $2a_i \neq 2a_j$. But what if $2 = 2a_i$ for some i? Then $\frac{1}{a_i} = 1$, so n cannot be greater than one despite the fact that $n \geq 3$. Hence, this is impossible.

Both the base and inductive steps have been verified, so we conclude $X = \mathbb{N}$ by the Induction Principle. This means that each natural $n \geq 3$ has got the respective numbers $a_1, \ldots, a_n$.

Another popular form of induction is the *Strong Induction Principle*. Basically, it allows us to use not only $\varphi(n)$ but all the statements $\varphi(0), \varphi(1), \ldots, \varphi(n)$ together as the inductive hypothesis to draw $\varphi(n + 1)$ from. Thus,

> For an arbitrary predicate φ, if $\varphi(0)$ is true and for every $n \in \mathbb{N}$, $\varphi(n + 1)$ holds whenever each of $\varphi(0), \varphi(1) \ldots, \varphi(n)$ holds, then $\varphi(n)$ is true for all $n \in \mathbb{N}$.

Let us give a more concise rewording of the Principle, eliminating any ‘predicates’ by the way. We call a set $X \subseteq \mathbb{N}$ *progressive*, if for each $n \in \mathbb{N}$, from $\forall m < n \; m \in X$, it follows that $n \in X$.[7] We write $Prog(X)$ if X is progressive. Then the principle reads this way:

> For an arbitrary set $X \subseteq \mathbb{N}$, if X is progressive, then $X = \mathbb{N}$.

Of course, the Instructor might prefer to call a *predicate* progressive.

[6] Evidently, from $1 = \frac{1}{a} + \frac{1}{b}$, it follows that $a = b = 2$.

[7] Recall that the bounded quantifier $\forall m < n \; \varphi$ means $\forall m \; (m < n \to \varphi)$.

Informally, a set X is progressive if it 'climbs' the set $\mathbb{N}$, 'conquering' n as soon as it has 'conquered' all lesser numbers. Most strikingly, this form of induction apparently lacks the base step. But this is not the case.

> In practice, the students usually wonder where the base case has gone to.

Lemma 5.2. *If X is progressive, then $0 \in X$.*

Proof. Suppose that for every n, $\forall m < n\ m \in X$ implies $n \in X$. Let us specify n as 0. Consider the assumption $\forall m < 0\ m \in X$, that is, $\forall m\,(m < 0 \to m \in X)$. This statement is vacuously true as $m < 0$ is false for each natural m. Hence, $0 \in X$ must be true as well. □

Example 5.3. Suppose $a + \dfrac{1}{a} \in \mathbb{Z}$ for some number $a \in \mathbb{R}$. Let's show that $a^n + \dfrac{1}{a^n} \in \mathbb{Z}$ for each $n \in \mathbb{N}$.

It suffices to prove the set $X = \{n \in \mathbb{N} \mid a^n + \dfrac{1}{a^n} \in \mathbb{Z}\}$ to be progressive. Consider an arbitrary $n \in \mathbb{N}$. Assume $m \in X$ for every $m < n$. If $n \geq 2$, then the naturals $n-1$ and $n-2$, which are lesser than n, belong to X. Hence,

$$a^n + \frac{1}{a^n} = \left(a + \frac{1}{a}\right)\left(a^{n-1} + \frac{1}{a^{n-1}}\right) - \left(a^{n-2} + \frac{1}{a^{n-2}}\right).$$

The product and difference of any integers are integer, so $n \in X$. The same is apparent when $n \leq 1$. Thus, we get $\mathit{Prog}(X)$. By the Strong Induction Principle, $X = \mathbb{N}$.

Furthermore, one easily sees that $a^x + \dfrac{1}{a^x} \in \mathbb{Z}$ for all $x \in \mathbb{Z}$ since $a^x + \dfrac{1}{a^x} = a^{|x|} + \dfrac{1}{a^{|x|}}$ and $|x| \in \mathbb{N}$ when $x \in \mathbb{Z}$.

The last form of induction for us to consider is the *Least Number Principle*, which states:

For an arbitrary set $X \subseteq \mathbb{N}$, if $X \neq \varnothing$, then there exists a least element $\min X$ of X.

If put in symbols (for a certain X fixed), the Principle reads:

$$\exists m\ m \in X \to \exists n\,\big(n \in X \land \forall m\,(m < n \to m \notin X)\big).$$

> As we have stated it, this is a '*minimal* element principle' despite its traditional name. Although, for the linear ordering of $\mathbb{N}$, the choice between 'minimal' and 'least' does not logically matter, just the 'minimal' form is equivalent to 'strong' (or transfinite) induction for an arbitrary poset. Therefore we prefer to keep the traditional name for this general form of the principle. On the other hand, students may have questions on this "misleading" name when they have learned about minima and maxima in posets. The notation $\min X$ will be redefined later as well.

Example 5.4. Let's find all the integer solutions to the equation $8a^4 + 4b^4 + 2c^4 = d^4$.

Clearly the tuple $(0,0,0,0)$ is a (integer) solution. Let us show this to be unique. If (a,b,c,d) is a solution, then $(|a|,|b|,|c|,|d|) \in \mathbb{N}^4$ is a solution as well. Thus, it suffices to prove that there is no non-zero solution from $\mathbb{N}$. Assume the contrary. Then, the set

$$Y = \{(a,b,c,d) \in \mathbb{N}^4 \mid 8a^4 + 4b^4 + 2c^4 = d^4 \land a + b + c + d > 0\}$$

is not empty. Hence, the set $X = \{a + b + c + d \mid (a,b,c,d) \in Y\} \subseteq \mathbb{N}_+$ is non-empty as well. By the Least Number Principle, there exists a certain $x = \min X > 0$, such that $x = a + b + c + d$ for some solution (a, b, c, d). Since $8a^4 + 4b^4 + 2c^4 = d^4$, the number d^4 is even. It is known from arithmetic, that d must be even too, i. e. $d = 2\delta$ for some number $\delta \in \mathbb{N}$.

Thus obtain $8a^4 + 4b^4 + 2c^4 = 16\delta^4$, whence $c^4 = 8\delta^4 - 4a^4 - 2b^4$. It is easy to see that $c = 2\gamma$ and, by a similar argument, $a = 2\alpha$, $b = 2\beta$ for certain $\alpha, \beta, \gamma \in \mathbb{N}$.

Replacing a with 2α, etc., and dividing both sides of the original equation by 16, we get $8\alpha^4 + 4\beta^4 + 2\gamma^4 = \delta^4$. So, the tuple $(\alpha, \beta, \gamma, \delta) \in \mathbb{N}^4$ is another solution. Yet $2(\alpha+\beta+\gamma+\delta) = a+b+c+d = x > 0$; hence, $\alpha + \beta + \gamma + \delta > 0$, and we have got a new *non-zero* solution. Then $(\alpha, \beta, \gamma, \delta) \in Y$ and $\alpha + \beta + \gamma + \delta \in X$, while $\alpha + \beta + \gamma + \delta < x = \min X$. A contradiction.

In fact, all the three forms of induction introduced hereunto are logically equivalent to each other. So, each may be looked upon as *the* principle of induction.

Theorem 5.5. *The following statements are equivalent:*

1. *the Strong Induction Principle;*

2. *the Least Number Principle;*

3. *the Induction Principle.*

> We use straightforward but abstract logical manipulations in order to prove this theorem. They pose a serious challenge for many students. Nevertheless, we prefer to make the students “eat the frog” now—just spending more time on this proof. In my experience, the ‘predicate’ version of this proof is no better, whereas the ‘predicate’ forms of the induction principle might be so for some audiences.

Proof. Suppose the Strong Induction Principle holds. We are to prove that each non-empty set $X \subseteq \mathbb{N}$ has a least element. Assume that some X lacks a least element. Let us see that the set $\bar{X}$ must be progressive then. Indeed, if $\forall m < n\ m \notin X$, then $n \notin X$, since n would be $\min X$ otherwise, which is impossible. By the Strong Induction Principle, $\bar{X} = \mathbb{N}$, whence $X = \varnothing$.

Suppose the Least Number Principle holds. We need to infer $X = \mathbb{N}$ from the assumptions $0 \in X$ and $\forall n\, (n \in X \to n + 1 \in X)$ for an arbitrary set $X \subseteq \mathbb{N}$. Consider the set $\bar{X}$ and suppose that $\bar{X} \neq \varnothing$. Then there exists a certain number $n = \min \bar{X}$. By the assumptions, $n \neq 0 \notin \bar{X}$. Hence, $n = m + 1$ for some $m \in \mathbb{N}$. As $m < n$, it is the case that $m \in X$. By the assumptions, $n = m + 1 \in X$, which is not so. Therefore, such n cannot exist and $\bar{X}$ must be empty. So, $X = \mathbb{N}$.

Suppose the Induction Principle holds. Let us prove that for every set $X \subseteq \mathbb{N}$, from the assumption $\mathit{Prog}(X)$, it follows that $X = \mathbb{N}$. Consider the set

$$Y = \{n \in \mathbb{N} \mid \forall m < n\ m \in X\}.$$

Clearly, $0 \in Y$. Suppose $n \in Y$. By definition, we have $\forall m < n\ m \in X$ then, which yields $n \in X$ for X being progressive. If $m < n + 1$, then either $m < n$ or $m = n$. In both cases, we get $m \in X$, whence $n+1 \in Y$. For Y, we have verified both the base and inductive step; by the Induction Principle, conclude $Y = \mathbb{N}$. For every $n \in \mathbb{N}$, it holds that $n < n + 1 \in Y$, whence $n \in X$. Consequently, $X = \mathbb{N}$. □

6 Divisibility

> This section opens a series of classical results on integer arithmetic. We looked upon them as a playground for the students to try induction principles in solving concrete problems and, of course, to have some rest before coming back to more abstract concepts.

It is time to apply our knowledge of induction to natural and integer arithmetic. As earlier, we will assume some facts without a proof, like the *commutativity law* for addition: $n + m = m + n$ for every $m, n \in \mathbb{Z}$. Such facts can be easily derived by induction from the so called *recursive definitions*[8] for arithmetical operations. But this topic is a bit bigger than the scope of our Course can contain.

Divisibility. From now on, the term 'number' will denote an integer number by default. We say that a number a *divides* a number b if there exists some $k \in \mathbb{Z}$ such that $b = ak$. Conversely, b is said to be *divisible* by a in this case. Also, we call a a *divisor* of b and call b a *multiple* of a. We write $a \,|\, b$ when a divides b.

> Most students feel quite uncomfortable with the fact that 0 divides 0. They tend to ignore the *definition* in favor of *connotations*: "as it is not possible to *divide by* zero, zero cannot *divide* anything". Of course, it is a general problem of mathematical education that the students just do not *read* what is written. The Instructor should use such examples to demonstrate the importance of clear and, perhaps, *slow* reading in mathematics (likewise the latter is important in philology, according to Nietzsche's famous maxim).

Example 6.1. So, $2 \,|\, 6$ as $6 = 2 \cdot 3$; $2 \,|\, 2$ as $2 = 2 \cdot 1$; $-2 \,|\, 6$ as $6 = (-2) \cdot (-3)$; $2 \,|\, 0$ as $0 = 2 \cdot 0$; and, finally, $0 \,|\, 0$ as $0 = 0 \cdot 2019$. In general, $a \,|\, 0$ and $1 \,|\, a$ for each a as $0 = a \cdot 0$ and $a = 1 \cdot a$.

Lemma 6.2. *For every a, b, c, the following hold:*

1. *$a \,|\, a$;*

2. *if $a \,|\, b$ and $b \,|\, c$, then $a \,|\, c$;*

3. *if $a \,|\, b$ and $b \,|\, a$, then $a = \pm b$.*

Proof. The fact that $a = a \cdot 1$ gives us the first statement. For the second one, assume $a \,|\, b$ and $b \,|\, c$, that is, $b = ak_1$ or $c = bk_2$ for some numbers k_1, k_2. Hence we get $c = bk_2 = (ak_1)k_2 = a(k_1k_2)$, which implies $a \,|\, c$. For the final one, assume both $a \,|\, b$ and $b \,|\, a$. Then $a = bk_1$ and $b = ak_2$ for some k_1, k_2. This yields $a = a(k_1k_2)$. Now, let us see if $a = 0$. If it is so, then $b = 0 \cdot k_2 = 0 = a$.

Otherwise, one can cancel a out to obtain $1 = k_1k_2$. We take it as a fact that 1 can be factorized either as $1 \cdot 1$ or as $(-1) \cdot (-1)$. Thus, either $a = b$ or $a = -b$. □

Lemma 6.3. *If $a \,|\, b$ and $a \,|\, c$, then $a \,|(b + c)$ and $a \,|(b - c)$.*

Proof. Assuming $b = ak_1$ and $c = ak_2$, we get $b \pm c = ak_1 + ak_2 = a(k_1 + k_2)$ by applying the distributivity law (taken without a proof). □

Corollary 6.4. *If $a \,|(b + c)$ or $a \,|(b - c)$ and $a \,|\, b$, then $a \,|\, c$.*

Proof. Clearly, $c = (b + c) - b$ and $c = b - (b - c)$. Then apply Lemma 6.3. □

[8] Given the *successor* function S as a primary object ($S0 = 1$, $S1 = 2$, etc.), consider the identities $0 + m = m$ and $Sn + m = S(n + m)$. One can prove that there exists a unique operation $+$ satisfying these identities for every $n, m \in \mathbb{N}$.

Now, let us review the familiar procedure of integer division.

Theorem 6.5. *For every natural numbers a and $b \neq 0$, there exists a unique pair $(q, r) \in \mathbb{N}^2$ such that $a = bq + r$ and $0 \leq r < b$.*

Proof. Consider the set $X = \{s \in \mathbb{N} \mid a < bs\}$. We take as a fact, that multiplying by a positive number b is *monotonic*, i. e., $bx < by$ when $x < y$. Hence, $b(a+1) > ba \geq 1 \cdot a = a$ and $a + 1 \in X$. So, X is non-empty. By the Least Number Principle, there exists some $s' = \min X$. If $s' = 0$, then $a < bs' = 0$, which is impossible for a natural a. Otherwise, $s' = q + 1$ for some $q \in \mathbb{N}$, where $bq \leq a$. Hence, $0 \leq a - bq$. If $a - bq \geq b$, there would be $a - bs' = a - b(q+1) \geq 0$, that is, $a \geq bs'$, which is not the case. Therefore, $0 \leq a - bq < b$ and one can safely put $r = a - bq$. The existence of a required pair is thus proved.

Suppose there are two such pairs (q, r) and (q', r'). We have both $a = bq + r$ and $a = bq' + r'$, while $0 \leq r, r' < b$. If $q = q'$, then, clearly, $r = r'$ as well. Otherwise, w. l. o. g.,[9] we assume that $q < q'$, that is, $q + 1 \leq q'$. Hence, $r - r' = bq' - bq = b(q' - q) \geq b$. Then $r \geq b + r' \geq b$, which is not so. □

Such a number q is called the *partial quotient* and r is called the *remainder* after division of a by b. This result can be easily generalized to arbitrary integers:

Corollary 6.6. *For every numbers a and $b \neq 0$, there exists a unique pair $(q, r) \in \mathbb{Z} \times \mathbb{N}$ such that $a = bq + r$ and $0 \leq r < |b|$.*

Proof. By Theorem 6.5, there exists a pair of naturals (q', r') such that $|a| = |b|q' + r'$, while $0 \leq r' < |b|$. It is clear that $c = |c| \cdot \operatorname{sgn} c$ for any integer c. So, $a = \operatorname{sgn} a(\frac{b}{\operatorname{sgn} b}q' + r') = \frac{\operatorname{sgn} a}{\operatorname{sgn} b}q'b + r' \operatorname{sgn} a$. When $a \geq 0$ or $r' = 0$, it suffices to put $(q, r) = (\frac{\operatorname{sgn} a}{\operatorname{sgn} b}q', r')$. Suppose that $a < 0$ and $r' > 0$. Then, we have

$$a = -\frac{q'}{\operatorname{sgn} b}b - r' = -\frac{q'}{\operatorname{sgn} b}b - \frac{b}{\operatorname{sgn} b} + \frac{b}{\operatorname{sgn} b} - r' = -\frac{1+q'}{\operatorname{sgn} b}b + (|b| - r'),$$

where $0 < |b| - r' < |b|$. Now, put $(q, r) = (-\frac{1+q'}{\operatorname{sgn} b}, |b| - r')$.

Let's prove the uniqueness. Suppose that $bq + r = bq' + r'$, while $0 \leq r, r' < |b|$. Then $|r - r'| = |bq' - bq| = |b(q' - q)| = |b| \cdot |q' - q|$. If $q \neq q'$, then $|r - r'| \geq |b|$. W. l. o. g., $r > r'$, whence $r \geq |b| + r' \geq |b|$, which is not the case. □

Example 6.7. One has $-5 = -2 \cdot 3 + 1$ and $-5 = 2 \cdot (-3) + 1$. Please notice that the remainder is never negative.

Modular arithmetic. Let m be a positive number. We say that a is *congruent* to b *modulo* m if $m \,|\, (a - b)$. We write $a \equiv b \pmod{m}$ and call the number m a *modulus* in such a case.

Example 6.8. We have $23 \equiv -31 \pmod{9}$ since $23 - (-31) = 54$ and $9 \,|\, 54$. It holds that $x \equiv y \pmod{2}$ iff $x - y$ is even, that is, either both x and y are even or both are odd (they say that x and y have the same *parity* in this case).

Remark 6.9. As each number is divisible by 1, $x \equiv y \pmod{1}$ for all numbers x and y. This makes congruence modulo 1 a trivial, uninteresting property. Therefore, we will usually suppose $m > 1$ in $x \equiv y \pmod{m}$.

While it is possible to consider congruence modulo 0, where $x \equiv y \pmod{0}$ means $0 \,|\, (x - y)$, that is, $x - y = 0$, this predicate is no more interesting: every number is only congruent to itself modulo 0. Congruence for negative moduli is not needed either since $m \,|\, (x - y)$ is equivalent to $-m \,|\, (x - y)$.

[9] *Without loss of generality*—this means that we are going to consider just one of all possible cases but are sure that all the rest can be treated similarly.

In practice, we also used simplified notation $a \equiv b\ (m)$ as well. Many students are already familiar with this concept but prefer to define it in terms of remainders. The Instructor should underline that each of these two equivalent definitions may be preferable to the other one in various situations.

Lemma 6.10. *For any numbers a, b, m, it holds that $a \equiv b \pmod m$ iff a and b leave the same remainder when divided by m.*

Proof. Suppose that $m \,|(a - b)$. By Corollary 6.6, one have $a = mq + r$ and $b = mq' + r'$, where $0 \le r, r' < m$. So, $m \,|(m(q - q') + (r - r'))$. By Corollary 6.4, this implies $m \,|(r - r')$. We are done if $r = r'$. Otherwise, w.l.o.g., assume that $r > r'$. Then $r = mk + r'$ for some $k > 0$, whence $r \ge mk \ge m$, which is not so.

For the other direction, suppose that a and b give the same remainder m, so $a = mq + r$ and $b = mq' + r$ for some q and r. Then $a - b = m(q - q')$, which is clearly a multiple of m. □

Corollary 6.11. *No two of the numbers $0, 1, \ldots, m - 1$ are congruent modulo m.*

Corollary 6.12. *Suppose r is the remainder after dividing a by m. Then $a \equiv r \pmod m$.*

Corollary 6.13. *For any numbers a, b, m, the following hold:*

1. *$a \equiv a \pmod m$;*
2. *if $a \equiv b \pmod m$, then $b \equiv a \pmod m$;*
3. *if $a \equiv b \pmod m$ and $b \equiv c \pmod m$, then $a \equiv c \pmod m$.*

The most interesting and most important point about congruence is that it ‘respects’ the arithmetical operations.

Lemma 6.14. *Suppose that $a \equiv b \pmod m$ and $c \equiv d \pmod m$. Then the following hold:*

1. *$a + c \equiv b + d \pmod m$;*
2. *$ac \equiv bd \pmod m$;*
3. *$a^n \equiv b^n \pmod m$ for every $n \in \mathbb{N}$.*

Proof. For the first statement, let us see that $a + c - (b + d) = (a - b) + (c - d)$. As both these summands are multiples of m, the sum is a multiple of m too by Lemma 6.3.

Consider the second one. We know that $a = b + mk_1$ and $c = d + mk_2$ for some numbers k_1, k_2. So, $ac = bd + m(bk_2 + dk_1 + mk_1k_2)$. Hence $m \,|(ac - bd)$.

The third is obtained from the second one by a trivial induction on n. □

Example 6.15. This fact simplifies computations modulo m a lot. Indeed, one may freely replace any summand or factor by any other one he sees convenient given the two are congruent modulo m. Often, one prefers positive or negative numbers small in their absolute value for this procedure (say, one can replace 99 by -1 in *any* computation modulo 100 that involves just addition and multiplication).

Let us compute the remainder after dividing 71^{59} by 97. As $71 \equiv -26 \pmod{97}$, we get

$$x = 71^{59} \equiv -2^{59} \cdot 13^{59} \equiv -2^{59} \cdot (-4 \cdot 21)^{59} \equiv 2^{177} \cdot 3^{59} \cdot 7^{59} \equiv 2^{177} \cdot (3^4)^{14} \cdot 3^3 \cdot (7^2)^{29} \cdot 7 \pmod{97}.$$

Furthermore, $7^2 \equiv -48 \equiv -7^2+1$, whence $2 \cdot 7^2 \equiv 1 \pmod{97}$; also, $3^4 \equiv -2^4$, and $27 \cdot 7 \equiv 3 \cdot 63 \equiv 3 \cdot (-34) \equiv -102 \equiv -5$. Then,

$$\begin{gathered}
x \equiv (2 \cdot 7^2)^{29} \cdot 2^{148} \cdot 2^{56} \cdot (-5) \equiv 1^{29} \cdot (2^6)^{34} \cdot (-5) \equiv -5 \cdot (-33)^{34} \equiv -5 \cdot 3^{34} \cdot 11^{34} \equiv \\
-45 \cdot (3^4)^8 \cdot (11^2)^{17} \equiv -45 \cdot 2^{32} \cdot (3 \cdot 2^3)^{17} \equiv -135 \cdot 2^{83} \cdot (3^4)^4 \equiv -38 \cdot 2^{99} \equiv -19 \cdot (2^6)^{16} \cdot 2^4 \equiv \\
-19 \cdot 3^{16} \cdot (11^2)^8 \cdot 2^4 \equiv -19 \cdot 3^{24} \cdot 2^{24} \cdot 2^4 \equiv -19 \cdot 2^{24} \cdot 2^{24} \cdot 2^4 \equiv -19 \cdot (2^6)^8 \cdot 2^4 \equiv -19 \cdot 3^8 \cdot (11^2)^4 \cdot 2^4 \equiv \\
-19 \cdot 3^{12} \cdot 2^{12} \cdot 2^4 \equiv -19 \cdot (-2^4)^3 \cdot 2^{12} \cdot 2^4 \equiv 19 \cdot 2^{24} \cdot 2^4 \equiv 19 \cdot (2^6)^4 \cdot 2^4 \equiv 19 \cdot 3^4 \cdot (11^2)^2 \cdot 2^4 \equiv \\
-19 \cdot 2^4 \cdot (24)^2 \cdot 2^4 \equiv -19 \cdot 2^2 \cdot 3^2 \cdot (2^6)^2 \equiv -19 \cdot 2^2 \cdot 3^4 \cdot (11)^2 \equiv 19 \cdot 2^2 \cdot 2^4 \cdot 3 \cdot 2^3 \equiv 19 \cdot 3 \cdot 2^9 \equiv -40 \cdot 2^9 \equiv \\
-5 \cdot 2^{12} \equiv -5 \cdot (2^6)^2 \equiv -5 \cdot 3^2 \cdot 11^2 \equiv -5 \cdot 3^3 \cdot 2^3 \equiv -10 \cdot 108 \equiv -10 \cdot 11 \equiv -13 \equiv 84 \pmod{97}.
\end{gathered}$$

So, the remainder in question is 84. This computation was far from optimal, as we ignored many noticeable facts (like $2^{48} \equiv 1 \pmod{97}$ or $2^{96} \equiv 1 \pmod{97}$). We shall soon see how it is possible to easily prove these. Nevertheless, this tedious computation was very easy in the respect that we had no need to manipulate any number greater than 204 in absolute value.

Prime numbers and factorization. A number $p > 1$ is called *prime* if it is divisible by ± 1 and by $\pm p$ but nothing else. Otherwise, a number greater than 1 is called *composite.*

Example 6.16. The number 101 is prime, while $91 = 7 \cdot 13$ is composite. The negatives, 0, and 1 are neither prime nor composite.

A representation of the form $n = p_1^{a_1} p_2^{a_2} \dots p_s^{a_s}$ for a number $n > 1$, where p_i are pairwise distinct primes and $a_i \in \mathbb{N}$, is called a *(prime) factorization* of n. One may naturally suppose that *every* prime number p takes part in a factorization of n but just finitely many of them have non-zero degrees a. It is possible to factorize 1 as well, but each prime shall have degree 0 then. Two factorization are *distinct* iff some prime number has different degrees in these.

> It would be very natural to define a factorization as a function from primes to naturals with a finite support. Unfortunately, we have not enough set theory up to this point to do so. However, the students are usually happy with these ugly notions of 'distinct' factorizations etc. for those are intuitive enough.

Example 6.17. The prime 11 has degree 0 in the factorization $3 \cdot 5 \cdot 7$ of the number 105. One may say 11 is *absent* from that factorization. The factorizations $2 \cdot 3^2 \cdot 5^3$ and $3^3 \cdot 5^2 \cdot 7$ are distinct.

Theorem 6.18 (Fundamental Theorem of Arithmetic)**.** *For each number $n > 1$, there exists a unique prime factorization.*

Proof. Firstly, we check existence of a factorization. Assume the contrary and consider the least number $n > 1$ lacking a factorization (such a number must exist by the Least Number Principle). If n is prime, there clearly is a trivial factorization. Hence, n is composite and has a non-trivial divisor m, that is, $n = mk$, where $1 < m, k < n$ (as multiplication is monotonic). Then $m = p_1^{a_1} p_2^{a_2} \dots p_s^{a_s}$ and $k = p_1^{b_1} p_2^{b_2} \dots p_s^{b_s}$ for some primes $p_1, p_2, \dots, p_s$ (if p_i is absent from the factorization of, say, m, just put $a_i = 0$). We can thus factorize n as $p_1^{a_1+b_1} p_2^{a_2+b_2} \dots p_s^{a_s+b_s}$. A contradiction.

Secondly, we should prove that no two factorizations of $n > 1$ are distinct. Assume the contrary and consider the least number $n > 1$ with two distinct factorizations: $n = p_1^{a_1} p_2^{a_2} \dots p_s^{a_s} = p_1^{b_1} p_2^{b_2} \dots p_s^{b_s}$. We have $\min(a_i, b_i) = 0$ for each i as otherwise one can cancel $p_i^{\min(a_i,b_i)}$ out thus decreasing n but keeping the factorizations distinct. Hence, in each factorization, there should be some prime absent from the other (for otherwise at least one of these equals 1). W. l. o. g., let it be that $a_1 > 0 = b_1$, $a_2 = 0 < b_2$, and $p_1 > p_2$. Consider the number

$$m = (p_1 - p_2)p_1^{a_1-1}p_2^{a_2} \dots p_s^{a_s} = n - p_2 p_1^{a_1-1}p_2^{a_2} \dots p_s^{a_s} = p_2(p_1^{b_1}p_2^{b_2-1} \dots p_s^{b_s} - p_1^{a_1-1}p_2^{a_2} \dots p_s^{a_s}) > 0.$$

Clearly, $m < n$, so m has a unique factorization, which is provided by multiplying some factorization of the number $p_1^{b_1}p_2^{b_2-1} \dots p_s^{b_s} - p_1^{a_1-1}p_2^{a_2} \dots p_s^{a_s}$ by p_2. As $a_2 = 0$, we have $m = (p_1 - p_2)p_1^{a_1-1}p_3^{a_3} \dots p_s^{a_s}$, so another factorization of m is provided by multiplying $p_1^{a_1-1}p_3^{a_3} \dots p_s^{a_s}$ by some factorization of $p_1 - p_2$. Since these two factorizations of m cannot be distinct, p_2 must be present in both. So, $p_2 \,|\, (p_2 - p_1)$, whence $p_2 \,|\, p_1$. This is impossible for p_1 and p_2 are distinct primes. A contradiction. □

As we have already said, 1 has a trivial factorization from which any prime is absent. Zero has infinitely many factorizations, but each containing zero itself: $0 = 673 \cdot 2 \cdot 3 \cdot 23 \cdot 0 \cdot 1$ etc. To factor a negative integer n, it is enough to factor its absolute value $|n|$ and multiply the result by -1. So, every number $a \neq 0$ can be identified with its prime factorization.

Theorem 6.19. *There are infinitely many prime numbers.*

Proof. Assume it is not so. Clearly, there exists at least one prime. Then some $p_1, p_2, \dots, p_n$ are the only primes in existence. Consider the number $N = p_1 p_2 \dots p_n + 1$. Since multiplication is monotonic, $N > p_i \geq 2$ for each i. Hence, N is not prime. Because of Theorem 6.18, N should be a multiple of some prime p_j. By Corollary 6.4, get the impossible conclusion $p_j \,|\, 1$. Our assumption thus fails. □

Lemma 6.20 (Divisibility Criterion)**.** *For any non-zero numbers a and b, a divides b iff $\alpha_i \leq \beta_i$ for each prime p_i, where α_i (or β_i) is the degree of p_i in the factorization of a (respectively, b).*

Proof. Let us factor both a and b to obtain $a = \varepsilon p_1^{\alpha_1} \dots p_s^{\alpha_s}$ and $b = \delta p_1^{\beta_1} \dots p_s^{\beta_s}$, where $\varepsilon, \delta = \pm 1$.

Suppose we have $b = ak$. By factoring k, get $k = \pm p_1^{\gamma_1} \dots p_s^{\gamma_s}$. As the factorization of b is unique, there should be $\beta_i = \alpha_i + \gamma_i \geq \alpha_i$.

For the other direction, one may put $\gamma_i = \beta_i - \alpha_i \geq 0$ to see that $b = a \cdot \frac{\delta}{\varepsilon} p_1^{\gamma_1} \dots p_s^{\gamma_s}$. □

7 A Few Classical Theorems

We say that a number $d \in \mathbb{N}$ is a *greatest common divisor* of numbers a and b if (1) $d \mid a$ and $d \mid b$, and (2) for each $d' \in \mathbb{N}$, from $d' \mid a$ and $d' \mid b$, it follows that $d' \mid d$. We write $\gcd(a,b) = d$ in such a case. In other words, d is such a common divisor of a and b that is a multiple of any (other) common divisor d'. A priori, it is not obvious why such a number d exists.

Lemma 7.1. *For every a and b, there exists a unique number d such that* $d = \gcd(a,b)$.

Proof. Firstly, we check uniqueness. If there are two such numbers d and d', we have both $d \mid d'$ and $d' \mid d$, whence $d = d'$ by Lemma 6.2.

Secondly, let us demonstrate existence. If $a = 0$, put $d = |b|$. Clearly, $|b| \mid 0$ and $|b| \mid b$. Whenever $d' \mid b$, we get $d' \mid d$. The case when $b = 0$ is similar.

Suppose that neither of a and b is zero. Then factor these to get $a = \varepsilon p_1^{\alpha_1} \dots p_s^{\alpha_s}$ and $b = \delta p_1^{\beta_1} \dots p_s^{\beta_s}$. Consider $d = p_1^{\min(\alpha_1,\beta_1)} \dots p_s^{\min(\alpha_s,\beta_s)}$. By Lemma 6.20, obtain $d \mid a$ and $d \mid b$. Assume both $d' \mid a$ and $d' \mid b$ for some $d' \in \mathbb{N}$. Clearly, $d' \neq 0$, so one factors d' as $p_1^{\gamma_1} \dots p_s^{\gamma_s}$. Applying Lemma 6.20, we have both $\gamma_i \leq \alpha_i$ and $\gamma_i \leq \beta_i$, whence $\gamma_i \leq \min(\alpha_i, \beta_i)$ for each i. Thus, $d' \mid d$. □

So, it is easy to find the number $\gcd(a,b)$ given some factorizations of these numbers. But it is not that easy to factor a number. We shall see a more practical way to compute $\gcd(a,b)$ soon.

Example 7.2. Inspecting the proof of the previous lemma, one observes that $\gcd(15,-12) = 3$, $\gcd(-14,-21) = 7$, $\gcd(1,a) = 1$, and $\gcd(0,a) = |a|$ for each a. In particular, $\gcd(0,0) = 0$. Evidently, $\gcd(a,b) = \gcd(|a|,|b|)$.

We say that a number $m \in \mathbb{N}$ is a *least common multiple* of numbers a and b if (1) $a \mid m$ and $b \mid m$ and (2) for each $m' \in \mathbb{N}$, from $a \mid m'$ and $b \mid m'$, it follows that $m \mid m'$. We write $\operatorname{lcm}(a,b) = m$ in such a case. In other words, m is such a common multiple of a and b that divides any (other) common multiple m'.

Exercise 7.3. For each a and b, their least common multiple truly exists and is unique, while $\operatorname{lcm}(0,a) = 0$ and $\operatorname{lcm}(a,b) = p_1^{\max(\alpha_1,\beta_1)} \dots p_s^{\max(\alpha_s,\beta_s)}$ if $a = \varepsilon p_1^{\alpha_1} \dots p_s^{\alpha_s}$ and $b = \delta p_1^{\beta_1} \dots p_s^{\beta_s}$.

> There is a clear similarity between gcd and minimum (respectively, lcm and maximum)—since both are infima for suitable posets (we will elaborate on the idea later). The Instructor might want to highlight this fact.

Exercise 7.4. For every a, b, c the following hold:

1. $\gcd(a,a) = |a| = \operatorname{lcm}(a,a)$;
2. $\gcd(a,b) = \gcd(b,a)$; $\operatorname{lcm}(a,b) = \operatorname{lcm}(b,a)$;
3. $\gcd(a,\gcd(b,c)) = \gcd(\gcd(a,b),c)$; $\operatorname{lcm}(a,\operatorname{lcm}(b,c)) = \operatorname{lcm}(\operatorname{lcm}(a,b),c)$;
4. $\gcd(a,\operatorname{lcm}(a,b)) = |a| = \operatorname{lcm}(a,\gcd(a,b))$;
5. $\gcd(a,b) \cdot \operatorname{lcm}(a,b) = |ab|$.

Lemma 7.5. *For any numbers a, b, and q,* $\gcd(a,b) = \gcd(a+bq,b)$.

Proof. Suppose that $d = \gcd(a,b)$ and $d' = \gcd(a+bq,b)$. Clearly, $d \mid (a+bq)$ and $d \mid b$, so $d \mid d'$ by definition. Conversely, $d' \mid b$, $d' \mid bq$, and $d' \mid a$, for $a = (a+bq) - bq$. Hence, $d' \mid d$. Therefore, $d = d'$. □

Corollary 7.6. *If r is the remainder after dividing a by b, then* $\gcd(r,b) = \gcd(a,b)$.

Proof. As $a = bq + r$, get $r = a - bq$, but $\gcd(a-bq,b) = \gcd(a,b)$ by the previous lemma. □

Coprimes. Numbers a and b are called *coprime* (to each other) if $\gcd(a,b) = 1$. That is, the only common natural divisor of a and b is 1.

Example 7.7. Clearly, every two distinct primes are coprime. The number 1 is coprime to any number. The numbers $2^2 \cdot 3$ and $-5 \cdot 7^3$ are coprime unlike $2^2 \cdot 3$ and $3 \cdot 5$. The number 0 is coprime to just ± 1.

Lemma 7.8. *For every numbers a and b, the following statements are equivalent:*

1. *a and b are coprime;*

2. *for every $d \in \mathbb{N}$, if $d \,|\, a$ and $d \,|\, b$, then $d = 1$;*

3. *for every n, if $n \,|\, a$ and $n \,|\, b$, then n is not prime;*

and, when a, b are non-zero, also

4. $\min(\alpha, \beta) = 0$ *for each prime p, where α and β are the degrees of p in the factorizations of a and b, respectively.*

Corollary 7.9. *If both a and b are non-zero, then the numbers $a' = \frac{a}{\gcd(a,b)}$ and $b' = \frac{b}{\gcd(a,b)}$ are integer and coprime.*

Proof. Let $a = \varepsilon p_1^{\alpha_1} \dots p_s^{\alpha_s}$ and $b = \delta p_1^{\beta_1} \dots p_s^{\beta_s}$. Then $a' = \varepsilon p_1^{\alpha'_1} \dots p_s^{\alpha'_s}$ and $b' = \delta p_1^{\beta'_1} \dots p_s^{\beta'_s}$, where $\alpha'_i = \alpha_i - \min(\alpha_i, \beta_i)$ and $\beta'_i = \beta_i - \min(\alpha_i, \beta_i)$ for each i. As $\alpha_i, \beta_i \geq \min(\alpha_i, \beta_i)$, both the numbers a', b' are integer. Since $\min(\alpha_i, \beta_i)$ is either α_i or β_i, then $\alpha'_i = 0$ or $\beta'_i = 0$, whence $\min(\alpha'_i, \beta'_i) = 0$. By Lemma 7.8, a' and b' are coprime. □

Lemma 7.10. *Suppose that $a \,|\, bc$, while a and b are coprime. Then $a \,|\, c$.*

Proof. If $c = 0$ or $a = \pm 1$, the conclusion is immediate. Assume $c \neq 0$ and $a \neq \pm 1$. Then $\gcd(a, 0) = |a| \neq 1 = \gcd(a, b)$, whence $b \neq 0$. Since $bc \neq 0$, a is non-zero as well. Let us factor our numbers: $a = \varepsilon p_1^{\alpha_1} \dots p_s^{\alpha_s}$, $b = \delta p_1^{\beta_1} \dots p_s^{\beta_s}$, and $c = \eta p_1^{\gamma_1} \dots p_s^{\gamma_s}$. From Lemmas 6.20 and 7.8, we know that $\alpha_i \leq \beta_i + \gamma_i$ and $\min(\alpha_i, \beta_i) = 0$ for each i. If $\alpha_i = 0$, we have $\alpha_i \leq \gamma_i$; otherwise, $\beta_i = 0$ and $\alpha_i \leq 0 + \gamma_i = \gamma_i$ as well. Therefore, $a \,|\, c$. □

This means that one can sometimes cancel numbers out from congruences.

Corollary 7.11. *If $ax \equiv ay \pmod{m}$ and $\gcd(a, m) = 1$, then $x \equiv y \pmod{m}$.*

Proof. Indeed, from $m \,|\, a(x - y)$, it follows $m \,|(x - y)$. □

Exercise 7.12. If a and b are coprime, then $\gcd(ac, b) = \gcd(c, b)$.

Exercise 7.13. If a and b are coprime, then $\gcd(ab, c) = \gcd(a, c) \cdot \gcd(b, c)$.

The following fact of fundamental import establishes a link between divisibility and addition.

Theorem 7.14 (Bézout's Identity)**.** *Let numbers a and b be coprime. Then there exist numbers u and v such that $au + bv = 1$.*

Proof. Consider the set $X = \{d \in \mathbb{N}_+ \mid d = as + bt \text{ for some } s, t \in \mathbb{Z}\}$. Clearly, at least one of a and b is non-zero. Let it be a. Then $d = a \operatorname{sgn} a + b \cdot 0 = |a| > 0$, so X is non-empty. By the Least Number Principle, the exists some $d = \min X$ and numbers s, t so that $d = as + bt > 0$.

Let us divide a by d to obtain q, r such that $a = dq + r$ and $0 \le r < d$. We get

$$r = a - dq = a - (as + bt)q = a(1 - sq) + b(-tq).$$

Therefore, either $r \in X$ or $r = 0$. The former case is not possible since $r < d = \min X$. Hence, $r = 0$ and $d \,|\, a$. One can prove that $d \,|\, b$ in a similar way.

So, $d > 0$ is a common divisor of coprime numbers. Hence, $as + bt = d = 1$. □

Corollary 7.15. *For every numbers a and b, there exist numbers u and v such that* $au + bv = \gcd(a, b)$.

Proof. This is trivial when either number is zero. Otherwise, the numbers $a' = \frac{a}{\gcd(a,b)}$ and $b' = \frac{b}{\gcd(a,b)}$ are coprime integers. Applying Bézout's Identity, obtain some u and v such that $a'u + b'v = 1$. Then multiply both sides by $\gcd(a, b)$ to get $au + bv = \gcd(a, b)$. □

Unlike rationals or reals, one cannot turn an arbitrary integer $b \neq 0$ into 1 by multiplying it by some integer c. Hence, it is not generally possible to construct an *integer* $\frac{a}{b}$ such that $\frac{a}{b}b = a$. But this may be possible modulo m.

Corollary 7.16. *Let a and $m > 1$ be some numbers. There exists x such that $ax \equiv 1 \pmod{m}$ iff a and m are coprime.*

Proof. If a and m are coprime, we obtain $1 = ax + my$ for some x, y by applying Theorem 7.14. Then $m \,|(ax - 1)$.

Conversely, if $m \,|(ax - 1)$, then $ax + my = 1$ for some y. For any $d \in \mathbb{N}$, from $d \,|\, a$ and $d \,|\, m$, it clearly follows that $d \,|\, 1$, that is, $d = 1$; hence, a and m are coprime. □

The number x is then called a *(multiplicative) inverse* of a modulo m.

Example 7.17. As 15 and 14 are coprime, there should be some x such that $14x \equiv 1 \pmod{15}$. Using Lemma 6.14, one sees that $14x \equiv -x \pmod{15}$. So, it suffices to take -1 (or $15 - 1 = 14$) for x. We shall see a uniform way to find the inverse x in the next section.

Theorems of Euler's and Fermat's. Let m be greater than 1. Consider the numbers $1, 2, \dots, m-1$. Some of these (including 1 and $m - 1$, of course) are coprime with m. Let $q_1, \dots, q_s$ be all such integers. The number s thereof is denoted by $\varphi(m)$. For this number depends on m, one may consider the function φ returning s given m. This function is called *Euler's totient function.*

Example 7.18. One has $\varphi(9) = 6$ for $1, 2, 4, 5, 7, 8$ are the only positive numbers lesser than 9 and coprime to it. Clearly, $\varphi(p) = p - 1$ for every prime p. We shall derive a general formula for φ later.

Theorem 7.19 (Euler's Theorem)**.** *If a and $m > 1$ are coprime, then $a^{\varphi(m)} \equiv 1 \pmod{m}$.*

Proof. Let s be $\varphi(m)$ and consider the numbers $q_1, \dots, q_s$ discussed in the above. By Corollary 6.11, $q_i \not\equiv q_j \pmod{m}$ if $i \neq j$. But then $aq_i \not\equiv aq_j \pmod{m}$, as Corollary 7.11 implies. This means the numbers $aq_1, \dots, aq_s$ have s pairwise distinct remainders when divided by m. Let us denote these remainders by $r_1, \dots, r_s$ respectively. Obviously, $0 \le r_i \le m - 1$ for each i.

Consider an arbitrary natural divisor d of both m and aq_i. It is easy to see that a and d are coprime, as it follows that $d' \,|\, a$ and $d' \,|\, m$ from $d' \,|\, a$ and $d' \,|\, d$. Then $d \,|\, aq_i$ implies $d \,|\, q_i$ by Lemma 7.10. For m and q_i being coprime, we obtain $d = 1$, that is, aq_i and m are coprime as well.

On the other hand, $\gcd(r_i, m) = \gcd(aq_i, m) = 1$ by Corollary 7.6. Thus, $r_1, \dots, r_s$ are pairwise distinct numbers from the set $\{1, \dots, m-1\}$, each being coprime with m. But $q_1, \dots, q_s$ are the only such numbers. This means that $\{r_1, \dots, r_s\} = \{q_1, \dots, q_s\}$ (but not necessarily that $q_i = r_i$ for every i).

Finally, we obtain

$$r_1 \cdot \ldots \cdot r_s \equiv aq_1 \cdot \ldots \cdot aq_s \equiv a^s \cdot q_1 \cdot \ldots \cdot q_s \equiv a^s \cdot r_1 \cdot \ldots \cdot r_s \pmod{m}.$$

Applying Corollary 7.11 to each r_i, we cancel them out to get $1 \equiv a^s \pmod{m}$, which was to be proved. □

Corollary 7.20 (Fermat's Little Theorem)**.** *If p is prime and $p \nmid a$, then $a^{p-1} \equiv 1 \pmod{p}$.*

Proof. For p is prime, $\varphi(p) = p - 1$ and $\gcd(p, a)$ is either 1 or p; but $p \nmid a$, hence $\gcd(p, a) = 1$. □

Remark 7.21. Another equivalent statement of Fermat's Little Theorem is as follows:

> If p is prime, then $a^p \equiv a \pmod{p}$.

These theorems could be handy in modular calculations and spare us a great deal of work.

Corollary 7.22. *If $\gcd(a, m) = 1$, $x, y \geq 0$, and $x \equiv y \pmod{\varphi(m)}$, then $a^x \equiv a^y \pmod{m}$.*

Proof. We have both $x = k\varphi(m) + r$ and $y = l\varphi(m) + r$, where $0 \leq r < m$. Then

$$a^x \equiv a^{k\varphi(m)+r} \equiv (a^{\varphi(m)})^k a^r \equiv 1^k \cdot a^r \equiv a^r \equiv 1^l \cdot a^r \equiv (a^{\varphi(m)})^l a^r \equiv a^{l\varphi(m)+r} \equiv a^y \pmod{m}.$$

□

Example 7.23. What is the last digit of the number 1234567^N in decimal notation, where $N = \underbrace{11\ldots1}_{2019}$?

Clearly, that last digit is just the remainder after dividing 1234567^N by 10. It is easy to see that $\varphi(10) = 4$, but $N \equiv 100K + 11 \equiv 0 + 11 \equiv 3 \pmod{4}$. Therefore, $1234567^N \equiv 7^N \equiv 7^3 \equiv (-3)^2 \cdot 7 \equiv -7 \equiv 3 \pmod{10}$.

8 Linear Equations and Congruences

From Corollary 7.15, we know that for every numbers a and b, there exist some u and v such that $au + bv = \gcd(a, b)$. Let us learn an elegant way to compute all the numbers u, v and $\gcd(a, b)$ simultaneously.

We begin with a particular example. Consider the numbers 26 and 34. Dividing the former by the latter, obtain

$$26 = 34 \cdot 0 + 26.$$

Here, 26 is the remainder. Let us divide the former divisor 34 by the remainder:

$$34 = 26 \cdot 1 + 8.$$

Then, we are to iterate such a procedure: *divide the divisor by the remainder*—obtaining

$$\begin{aligned} 26 &= 8 \cdot 3 + 2, \\ 8 &= 2 \cdot 4 + 0. \end{aligned}$$

No more divisions are possible as the last remainder is zero. The last *non-zero* remainder is 2, while $\gcd(26, 34) = 2$. This is no coincidence.

We need a more general setting to see why it is so. Let a and b be arbitrary integers such that $b \neq 0$. Dividing a by b first and *the divisor by the remainder* hereafter, one gets

$$\begin{array}{lcll} a &=& bq_1 + r_2, & 0 < r_2 < |b|; \\ b &=& r_2q_2 + r_3, & 0 < r_3 < r_2; \\ r_2 &=& r_3q_3 + r_4, & 0 < r_4 < r_3; \\ \ldots & & & \\ r_k &=& r_{k+1}q_{k+1} + r_{k+2}; & 0 \leqslant r_{k+2} < r_{k+1}; \\ \ldots & & & \end{array}$$

Since $0 \leq r_k$ for each k, there must exist a *least* such number r_{s+1} (by the Least Number Principle). If $r_{s+1} > 0$, it is still possible to divide r_s by r_{s+1} to obtain a lesser remainder $r_{s+2} < r_{s+1}$; thus, r_{s+1} would not be the least remainder. Hence, $r_{s+1} = 0$ while $r_s > 0$.

The procedure presented is known as the *Euclidean Algorithm.* We have just proved

Lemma 8.1. *The Euclidean Algorithm terminates for each input (a, b) where $b \neq 0$.*

Lemma 8.2. *Let r_s be the last non-zero remainder resulting from applying the Euclidean Algorithm to numbers a and $b \neq 0$. Then $\gcd(a, b) = r_s$.*

Proof. Applying Corollary 7.6, we obtain consecutively:

$$\gcd(a, b) = \gcd(b, r_2) = \gcd(r_2, r_3) = \gcd(r_3, r_4) = \ldots = \\ \gcd(r_k, r_{k+1}) = \gcd(r_{k+1}, r_{k+2}) = \ldots = \gcd(r_s, r_{s+1}) = \gcd(r_s, 0) = r_s.$$

□

So, we have got a way to calculate $\gcd(a, b)$ without factoring either number. But how can one extract numbers u and v such that $au + bv = \gcd(a, b)$ from this procedure? Let us see!

Given numbers a and b, the Euclidean Algorithm constructs the sequences of numbers r_k and q_k. Denoting a by r_0 and b by r_1, we can rearrange all the equations from the algorithm to the form

$$r_{k+2} = r_k - r_{k+1}q_{k+1}.$$

Let us define two more number sequences: those of u_k and v_k. It is easy to see that the equations

$$\begin{array}{rcrcl} a & = & r_0 & = & au_0 + bv_0, \\ b & = & r_1 & = & au_1 + bv_1. \end{array}$$

have a solution $u_0 = 1$, $v_0 = 0$, $u_1 = 0$, and $v_1 = 1$. Let this be the *definition* for u_0, v_0, u_1, v_1. We want to extend those equations to all values of k so that

$$r_k \quad = \quad au_k + bv_k.$$

With this in view, we define

$$\begin{array}{rcl} u_{k+2} & = & u_k - u_{k+1}q_{k+1}, \\ v_{k+2} & = & v_k - v_{k+1}q_{k+1}, \end{array}$$

so that u_k and v_k mimic the behavior of the sequence of r_k. Let us check the identity $r_k = au_k + bv_k$ by (strong) induction on k. This boils down to inferring $r_{k+2} = au_{k+2} + bv_{k+2}$ from the hypotheses $r_{k+1} = au_{k+1} + bv_{k+1}$ and $r_k = au_k + bv_k$. Indeed,

$$au_{k+2} + bv_{k+2} = a(u_k - u_{k+1}q_{k+1}) + b(v_k - v_{k+1}q_{k+1}) = \\ (au_k + bv_k) - (au_{k+1} + bv_{k+1})q_{k+1} = r_k - r_{k+1}q_{k+1} = r_{k+2}.$$

In particular, we obtain $\gcd(a, b) = r_s = au_s + bv_s$. This is called the *Extended Euclidean Algorithm.*

Let us come back to our numeric example. We know that $q_1 = 0$, $q_2 = 1$, $q_3 = 3$, $q_4 = 4$, and $\gcd(26, 34) = r_4 = 2$. Then,

$$\begin{array}{rcrcl} u_2 & = & u_0 - u_1q_1 & = & 1; \\ v_2 & = & v_0 - v_1q_1 & = & 0; \\ u_3 & = & u_1 - u_2q_2 & = & -1; \\ v_3 & = & v_1 - v_2q_2 & = & 1; \\ u_4 & = & u_2 - u_3q_3 & = & 4; \\ v_4 & = & v_2 - v_3q_3 & = & -3. \end{array}$$

So, there should be $2 = 26 \cdot 4 + 34 \cdot (-3)$, which is surely the case.

Linear equations. The Extended Euclidean Algorithm allows us to solve any equation of the form $ax + by = c$ in integer numbers. This equation is trivial when a or b equals zero. For example, if $a = 0 \neq b$, the equation $bx = c$ has a (unique) solution $\frac{c}{b}$ iff $b \mid c$.

Theorem 8.3. *For any numbers $a \neq 0$, $b \neq 0$, and c, the equation*

$$ax + by = c \tag{1}$$

has a solution iff $d \mid c$, where $d = \gcd(a, b)$. If $d \mid c$, the set of solutions to (1) is

$$\{(x_0 - b't,\ y_0 + a't) \mid t \in \mathbb{Z}\},$$

where $a' = \frac{a}{d}$, $b' = \frac{b}{d}$, $c' = \frac{c}{d}$, and (x_0, y_0) is an arbitrary solution to the equation

$$a'x + b'y = c'. \tag{2}$$

Proof. Clearly, equation (1) has no solution unless $d \,|\, c$. Suppose that $d \,|\, c$; then $d \neq 0$ cancels out from (1), which results in the equivalent equation (2). Hence, it suffices to consider just the latter.

Let (x_0, y_0) be an arbitrary solution (to (2)). Then the pair $(x_0 - b't, y_0 + a't)$ is a solution as well for whatever $t \in \mathbb{Z}$. Clearly,

$$a'(x_0 - b't) + b'(y_0 + a't) = a'x_0 + b'y_0 - a'b't + b'a't = a'x_0 + b'y_0 = c.$$

Moreover, any solution relates this way to (x_0, y_0). Indeed, let (x, y) be a solution. Then

$$a'(x_0 - x) + b'(y_0 - y) = c' - c' = 0,$$

i. e., $a'(x_0 - x) = b'(y - y_0)$. As $\gcd(a', b') = 1$, apply Lemma 7.10 to conclude $b' \,|(x_0 - x)$ and $a' \,|(y - y_0)$, that is, $x = x_0 - b't_1$ and $y = y_0 + a't_2$ for some $t_1, t_2 \in \mathbb{Z}$. Replacing x and y in (2) by the respective right-hand sides, one gets

$$c' = a'(x_0 - b't_1) + b'(y_0 + a't_2) = a'x_0 - a'b't_1 + b'y_0 + b'a't_2 = c' + a'b'(t_2 - t_1),$$

whence $a'b'(t_2 - t_1) = 0$ and $t_1 = t_2$ (for $a', b' \neq 0$).

It remains to show that equation (2) has a solution indeed. By Corollary 7.15, there are some numbers u and v such that

$$a'u + b'v = 1.$$

(In practice, these numbers can be found with the Extended Euclidean Algorithm.) If $x_0 = uc'$ and $y_0 = vc'$, then (x_0, y_0) is clearly a solution to (2). □

Example 8.4. Solve the equation $45x - 37y = 25$ in integer numbers.

As $\gcd(45, -37) = 1$, the equation has a solution (x_0, y_0). Moreover, the solutions are exactly the pairs $(x_0 + 37t, y_0 + 45t)$ for all possible $t \in \mathbb{Z}$. Given u and v with $45u - 37v = 1$, it is enough to put $(x_0, y_0) = (25u, 25v)$.

Let us use the Extended Euclidean Algorithm to obtain these numbers u and v (as well as $\gcd(45, -37)$, indeed). It might be somewhat easier for a human to divide *positive* integers, so we shall seek for u' and v' satisfying $45u' + 37v' = 1$ to change the sign thereafter.

$$\begin{array}{rcl@{\qquad}rcl@{\qquad}rcl}
45 & = & 37 \cdot 1 + 8; & u_2 & = & u_0 - u_1q_1 = 1 - 0 \cdot 1 = 1; & v_2 & = & v_0 - v_1q_1 = 0 - 1 \cdot 1 = -1; \\
37 & = & 8 \cdot 4 + 5; & u_3 & = & u_1 - u_2q_2 = 0 - 1 \cdot 4 = -4; & v_3 & = & v_1 - v_2q_2 = 1 - (-1) \cdot 4 = 5; \\
8 & = & 5 \cdot 1 + 3; & u_4 & = & u_2 - u_3q_3 = 1 - (-4) \cdot 1 = 5; & v_4 & = & v_2 - v_3q_3 = -1 - 5 \cdot 1 = -6; \\
5 & = & 3 \cdot 1 + 2; & u_5 & = & u_3 - u_4q_4 = -4 - 5 \cdot 1 = -9; & v_5 & = & v_3 - v_4q_4 = 5 - (-6) \cdot 1 = 11; \\
3 & = & 2 \cdot 1 + 1; & u_6 & = & u_4 - u_5q_5 = 5 - (-9) \cdot 1 = 14; & v_6 & = & v_4 - v_5q_5 = -6 - 11 \cdot 1 = -17; \\
2 & = & 1 \cdot 2 + 0. & & & & & &
\end{array}$$

Thus $1 = r_6 = 45u_6 + 37v_6 = 45 \cdot 14 + 37 \cdot (-17)$. Changing the sign, we see that $1 = 45u - 37v$ when $u = 14$ and $v = 17$. Hence, $(x_0, y_0) = (350, 425)$ and the set of solutions of the equation in question is just $\{(350 + 37t, 425 + 45t) \mid t \in \mathbb{Z}\}$.

Linear congruences. Given a, c, and m, we say that the expression $ax \equiv c \pmod{m}$ is a *congruence* in the variable x. This congruence can be solved for x. Namely, we say that a number x_0 is a *solution* to the congruence if $ax_0 \equiv b \pmod{m}$ and $0 \leq x_0 < m$. In other words, we are exclusively interested in the remainders modulo m of possible values for x. Why?

Indeed, knowing all the solutions suffices to find *every* number x' such that $ax' \equiv c \pmod{m}$. On the one hand, if x' is such a number and x'' is its remainder after dividing by m, we have $x' \equiv x''$

(mod m) by Corollary 6.12, whence $ax'' \equiv ax' \equiv c \pmod{m}$ by Lemma 6.14, so x'' is a solution. On the other hand, if x is a solution and $x' \equiv x \pmod{m}$, we see that $ax' \equiv ax \equiv c \pmod{m}$. Finally, for every number x', $ax' \equiv c \pmod{m}$ iff there exists a *solution* x to this congruence such that $x' \equiv x$ (mod m).

Theorem 8.5. *For any numbers $a \neq 0$, c, $m > 1$, the congruence*

$$ax \equiv c \pmod{m} \tag{3}$$

has a solution iff $d \,|\, c$, where $d = \gcd(a, m)$. If $d \,|\, c$, there are exactly d pairwise distict solutions:

$$x_0,\ x_0 + m',\ x_0 + m'2,\ \dots,\ x_0 + m'(d-1),$$

where x_0 is a unique solution to the congruence

$$a'x \equiv c' \pmod{m'}, \tag{4}$$

and $a' = \frac{a}{d}$, $c' = \frac{c}{d}$, and $m' = \frac{m}{d}$.

Proof. For each x, the number x satisfies (3) iff there exists some k such that $ax = c + mk$. This is impossible unless $d \,|\, c$. Suppose $d \,|\, c$. Then the latter equation is equivalent to $a'x = c' + m'k$ (as $d \neq 0$ cancels out), which indeed has a solution (x_1, k_1) due to Theorem 8.3. Consequently, x satisfies congruence (3) iff it satisfies (4). However, this does not mean their solutions are the same since the condition $0 \leq x < m$ does not imply $0 \leq x < m'$.

At first, we observe that if x satisfies (3) (or, equivalently, (4)), then the remainder x' after dividing x by m' is a solution to (4) as $x \equiv x' \pmod{m'}$ and $0 \leq x' < m'$. As we know, some x_1 indeed satisfies (4). Therefore, (4) has a solution x_0.

Let us show that this solution is unique. For, if x is another solution to (4), then, obviously, $a'x \equiv a'x_0 \pmod{m'}$, whence $x \equiv x_0 \pmod{m'}$ by Corollary 7.11. On the other hand, $0 \leq x, x_0 < m'$; one concludes that $x = x_0$ by Corollary 6.11.

Clearly, $x_0 + m'k \equiv x_0 \pmod{m'}$ for each k, hence any number of the form $x_0 + m'k$ satisfies both (4) and (3). Among such numbers, just the following satisfy $0 \leq x < m = m'd$:

$$x_0,\ x_0 + m',\ x_0 + m'2,\ \dots,\ x_0 + m'(d-1).$$

So, these are distinct solutions to (3). Conversely, let x be a solution to (3). Then x satisfies (4) as well, whence $a'x \equiv c' \equiv a'x_0 \pmod{m'}$. By Corollary 7.11, this yields $x \equiv x_0 \pmod{m'}$, i. e., $x = x_0 + m'k$ for some k, whereas $0 \leq x_0 < m'$ and $0 \leq x < m = m'd$. This results in $0 \leq k < d$. Hence, congruence (3) has no solution except those listed above. □

Example 8.6. Solve the congruence $12x \equiv -15 \pmod{9}$.

Clearly, $d = \gcd(12, 9) = 3$. So, $a' = 4$, $c' = -5$, and $m' = 3$. The only solutions to this congruence are the numbers: x_0, $x_0 + 3 \cdot 1$, $x_0 + 3 \cdot 2$, where x_0 is a unique solution to $4x \equiv -5 \equiv 1 \pmod{3}$. One can easily guess this unique solution, so $x_0 = 1$. Hence, $\{1, 4, 7\}$ is the set of solutions to the original congruence. Should it be harder to guess, one might use the Extended Euclidean Algorithm to find x_0 likewise the case of equation.

Remark 8.7. Notice that any polynomial congruence, that is, of the form $P(x_1, \dots, x_n) \equiv 0 \pmod{m}$, where P is a polynomial in variables $x_1, \dots, x_n$ with integer coefficients,—not just a linear one—could be solved by applying brute force: it suffices to test $m \,|\, P(x_1, \dots, x_n)$ for all possible tuples $(x_1, \dots, x_n)$ of numbers $x_i \in \{0, 1, \dots m-1\}$. Lemma 6.14 then guarantees that $P(x'_1, \dots, x'_n) \equiv 0 \pmod{m}$ iff there exists a solution $(x_1, \dots, x_n)$ such that $x'_i \equiv x_i$ for each i (for x_i, take the remainder after dividing x'_i by m). Of course, applying brute force is generally inefficient for moduli big enough—even for the simplest linear congruence we have just considered.

Example 8.8. Let us solve the congruence $P(x) = 7x^3 - 3x^2 + x - 4 \equiv 0 \pmod{6}$. Clearly, $P(x) \equiv x^3 - 3x^2 + x - 4 \equiv x^2(x-3) + (x-3) - 1 \equiv (x^2+1)(x-3) - 1 \pmod{6}$.

$$\begin{aligned}
P(0) &\equiv -4 \equiv 2 \pmod{6};\\
P(1) &\equiv 2 \cdot (-2) - 1 \equiv -5 \equiv 1 \pmod{6};\\
P(2) &\equiv 5 \cdot (-1) - 1 \equiv -6 \equiv 0 \pmod{6};\\
P(3) &\equiv 10 \cdot 0 - 1 \equiv -1 \equiv 5 \pmod{6};\\
P(4) &\equiv P(-2) \equiv 5 \cdot (-5) - 1 \equiv -1 \cdot 1 - 1 \equiv -2 \equiv 4 \pmod{6};\\
P(5) &\equiv P(-1) \equiv 2 \cdot (-4) - 1 \equiv -9 \equiv 3 \pmod{6}.
\end{aligned}$$

Thus, 2 is the only solution to this congruence.

Simultaneous congruences. Consider some moduli $m_1, \dots, m_n$ and numbers $a_1, \dots, a_n$. Is it always possible to find some number x congruent to a_i modulo m_i for each i? In other words, is it always possible to recover a number given all the remainders after dividing it by m_i?

It is easy to see that the answer is negative as no number x satisfies both $x \equiv 4 \pmod{6}$ and $x \equiv 1 \pmod{8}$. Indeed, from the first congruence it follows that x is even, so 1 should be even as well when taking into account the second one.

However, the answer becomes positive assuming the numbers $m_1, \dots, m_n$ are pairwise coprime. We say that a number x_0 is a *solution* to the system of simultaneous congruences

$$\begin{cases}
x \equiv a_1 \pmod{m_1}\\
x \equiv a_2 \pmod{m_2}\\
\dots\\
x \equiv a_n \pmod{m_n},
\end{cases} \tag{5}$$

iff x_0 satisfies each congruence and $0 \le x_0 < M$, where $M = m_1 m_2 \dots m_n$.

Remark 8.9. Again, why are we interested in solutions from $\{0, \dots, M-1\}$ solely? Do they suffice to find *all* the numbers which satisfy the system?

Let us consider a more general situation. We say that μ is a *least common multiple* of numbers $c_1, \dots, c_n$ iff (1) $c_i \mid \mu$ for each i; and (2) for every number μ', if $c_i \mid \mu'$ for each i, then $\mu \mid \mu'$. This is a straightforward generalization of lcm for two numbers, so we write $\mu = \operatorname{lcm}(c_1, \dots, c_n)$.

Exercise 8.10. Prove that for every $c_1, \dots, c_n$, there exists a unique such μ that $\mu = \operatorname{lcm}(c_1, \dots, c_n)$. Moreover, if every c_i is non-zero, so that $c_i = p_1^{\gamma_{i\,1}} \dots p_s^{\gamma_{i\,s}}$, then $\mu = p_1^{\max_i \gamma_{i\,1}} \dots p_s^{\max_i \gamma_{i\,s}}$.

Consider a system

$$\begin{cases}
P_1(x_1, \dots, x_k) \equiv 0 \pmod{c_1}\\
P_2(x_1, \dots, x_k) \equiv 0 \pmod{c_2}\\
\dots\\
P_n(x_1, \dots, x_k) \equiv 0 \pmod{c_n},
\end{cases}$$

where every P_i is a polynomial in variables $x_1, \dots, x_k$ with integer coefficients. We call a tuple $(x_1, \dots, x_k)$ a *solution* to this system if it satisfies each congruence and $0 \le x_1, \dots, x_k < \mu$, where $\mu = \operatorname{lcm}(c_1, \dots, c_n)$. We shall show that a tuple $(x_1', \dots, x_k')$ satisfies the system iff there exists a solution $(x_1, \dots, x_k)$ to the latter such that $x_j' \equiv x_j \pmod{\mu}$ for every j.

Indeed, if a tuple $(x'_1, \ldots, x'_k)$ satisfies the system, one can consider another tuple $(x''_1, \ldots, x''_k)$, where every x''_j is the remainder after dividing x'_j by μ. Then $\mu \,|\, (x'_j - x''_j)$ and $c_i \,|\, \mu$, whence $x''_j \equiv x'_j \pmod{c_i}$ for every i and j. By Lemma 6.14, $P_i(x''_1, \ldots, x''_k) \equiv P_i(x'_1, \ldots, x'_k) \equiv 0 \pmod{c_i}$ for each i. Hence, $(x''_1, \ldots, x''_k)$ is a solution to the system. The other direction is similar. Thus, solutions are sufficient to restore all the satisfying tuples.

Lemma 8.11. *If the numbers $c_1, \ldots, c_n$ are pairwise coprime, then* $\operatorname{lcm}(c_1, \ldots, c_n) = c_1 \cdot \ldots \cdot c_n$.

Proof. The case when some c_i is zero is trivial. Assume none is. Then every prime p_j has degree $\gamma_{1\,j} + \ldots + \gamma_{n\,j}$ in the right-hand side and degree $\max_{1 \le i \le n} \gamma_{i\,j}$ in the left-hand side (see Exercise 8.10). When $i \ne k$, we have $\gcd(c_i, c_k) = 1$, whence $\min(\gamma_{i\,j}, \gamma_{k\,j}) = 0$ for each j. This means that all the numbers $\gamma_{1\,j}, \ldots, \gamma_{n\,j}$, except at most one—the greatest among them, equal zero. Therefore, $\gamma_{1\,j} + \ldots + \gamma_{n\,j} = \max_{1 \le i \le n} \gamma_{i\,j}$ for each j; hence, $\operatorname{lcm}(c_1, \ldots, c_n) = c_1 \cdot \ldots \cdot c_n$. □

This explains why we have taken $M = m_1 m_2 \ldots m_n$ for system (5).

Theorem 8.12 (Chinese Remainder Theorem)**.** *Given the numbers $m_1, \ldots, m_n$ are pairwise coprime, system (5) has a unique solution.*

Proof. We check the uniqueness first. Let x and x' be two solutions to (5). Clearly, $x \equiv x' \pmod{m_i}$, that is, $m_i \,|\, (x - x')$ for each i. By Lemma 8.11, $M = \operatorname{lcm}(m_1, \ldots, m_n)$; hence, $M \,|\, (x - x')$. Therefore, $x \equiv x' \pmod{M}$, whence $x = x'$ by Corollary 6.11.

We prove the existence of a solution by induction on n. For $n = 1$, the remainder after dividing a_1 by m_1 is a sure solution. Suppose that there is a solution to any system of n congruences of the form (5), while we are given with an arbitrary system of the form:

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ \ldots \\ x \equiv a_n \pmod{m_n} \\ x \equiv a \pmod{m}, \end{cases} \tag{6}$$

where $m, m_1, \ldots, m_n$ are pairwise coprime. Let x_0 be a solution to the first n simultaneous congruences. Those congruences are thus equivalent to the system

$$\begin{cases} x \equiv x_0 \pmod{m_1} \\ x \equiv x_0 \pmod{m_2} \\ \ldots \\ x \equiv x_0 \pmod{m_n}, \end{cases} \tag{7}$$

since given $x_0 \equiv a_i \pmod{m_i}$, for each x one has $x \equiv a_i \pmod{m_i}$ iff $x \equiv x_0 \pmod{m_i}$. Let μ be $m_1 m_2 \ldots m_n$. By the proven uniqueness statement, system (7) has a unique solution among $0, 1, \ldots, \mu - 1$, that is, if x satisfies (7), then $x \equiv x_0 \pmod{\mu}$. Clearly, the latter is enough for x to satisfy (7). Finally, one sees that system (6) is equivalent to

$$\begin{cases} x \equiv x_0 \pmod{\mu} \\ x \equiv a \pmod{m}. \end{cases} \tag{8}$$

As μ and m are coprime, there exist some numbers u and v such that $\mu u+mv=1$ due to Corollary 7.15. Consider the number $x_1=a\mu u+x_0mv$. We see that $x_1-x_0=a\mu u+x_0(mv-1)=a\mu u+x_0(-\mu u)=\mu u(a-x_0)$, whence $\mu\,|(x_1-x_0)$. Likewise, $x_1-a=a(\mu u-1)+x_0mv=a(-mv)+x_0mv=mv(x_0-a)$, which yields $m\,|(x_1-a)$. Therefore, x_1 satisfies both systems (8) and (6). Let x_2 be the remainder after dividing x_1 by $M=\mu m$. As $x_2\equiv x_1\pmod{m}$ and $x_2\equiv x_1\pmod{m_i}$ for each i, the number x_2 is a solution to (6). □

In fact, one can easily extract a recursive algorithm for solving a system of the form (5) from the proof presented. The numbers u and v could surely be found using the Extended Euclidean Algorithm.

Example 8.13. Solve the simultaneous congruences:

$$\begin{cases} x\equiv 12 \pmod{15}\\ x\equiv 8 \pmod{17}\\ x\equiv 3 \pmod{8}. \end{cases}$$

Let us solve the first two congruences. Firstly, we are to find some numbers u and v such that $15u+17v=1$. Applying the Extended Euclidean Algorithm, obtain $u=8$ and $v=-7$. According to the proof of Theorem 8.12, the number $x_1=8\cdot15\cdot8+12\cdot17\cdot(-7)\equiv-15\cdot4-51\cdot(-7)\equiv357-60\equiv42 \pmod{15\cdot17}$ satisfies those congruences and 42 is the only solution to the system thereof. We have thus reduced the problem to solving the following simultaneous congruences:

$$\begin{cases} x\equiv 42 \pmod{255}\\ x\equiv 3 \pmod{8}. \end{cases}$$

It is easy to guess the values $u'=-1$ and $v'=32$ for $255u'+8v'=1$ to hold. Hence, $x_2=3\cdot(-255)+42\cdot32\cdot8=3\cdot(14\cdot256-255)=3\cdot(13\cdot255+14)\equiv7\cdot255+3\cdot14\equiv1827 \pmod{255\cdot8}$ satisfies the latter system and 1827 is the unique solution to the original problem.

Example 8.14. This method can be easily generalized to the case when some congruences are of the form $bx\equiv a\pmod{m}$ (or any other form we can however solve). For example, let us consider the system

$$\begin{cases} 2x\equiv 10 \pmod{22}\\ x\equiv 8 \pmod{31}. \end{cases}$$

By Theorem 8.5, the first congruence has solutions 5 and 16 and is thus equivalent to the *disjunction* $x\equiv5\pmod{22}\vee x\equiv16\pmod{22}$, which may be also written as

$$\left[\begin{array}{l} x\equiv 5 \pmod{22}\\ x\equiv 16 \pmod{22}. \end{array}\right.$$

As simultaneous congruences form a *conjunction*, one can apply $(A\vee B)\wedge C\equiv(A\wedge C)\vee(B\wedge C)$ to obtain

$$\left[\begin{array}{l} \begin{cases} x\equiv 5 \pmod{22}\\ x\equiv 8 \pmod{31} \end{cases}\\ \begin{cases} x\equiv 16 \pmod{22}\\ x\equiv 8 \pmod{31}. \end{cases} \end{array}\right.$$

Now, it suffices to solve each subsystem separately.

9 Binary Relations

We introduce the algebra of binary relations to provide students with a unified formalism for functions, orders and equivalences. First, this allows to solve many problems via a straight-forward symbolic computation. Second, we try to overcome the well-known difficulties many students experience when learning images and pre-images (which are typically presented in an asymmetric manner and for the function case only) and various concepts of a function's 'inverse' as well.

Let A and B be some sets. Any set R such that $R \subseteq A \times B$ is called a *(binary) relation between* sets A and B. This simple notion forms the base for modeling functions and orderings in the formalism of sets. The set

$$\operatorname{dom} R = \{a \in A \mid \exists b\, (a, b) \in R\}$$

is called the *domain* of the relation R while the set

$$\operatorname{rng} R = \{b \in B \mid \exists a\, (a, b) \in R\}$$

is called the *range* of the relation R. The set $\operatorname{dom} R \cup \operatorname{rng} R$ is called the *field* of the relation R. Clearly, $\operatorname{dom} R \subseteq A$ and $\operatorname{rng} R \subseteq B$. We see that $\operatorname{dom} R$ ($\operatorname{rng} R$) is the set of the first (second) coordinates of the pairs from R. If we mention just a *binary relation* R, we imply that some sets A and B with $R \subseteq A \times B$ are fixed yet their exact specification is immaterial.

We prefer this 'parametric' definition to the more natural one (where every set R of ordered pairs is called a relation) since the latter would require taking $\cup\cup R$ for the relation's field, whereas we want to avoid infinite unions generally.

Example 9.1. The sets $\varnothing$ and $A \times B$ are binary relations between A and B.

Let $A = \{0, 2, x, y\}$ and $B = \{z, 0, 1\}$ for some sets x, y, z. The set

$$R = \{(0, 0), (0, z), (x, 1), (y, 1)\}$$

is a binary relation between A and B. Clearly, $\operatorname{dom} R = \{0, x, y\} \subseteq A$ (but $\operatorname{dom} R \neq A$ when neither x nor y equals 2) and $\operatorname{rng} R = B$.

If $R \subseteq A \times A$ for a set A, the relation R is called a *binary relation on* the set A.

Example 9.2. The set $\varnothing$ is a binary relation on any set. The sets $A^2 = A \times A$ and

$$\mathrm{id}_A = \{(x, x) \mid x \in A\}$$

are binary relations on A. The *identity relation* id_A is of some import. Clearly, $(x, y) \in \mathrm{id}_A$ iff $x = y$ and $x, y \in A$. One may think of id_A as of the set-theoretic representation for equality predicate (restricted to the set A).

Likewise, one can consider the predicate 'less than' over natural numbers as a *set*—namely, the binary relation $< \,= \{(m, n) \in \mathbb{N}^2 \mid m \text{ is less than } n\}$. This way, one gets $(2, 3) \in \,<$ (as $2 < 3$) yet $(2, 2) \notin \,<$ (as $2 \not< 2$). In fact, *every* binary predicate over a set may be identified with a binary relation. Thus, binary relations comprise the set-theoretic model for binary predicates.

On the other hand, for any given binary relation R, we will use a 'predicate-style' notation, that is, we will often write xRy instead of $(x, y) \in R$.

Every geometric figure in the coordinate plane $\mathbb{R}^2$ is a binary relation on $\mathbb{R}$. Its domain and range are the projections thereof onto the abscissa and ordinate axes, respectively. (See Figure 4).

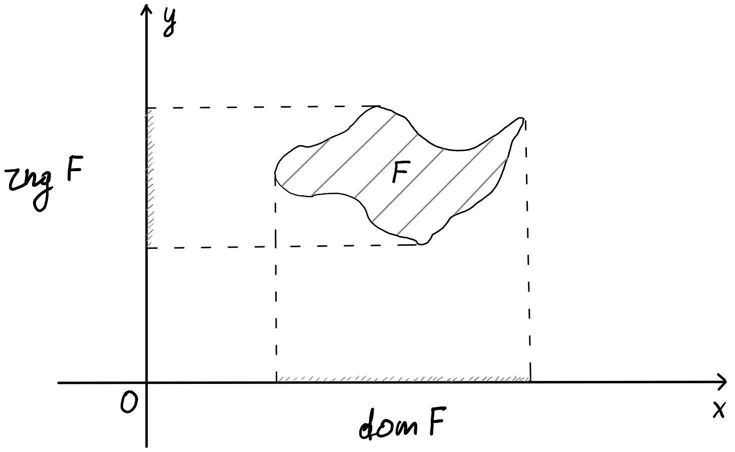

Figure 4: A planar figure (equivalently, a binary relation on $\mathbb{R}$) F with its two projections $\operatorname{dom} F$ and $\operatorname{rng} F$ are highlighted.

For any finite relation R one can draw a *diagram* (in principle, at least). That is, one labels some plane points with elements of the field of R (or a superset thereof). Sometimes, they allow multiple points with identical labels or multiple labels for the same point (if the respective elements are equal). Then one draws an arrow from the point (labeled by) a to the point b iff $(a, b) \in R$. When plotting two or more relations on one diagram, it is useful to label the arrows themselves with relation symbols such as 'R'.

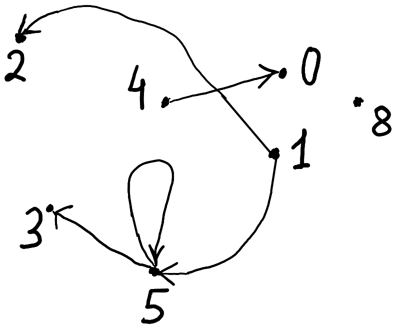

Figure 5: A diagram for the relation $\{(5,5), (5,3), (1,5), (1,2), (4,0)\}$ on the set $\{0, 1, 2, 3, 4, 5, 8\}$.

Exercise 9.3. Let $A = \{1, 2, 3, 4\}$. Draw a diagram depicting relations $P = A^2$, $Q = \operatorname{id}_A$, and $R = \{(m, n) \in A^2 \mid m < n\}$.

Algebra of relations. Let $R \subseteq A \times B$. The *converse* relation of R is the relation

$$R^{-1} = \{(b, a) \in \operatorname{rng} R \times \operatorname{dom} R \mid (a, b) \in R\},$$

that is, each pair from R is overturned in R^{-1}.

Exercise 9.4. Prove that $\operatorname{dom} R^{-1} = \operatorname{rng} R$ and $\operatorname{rng} R^{-1} = \operatorname{dom} R$.

Example 9.5. For every sets A, B, it is clear that $\varnothing^{-1} = \varnothing$, $\mathrm{id}_A^{-1} = \mathrm{id}_A$, and $(A \times B)^{-1} = B \times A$.

As we have already seen, the predicates 'less than', 'less or equal' (for natural arguments) etc. may be treated as binary relations on $\mathbb{N}$. We have $<^{-1} = >$, $>^{-1} = <$, and $\leq^{-1} = \geq$.

Since binary relations are sets, every set algebra operation makes sense for them. For example, one has $< \cup \, \mathrm{id}_{\mathbb{N}} = \leq$. Indeed, a pair $(x, y) \in \mathbb{N}^2$ belongs to the left-hand side iff $x < y$ or $x = y$, which is equivalent to $x \leq y$. As usual, to define the *complement* of a relation R, one must fix a *universe* first. Since we have just *relations between two sets* formally, one has certain sets A and B with $R \subseteq A \times B$ already fixed. In general, we put $\bar{R} = (A \times B) \smallsetminus R$, but caution is recommended at this point.

Remark 9.6. It is important to realize that *conversion* and *complement* operations are not the same. (We shall see in a moment that they result in distinct relations 'almost surely'.) In particular, $\bar{<} = \geq$ (since $x \not< y$ is equivalent to $x \geq y$ for all $x, y \in \mathbb{N}$) whereas $<^{-1} = >$.

> In my experience, the students often confuse the two.

Lemma 9.7. *Let $P, Q \subseteq A \times B$ be arbitrary binary relations. Then*

1. $(P^{-1})^{-1} = P$;
2. $(P \cup Q)^{-1} = P^{-1} \cup Q^{-1}$;

Proof. For the first statement, for an arbitrary pair (a, b) obtain

$$\begin{aligned} (a, b) \in (P^{-1})^{-1} \quad &\Longleftrightarrow \quad (b, a) \in P^{-1} \\ &\Longleftrightarrow \quad (a, b) \in P. \end{aligned}$$

For the second one,

$$\begin{aligned} (a, b) \in (P \cup Q)^{-1} \quad &\Longleftrightarrow \quad (b, a) \in P \cup Q \\ &\Longleftrightarrow \quad (b, a) \in P \vee (b, a) \in Q \\ &\Longleftrightarrow \quad (a, b) \in P^{-1} \vee (a, b) \in Q^{-1} \\ &\Longleftrightarrow \quad (a, b) \in P^{-1} \cup Q^{-1}. \end{aligned}$$

□

Corollary 9.8. *If $P \subseteq Q$, then $P^{-1} \subseteq Q^{-1}$.*

Proof. Suppose that $P \subseteq Q$. By Lemma 4.19, get $Q = P \cup Q$. Hence, $Q^{-1} = (P \cup Q)^{-1} = P^{-1} \cup Q^{-1}$ and $P^{-1} \subseteq Q^{-1}$ by the same Lemma. □

Exercise 9.9. Prove that $(P \cap Q)^{-1} = P^{-1} \cap Q^{-1}$.

Example 9.10. Suppose that $R \subseteq A \times B$ for non-empty sets A and B. Then $R^{-1} \neq \bar{R}$.

Assuming the contrary, let $R^{-1} = \bar{R}$. First, we are to prove that $A \cap B \neq \varnothing$. If $R = \varnothing$, then $R^{-1} = \varnothing$. On the other hand, $\bar{R} = (A \times B) \smallsetminus \varnothing = A \times B \neq \varnothing$. The contradiction implies $R \neq \varnothing$, i. e., $(a, b) \in R$ for some $a \in A$ and $b \in B$. Then $(b, a) \in R^{-1}$, whence $(b, a) \in (A \times B) \smallsetminus R$. Hence, $b \in A$ and $b \in A \cap B$.

Let x be an element of $A \cap B$. If $(x, x) \in R$, then $(x, x) \in R^{-1} = \bar{R}$, i. e., $(x, x) \notin R$. The contradiction yields $(x, x) \notin R$. But then $(x, x) \in \bar{R} = R^{-1}$. We thus get $(x, x) \in R$ and a new contradiction. The only remaining assumption $R^{-1} = \bar{R}$ must be false.

Let P and Q be arbitrary binary relations. The set

$$Q \circ P = \{(a, c) \in \operatorname{dom} P \times \operatorname{rng} Q \mid \exists b \left(aPb \wedge bQc\right)\}$$

is then called the *composition* of the relations P and Q. Clearly, $Q \circ P \subseteq A \times C$ if $P \subseteq A \times B$ and $Q \subseteq B \times C$. On the other hand, we have

$$(a, c) \in Q \circ P \iff \exists b \left(aPb \wedge bQc\right)$$

for *any* pair (a, c) since the right-hand side implies that $a \in \operatorname{dom} P$ and $c \in \operatorname{rng} Q$.

Remark 9.11. Note that the relation P 'acts' first but comes second (from left to right) in the expression $Q \circ P$. This counter-intuitive convention is motivated by the traditional notation for the composition of functions (which we discuss later), where $(g \circ f)(x) = g(f(x))$.

In a diagram, points a and c are connected by a $(Q \circ P)$-arrow iff there exists a two arrow path from a to c first going along a P-arrow from a to some b and then along a Q-arrow from b to c. You can think about composition as the set of pairs of places connected by a route with a change.

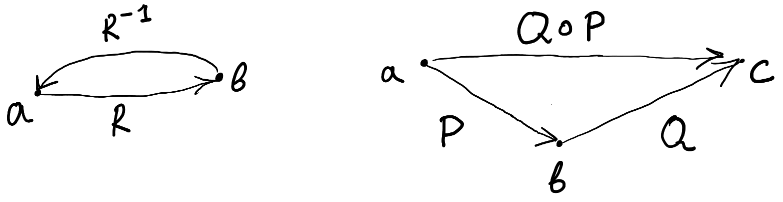

Figure 6: Conversion and composition.

Example 9.12. Another illustration. Let A be a group of authors, C be a group of critics, and B be a set of books. Assume that aPb means that *Author a has authored Book b*, while bQc means that *Book b is praised by Critic c*. Then $a(Q \circ P)c$ means that Author a has authored a book praised by Critic c.

Exercise 9.13. Let A be a group of men and let aPb mean that *a is a son of b*. Which concepts are expressed by the relations $P \circ P$, P^{-1}, $P^{-1} \circ P$, and $P \circ P^{-1}$?

Example 9.14. On $\mathbb{N}$, we have $\leq \circ < \, = \{(m, n) \in \mathbb{N}^2 \mid \exists k \in \mathbb{N} \, (m < k \wedge k \leq n)\}$. If $m < k$ and $k \leq n$, then $m < n$. If $m < n$, then we get $m < k$ and $k \leq n$ by assigning $k = n$. Hence, $\leq \circ < \, = \, <$.

As we shall see, composition is similar to numerical multiplication in many respects. For example, there is a 'zero' element.

Exercise 9.15. Prove that $R \circ \varnothing = \varnothing \circ R = \varnothing$.

Also, there is an analogue for unity, or 'neutral element' w. r. t. composition (a quantity does not change if you multiply it by unity). For relations between A and B, we have distinct 'left' and 'right' unities indeed.

Example 9.16. If $R \subseteq A \times B$, then $R \circ \mathrm{id}_A = R$ and $\mathrm{id}_B \circ R = R$.

Let us prove the first equality. If $(x, y) \in R$ then $x \in A$ and $(x, x) \in \mathrm{id}_A$. Thus we have $(x, z) \in \mathrm{id}_A$ and $(z, y) \in R$ for a suitable z—namely, for $z = x$. Hence $(x, y) \in R \circ \mathrm{id}_A$ by the definition of composition. For the other direction, suppose that $(x, y) \in R \circ \mathrm{id}_A$. By the same definition, we conclude that $(x, z) \in \mathrm{id}_A$ and $(z, y) \in R$ for some z. But the former means that $z = x$, whence $(x, y) \in R$.

Theorem 9.17 (Composition associativity)**.** *Let P, Q, R be arbitrary binary relations. Then*

$$R \circ (Q \circ P) = (R \circ Q) \circ P.$$

Proof. For any pair (a, d), obtain:

$$\begin{aligned}
(a,d) \in R \circ (Q \circ P) \quad &\iff \quad \exists c\, \big(a(Q \circ P)c \wedge cRd\big) \\
&\iff \quad \exists c\, \big(\exists b\, (aPb \wedge bQc) \wedge cRd\big) \\
&\iff \quad \exists c\, \exists b\, \big((aPb \wedge bQc) \wedge cRd\big) \\
&\iff \quad \exists b\, \exists c\, \big(aPb \wedge (bQc \wedge cRd)\big) \\
&\iff \quad \exists b\, \big(aPb \wedge \exists c\, (bQc \wedge cRd)\big) \\
&\iff \quad \exists b\, \big(aPb \wedge b(R \circ Q)d\big) \\
&\iff \quad (a,d) \in (R \circ Q) \circ P.
\end{aligned}$$

Here, we have observed that cRd does not depend on b (and does not mention at all); therefore the quantifier on b does not change the meaning of this statement, so the later can be safely placed in or out of the scope of that quantifier.

□

> For this argument, the Instructor might wish to explain the logical equivalences $\exists x\, (A(x) \wedge B) \equiv \exists x\, A(x) \wedge B$ (where B does not mention x) and $\exists x \exists y\, A \equiv \exists y \exists x\, A$ in more detail.

Remark 9.18. Associativity allows us to omit parentheses in expressions like $P_1 \circ P_2 \circ \ldots \circ P_n$ as it does not matter how one places them.

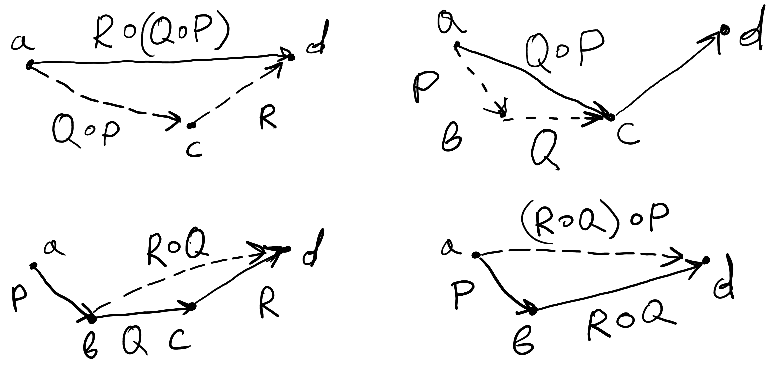

Figure 7: Proving Theorem 9.17.

Conversion is *somewhat* similar to taking a multiplicative inverse for numbers: one has $\frac{1}{xy} = \frac{1}{y} \cdot \frac{1}{x}$.

> The Instructor might wish to compare conversion to matrix transposition or string reversion as well.

Theorem 9.19. *Let P, Q be arbitrary binary relations. Then $(Q \circ P)^{-1} = P^{-1} \circ Q^{-1}$.*

Proof. For any pair (a, c), obtain:

$$\begin{aligned} (a,c) \in (Q \circ P)^{-1} &\iff (c,a) \in Q \circ P \\ &\iff \exists b\,(cPb \wedge bQa) \\ &\iff \exists b\,(aQ^{-1}b \wedge bP^{-1}c) \\ &\iff (a,c) \in P^{-1} \circ Q^{-1}. \end{aligned}$$

□

Remark 9.20. For numbers, one also has $x \cdot \frac{1}{x} = x$. Nevertheless, a similar statement fails for binary relations. Indeed, consider the relation $R = \{(1,2)\}$ on the set $A = \{1,2\}$. Clearly, $R^{-1} = \{(2,1)\}$, $R \circ R^{-1} = \{(2,2)\}$, and $R^{-1} \circ R = \{(1,1)\}$. Neither of the relations $R \circ R^{-1}$ and $R^{-1} \circ R$ equals id_A, which is the analogue of unity as we have seen.

For this reason, we have preferred to call R^{-1} the *converse* rather than the *inverse* of the relation R.

Exercise 9.21. The analogues of the following statements hold for numerical multiplication. Do they hold for arbitrary binary relations?

1. $P \circ Q = Q \circ P$;

2. if $P \neq \varnothing$ and $P \circ Q = P \circ R$, then $Q = R$.

Lemma 9.22. *Let R, P, Q be arbitrary binary relations. Then*

1. *$(P \cup Q) \circ R = (P \circ R) \cup (Q \circ R)$;*

2. *$(P \cap Q) \circ R \subseteq (P \circ R) \cap (Q \circ R)$.*

Proof. For any pair (a, c),

$$\begin{aligned} (a,c) \in (P \cup Q) \circ R &\iff \exists b\,\big(aRb \wedge (b,c) \in P \cup Q\big) \\ &\iff \exists b\,\big(aRb \wedge (bPc \text{ or } bQc)\big) \\ &\iff \exists b\,\big((aRb \wedge bPc) \text{ or } (aRb \wedge bQc)\big) \\ &\iff \exists b\,\big(aRb \wedge bPc\big) \text{ or } \exists b\,\big(aRb \wedge bQc\big) \\ &\iff (a,c) \in P \circ R \text{ or } (a,c) \in Q \circ R \\ &\iff (a,c) \in (P \circ R) \cup (Q \circ R). \end{aligned}$$

Here we have applied a certain law of logic. Indeed, $\exists x\,(A \vee B)$ ("there exists a unicorn either black or tame") iff $\exists x\,A \vee \exists x\,B$ ("there exists a black unicorn or there exists a tame unicorn").

Consider the second statement.

$$\begin{aligned} (a,c) \in (P \cap Q) \circ R &\iff \exists b\,\big(aRb \wedge (b,c) \in P \cap Q\big) \\ &\iff \exists b\,\big(aRb \wedge (bPc \wedge bQc)\big) \\ &\iff \exists b\,\big((aRb \wedge bPc) \wedge (aRb \wedge bQc)\big) \\ &\implies \exists b\,\big(aRb \wedge bPc\big) \wedge \exists b\,\big(aRb \wedge bQc\big) \\ &\iff (a,c) \in P \circ R \wedge (a,c) \in Q \circ R \\ &\iff (a,c) \in (P \circ R) \cap (Q \circ R). \end{aligned}$$

Thus, from $(a,c) \in (P \cap Q) \circ R$, it follows that $(a,c) \in (P \circ R) \cap (Q \circ R)$. We have applied the fact that $\exists x\,(A(x) \wedge B(x))$ ("there exists a unicorn both black and tame") implies $\exists x\,A(x) \wedge \exists x\,B(x)$ ("there exists a black unicorn and there exists a tame unicorn"). The reverse implication does not hold in general. □

Example 9.23. Here is an example where the last inclusion of Lemma 9.22 cannot be strengthened to become an equality. Consider relations $R = \{(0,1),(0,2)\}$, $Q = \{(1,3)\}$ and $P = \{(2,3)\}$. Clearly, $(0,3) \in (P \circ R) \cap (Q \circ R)$, while $P \cap Q$ and $(P \cap Q) \circ R$ are empty.

Example 9.24. Suppose that $P \subseteq Q$. Then $P \circ R \subseteq Q \circ R$.

Indeed, we get $Q = P \cup Q$, whence $Q \circ R = (P \cup Q) \circ R = (P \circ R) \cup (Q \circ R) \supseteq P \circ R$ by Lemma 9.22.

Exercise 9.25. Apply the above example to derive Claim 2 of Lemma 9.22.

Example 9.26. By Lemma 9.22 and Theorem 9.19, it holds that

$$\begin{aligned}(R \circ (P \cup Q))^{-1} = (P \cup Q)^{-1} \circ R^{-1} = \\ = (P^{-1} \cup Q^{-1}) \circ R^{-1} = (P^{-1} \circ R^{-1}) \cup (Q^{-1} \circ R^{-1}) = \\ = (R \circ P)^{-1} \cup (R \circ Q)^{-1} = ((R \circ P) \cup (R \circ Q))^{-1}.\end{aligned}$$

Hence,

$$R \circ (P \cup Q) = ((R \circ (P \cup Q))^{-1})^{-1} = ((R \circ P) \cup (R \circ Q))^{-1})^{-1} = (R \circ P) \cup (R \circ Q).$$

One can derive $R \circ (P \cap Q) \subseteq (R \circ P) \cap (R \circ Q)$ in a similar manner.

Image of set. Let R be a binary relation and X be some set. Then the set

$$R[X] = \{b \in \operatorname{rng} R \mid \exists a \in X\, aRb\}$$

is called the *image of X under R* (or simply the *R-image of X*). It is easy to see that

$$b \in R[X] \iff \exists a \in X\, aRb$$

for the right-hand side clearly implies $b \in \operatorname{rng} R$. The set $R^{-1}[X]$ (that is, the image of X under R^{-1}) is usually called the *preimage of X under R*. Clearly,

$$R^{-1}[X] = \{a \in \operatorname{rng} R^{-1} \mid \exists b \in X\, bR^{-1}a\} = \{a \in \operatorname{dom} R \mid \exists b \in X\, aRb\}.$$

Considering arrow diagrams, we see that $R[X]$ is the set of endings of R-arrows beginning somewhere in X, while $R^{-1}[X]$ is the set of beginnings of R-arrows with endings in X.

Remark 9.27. If $R \subseteq A \times B$, then $\operatorname{dom} R = R^{-1}[B]$ и $\operatorname{rng} R = R[A]$.

Example 9.28. Let A be a set of cities and let xRy mean that there is some route from x to y along existing highways for a relation $R \subseteq A^2$. Then $R[X]$ is the set of cities accessible from at least one city that belongs to X.

Example 9.29. Let us show that $R[X \cup Y] = R[X] \cup R[Y]$. For any set b, obtain

$$\begin{aligned} b \in R[X \cup Y] &\iff \exists a \big(a \in X \cup Y \land aRb\big) \\ &\iff \exists a \big((a \in X \lor a \in Y) \land aRb\big) \\ &\iff \exists a \big((a \in X \land aRb) \lor (a \in Y \land aRb)\big) \\ &\iff \exists a \big(a \in X \land aRb) \lor \exists a\, (a \in Y \land aRb) \\ &\iff b \in R[X] \lor b \in R[Y] \\ &\iff b \in R[X] \cup R[Y]. \end{aligned}$$

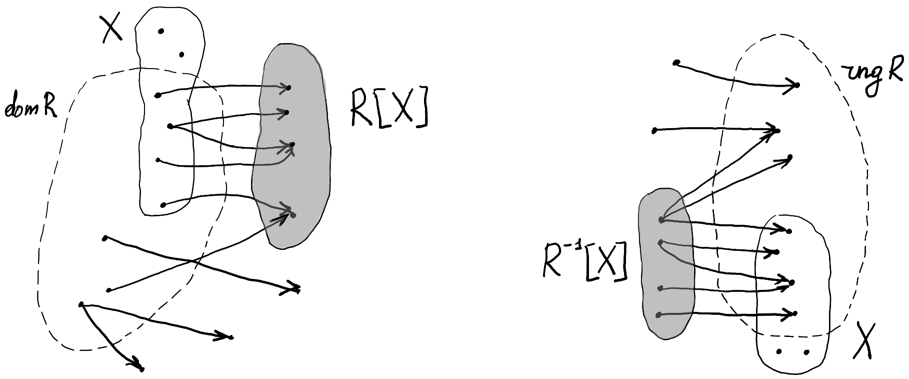

Figure 8: An image and a pre-image.

Exercise 9.30. Prove that $X \subseteq Y$ implies $R[X] \subseteq R[Y]$.

Exercise 9.31. Prove that $R[\varnothing] = \varnothing$.

Example 9.32. $R[X \cap Y] \subseteq R[X] \cap R[Y]$.

From $X \cap Y \subseteq X$ and $X \cap Y \subseteq Y$, it follows that $R[X \cap Y] \subseteq R[X]$ and $R[X \cap Y] \subseteq R[Y]$ by Exercise 9.30, which implies the required inclusion.

The reverse inclusion is not yet necessary. E. g., consider the relation $R = \{(0,2),(1,2)\}$. For $X = \{0\}$ and $Y = \{1\}$, we have $R[X] = R[Y] = R[X] \cap R[Y] = \{2\}$ but $R[X \cap Y] = R[\varnothing] = \varnothing$.

Example 9.33. $(R \circ Q)[X] = R[Q[X]]$.

For any c, we have

$$\begin{aligned}
c \in (R \circ Q)[X] \quad &\iff \quad \exists a \left(a \in X \wedge (a,c) \in R \circ Q\right) \\
&\iff \quad \exists a \left(a \in X \wedge \exists b\,(aQb \wedge bRc)\right) \\
&\iff \quad \exists a\,\exists b \left((a \in X \wedge aQb) \wedge bRc\right) \\
&\iff \quad \exists b \left(\exists a\,(a \in X \wedge aQb) \wedge bRc\right) \\
&\iff \quad \exists b \left(b \in Q[X] \wedge bRc\right) \\
&\iff \quad c \in R[Q[X]].
\end{aligned}$$

The above equation gives another reason for that strange “acting first is written last” rule for composition as long as we prefer the notation $R[X]$ to $[X]R$ for image.

10 Functions

Function is one of the most basic concepts in mathematics. Loosely speaking, this notion expresses the idea of *dependence* of one 'quantity' on another, as in the laws of nature. This idea can be modeled in set-theoretic framework by using binary relations of a special kind.

A binary relation $R \subseteq A \times B$ is called:

1. *functional* iff $\forall x \forall y \forall z\, ((xRy \wedge xRz) \implies y = z)$;

2. *injective* iff $\forall x \forall y \forall z\, ((yRx \wedge zRx) \implies y = z)$;

3. *total for a set* Z iff $\forall x \in Z\, \exists y\; xRy$;

4. *surjective for a set* Z iff $\forall x \in Z\, \exists y\; yRx$.

By default, we say that R is *total* if R is total for A and we say that R is *surjective* when R is surjective for B.

Example 10.1. Let $R = \{(0,1),(0,2),(1,1),(3,4)\}$. The relation R is not functional as $0R1$ and $0R2$ hold but $1 \neq 2$. That is, we have two R-arrows sharing their beginning whose endings are yet separate. Nor is the relation R injective as $(0,1)$ and $(1,1)$ are two R-arrows with same end but different beginnings. Clearly, R is total for $\operatorname{dom} R = \{0,1,3\}$ and surjective for $\operatorname{rng} R = \{1,2,4\}$ but R is neither total nor surjective for its field $\{0,1,2,3,4\}$ as there is no R-arrow beginning at 2 nor an R-arrow ending at 0.

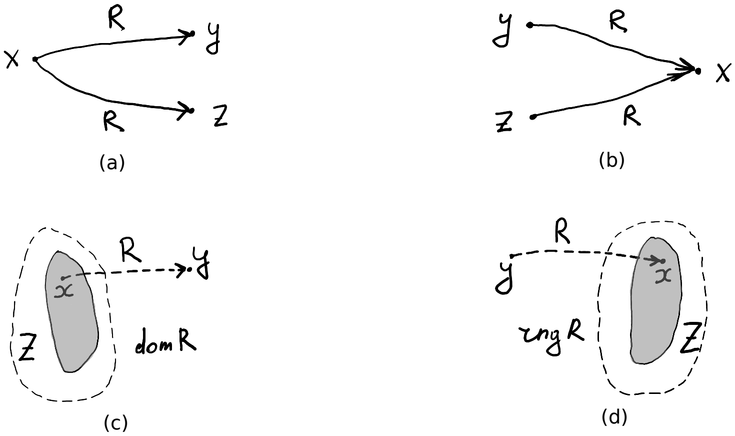

Figure 9: (a) R is *not* functional when $y \neq z$; (a) R is *not* injective when $y \neq z$; (c) R is total for Z; (d) R is surjective for Z.

Example 10.2. The relation $<$ on $\mathbb{N}$ is total since for any natural m there exists a greater number n. But $<$ is not functional as such a number n is not unique. This relation is not injective for $0 < 2$, $1 < 2$ but $0 \neq 1$, nor surjective as $n < 0$ holds for no n.

Let $R = \{(x,y) \in \mathbb{R}_+ \times \mathbb{R} \mid x = y^2\}$ for $\mathbb{R}_+ = \{a \in \mathbb{R} \mid a > 0\}$. Then the relation R is total but not surjective ($xR0$ is not possible). R is injective as from $x = y^2$ and $z = y^2$, it follows that $x = z$, but R is not functional for $(1,1) \in R$ and $(1,-1) \in R$.

Remark 10.3. Clearly, R is total for Z iff $Z \subseteq \operatorname{dom} R$ and R is surjective for Z iff $Z \subseteq \operatorname{rng} R$.

The notions of functionality and injectivity (as well as those of totality and surjectivity) are dual to each other.

Lemma 10.4.

1. *R is functional $\iff$ R^{-1} is injective; R is injective $\iff$ R^{-1} is functional;*

2. *R is total for Z $\iff$ R^{-1} is surjective for Z; R is surjective for Z $\iff$ R^{-1} is total for Z.*

Proof. By definition,

$$R \text{ is functional } \iff \forall x \forall y \forall z\, ((xRy \wedge xRz) \Longrightarrow y = z) \iff$$
$$\forall x \forall y \forall z\, ((yR^{-1}x \wedge zR^{-1}x) \Longrightarrow y = z) \iff R^{-1} \text{ is injective.}$$

Now, we apply the statement just proved to R^{-1}:

$$R \text{ is injective } \iff (R^{-1})^{-1} \text{ is injective } \iff R^{-1} \text{ is functional.}$$

The second claim can be proved similarly. □

All four notions are invariant w. r. t. the relation composition.

Lemma 10.5. *Let $Q \subseteq A \times B$ and $R \subseteq B \times C$. If Q and R are functional (injective, total, surjective), then so is $R \circ Q$.*

Proof. Firstly, let us check functionality. Suppose that $x(R \circ Q)y$ and $x(R \circ Q)z$. Then there exist $u, v \in B$ such that xQu, xQv, uRy, and vRz. As Q is functional, obtain $u = v$, whence uRy and uRz. Thus, $y = z$ by functionality of R.

We use duality (Lemma 10.4) to check injectivity. As $Q = (Q^{-1})^{-1}$ and $R = (R^{-1})^{-1}$ are injective, the relations Q^{-1} and R^{-1} are functional as well as $Q^{-1} \circ R^{-1} = (R \circ Q)^{-1}$ is. Hence, $R \circ Q$ is injective.

Now, assume that Q is total for A and R is total for B. We have to show that $R \circ Q$ is total for A. For any $x \in A$, there exists some u such that xQu. As $Q \subseteq A \times B$, we have $u \in B$. Hence, there exists some y such that uRy. So, xQu and uRy for a certain u, that is, $x(R \circ Q)y$.

Surjectivity can be easily proved by Lemma 10.4 now. □

Example 10.6. Let a relation R be functional. Then $R^{-1}[X \cap Y] = R^{-1}[X] \cap R^{-1}[Y]$ for every sets X and Y.

The inclusion from left to right is due to Example 9.32. For the opposite direction, suppose that $a \in R^{-1}[X]$ and $a \in R^{-1}[Y]$. Then there are some $b \in X$ and $c \in Y$ such that $bR^{-1}a$ and $cR^{-1}a$, i. e., aRb and aRc. Since R is functional, we get $b = c$. Then $b \in X \cap Y$; hence $a \in R^{-1}[X \cap Y]$.

Example 10.7. Let a relation $R \subseteq A \times B$ be total. Then $X \subseteq R^{-1}[R[X]]$ for any $X \subseteq A$.

Indeed, let $a \in X$. By totality, get aRb for some $b \in B$. See that $b \in R[X]$ and $bR^{-1}a$, whence $a \in R^{-1}[R[X]]$.

Is the converse inclusion necessary? No, it is not. Indeed, consider the sets $A = \{0, 1\}$, $B = \{2\}$, $R = \{(0, 2), (1, 2)\}$, and $X = \{1\}$. Then $R[X] = \{2\}$ and $R^{-1}[\{2\}] = \{0, 1\} \not\subseteq X$. Moreover, the relation R is functional in this example.

Functions and their values We are ready now to give a set-theoretic definition of function. Let A and B be some sets. A binary relation $f \subseteq A \times B$ is called a *function from the set A to the set B* iff f is functional and total (for A). The symbol $f \colon A \to B$ reads: "f is a function from A to B". Clearly, $\operatorname{dom} f = A$ and $\operatorname{rng} f \subseteq B$.

Example 10.8. The relation $f = \{(x, y) \in \mathbb{R}_+ \times \mathbb{R} \mid x = y^2 \land y > 0\}$ is the well-know numeric function from $\mathbb{R}_+$ to $\mathbb{R}$ called "the principal square root" (of a positive number). As the principal square root of a positive number is positive, the statement $f \colon \mathbb{R}_+ \to \mathbb{R}_+$ holds as well.

The relation $\{(1, 2), (2, 2)\}$ is a function from $\{1, 2\}$ to $\{2\}$. In fact, every functional relation is a function from its domain to its range. For each set A, the relation id_A is a function from A to A.

The relation $\sin = \{(x, y) \in \mathbb{R}^2 \mid \text{the sine of } x \text{ equals } y\}$ on the set $\mathbb{R}^2$ is a function from $\mathbb{R}$ to $\mathbb{R}$ and also from $\mathbb{R}$ to the segment $[-1, 1]$. Notice that we have identified the well-known sine function with its *graph* (that is, the sinusoid, the curve of sines)—a part of the plane, hence, a binary relation on $\mathbb{R}$. Here lies the main idea of modeling functions in set theory.

The relation $<$ on the set $\mathbb{N}$ is not a function for it is not functional ($1 < 2$ and $1 < 3$ while $2 \neq 3$). The relation $\tan = \{(x, y) \in \mathbb{R}^2 \mid \text{the tangent of } x \text{ equals } y\}$ is *not* a function from $\mathbb{R}$ to $\mathbb{R}$ since there is no such y that $\tan \frac{\pi}{2} = y$, so this relation is not total for $\mathbb{R}$. Being functional, tan is nevertheless a function from its domain to $\mathbb{R}$. They usually reserve the term *partial function* for such cases but we will not use it.

The main idea of a function is that it "maps" each element of its domain A to a *unique* element of B, so the latter "depends on" or "is determined by" the former solely. More formally, for every $x \in A$, there exists $y \in B$ such that $(x, y) \in f$ (by totality), while such y is unique, i. e., for any z, it follows from $(x, z) \in f$ that $z = y$ (by functionality).

This allows to denote the element y by the symbol $f(x)$ implying its dependence on x. In particular, $f(x) = f(x')$ when $x = x'$. The symbol $f(x)$ reads: "the value of the function f at the element x". Clearly, this notation makes sense iff $x \in \operatorname{dom} f$.

Example 10.9. Is it possible that $f \colon \varnothing \to B$ for some set B?

Suppose this holds. Then $f \subseteq \varnothing \times B = \varnothing$, hence $f = \varnothing$. On the other hand, the empty relation $f = \varnothing$ is, indeed, functional, injective, and total for the set $\varnothing$. Clearly, $\varnothing$ is surjective for B iff $B = \varnothing$. Finally, $\varnothing$ is the only function from $\varnothing$ to B for each B.

Intuitively, a function with a fixed domain is fully determined by its values. In particular, to establish the equality of two such functions, it suffices to check if their respective values are equal at every point. Indeed, the following holds.

Lemma 10.10. *Let $f \colon A \to B$ and $g \colon C \to D$. Then*

$$f = g \iff A = C \ \land\ \forall x \in A\ f(x) = g(x).$$

Proof. Assume that $f = g$. Then $A = \operatorname{dom} f = \operatorname{dom} g = C$. Consider an arbitrary $x \in A = C$. By totality, there exist such $y \in B$ and $z \in C$ that $(x, y) \in f$ and $(x, z) \in g$. By $f = g$, obtain $(x, y), (x, z) \in f$, whence $y = z$ by functionality. Thus, $f(x) = y = z = g(x)$.

For the other direction, suppose that $f(x) = g(x)$ for all $x \in A = C$. Let $(x, y) \in f$. We have $x \in A$ and $f(x) = y$. On the other hand, $x \in C$ and $g(x) = f(x) = y$. Hence $(x, y) \in g$. The opposite inclusion is similar. □

Remark 10.11. As we have already said, to define a function it suffices to describe its domain and value at each element thereof. In practice, this results in the following abbreviation. For example, they write $f\colon \mathbb{N} \to \mathbb{N}$, $f\colon n \mapsto n+1$ instead of

$$f = \{(n, m) \in \mathbb{N}^2 \mid m = n + 1\}.$$

Sometimes, the function defined is still 'anonymous', as in the statement "the function $x \mapsto x^2$ from $\mathbb{R}$ to $\mathbb{R}$ takes only non-negative values". Such abbreviations should be used with care: say, the symbol $a^2 \mapsto a$ does not define a function $\mathbb{R} \to \mathbb{R}$, since it puts both 1 and -1 into correspondence to 1.

Remark 10.12. Obviously, $\{f(a)\} = f[\{a\}]$ if $f\colon A \to B$ and $a \in A$.

> The students may have already defined composition of functions in their calculus course. So, it is important to show them the identity of 'our' composition (when restricted to functions) with that of calculus.

Remark 10.13. Suppose that $f\colon A \to B$ and $g\colon B \to C$. From Lemma 10.5, it follows that the composition $g \circ f$ is a function from A to C. It is easy to see that $(g \circ f)(a) = g(f(a))$ for all $a \in A$.

Indeed, let $b = (g \circ f)(a)$; then $(a, b) \in g \circ f$. Hence, $a\, f\, c$ and $c\, g\, b$ for some c. This implies $c = f(a)$ and $b = g(c) = g(f(a))$, as required.

The equation $(g \circ f)(a) = g(f(a))$ is the main reason for the 'strange' definition of relation composition, where the rightmost relation 'acts' first. The natural alternative would be to denote the value of a function at a as $(a)f$ but this seems way too radical.

Example 10.14. Let $f\colon A \to B$ and $g\colon A \to B$. Then $f \cap g\colon A \to B$ iff $f = g$. That is, the intersection of two functions with the same domain can be a function when trivial only.

If $f = g$, then $f \cap g = f\colon A \to B$. Assume now that $f \cap g\colon A \to B$. Let $(a, b) \in f$. Since $f \cap g$ is total, there exists some $c \in B$ with $(a, c) \in f \cap g$, whence $(a, c) \in f$ and $(a, c) \in g$. As f is functional, we have $b = c$; hence, $(a, b) \in g$. Thus, $f \subseteq g$. The other inclusion is similar.

Exercise 10.15. Consider a similar statement for the union of two functions. Does it hold true?

For arbitrary sets A and B, the set $\{f \in \mathcal{P}(A \times B) \mid f \text{ is a function from } A \text{ to } B\}$ is denoted by B^A. According to Example 10.9, $B^\varnothing = \{\varnothing\}$ (which is the set of exactly *one* element) for any B just like $n^0 = 1$ in the case of numerical exponentiation. More parallels between the set B^A and numerical powers will be drawn later.

Bijections and friends. Now, let us consider some special classes of functions. A function $f\colon A \to B$ is an *injection from A to B* if f is injective. Likewise, if f is surjective (for B), it is called a *surjection from A to B*. Finally, if the function $f\colon A \to B$ is both injective and surjective, it is called a *bijection from A to B* (or a *one-to-one correspondence between A and B*, or a *bijective* function from A to B).

Example 10.16. The relation id_A is a bijection from A to A for any set A.

Example 10.17. The function $x \mapsto e^x$ is an injection $\mathbb{R} \to \mathbb{R}$ and a bijection $\mathbb{R} \to \mathbb{R}_+$. The function $x \mapsto x^2$ is a surjection $\mathbb{R} \to (\mathbb{R}_+ \cup \{0\})$, but not an injection. Likewise, the function $(n, m) \mapsto n + m$ is a surjection $\mathbb{N}^2 \to \mathbb{N}$, but not an injection.

Remark 10.18. Given a function $f\colon A \to B$, f is injective iff it follows that $x = y$ from $f(x) = f(y)$, for any x and y. Also, f is surjective (for B) iff for every $y \in B$, there exists such x that $f(x) = y$.

| This point may be not that obvious for some students. It makes sense to give a detailed proof.

Exercise 10.19. Let $f\colon A \to B$ and $g\colon B \to C$. Prove that f is an injection if $g \circ f$ is, and g is a surjection when $g \circ f$ is such.

Lemma 10.20. *A relation $R \subseteq A \times B$ is a bijection from A to B iff R^{-1} is a bijection from B to A.*

Proof. By Lemma 10.4. □

Exercise 10.21. From $f\colon A \to B$ and $f^{-1}\colon B \to A$ derive that f is a bijection from A to B.

Sometimes, it proves convenient to describe functionality, injectivity, etc. in terms of operations on binary relations.

Lemma 10.22. *Let $R \subseteq A \times B$. Then*

1. *R is functional $\iff$ $R \circ R^{-1} \subseteq \mathrm{id}_B$;*

2. *R is injective $\iff$ $R^{-1} \circ R \subseteq \mathrm{id}_A$;*

3. *R is surjective $\iff$ $\mathrm{id}_B \subseteq R \circ R^{-1}$;*

4. *R is total $\iff$ $\mathrm{id}_A \subseteq R^{-1} \circ R$.*

Proof. Suppose that R is functional and $(x, y) \in R \circ R^{-1}$. Then $x, y \in B$, $xR^{-1}z$, and zRy for some z. By functionality, zRx and zRy imply $x = y$, whence $(x, y) \in \mathrm{id}_B$. For the other direction, assume the inclusion $R \circ R^{-1} \subseteq \mathrm{id}_B$ and zRx and zRy for arbitrary x, y, z. Clearly, $xR^{-1}z$, whence $(x, y) \in R \circ R^{-1} \subseteq \mathrm{id}_B$; so, $x = y$.

For injectivity, we shall apply the first claim to the relation R^{-1} (notice changing A to B and vice versa). Clearly,

$$R \text{ is injective} \iff R^{-1} \text{ is functional} \iff R^{-1} \circ (R^{-1})^{-1} \subseteq \mathrm{id}_A \iff R^{-1} \circ R \subseteq \mathrm{id}_A.$$

Now, let R be surjective (for B). Consider an arbitrary $x \in B$. By surjectivity, there exists y such that yRx, whence $xR^{-1}y$. These two imply $(x, x) \in R \circ R^{-1}$. For the other direction, assume $\mathrm{id}_B \subseteq R \circ R^{-1}$ and consider an arbitrary $x \in B$. Then $(x, x) \in R \circ R^{-1}$; so there exists y such that $xR^{-1}y$ and yRx. Each of these two statements implies surjectivity of R.

For totality, we have

$$R \text{ is total for } A \iff R^{-1} \text{ is surjective for } A \iff \mathrm{id}_A \subseteq R^{-1} \circ (R^{-1})^{-1} \iff \mathrm{id}_A \subseteq R^{-1} \circ R$$

by applying the previous claim to R^{-1} (and changing B to A, of course). □

Corollary 10.23. *The relation $R \subseteq A \times B$ is a bijection from A to B iff*

$$R^{-1} \circ R = \mathrm{id}_A \quad \textit{and} \quad R \circ R^{-1} = \mathrm{id}_B.$$

In particular, we see that for a *bijection* on a set A, the converse relation is indeed the *inverse* w. r. t. composition and id_A as the unity. The set of bijections on a given set is thus similar to the set of (say) rational non-zero numbers, where for every x one has $\frac{1}{x}$ such that $x \cdot \frac{1}{x} = 1 = \frac{1}{x} \cdot x$.

11 Equivalent Sets

The main goal of this section is to define set equivalence and embedding formally. The Instructor should try to persuade the students that these notions are natural models for comparing sets by their 'sizes'. Here we present some basic facts (including Cantor—Schröder—Bernstein Theorem) applicable to both the finite and infinite cases.

Bijections are of utmost importance in mathematics since they allow to formalize the intuitive notion of the 'number of elements' in a set.

Our intuition does not serve us well when the set we consider is infinite. Indeed, not every natural number is a square of a natural (like $9 = 3^2$ or $25 = 5^2$ are). So the set of squares is a proper subset of $\mathbb{N}$. Intuitively, 'a part is less than the whole', hence the 'number' (amount, quantity) of squares should be less than that of all natural numbers. On the other hand, each natural number has its 'own' square, which differs from the square of any other natural. That is, there are at least as many squares as there are natural numbers. Thus, the 'number' of squares is greater or equal to that of all naturals.

This is known as *Galileo's Paradox.* The first step to tame our paradox-bearing intuition is to avoid mentioning a 'number of elements' as some distinct entity[10]. Rather than that, we will say that *two sets have the same number of elements* or *one set has no more elements than the other.*

Formally, we say that sets A and B are *equivalent* (or *equinumerous*) iff there exists a bijection from A to B. Then we write $A \sim B$ or $A \overset{f}{\sim} B$ if f is such a bijection.

Intuitively, equivalent sets must have the same number of elements, must be of the same 'size'. How can one show that this *formal* definition is adequate to the *intuitive* idea? (At least, any 'number' is absent from the former, which clearly violates our intuition...) Again, we shall do that by *proving* that equivalent sets 'behave' the same way we would expect from "sets of the same size". This sounds much like the 'axiomatic method' we have employed for building a theory of sets.

Of course, the formal notion is not the same as the intuitive one. Indeed, it is easy to see that $\mathbb{N}$ is equivalent to the set of squares from Galileo's Paradox. Hence, a proper part is not necessarily less than the whole when infinite sets are involved. This fact is not very intuitive; one the other hand, we have got rid of the paradox!

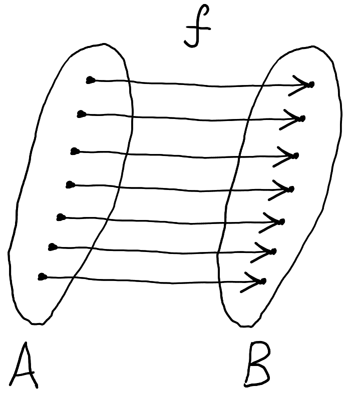

Figure 10: Equivalent sets: $A \overset{f}{\sim} B$.

[10] Although, this can be redefined rigorously at some later stage of set-theoretic developments.

Clearly, A must be of the same size as A; if A has as many elements as B, then B must have the same number of elements as A does, etc. These intuitive expectations are indeed met by set equivalence.

Lemma 11.1. *For any sets A, B, C,*

1. $A \sim A$;

2. *if* $A \sim B$, *then* $B \sim A$;

3. *if* $A \sim B$ *and* $B \sim C$, *then* $A \sim C$.

Proof. Clearly, $A \overset{\mathrm{id}_A}{\sim} A$. If $A \overset{f}{\sim} B$, then $B \overset{f^{-1}}{\sim} A$ by Lemma 10.20. If $A \overset{f}{\sim} B$ and $B \overset{g}{\sim} C$, then $A \overset{g\circ f}{\sim} C$ by Lemma 10.5. □

Remark 11.2. Let $f\colon A \to B$ be an injection. Then $A \overset{f}{\sim} f[A]$ and, in general, $X \sim f[X]$ for each $X \subseteq A$.

Example 11.3. Let us show that $\mathbb{N}^2 \sim \mathbb{N}$ by constructing an appropriate bijection. This fact is quite important for computer science since it makes possible to encode a finite sequence of numbers (e.g., a program code) with one natural number.

Define a function f such that $f(m,n) = 2^m(2n+1) - 1$ for all $(m,n) \in \mathbb{N}^2$. If $f(m,n) = f(m',n')$, then $2^m(2n+1) = 2^{m'}(2n'+1)$. Suppose that $m \neq m'$. W. l. o. g.[11], assume $m < m'$. Then $2n+1 = 2^{m'-m}(2n'+1)$. Here the left-hand side is odd, whereas the right-hand side is even. By contradiction, $m = m'$. Hence, $2n+1 = 2n'+1$ and $n = n'$. Thus, f is an injection.

Now, let us check if f is surjective. Assume that there is some *positive* natural number which *cannot* be presented in the form $2^m(2n+1)$. Consider the *least* such number k. The number k is even, for otherwise it would be of the form $2^0(2n+1)$. So, $k = 2k'$ for suitable $k' < k$. By our choice of k, it holds that $k' = 2^{m'}(2n'+1)$ for some $m', n' \in \mathbb{N}$. But then $k = 2^{m'+1}(2n'+1)$. A contradiction. Thus, every positive natural is of the form $f(m,n)+1$. Hence every natural equals $f(m,n)$ for some numbers $m, n \in \mathbb{N}$.

It is clear that there exist non-equivalent sets. For example, $\varnothing \nsim \{\varnothing\}$ as the only function $\varnothing$ from $\varnothing$ to $\{\varnothing\}$ is not surjective. Actually, for every set A there exists a non-equivalent set $\mathcal{P}(A)$, which is 'strictly greater' in 'number of elements' as we shall see a little bit later.

Theorem 11.4 (Cantor)**.** *For every set A, $A \nsim \mathcal{P}(A)$.*

Proof. Assume the contrary and let φ be a bijection from A to $\mathcal{P}(A)$. Consider the set

$$X = \{a \in A \mid a \notin \varphi(a)\}.$$

Since $X \in \mathcal{P}(A)$ and φ is surjective, there must be some $a_0 \in A$ such that $\varphi(a_0) = X$. If $a_0 \in X$, then $a_0 \notin \varphi(a_0) = X$. The contradiction shows that $a_0 \notin X = \varphi(a_0)$, which implies $a_0 \in X$ by the definition of X. So, our first assumption must be false: no such bijection φ is possible. □

Remark 11.5. We have proved more indeed: there is no surjection from $\mathcal{P}(A)$ to A for any set A.

[11] *Without loss of generality*, that is, the cases remaining are similar to the cases being considered.

As we have already seen, $A \times B \neq B \times A$ generally. But if one replaces $=$ with $\sim$, Cartesian product behaves much like the familiar numerical product; moreover, the operation B^A becomes similar to numerical exponentiation.

> The following theorem can spare a lot of effort when proving set equivalence as Example 11.21 shows. So, it is recommended to highlight this result and make the students remember its statement. It depends on the audience whether it is worth a detailed proof.

Theorem 11.6. *For any sets A, B, C,*

1. *if $A \sim B$, then $A \times C \sim B \times C$, $A^C \sim B^C$ and $C^A \sim C^B$;*
2. *$A \times B \sim B \times A$;*
3. *$(A \times B) \times C \sim A \times (B \times C)$;*
4. *$(A \times B)^C \sim A^C \times B^C$;*
5. *$(C^B)^A \sim C^{A \times B}$.*

Proof sketch. We omit the proof, but give some directions. For the first statement, given that $A \overset{\varphi}{\sim} B$, one can consider the bijections $(a, c) \mapsto (\varphi(a), c)$ from $A \times C$ to $B \times C$, $f \mapsto \varphi \circ f$ from A^C to B^C, and $f \mapsto f \circ \varphi^{-1}$ from C^A to C^B. The bijections $(a, b) \mapsto (b, a)$ and $((a, b), c) \mapsto (a, (b, c))$ certify the second and third equivalences, respectively. For the fourth statement, apply the bijection $f \mapsto (\pi_1 \circ f, \pi_2 \circ f)$, where the functions $\pi_1 \colon (a, b) \mapsto a$ and $\pi_2 \colon (a, b) \mapsto b$ are known as *projectors*.

The fifth statement is the trickiest one. If $f \in (C^B)^A$, then f is a function which returns a function $f(a)$ from B to C given an element $a \in A$. One can map f to the function $g_f \colon A \times B \to C$ such that $g_f(a, b) = (f(a))(b)$. This mapping $f \mapsto g_f$ is the required bijection.

For injectivity, we assume that $g_f = g_{f'}$ and, for the sake of contradiction, $f \neq f'$. Then there exists a point a such that $f(a) \neq f'(a)$ (by Lemma 10.10) and, in its turn, there is such b that $(f(a))(b) \neq (f'(a))(b)$. By definition, $g_f(a, b) \neq g_{f'}(a, b)$, whence $g_f \neq g_{f'}$. For surjectivity, consider an arbitrary function $h \colon A \times B \to C$. We need to find such a function f that $g_f = h$. Let us put $f(a) = (b' \mapsto h(a, b'))$. Clearly, $f(a) \in C^B$ for every $a \in A$, so $f \in (C^B)^A$. Finally, $g_f(a, b) = (b' \mapsto h(a, b'))(b) = h(a, b)$ for all a, b, whence $g_f = h$.

The fifth statement establishes equivalence between the set of function-valued functions and the set functions of two arguments (i. e., one from A, the other from B). Thus, in principle, functions of two (or more) arguments are not necessary as they can be encoded by functions of just one argument. This idea, known as *currying*[12], is important for *functional* programming paradigm.

Let us give an example of currying. Let $+ \colon \mathbb{N}^2 \to \mathbb{N}$ be the usual numeric addition, which clearly takes two natural arguments. For each $k \in \mathbb{N}$, consider the function $f_k \colon \mathbb{N} \to \mathbb{N}$ such that $f_k(n) = k + n$ for all $n \in \mathbb{N}$ (in other words, $f_k = (n \mapsto k + n)$; in particular, $f_0 = \mathrm{id}_{\mathbb{N}}$). Now, define the function $f \colon \mathbb{N} \to \mathbb{N}^{\mathbb{N}}$ so that $f(n) = f_n$. Then $n + m = f_n(m) = (f(n))(m)$. There are no two-argument functions in the right-hand side. □

Indicator function. For the further discussion, we need some 'exemplary' finite sets. These are the sets

$$\underline{n} = \{k \in \mathbb{N} \mid k < n\}$$

[12] After Haskell B. Curry, a logic pioneer.

for all possible $n \in \mathbb{N}$. Clearly, $\underline{0} = \varnothing$, $\underline{1} = \{0\}$, and $\underline{2} = \{0, 1\}$. In general, $\underline{n+1} = \underline{n} \cup \{n\}$. It is intuitively clear that the set $\underline{n}$ has exactly n elements; in fact, we shall later *use* these sets to *define* the property of having exactly n elements.

Remark 11.7. From the standard formal definition of the set $\mathbb{N}$, which is omitted[13] from this Course, it follows that $0 = \varnothing$, $1 = \{0\} = \{\varnothing\}$, $2 = \{0, 1\} = \{\varnothing, \{\varnothing\}\}$, and $n = \underline{n}$ for every $n \in \mathbb{N}$ in general.

> We usually ask students not to use the inner structure of naturals in any argument since this has not been "officially" defined in our Course.

Let X be a set and $A \subseteq X$. Define the function $\mathbf{1}_A \colon X \to \underline{2}$ by the equation:

$$\mathbf{1}_A(x) \quad = \quad \begin{cases} 1, & \text{if } x \in A; \\ 0, & \text{if } x \notin A. \end{cases}$$

The function $\mathbf{1}_A$ is called the *indicator* (or *characteristic*) function of the set A. This function relates to A very closely as it discriminates elements thereof from non-elements, thus 'containing' the same 'information' as A itself. This fact might be formalized as follows.

Lemma 11.8. *For every sets $A, B \subseteq X$, $A = B$ iff $\mathbf{1}_A = \mathbf{1}_B$.*

Proof. The implication from the left to the right is obvious. For the other direction, suppose that $\mathbf{1}_A = \mathbf{1}_B$ and $x \in A$. Then $\mathbf{1}_A(x) = 1$, whence $\mathbf{1}_B(x) = 1$, which means $x \in B$. Thus, $A \subseteq B$. The converse inclusion is similar. □

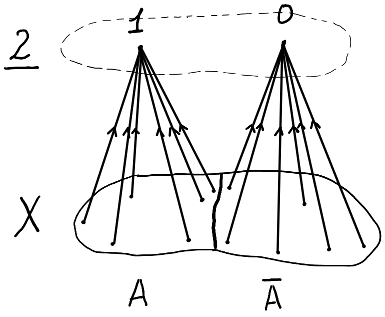

Figure 11: The indicator $\mathbf{1}_A$ of a set A.

Consider the set $\underline{2}^X$ comprising all possible functions from X to $\underline{2} = \{0, 1\}$. In fact, all these functions are indicators for certain subsets of X.

Lemma 11.9. *For every set X, $\mathcal{P}(X) \sim \underline{2}^X$.*

[13] Otherwise, we would have to introduce a new axiom and develop some 'recursive' techniques, which would comprise a huge detour from the main line of our Course.

Proof. Let a function $\varphi\colon \mathcal{P}(X) \to \underline{2}^X$ be such that $\varphi(A) = \mathbf{1}_A$ for each $A \in \mathcal{P}(X)$. By Lemma 11.8, from $\varphi(A) = \varphi(B)$, it follows $A = B$. So, φ is an injection. For an arbitrary function $g \in \underline{2}^X$, consider the set $A = g^{-1}[\{1\}] = \{a \in X \mid g(a) = 1\}$. Clearly, $g(x) = 1$ if $x \in A$, and $g(x) = 0$ otherwise. Hence, $g = \mathbf{1}_A = \varphi(A)$, and φ is a surjection. □

This explains a widespread notation 2^X for the power-set $\mathcal{P}(X)$.

In practice, indicator functions are used to replace complicated arguments employing set-theoretic operations with numerical calculations.

Exercise 11.10. Prove that for any sets $B, C \in \mathcal{P}(X)$ and $x \in X$, the following hold:

$$\begin{aligned}
\mathbf{1}_{B\cap C}(x) &= \mathbf{1}_B(x) \cdot \mathbf{1}_C(x);\\
\mathbf{1}_{B\cup C}(x) &= \mathbf{1}_B(x) + \mathbf{1}_C(x) - \mathbf{1}_B(x) \cdot \mathbf{1}_C(x);\\
\mathbf{1}_{\bar{B}}(x) &= 1 - \mathbf{1}_B(x),
\end{aligned}$$

and $B \subseteq C$ is equivalent to $\mathbf{1}_B(x) \leq \mathbf{1}_C(x)$ for each $x \in X$.

> In practice, we omit indicators' argument for it typically does not change throughout such proofs.

Example 11.11. Putting $X = B \cup C$, one can easily prove that $\bar{B} \cap \bar{C} = \overline{B \cup C}$. Indeed, for each $x \in X$ one has

$$\mathbf{1}_{\bar{B}\cap\bar{C}}(x) = (1 - \mathbf{1}_B(x))(1 - \mathbf{1}_C(x)) = 1 - (\mathbf{1}_B(x) + \mathbf{1}_C(x) - \mathbf{1}_B(x)\mathbf{1}_C(x)) = \mathbf{1}_{\overline{B\cup C}}(x).$$

Example 11.12. Let us infer $B = C$ from $B \cap C = B \cup C$. By the assumption, for each $x \in X$ we have

$$\begin{aligned}
0 = \mathbf{1}_{B\cup C}(x) - \mathbf{1}_{B\cap C}(x) = \mathbf{1}_B(x) + \mathbf{1}_C(x) - 2 \cdot \mathbf{1}_B(x)\mathbf{1}_C(x) = \\
\mathbf{1}_B^2(x) + \mathbf{1}_C^2(x) - 2 \cdot \mathbf{1}_B(x)\mathbf{1}_C(x) = (\mathbf{1}_B(x) - \mathbf{1}_C(x))^2.
\end{aligned}$$

Hence $\mathbf{1}_B(x) = \mathbf{1}_C(x)$ for each $x \in X$. Therefore, $B = C$.

Tuples and functions. Previously, we introduced n-tuples of elements of a set A as elements of the set A^n. For example, $\underline{2}^3$ is the set of all possible triplets (3-tuples) formed by elements of the set $\underline{2}$. Clearly,

$$\underline{2}^3 = \{(0,0,0),\ (0,0,1),\ (0,1,0),\ (0,1,1),\ (1,0,0),\ (1,0,1),\ (1,1,0),\ (1,1,1)\}.$$

On the other hand, let us consider the set $\underline{2}^{\underline{3}}$. By definition, this is the set of all possible functions from $\underline{3}$ to $\underline{2}$. One such function f is defined by the equations $f(0) = 1$, $f(1) = 0$, $f(2) = 1$. Taking into account that each function is a set, one thus has

$$f = \{(0,1), (1,0), (2,1)\}.$$

Likewise, the set of all functions of this form is just

$$\begin{aligned}
\underline{2}^{\underline{3}} = \{\ & \{(0,0),(1,0),(2,0)\},\ \{(0,0),(1,0),(2,1)\},\\
& \{(0,0),(1,1),(2,0)\},\ \{(0,0),(1,1),(2,1)\},\\
& \{(0,1),(1,0),(2,0)\},\ \{(0,1),(1,0),(2,1)\},\\
& \{(0,1),(1,1),(2,0)\},\ \{(0,1),(1,1),(2,1)\}\\
\}.&
\end{aligned}$$

It is easy to see that if one takes the *second* coordinate from each pair in one function and order them as the respective *first* coordinate increases, he obtains a triplet from $\underline{2}^3$. In particular, the function f results in the triplet $(1,0,1) = (f(0), f(1), f(2))$. Conversely, each triplet (a_1, a_2, a_3) gives rise to a function $\{(0,a_1),(1,a_2),(2,a_2)\}$. In fact, such a correspondence is bijective and far from accidental.

> This easy theorem proves handy when formalizing combinatorial arguments; it is thus unwise to skip it.

Theorem 11.13. *For every set A and each $n \in \mathbb{N}$, $A^{\underline{n}} \sim A^n$.*

Proof sketch. If $n = 0$, one clearly gets $A^0 = \{\varnothing\}$ by definition and $A^{\underline{0}} = A^{\varnothing} = \{\varnothing\}$ by Example 10.9. If $n = 1$, $A^1 = A$ by the same definition. In this case, $A^{\underline{1}} = A^{\{0\}} = \{\{(0,a)\} \mid a \in A\}$ since any function from $\{0\}$ to A contains exactly one pair $(0,a)$ for some $a \in A$. Clearly, the function $\pi_2 \colon (0,a) \mapsto a$ is a bijection from $A^{\underline{1}}$ to A whereas $A = A^1$.

Let $n \geq 2$. Then $\underline{n} = \{0, \ldots, n-1\}$. We are to define a bijection $\varphi \colon A^{\underline{n}} \to A^n$. For each $f \in A^{\underline{n}}$, put $\varphi(f) = (f(0), \ldots, f(n-1))$. As $f \colon \underline{n} \to A$, it is clear that $\varphi(f) \in A^n$.

The function φ is injective. Indeed, suppose that $\varphi(f) = (f(0), \ldots, f(n-1)) = (g(0), \ldots, g(n-1)) = \varphi(g)$. Then $f(k) = g(k)$ for each $k \in \underline{n}$ by Lemma 4.35, which implies $f = g$ by Lemma 10.10.

Let us see that the function φ is surjective as well. Suppose that $(a_0, \ldots, a_{n-1}) \in A^n$ and define a function $f \colon \underline{n} \to A$ by the equations $f(k) = a_k$ for each $k \in \underline{n}$. Then $\varphi(f) = (f(0), \ldots, f(a_{n-1})) = (a_0, \ldots, a_{n-1})$.[14] □

This way, we can see that n-tuples and functions from $\underline{n}$ (also know as *finite sequences of length n*) are 'almost the same' for there is a natural bijection between the two. In mathematical practice, tuples and finite sequences are routinely identified (i. e., they view a sequence as a tuple and vice versa, usually without any comment).

Embeddings. As we know, the intended meaning of $A \sim B$ is just that: 'A has the same number of elements as B'—and this is supposed to work for finite and infinite sets alike. Let us see what would be the mathematician's interpretation for 'A has no more elements than B'.

By definition, a set A *embeddable* into a set B iff there is an injection $f \colon A \to B$. The function f is called an *embedding* in this case. We write $A \overset{f}{\lesssim} B$ when f is an embedding of A into B, and $A \lesssim B$ when such an embedding exists.

Clearly, the ends of f-arrows form a 'copy' of A in B since any two arrows sharing their end or their beginning must coincide. So, it should be intuitively plausible that B has no less elements than A does. But how can one make sure that the formal notion reflects the intuitive one adequately? As with set equivalence, we are to *prove* that embedding behaves as one could expect intuitively.

Example 11.14. If $A \subseteq B$, then $A \overset{\mathrm{id}_A}{\lesssim} B$. Let $2\mathbb{N}$ be the set of all even naturals. Then $\mathbb{N} \lesssim 2\mathbb{N}$ and, surely, $2\mathbb{N} \lesssim \mathbb{N}$. Observe that $\mathbb{N} \neq 2\mathbb{N}$ yet $\mathbb{N} \sim 2\mathbb{N}$.

Lemma 11.15. *For any sets A, B, C the following hold:*

1. *$A \lesssim A$;*

2. *if $A \lesssim B$ and $B \lesssim C$, then $A \lesssim C$;*

3. *if $A \sim B$, then $A \lesssim B$ and $B \lesssim A$;*

[14] This is just a sketch as a rigorous definition for the tuple $(f(0), \ldots, f(n-1))$ would require some recursion.

4. $A \lesssim B \iff \exists D \subseteq B\ A \sim D.$

Proof. For the second statement, recall that the composition of two injections is an injection by Lemma 10.5. For the third one, we know that both f and f^{-1} are injections given f is a bijection.

In view of Remark 11.2, it suffices to take $f[A]$ as D for the last statement, where f is an arbitrary injection $f\colon A \to B$. □

The intuitive validity of the first three statements is clear: say, if A and B has the same number of elements, none of the sets has more elements than the other one. The last property is also very natural: A has no more elements than B iff A is equinumerous with a *part* (i. e., a subset) of B.

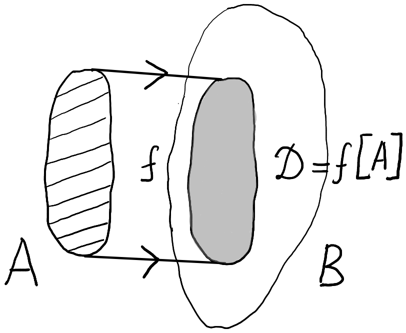

Figure 12: Proving Claim 4 of Lemma 11.15.

> Clearly, it depends on the calculus course taught which definition of $\mathbb{R}$ they use (if any). An informal discussion of "infinite binary fractions" might be helpful here. I recommend to stress the point that there exist non-equivalent infinite sets. Some students manage to ignore this fact even after having studied many mathematical courses.

Remark 11.16. It is well-known from Calculus courses that $\mathbb{R} \sim \mathcal{P}(\mathbb{N})$. The set $\mathcal{P}(\mathbb{N})$ is called the *continuum* for being equivalent to the 'continuous' real line. By Theorem 11.4, $\mathbb{N} \nsim \mathbb{R}$, that is, the 'infinite numbers of elements' for $\mathbb{N}$ and $\mathbb{R}$ cannot be the same; not all 'infinities' are thus 'equal' (equivalent).

The famous *Continuum Hypothesis* (CH) says that from $\mathbb{N} \lesssim X \lesssim \mathcal{P}(\mathbb{N})$, it follows either $X \sim \mathbb{N}$ or $X \sim \mathcal{P}(\mathbb{N})$. In other words, there is no 'number of elements' strictly between $\mathbb{N}$ and the continuum. It is proved that neither CH nor its negation follows from the set-theoretic axioms usually assumed (provided those axioms are consistent[15]). Hence, either statement is consistent with the axioms.

So, the question about 'intermediate sets' is fundamentally undecidable if one sticks to the well-tested, widely accepted, and intuitively valid axioms. It is an open philosophical question whether CH is intuitively valid.

In practice, it is not thus justified to assume either CH or its negation.

A set X is called *countable* if $X \sim \mathbb{N}$. If an infinite set is not countable, it is then *uncountable*. Thus, the set $\mathbb{R}$ is uncountable.

[15] It is unprovable that the axioms are consistent given (1) the axioms are indeed consistent; (2) each proof must be based on the axioms themselves.

Interestingly, it is enough to have two 'unrelated' injections from A to B and from B to A for these two sets to be equivalent. On the other hand, it is very natural that A and B has the same number of elements when neither set has more elements than the other one. Nevertheless, all known proofs of this theorem are somewhat tricky and employ a 'limit' construction.

Theorem 11.17 (Cantor—Schröder—Bernstein)**.** *For any sets A and B, if $A \lesssim B$ and $B \lesssim A$, then $A \sim B$.*

The proof is omitted for being technically demanding.

Corollary 11.18. *For no set A is an embedding $\mathcal{P}(A) \lesssim A$ possible.*[16]

Proof. Clearly, $A \overset{f}{\lesssim} \mathcal{P}(A)$, where $f\colon x \mapsto \{x\}$ for every $x \in A$. By Theorem 11.4, it is not the case that $A \sim \mathcal{P}(A)$. Hence, $\mathcal{P}(A) \not\lesssim A$. □

By $A \lnsim B$, we shall denote the situation when $A \lesssim B$ but $A \nsim B$. This is the formal way to say that A has *fewer* elements than B does. By Theorem 11.17, $A \lnsim B$ iff $A \lesssim B$ but $B \not\lesssim A$. In particular, $A \lnsim \mathcal{P}(A)$; so, for every set A, the set $\mathcal{P}(A)$ is 'strictly greater'. There is no 'greatest set' in existence.

Cantor—Schröder—Bernstein Theorem is very practical when a set equivalence is to be proved. Usually, it is *much* easier to construct two injections instead of an explicit bijection.

Example 11.19. Clearly, $\mathbb{N} \lesssim \mathbb{Q}$. On the other hand, $\mathbb{Q} \lesssim \mathbb{N}^3$. For, indeed, each *positive* rational q is uniquely expressible as an *irreducible* fraction[17] $\frac{m}{n}$, where $m, n \in \mathbb{N}$; the function $f\colon q \mapsto (m, n, 0), 0 \mapsto (0, 1, 0), -q \mapsto (m, n, 1)$ is then a required injecton.

Further, one obtains $\mathbb{N}^3 = \mathbb{N}^2 \times \mathbb{N} \sim \mathbb{N} \times \mathbb{N} \sim \mathbb{N}$ by Theorem 11.6 and Example 11.3. Hence $\mathbb{Q} \lesssim \mathbb{N}$. Applying Theorem 11.17, one finally gets $\mathbb{Q} \sim \mathbb{N}$, that is, the set of rationals $\mathbb{Q}$ is countable. Since $\mathbb{N} \lesssim \mathbb{Z} \lesssim \mathbb{Q} \lesssim \mathbb{N}$, one can observe that the set $\mathbb{Z}$ of integers is countable as well.

Exercise 11.20. Prove that $\underline{2}^{\mathbb{N}} \sim \underline{3}^{\mathbb{N}}$.

Example 11.21. As we know, $\mathbb{R} \sim \mathcal{P}(\mathbb{N})$. By Lemma 11.9, this implies $\mathbb{R} \sim \underline{2}^{\mathbb{N}}$, whence

$$\mathbb{R} \sim \mathbb{R} \times \{0\} \lesssim \mathbb{R} \times \mathbb{R} \sim \underline{2}^{\mathbb{N}} \times \underline{2}^{\mathbb{N}} \sim (\underline{2} \times \underline{2})^{\mathbb{N}} \sim$$
$$\underline{4}^{\mathbb{N}} \lesssim \mathbb{N}^{\mathbb{N}} \lesssim \mathbb{R}^{\mathbb{N}} \sim (\underline{2}^{\mathbb{N}})^{\mathbb{N}} \sim \underline{2}^{\mathbb{N} \times \mathbb{N}} \sim \underline{2}^{\mathbb{N}} \sim \mathbb{R},$$

with the help of Theorem 11.6 and Example 11.3. We conclude that $\mathbb{R}^2 \sim \mathbb{N}^{\mathbb{N}} \sim \mathbb{R}^{\mathbb{N}} \sim \mathbb{R}$ applying Theorem 11.17. The statement $\mathbb{R}^2 \sim \mathbb{R}$ means that the line has 'as many' points as the plane. In the 1870s, G. Cantor remarked on a similar fact: "I see it, but I do not believe it."

Example 11.22. The set C of all possible circles in the plane is equivalent to $\mathbb{R}$.

First, we construct an injection $\mathbb{R} \to C$. We map each number $x \in \mathbb{R}$ to the circle with center $(x, 0)$ and of radius 1. Clearly, this is a well-defined function (total for $\mathbb{R}$ and functional) and an injection indeed as two circles with distinct centers are not equal.

An injection from C to $\mathbb{R}$ is harder. It is natural to map each circle to the triplet $(x, y, r) \in \mathbb{R}^3$, where (x, y) is the center and r is the radius. This is a well-defined function for each circle has exactly one center and one radius. Injectivity is clear: if both the centers and the radii coincide, so do the circles. Thus, $C \lesssim \mathbb{R}^3$. Please notice that this injection is not surjective since radii must be positive. Now, it suffices to observe that $\mathbb{R}^3 = \mathbb{R}^2 \times \mathbb{R} \sim \mathbb{R} \times \mathbb{R} \sim \mathbb{R}$, whence $C \lesssim \mathbb{R}$.

From $\mathbb{R} \lesssim C$ and $C \lesssim \mathbb{R}$, follows that $\mathbb{R} \sim C$ by Theorem 11.17.

[16] This fact can be proved quite easily without any reference to Theorem 11.17.

[17] That is, one whose numerator and denominator are coprime.

Example 11.23. Let X be some set of pairwise disjoint discs[18] in the plane. Prove that $X \lesssim \mathbb{N}$. So, we are to prove here that it is impossible to choose more than countably many pairwise disjoint discs 'simultaneously'.

The main idea is to set a 'mark' upon each disc so that (1) distinct discs have distinct marks; (2) marks come from a countable set. To implement this plan, let us recall the following fact from Calculus: for every numbers $a, b \in \mathbb{R}$, if $a < b$, then there exists a *rational* number $q \in \mathbb{Q}$ such that $a < q < b$.

Consider an arbitrary disc $D \in X$ with center (x, y) and of radius r. Let S_D be the square with vertices $(x - \frac{r}{2}, y - \frac{r}{2})$, $(x - \frac{r}{2}, y + \frac{r}{2})$, $(x + \frac{r}{2}, y + \frac{r}{2})$, $(x + \frac{r}{2}, y - \frac{r}{2})$. It is easy to see that all points from S_D lie inside the disc D.

Let q and q' be rational numbers with $x - \frac{r}{2} < q < x + \frac{r}{2}$ and $y - \frac{r}{2} < q' < y + \frac{r}{2}$. Then the point $(q, q') \in \mathbb{Q}^2$ belongs to D. It is quite believable that there exists a function f mapping each disc from X to some point from $\mathbb{Q}^2$ which belongs to that disc.[19] As any two distinct discs are disjoint, one gets $f(D) \neq f(D')$ when $D \neq D'$. Hence, $X \overset{f}{\lesssim} \mathbb{Q}^2$. As $\mathbb{Q}^2 = \mathbb{Q} \times \mathbb{Q} \sim \mathbb{N} \times \mathbb{N} \sim \mathbb{N}$, we have proved that $X \lesssim \mathbb{N}$.

So far, we have demonstrated that the formal notions of set equivalence and embedding enjoy many intuitively valid properties of their informal counterparts. For other properties, we have not yet check it: e. g., for every two sets A and B, either one must have more elements than the other or they must be of the same size (intuitively). Is it true that for every two sets A and B, either $A \lesssim B$ or $B \lesssim A$ holds? Assuming one more axiom (the Axiom of Choice), one can prove this claim. Nevertheless, these statements are beyond the scope of our Course. In fact, one rarely considers *arbitrary infinite sets* in "discrete mathematics", while the Axiom of Choice is not that urgently needed otherwise.

On the other hand, some facts we have established for set equivalence and embedding are counter-intuitive. Say, that the plane $\mathbb{R}^2$ is equinumerous with the line $\mathbb{R}$. Thus, the formal notions live their own lives and cannot be judged by their informal analogues.

[18] A *disc* is the region of the plane bounded by a circle and containing the center of that circle. We assume here that all discs in X are non-degenerate, i. e. of *positive* radii.

[19] In fact, no form of the Axiom of Choice is needed for such a function f to exist. As $\mathbb{Q}^2 \sim \mathbb{N}$, each rational point (q, q') from D corresponds to a natural number. One can just map D to such a point with the *least* possible number.

12 Counting: the Basics

This section contains a (reasonably) rigorous presentation of combinatorics' foundations including the Pigeonhole Principle, the Rules of Sum and Product. We try to prove as much as possible when avoiding explicit recursion and the Axiom of Choice.

Our next goal is to see how it is possible to construct a coherent theory of *finite* sets and, in particular, to *define* the 'number of elements' of such a set. To this end, we will use the sets $\underline{n} = \{k \in \mathbb{N} \mid k < n\}$ as 'exemplary' finite sets. Namely, a set A is called *finite* iff $A \sim \underline{n}$ for some $n \in \mathbb{N}$. Otherwise, the set A is called *infinite*.

Example 12.1. The sets $\varnothing \sim \underline{0}$, $\{\varnothing\} \sim \underline{1}$, and $\{\varnothing, \{\varnothing\}\} \sim \underline{2}$ are finite. The set $A = \{x, y, z\}$ is finite. Depending on which of the elements x, y, z are identical, we have either $A \sim \underline{1}$ or $A \sim \underline{2}$ or $A \sim \underline{3}$. Say, if $x = y \neq z$, then $A \sim \underline{2}$.

We have assumed as known that $0 \neq 1$. Therefore $\underline{2} = \{0, 1\} \neq \{0\} = \underline{1}$. But is it the case that $\underline{2} \sim \underline{1}$? Were it so, our definition of finite set would not be adequate to the 'number of elements' intuition as $2 \neq 1$.

Fortunately, this case is easy: if $\{0, 1\} \overset{\varphi}{\precsim} \{0\}$, then $\varphi(0) = 0 = \varphi(1)$, which implies $0 = 1$ by injectivity, which is false. But what if $\underline{n} \sim \underline{m}$ holds for some other distinct numbers n and m? We need induction in order to exclude such a possibility.

Lemma 12.2. *For each $n \in \mathbb{N}$, if $f \colon \underline{n+1} \to \underline{n}$, then f is* not *injective.*

Proof. Assume the contrary. So, there is such a number $n \in \mathbb{N}$ that there exists an injection $f \colon \underline{n+1} \to \underline{n}$. Due to the Least Number Principle, we may consider the *least* such n. No injection (nor function, in general) $f \colon \underline{1} \to \underline{0}$ is possible for $f(0) \notin \underline{0}$. Hence, $n \neq 0$, i. e., $n = m + 1$ for some $m \in \mathbb{N}$.

Let $f(n) = x \in \underline{n}$ and consider the function $g \colon \underline{n} \to \underline{n}$ that permutes m and x. More accurately,

$$g(k) \quad = \quad \begin{cases} m & \text{if } k = x; \\ x & \text{if } k = m; \\ k & \text{otherwise.} \end{cases}$$

It is clear that g is an injection (and even a bijection). The function $f \restriction \underline{n} \colon \underline{n} \to \underline{n}$, which is defined by the trivial equation $(f \restriction \underline{n})(x) = f(x)$ for each $x \in \underline{n}$ and called the *restriction* of f to $\underline{n}$, is clearly injective as well. Therefore, the composition $h = g \circ (f \restriction \underline{n})$ is an injection $\underline{n} \to \underline{n}$.

If $h(k) = m$, then $(f \restriction \underline{n})(k) = x$, which implies $f(n) = x = f(k)$ despite $n \neq k \in \underline{n}$. This contradicts the injectivity of f. So, h does not take the value m and $\operatorname{rng} h \subseteq \underline{m}$. Yet then h is an injection $\underline{m+1} \to \underline{m}$. This is not possible due to the choice of n and the fact that $m < n$. A contradiction. □

Theorem 12.3 (Pigeonhole Principle)**.** *If $m > n$ and $f \colon \underline{m} \to \underline{n}$, then the function f is not injective (i. e., $\underline{m} \not\precsim \underline{n}$).*

Proof. Suppose that a certain injection $f \colon \underline{m} \to \underline{n}$ does exist. Since $m > n$, we get $m \geq n + 1$, whence $\underline{n+1} \subseteq \underline{m}$. Consequently, $\underline{n+1} \precsim \underline{m} \precsim \underline{n}$. It follows then that $\underline{n+1} \precsim \underline{n}$, which is not possible. □

The Pigeonhole Principle can be informally stated the following way:

> If $m > n$, it is not possible to place m distinct objects into n distinct boxes in such a manner that each box contains at most one object.

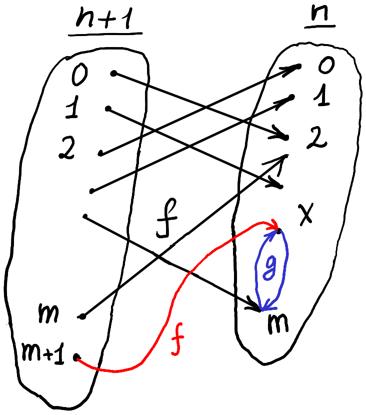

Figure 13: Proving Lemma 12.2.

Evidently, this statement is reducible to the first one if one labels the objects and boxes with numbers from $\underline{m}$ and $\underline{n}$, respectively, and then considers a function that maps every object's label to the label of the box containing that object.

Corollary 12.4. *If $m \neq n$, then $\underline{m} \nsim \underline{n}$.*

Proof. If $m \neq n$, then either $m > n$ or $m < n$. In the first case, by the Pigeonhole Principle, it is impossible that $\underline{m} \lesssim \underline{n}$. In the second case, it is impossible that $\underline{n} \lesssim \underline{m}$. Each case but excludes the equivalence $\underline{m} \sim \underline{n}$. □

Corollary 12.5. *For every finite set A, there exists a* unique *number $n \in \mathbb{N}$ such that $A \sim \underline{n}$.*

This number n with $A \sim \underline{n}$ is called the *cardinality* (or the number of elements, the size) of the finite set A. We shall write $n = |A|$ is this case. Clearly, for finite sets A and B, one has $A \sim B$ iff $|A| = |B|$.

Example 12.6. The set $\mathbb{N}$ is infinite. Otherwise, $\mathbb{N} \sim \underline{n}$ for some $n \in \mathbb{N}$. Yet $\underline{n+1} \subseteq \mathbb{N}$, which results in $\underline{n+1} \lesssim \mathbb{N}$. Then $\underline{n+1} \lesssim \underline{n}$, contrary to the Pigeonhole Principle.

It is not hard to establish a dual result to the Pigeonhole Principle. Namely, the following theorem holds.

Theorem 12.7. *If $m > n$ and $f \colon \underline{n} \to \underline{m}$, then the function f is not surjective.*

Proof. Assume it is. Consider the function $g \colon \underline{m} \to \underline{n}$ defined in the following way:

$$g(k) = \text{the least such } x \in \underline{n} \text{ that } f(x) = k.$$

Surjectivity of f and the Least Number Principle guarantee totality of g, while g is functional as a least element is unique. So, g is a well-defined function. Moreover, it is injective. Indeed, if $g(k) = g(l)$, then $k = f(g(k)) = f(g(l)) = l$. But no injection from $\underline{m}$ to $\underline{n}$ is possible according to the Pigeonhole Principle. A contradiction. □

Another useful result closely related to the Pigeonhole Principle is as follows. (We omit the proof.)

Theorem 12.8. *Let sets A, B be finite and $A \sim B$. Then for every function $f \colon A \to B$, the following holds: f is injective iff f is surjective.*

In fact, one can easily prove that every injection $A \to B$ is surjective under assumption that a proper subset of a finite set is smaller than that set. The latter statement follows from the Rule of Sum, which we prove below. A forward reference is unwanted here, so we omit the proof, which can be postponed for a seminar class, though. Another—far less elegant—way to prove surjectivity is similar to that of Theorem 12.3. Next, one can prove injectivity of every surjection $f\colon A \to B$ by constructing its right inverse g (like we have done when proving Theorem 12.7), which is clearly injective. By the first claim, g must be bijective and $f = g^{-1}$.

Example 12.9. If 10 students have got their marks (from 1 to 10 points) in such a way that no two distinct students have identical marks, then each mark has been used.

Finite and countable sets. Recall that a set A is called *countable*[20] if $A \sim \mathbb{N}$. Now we want to establish some general facts about finite and countable sets. Most of these are 'obvious' and in many cases, we will not bother ourselves with detailed proofs—for *rigorous* proofs are *not* that obvious. But anyway, we should at least put our knowledge of those facts in order.

Lemma 12.10. *If a set A is countable (or finite, infinite) and $A \sim B$, then B enjoys the same property.*

This is truly obvious.

Lemma 12.11. *If a set A is countable and $A \lesssim B$, then the set B is infinite.*

Proof. Otherwise, one gets $\underline{n+1} \lesssim \mathbb{N} \lesssim B \sim \underline{n}$ for some $n \in \mathbb{N}$. Yet then $\underline{n+1} \lesssim \underline{n}$ contrary to the Pigeonhole Principle. □

Lemma 12.12. *If $A \subseteq \mathbb{N}$, then the set A is either finite or countable.*

Proof sketch. The idea is simple: let us enumerate the elements of A in increasing order, so that $A = \{a_0, a_1, \ldots\}$ and $a_0 < a_1 < \ldots$ If we run out of elements of A at some stage, we have got a bijection $k \mapsto a_k$ from $\underline{n}$ to A for some $n \in \mathbb{N}$. Otherwise, we have a bijection $\mathbb{N} \to A$. □

Corollary 12.13. *If $A \lesssim B$ and the set B is countable, then the set A is either finite our countable.*

Corollary 12.14. *If $A \lesssim B$ and the set B is finite, then set A is also finite and $|A| \leq |B|$.*

Proof. By the assumption, $A \lesssim B \sim \underline{n} \lesssim \mathbb{N}$ for some $n \in \mathbb{N}$. Then the set A is either finite or countable by the previous corollary. However, A cannot be countable in view of Lemma 12.11. If $A \sim \underline{m}$ and $m > n$, then we get $\underline{m} \lesssim \underline{n}$ contrary to the Pigeonhole Principle. Hence, $m \leq n$. □

The following *Rules of Sum and Product* form a basis for the *enumerative combinatorics*, which deals with counting of various finite objects (like permutations, arrangement, partitions etc.)

Theorem 12.15 (Rule of Sum)**.** *Let sets A, B be finite and $A \cap B = \varnothing$. Then the set $A \cup B$ is also finite and $|A \cup B| = |A| + |B|$.*

Proof. Assume that $A \overset{f}{\sim} \underline{n}$ and $B \overset{g}{\sim} \underline{m}$. We want to define a bijection $h\colon A \cup B \to \underline{n+m}$ in the following way:

$$h(x) \quad = \quad \begin{cases} f(x) & \text{if } x \in A; \\ n + g(x) & \text{if } x \in B. \end{cases}$$

[20] Sometimes, they call finite sets countable as well.

As $A \cap B = \varnothing$, h is indeed a function with $h(x) < n + m$. Let $h(x) = h(y)$. If $x, y \in A$, then $x = y$ by injectivity of f. If $x, y \in B$, we get $n + g(x) = n + g(y)$, whence $g(x) = g(y)$ by the properties of addition, which, in turn, implies that $x = y$ for g is injective. Now assume that $x \in A$ and $y \in B$. Then $h(x) = f(x) < n \leq n + g(y) = h(y)$ despite $h(x) = h(y)$. The function h is thus injective.

Let us verify its being surjective. Let $k \in \underline{n + m}$. Then either $k < n$ or $n \leq k < n + m$. In the first case, obtain $k = f(x) = h(x)$ for some $x \in A$ as f is surjective. In the second case, we have $k = n + k'$ for a suitable $k' < m$ due to the properties of addition. By surjectivity of g, there exists such $y \in B$ that $k' = g(y)$, which yields $k = n + g(y) = h(y)$. □

Corollary 12.16. *If sets A, B are finite, then the set $A \cup B$ is also finite and $|A \cup B| = |A| + |B| - |A \cap B|$.*

Proof. One has both $A = (A \smallsetminus B) \cup (A \cap B)$ and $A \cup B = (A \smallsetminus B) \cup B$.

The sets $A \smallsetminus B$, $A \cap B \subseteq A$ are finite by Corollary 12.14. The set $A \smallsetminus B$ intersects neither $A \cap B$ nor B. Hence $|A| = |A \smallsetminus B| + |A \cap B|$, $A \cup B$ is finite, and

$$|A \cup B| = |A \smallsetminus B| + |B| = (|A| - |A \cap B|) + |B| = |A| + |B| - |A \cap B|.$$

□

Corollary 12.17. *If sets A, B are finite, then $|A \cup B| \leq |A| + |B|$.*

Theorem 12.18 (Rule of Product)**.** *Let sets A and B be finite. Then the set $A \times B$ is also finite and $|A \times B| = |A| \cdot |B|$.*

Proof. Let $A \overset{f}{\sim} \underline{n}$ and $B \overset{g}{\sim} \underline{m}$. If $m = 0$, then $B = \varnothing$ and $A \times B = \varnothing \sim \underline{0}$. So, assume $m \neq 0$. We want to construct a bijection $h \colon A \times B \to \underline{nm}$. In order to do so, put

$$h(x, y) = mf(x) + g(y)$$

for every $x \in A$, $y \in B$.

We recall Theorem 6.5 (on integer division), which states that for every naturals u and $v \neq 0$, there exists a unique pair $(q, r) \in \mathbb{N}^2$ such that $u = vq + r$ and $r < v$.

Let us check that h is surjective. Suppose that $z \in \underline{nm}$. Then $z = mq + r$ for some $q \in \mathbb{N}$ and $r \in \underline{m}$. Hence, there is such $y \in B$ that $r = g(y)$. We also have $q \in \underline{n}$ for $z \geq nm$ otherwise; it follows that there is an element $x \in A$ with $q = f(x)$. Thus, $z = mf(x) + g(y) = h(x, y)$.

Now, let us check injectivity. Assume that $mf(x) + g(y) = mf(x') + g(y') = z = mq + r$. As $g(y), g(y') < m$, both $g(y) = g(y')$ and $f(x) = f(x')$ must hold for the uniqueness requirement of Theorem 6.5. Hence, $x = x'$ and $y = y'$ for both f and g are injective. □

Corollary 12.19. *If a set A is finite, then for each $n \in \mathbb{N}$, the set A^n is finite as well and $|A^n| = |A|^n$.*

Proof. By induction on n. Use the identity $A^{n+1} = A^n \times A$ when $n \geq 1$. □

Let us count all possible functions between two finite sets.

Corollary 12.20. *If sets A and B are finite, then the set B^A is also finite and $|B^A| = |B|^{|A|}$.*

Proof. Let $A \sim \underline{n}$ and $B \sim \underline{m}$. By Theorems 11.13 and 11.6, get

$$B^A \sim \underline{m}^{\underline{n}} \sim \underline{m}^n,$$

whence $|B^A| = |\underline{m}|^n = m^n$. □

Corollary 12.21. *If a set A is finite, then the set $\mathcal{P}(A)$ is finite as well and $|\mathcal{P}(A)| = 2^{|A|}$.*

Proof. By Lemma 11.9, obtain $\mathcal{P}(A) \sim \underline{2}^A$, whence $|\mathcal{P}(A)| = 2^{|A|}$ in view of Corollary 12.20. □

Exercise 12.22. Let $f\colon A \to B$ and a set A be finite. Then the set $f[A]$ is also finite and $|f[A]| \leq |A|$.

Example 12.23. If a set A is countable, while B is either countable or finite, then the set $A \cup B$ is countable.

Clearly, $A \subseteq A \cup B$, whence $\mathbb{N} \lesssim A \cup B$. On the other hand, $A \cup B \lesssim (\mathbb{N} \times \{0\}) \cup (\mathbb{N} \times \{1\})$. Indeed, assume $A \overset{f}{\sim} \mathbb{N}$ and $B \overset{g}{\lesssim} \mathbb{N}$. For every $x \in A \cup B$ we let

$$h(x) \quad = \quad \begin{cases} (f(x), 0) & \text{if } x \in A; \\ (g(x), 1) & \text{if } x \in B \smallsetminus A. \end{cases}$$

This yields an injection $h\colon A \cup B \to (\mathbb{N} \times \{0\}) \cup (\mathbb{N} \times \{1\})$. Furthermore, we have $(\mathbb{N} \times \{0\}) \cup (\mathbb{N} \times \{1\}) = \mathbb{N} \times \underline{2} \lesssim \mathbb{N} \times \mathbb{N} \sim \mathbb{N}$. So, $A \cup B \lesssim \mathbb{N}$, whence $A \cup B \sim \mathbb{N}$.

Up to now, we have had only a 'negative' definition for infinity: a set A is infinite iff it is *not* finite, i. e. there is no $\underline{n}$ with $A \sim \underline{n}$. Clearly, this makes assuming some set to be infinite not very helpful in a proof. Using the Axiom of Choice, it is possible to turn infinity into a more explicit, 'positive' notion.

Lemma 12.24. *If a set A is infinite, then $\mathbb{N} \lesssim A$.*

Proof sketch. As A is infinite, then $A \smallsetminus B$ is also infinite, whence non-empty for every finite $B \subseteq A$. (Otherwise, $A = (A \smallsetminus B) \cup B$ would be finite by the Rule of Sum.) Therefore it is possible to choose some a_0 from A, then $a_1 \in A \smallsetminus \{a_0\}$, $a_2 \in A \smallsetminus \{a_0, a_1\}$, etc. (an *infinite* sequence of choices needs a special axiom to be well-defined). This procedure gives rise to a function $f\colon \mathbb{N} \to A$ such that $f(n) = a_n$, which is clearly injective.

□

Corollary 12.25. *If a set A is infinite, then there exists a countable set B such that $B \subseteq A$.*

Corollary 12.26. *A set A is infinite iff $\mathbb{N} \lesssim A$.*

The set $\mathbb{N}$ is thus the 'least' infinite set (in the sense of the 'relation' $\lesssim$).

Example 12.27. Let a set A be infinite and B be either finite or countable. Then $A \cup B \sim A$.

Notice that $A \cup B = A \cup (B \smallsetminus A)$. By Corollaries 12.14 and 12.13, the set $B \smallsetminus A \subseteq B$ is either finite or countable. We know from the above that A has a countable subset B'. Then

$$A \cup B = (A \smallsetminus B') \cup B' \cup (B \smallsetminus A),$$

where the three sets in the right hand side union are pairwise disjoint. By Example 12.23, the set $C = B' \cup (B \smallsetminus A)$ is countable; hence, there is a bijection $f\colon C \to B'$. As $A \cup B = (A \smallsetminus B') \cup C$, we can extend f to a function $g\colon A \cup B \to (A \smallsetminus B') \cup B'$ in the following way:

$$g(x) \quad = \quad \begin{cases} f(x) & \text{if } x \in C; \\ x & \text{if } x \in A \smallsetminus B'. \end{cases}$$

Clearly, g is bijective. Since $(A \smallsetminus B') \cup B' = A$, the bijection g is as required.

Exercise 12.28. Prove that if $A \smallsetminus B$ is infinite, while B is finite or countable, then $A \sim A \smallsetminus B$.

13 More Combinatorics

> Traditionally, the elements of combinatorics are presented informally, with very little reference to set theory. The benefits of this approach are well-known, but it has clear drawbacks in the context of a set-based course: first, it breaks the logical sequence of presentation (so that the Student may think set theory axioms are useless and inadequate for "everyday mathematics" besides being abstract and possibly indigestible); second, our intuition is limited, so one may lose comprehension beyond some point with little chance to dissect a complicated intuitive argument into primitive steps. Therefore, our approach is to make combinatorial computations as close to set-theoretic primitives as reasonably possible. Of course, enough practice will develop the Student's intuition so that he can see traditional arguments as shorthands for formal ones.

Our next goal is to gather more simple facts about finite set cardinalities. These may be looked upon as a kind of 'primitives' for solving traditional combinatorial problems (those about arrangements, permutations, and combinations) in a more formal manner.

Counting injections. Let A, B be finite sets and let $\mathrm{Inj}(A, B)$ be the set of all possible *injections* from A to B. How many of these are possible? As $\mathrm{Inj}(A, B) \subseteq B^A$, we see that the set $\mathrm{Inj}(A, B)$ is necessarily finite and $|\mathrm{Inj}(A, B)| \leq |B|^{|A|}$ (by Lemma 12.20). Then, we observe that the number $|\mathrm{Inj}(A, B)|$ *does not* depend on the sets A, B themselves but just on their sizes.

> Given enough time (which is unlikely in practice), the Instructor might prove similar results for all the following 'combinatorial numbers' (which are omitted traditionally). We do not do it nevertheless.

Lemma 13.1 ("just the size matters"). *If $A' \sim A$ and $B' \sim B$, then $\mathrm{Inj}(A', B') \sim \mathrm{Inj}(A, B)$.*

Proof. Assume that $A' \overset{\varphi}{\sim} A$ and $B' \overset{\psi}{\sim} B$. We need to get a bijection $\theta \colon \mathrm{Inj}(A', B') \to \mathrm{Inj}(A, B)$. So, for each $f \in \mathrm{Inj}(A', B')$, we define $\theta(f) = \psi \circ f \circ \varphi^{-1}$. As a composition of injections is an injection itself, we have $\theta(f) \in \mathrm{Inj}(A, B)$.

But why is θ injective? Indeed, suppose that $\theta(f) = \theta(g)$. Then,

$$f = (\psi^{-1} \circ \psi) \circ f \circ (\varphi^{-1} \circ \varphi) = \psi^{-1} \circ \theta(f) \circ \varphi = \psi^{-1} \circ \theta(g) \circ \varphi = (\psi^{-1} \circ \psi) \circ g \circ (\varphi^{-1} \circ \varphi) = g.$$

On the other hand, for each $h \in \mathrm{Inj}(A', B')$, we obtain

$$\theta(\psi^{-1} \circ h \circ \varphi) = (\psi \circ \psi^{-1}) \circ h \circ (\varphi \circ \varphi^{-1}) = h,$$

where $\psi^{-1} \circ h \circ \varphi \in \mathrm{Inj}(A', B')$. Thus, θ is surjective. □

This means that $|\mathrm{Inj}(A, B)| = |\mathrm{Inj}(\underline{m}, \underline{n})|$ if $m = |A|$ and $n = |B|$, and it suffices to do all counting work just for our 'exemplary' finite sets. Such a situation is common for traditional combinatorial numbers: *just the size matters.* Now let us do the actual counting.

Lemma 13.2. *For every numbers $n, m \in \mathbb{N}$, we have*

$$|\mathrm{Inj}(\underline{m}, \underline{n})| \;=\; \begin{cases} 0 & \text{if } m > n; \\ \dfrac{n!}{(n-m)!} & \text{if } m \leq n, \end{cases}$$

where $0! = 1$ and $(n+1)! = (n+1) \cdot n! = (n+1) \cdot n \cdot (n-1) \cdot \ldots 2 \cdot 1$ (such a number $n!$ is called the factorial *of n).*

Proof. If $m > n$, then $\mathrm{Inj}(\underline{m}, \underline{n}) = \varnothing$ due to the Pigeonhole Principle. Assume that $m \leq n$. Given that $\frac{n!}{(n-m)!} = n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot (n-m+1)$, it is not hard to explain the required statement informally. Indeed, in order to specify an injection f from $\underline{m}$ to $\underline{n}$, one has to choose its value at each point from 0 to $m-1$. There are n ways to choose the value $f(0)$ (it can be anything from $\underline{n}$). But $f(1)$ cannot be the same value, so one has just $n-1$ options for *whatever* $f(0)$ has been chosen, etc.

Yet let us give a more formal proof. We shall use induction on m. As $\mathrm{Inj}(\varnothing, \underline{n}) = \{\varnothing\}$, for $m = 0$, we get $|\mathrm{Inj}(\underline{m}, \underline{n})| = 1 = \frac{n!}{(n-0)!}$. Consider $m+1$ and $n \geq m+1 > 0$; suppose that $|\mathrm{Inj}(\underline{m'}, \underline{n'})| = \frac{n'!}{(n'-m')!}$ for every $m' \leq m$ and for *whatever* n' with $n' \geq m'$.

Consider the set $X = \mathrm{Inj}(\underline{m+1}, \underline{n})$. We can partition it into subsets according to the value $f(m)$. Indeed, let $X_k = \{f \in X \mid f(m) = k\}$ for each $k \in \underline{n}$. Clearly, $X = X_0 \cup X_1 \cup \ldots \cup X_{n-1}$, while every two sets of the form X_k are disjoint. By the Rule of Sum, $|X| = |X_0| + |X_1| + \ldots + |X_{n-1}|$.

But how many elements does X_k have? In fact, every $f \in X_k$ is completely determined by its values at the points $0, \ldots, m-1$, that is, by the function $f \restriction \underline{m} \in \mathrm{Inj}(\underline{m}, \underline{n} \smallsetminus \{k\})$ (k is not a value of $f \restriction \underline{m}$, for otherwise f would not be injective). So, taking Lemma 13.1 ("just the size matters") into account, we obtain:

$$X_k \sim \mathrm{Inj}(\underline{m}, \underline{n} \smallsetminus \{k\}) \sim \mathrm{Inj}(\underline{m}, \underline{n-1}).$$

As $n \geq m+1$, then $n-1 \geq m$, so the inductive hypothesis is applicable, which gives us

$$|X_k| = |\mathrm{Inj}(\underline{m}, \underline{n-1})| = \frac{(n-1)!}{((n-1)-m)!}.$$

Remarkably, $|X_k|$ does not depend on k, so

$$|\mathrm{Inj}(\underline{m+1}, \underline{n})| = |X_0| + |X_1| + \ldots + |X_{n-1}| = n \cdot \frac{(n-1)!}{((n-1)-m)!} = \frac{n!}{(n-(m+1))!},$$

as required. □

They traditionally call $|\mathrm{Inj}(\underline{m}, \underline{n})|$ *the number of partial permutations* or *(ordered) arrangements* of length m of n elements. This is then denoted by P_n^m or A_n^m, etc. Let us see what idea is behind such terminology. An *arrangement* of length m of a set A elements is a tuple $(a_0, \ldots, a_{m-1})$ where each $a_i \in A$ and $a_i \neq a_j$ if $i \neq j$. As we know from Theorem 11.13, the set A^m of *all* tuples is equivalent to the set $A^{\underline{m}}$ of functions $\underline{m} \to A$ via the natural bijection which maps $(a_0, \ldots, a_{m-1})$ to the function $f \colon i \mapsto a_i$. Clearly, for an *arrangement*, the function f will be an injection, hence the set of arrangements is equivalent to $\mathrm{Inj}(\underline{m}, A)$. When combined with the "just the size matters" principle, this justifies the equation $A_n^m = |\mathrm{Inj}(\underline{m}, \underline{n})|$.

Any bijection from a set A to A is usually called a *permutation* of A (particularly, when A is finite). How many permutations of a finite set are possible? In analogy with Lemma 13.1, this question is reducible to the following one: how many permutations of $\underline{n}$ are possible?

Lemma 13.3. *For every number $n \in \mathbb{N}$, there are exactly $n!$ distinct permutations of the set $\underline{n}$.*

Proof. Let us denote the set of all possible permutations of $\underline{n}$ by X. Clearly, $X \subseteq \mathrm{Inj}(\underline{n}, \underline{n})$. On the other hand, every injection from $\underline{n}$ to $\underline{n}$ is surjective by Theorem 12.8; hence, it is a bijection. So, $\mathrm{Inj}(\underline{n}, \underline{n}) \subseteq X$. Finally, we apply Lemma 13.2 to obtain

$$|X| = |\mathrm{Inj}(\underline{n}, \underline{n})| = \frac{n!}{(n-n)!} = \frac{n!}{1} = n!.$$

□

Counting subsets. As we know, a finite set A has as many as $2^{|A|}$ distinct subsets. But what is the count for the subsets of some fixed size k? We denote by $\mathcal{P}_k(A)$ the set of all subsets of A that are of cardinality k. Similarly to Lemma 13.1, it is easy to check that $\mathcal{P}_k(A) \sim \mathcal{P}_k(B)$ when $A \sim B$. Again, just the size matters. Hence, it suffices to know $|\mathcal{P}_k(\underline{n})|$ for diverse $n \in \mathbb{N}$, which number is traditionally denoted by C_n^k ("n choose k") since *combinations* is the traditional name for fixed size subsets.

Clearly, $C_n^k = 0$ if $k > n$, since the Pigeonhole Principle forbids a subset to be greater in size than the whole set. Likewise, $C_n^0 = 1$ as the empty set is unique, and $C_n^n = 1$ for if there were two n-sized *distinct* subsets X and Y of $\underline{n}$, then it would be that $\bar{X} \neq \bar{Y}$ despite $X = \varnothing = Y$ (clearly, $|\bar{X}| = |\bar{Y}| = n - n = 0$ by the Rule of Sum).

Lemma 13.4 (Pascal's Identity)**.** *For every $n, k \in \mathbb{N}$, it holds that*

$$C_{n+1}^{k+1} = C_n^{k+1} + C_n^k.$$

Proof. We will give a so-called *combinatorial proof* for this identity, that is, one based on counting the same quantity in two distinct ways.

Consider the set $A = \{X \in \mathcal{P}_{k+1}(\underline{n+1}) \mid n \in X\}$ and its complement $\bar{A} = \{X \in \mathcal{P}_{k+1}(\underline{n+1}) \mid n \notin X\}$ (i. e., we form two classes of subsets based on whether they contain $n \in \underline{n+1}$ or not). By the Rule of Sum, $C_{n+1}^{k+1} = |\mathcal{P}_{k+1}(\underline{n+1})| = |A| + |\bar{A}|$.

Clearly, $\bar{A} = \mathcal{P}_{k+1}(\underline{n})$ (as $X \in \bar{A}$ implies $X \subseteq \underline{n}$), whence $|\bar{A}| = C_n^{k+1}$. On the other hand, one has $X = (X \cap \underline{n}) \cup \{n\}$ for every $X \in A$. By the Rule of Sum, the set $X' = X \cap \underline{n}$ is of cardinality k. Then the mapping $X \mapsto X'$ is a clear bijection from A to $\mathcal{P}_k(\underline{n})$. So, $|A| = C_n^k$. The required identity is now immediate. □

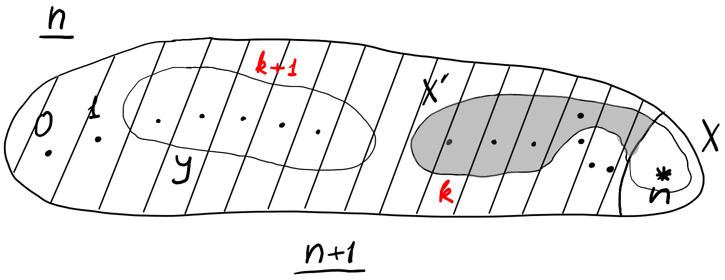

Figure 14: Proving Lemma 13.4: here $X \in A$ and $Y \in \bar{A}$; the k-element set X' is highlighted.

The above lemma allows to obtain an explicit formula for the number C_n^k.

Lemma 13.5. *For every $n, k \in \mathbb{N}$, if $k \leq n$, then*

$$C_n^k = \frac{n!}{(n-k)! \cdot k!}.$$

Proof. First, we notice that $1 = C_n^0 = \frac{n!}{n!} = \frac{n!}{(n-0)! \cdot 0!}$ for every n. Let us assume that $k > 0$. We prove the formula by induction on n. If $n = 0$, then $k = 0$, which contradicts our assumption.

Consider numbers $n+1$ and k such that $k \leq n+1$, and assume that $C_n^{k'} = \dfrac{n!}{(n-k')! \cdot k'!}$ for whatever k' with $k' \leq n$ (the inductive hypothesis).

As $k > 0$, we have $k = k'+1$ and $k'+1 \leq n+1$ for a suitable k'. Hence, $k' \leq n$, so the inductive hypothesis applies. By Pascal's Identity, we get:

$$C_{n+1}^{k'+1} = C_n^{k'+1} + C_n^{k'}.$$

If $k'+1 \leq n$, the IH is also applicable to the term $C_n^{k'+1}$, whence

$$C_{n+1}^{k} = C_{n+1}^{k'+1} = \frac{n!}{(n-k'-1)! \cdot (k'+1)!} + \frac{n!}{(n-k')! \cdot k'!} = \frac{n! \cdot (n-k') + n! \cdot (k'+1)}{(n-k')! \cdot (k'+1)!} =$$
$$\frac{n! \cdot (n+1)}{((n+1)-(k'+1))! \cdot (k'+1)!} = \frac{(n+1)!}{((n+1)-(k'+1))! \cdot (k'+1)!} = \frac{(n+1)!}{((n+1)-k)! \cdot k!},$$

as it is required.

Now consider the case when $k'+1 > n$, whence $n+1 \leq k'+1 \leq n+1$. Then $k' = n$, $k = n+1$, and

$$C_{n+1}^{k} = C_{n+1}^{n+1} = 1 = \frac{(n+1)!}{((n+1)-(n+1))! \cdot (n+1)!} = \frac{(n+1)!}{((n+1)-k)! \cdot k!}.$$

□

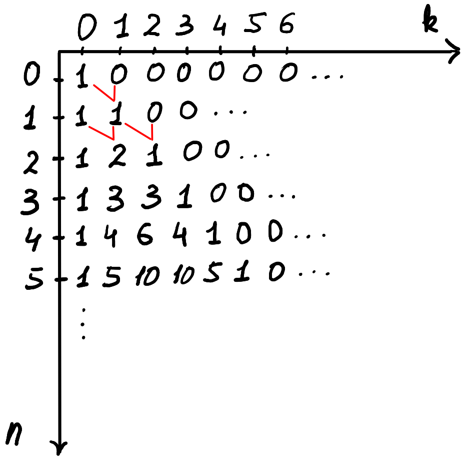

Figure 15: This *Pascal's Triangle* arranges C_n^k numbers. The equations $C_n^0 = 1$ and $C_n^k = 0$ for $k > 0$ describe the first column and row, respectively. All the other numbers are determined by Lemma 13.4.

Remark 13.6. It is easy to see that for $k \leq n$, one has $C_n^k = \dfrac{|\mathrm{Inj}(k,n)|}{k!}$. So, there are $k!$ times more injections from $\underline{k}$ to $\underline{n}$ than there are k-sized subsets of $\underline{n}$. If proved independently, this gives another way to obtain the formula for C_n^k. Such a proof is usually presented as follows: we have $k!$ times more *arrangements* than subsets, as each subset of size k can be arranged in exactly $k!$ distinct ways (k-sized arrangements of k elements are *permutations*). One, however, needs *equivalence relations* and *classes* described below in order to make this argument rigorous.

One easy property of numbers C_n^k with a combinatorial proof is as follows.

Lemma 13.7. *For every $n, k \in \mathbb{N}$, if $k \leq n$, then $C_n^k = C_n^{n-k}$.*

Proof. It suffices to prove that $\mathcal{P}_k(\underline{n}) \sim \mathcal{P}_{n-k}(\underline{n})$. Indeed, consider a function φ such that $\varphi(X) = \bar{X}$ for every $X \in \mathcal{P}_k(\underline{n})$. By the Rule of Sum, $|X| + |\bar{X}| = n$, hence $\varphi\colon \mathcal{P}_k(\underline{n}) \to \mathcal{P}_{n-k}(\underline{n})$. If $\varphi(X) = \varphi(Y)$, then $X = \overline{\varphi(X)} = \overline{\varphi(Y)} = Y$, so φ is injective. On the other hand, $Z = \varphi(\bar{Z})$ for each $Z \in \mathcal{P}_{n-k}(\underline{n})$, so φ is surjective. □

Corollary 13.8. $C_{a+b}^a = C_{a+b}^b$.

From high-school mathematics, we know the *binomial formula* $(a+b)^2 = a^2 + 2ab + b^2$. Let us generalize this formula to an arbitrary natural exponent.

Lemma 13.9. *For every numbers $x \in \mathbb{R}$ and $n \in \mathbb{N}$,*

$$(1+x)^n = 1 + C_n^1 x + C_n^2 x^2 + \ldots + C_n^{n-1} x^{n-1} + x^n = \sum_{k=0}^{n} C_n^k x^k.$$

Proof. Induction on n. For $n = 0$, this is obvious. Assume that $(1+x)^n = \sum_{k=0}^n C_n^k x^k$. Then, applying the IH, re-indexing (we replace $k+1$ with k), applying 13.4 and the fact that $C_m^0 = C_m^m = 1$ for every m, we obtain:

$$(1+x)^{n+1} = (1+x)(1+x)^n = (1+x) \cdot \sum_{k=0}^{n} C_n^k x^k = \sum_{k=0}^{n} C_n^k x^k + \sum_{k=0}^{n} C_n^k x^{k+1} =$$

$$\sum_{k=0}^{n} C_n^k x^k + \sum_{k=1}^{n+1} C_n^{k-1} x^k = C_n^0 x^0 + \sum_{k=1}^{n} (C_n^k + C_n^{k-1}) x^k + C_n^n x^{n+1} =$$

$$C_{n+1}^0 x^0 + \sum_{k=1}^{n} C_{n+1}^k x^k + C_{n+1}^{n+1} x^{n+1} = \sum_{k=1}^{n+1} C_{n+1}^k x^k,$$

as it is required. □

Corollary 13.10 (Binomial Theorem)**.** *For every numbers $a, b \in \mathbb{R}$ and $n \in \mathbb{N}$,*

$$(a+b)^n = \sum_{k=0}^{n} C_n^k a^k b^{n-k}.$$

Proof. If $b = 0$, then the right-hand side equals $C_n^n a^n b^0 = a^n$, that is, the left-hand side.

Suppose that $b \neq 0$. Then

$$(a+b)^n = (b+a)^n = b^n \cdot \left(1 + \frac{a}{b}\right)^n = b^n \cdot \sum_{k=0}^{n} C_n^k \left(\frac{a}{b}\right)^k = \sum_{k=0}^{n} C_n^k a^k b^{n-k}.$$

□

The above theorem justifies the name *binomial coefficients* for the numbers C_n^k.

Example 13.11. For every $n \in \mathbb{N}$, we have $\sum_{k=0}^n C_n^k = 2^n$.

Indeed, it is enough to apply the Binomial Theorem to $a = 1$ and $b = 1$. Another way to verify this identity is a combinatorial one: both sides clearly denote the number of all subsets of the set $\underline{n}$ since $\mathcal{P}(\underline{n}) = \mathcal{P}_0(\underline{n}) \cup \mathcal{P}_1(\underline{n}) \cup \ldots \cup \mathcal{P}_n(\underline{n})$ and the Rule of Sum is applicable here.

Exercise 13.12. For every $n \in \mathbb{N}$, prove that $\sum_{k=0}^{n}(-1)^k C_n^k = 0$.

Example 13.13. Yet another typical application of binomial coefficients (which many others can be reduced to) is as follows. Consider the equation

$$x_1 + x_2 + \ldots + x_m = n,$$

where $m \in \mathbb{N}_+$ and $n \in \mathbb{N}$. A *solution* to this equation is just a tuple $(x_1, \ldots, x_m) \in \mathbb{N}^m$. How many solutions does the equation have?

Let us encode *every* m-tuple $\vec{y} = (y_1, y_2, \ldots, y_m)$ of naturals by a binary word (i. e., a tuple of the set $\underline{2}$ elements) $b(\vec{y}) = \underbrace{11\ldots1}_{y_1}0\underbrace{11\ldots1}_{y_2}0\ldots0\underbrace{11\ldots1}_{y_m}$. Clearly, this encoding is injective. The symbol 0 occurs in $b(\vec{y})$ just $m-1$ times, while 1 has exactly $y_1 + y_2 + \ldots + y_m$ many occurences. So, $\vec{y}$ is a solution to our equation iff $b(\vec{y})$ is a binary word of length $n+m-1$ with exactly $m-1$ zeroes. Since b is injective, this mapping provides a bijection from the set of solutions to the set of binary words of the said form. Let us count the latter.

In its turn, each binary word may be bijectively encoded by the set of *position* numbers for zeroes. This is an $(m-1)$-sized subset of a $(n+m-1)$-element set (we choose $m-1$ positions among $n+m-1$ possible). Hence, there are C_{n+m-1}^{m-1} binary words of the form in question. Finally, in view of Corollary 13.8, this gives us $C_{n+m-1}^{m-1} = C_{n+m-1}^{n}$ distinct solutions to the original equation.

> It is important to highlight diverse bijections and 'encodings' which reduce a problem to formal primitives. For a seminar class, we recommend to make them explicit in a few introductory problems and switch to the traditional "intuitive" style afterward (except for the most tricky arguments). Ideally, students should be able to translate intuitive combinatorial proofs into the set-theoretic formalism freely. The fact that most traditional combinatorial problems are stated informally makes this task quite hard.

Example 13.14. What is the number of *positive* integer solutions to the equation $x_1+x_2+\ldots+x_m = n$, i. e., those with all $x_i > 0$?

Each natural number $x_i > 0$ can be uniquely expressed as $y_i + 1$ for some $y_i \geq 0$. Then $\vec{x}$ is a positive solution to the equation iff one has $(y_1+1)+(y_2+1)+\ldots+(y_m+1) = n$. The latter equation has no natural solution when $n < m$ but is equivalent to $y_1 + y_2 + \ldots y_m = n - m$ otherwise. By the above, there are $C_{(n-m)+m-1}^{m-1} = C_{n-1}^{m-1}$ distinct solutions $\vec{y}$. As $\vec{x}$ bijectively corresponds to $\vec{y}$, there are exactly C_{n-1}^{m-1} distinct positive solutions to the original equation when $n, m > 0$.

Inclusion–Exclusion Principle. From Corollary 12.16, we know that $|A \cup B| = |A| + |B| - |A \cap B|$ for any two finite sets A and B. It is not hard to extend this fact to three finite sets A, B, C. Indeed, since $(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$ and $(A \cap C) \cap (B \cap C) = A \cap B \cap C$, we obtain

$$\begin{aligned}
&|(A \cup B) \cup C| = |A \cup B| + |C| - |(A \cup B) \cap C| = \\
&\qquad |A| + |B| - |A \cap B| + |C| - |(A \cap C) \cup (B \cap C)| = \\
&\qquad |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |(A \cap C) \cap (B \cap C)| = \\
&\qquad\qquad |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|.
\end{aligned}$$

It is possible to further generalize this formula, yet we will omit the proof (for its being somewhat clumsy), which is based on Corollary 12.16 and induction on n.

Theorem 13.15 (Inclusion–Exclusion Principle)**.** *For arbitrary finite sets $A_1, A_2, \ldots, A_n$, it holds that*

$$|A_1 \cup A_2 \cup \ldots \cup A_n| = \sum_{k=1}^{n} (-1)^{k-1} \sum_{1 \leqslant i_1 < i_2 < \ldots < i_k \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_k}| =$$

$$\sum_{1 \leqslant i_1 \leqslant n} |A_{i_1}| - \sum_{1 \leqslant i_1 < i_2 \leqslant n} |A_{i_1} \cap A_{i_2}| + \sum_{1 \leqslant i_1 < i_2 < i_3 \leqslant n} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| - \ldots$$

$$+ (-1)^{n-1} \sum_{1 \leqslant i_1 < i_2 < \ldots < i_n \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_n}| =$$

$$(|A_1| + |A_2| + \ldots + |A_n|) - (|A_1 \cap A_2| + |A_1 \cap A_3| + \ldots + |A_2 \cap A_3| + \ldots + |A_{n-1} \cap A_n|) +$$

$$+ (|A_1 \cap A_2 \cap A_3| + |A_1 \cap A_2 \cap A_4| + \ldots + |A_2 \cap A_3 \cap A_4| + \ldots + |A_{n-2} \cap A_{n-1} \cap A_n|) - \ldots$$

$$+ (-1)^{n-1} |A_1 \cap A_2 \cap \ldots \cap A_n|.$$

Example 13.16. Let us count the number of *surjections* between two finite sets. Likewise the "just the size matters" principle, it suffices to know $|\mathrm{Sur}(\underline{m}, \underline{n})|$ for every $n, m \in \mathbb{N}$, where $\mathrm{Sur}(A, B)$ is the set of all surjections from a set A to a set B.

We know the total number of functions $\underline{m} \to \underline{n}$ to equal n^m. The idea is to count *non-surjections* first and then subtract the number thereof from n^m (by the Rule of Sum). Let A_{k+1}, for $k \in \underline{n}$, be the set of all functions $\underline{m} \to \underline{n}$ that *do not* take the value k, that is, $A_{k+1} = \{f \in \underline{n}^{\underline{m}} \mid k \notin \operatorname{rng} f\} = (\underline{n} \setminus \{k\})^{\underline{m}}$. Clearly, a function must belong to at least one of A_k in order to be a non-surjection. Then, by the Inclusion–Exclusion Principle,

$$|\overline{\mathrm{Sur}(\underline{m}, \underline{n})}| = |A_1 \cup A_2 \cup \ldots \cup A_n| =$$

$$\sum_{s=1}^{n} (-1)^{s-1} \sum_{1 \leqslant i_1 < i_2 < \ldots < i_s \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}| =$$

$$\sum_{1 \leqslant i_1 \leqslant n} |A_{i_1}| - \sum_{1 \leqslant i_1 < i_2 \leqslant n} |A_{i_1} \cap A_{i_2}| + \sum_{1 \leqslant i_1 < i_2 < i_3 \leqslant n} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| - \ldots$$

$$+ (-1)^{n-1} \sum_{1 \leqslant i_1 < i_2 < \ldots < i_n \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_n}|.$$

What is the set $A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}$ for pairwise distinct indices $i_1, i_2, \ldots, i_s$? Clearly, it is the set of functions from $\underline{m}$ to $\underline{n}$ which take no value among $i_1, i_2, \ldots, i_s$, i. e., it the set $(\underline{n} \setminus \{i_1, i_2, \ldots, i_s\})^{\underline{m}}$. According to Theorem 12.20 and the Rule of Sum, this set has cardinality $(n-s)^m$ for whatever choice of $i_1, i_2, \ldots, i_s$.

Then

$$\sum_{1 \leqslant i_1 < i_2 < \ldots < i_s \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}| = (n-s)^m \cdot \sum_{1 \leqslant i_1 < i_2 < \ldots < i_s \leqslant n} 1 = C_n^s (n-s)^m$$

for there are C_n^s many ways to choose a subset $\{i_1, i_2, \ldots, i_s\}$ from $\underline{n}$ (and just one way to sort it in ascending order). Combined with the Inclusion–Exclusion Principle, this yields:

$$|\mathrm{Sur}(\underline{m}, \underline{n})| = n^m - |A_1 \cup A_2 \cup \ldots \cup A_n| =$$

$$n^m - \sum_{s=1}^{n} (-1)^{s-1} C_n^s (n-s)^m = C_n^0 (n-0)^m + \sum_{s=1}^{n} (-1)^s C_n^s (n-s)^m = \sum_{s=0}^{n} (-1)^s C_n^s (n-s)^m.$$

Example 13.17. From Theorem 12.7, we know that $|\mathrm{Sur}(\underline{m}, \underline{n})| = 0$ if $n > m$. Hence we obtain a *purely arithmetical* non-trivial fact:

$$\sum_{s=0}^{n} (-1)^s C_n^s (n-s)^m = 0$$

when $n > m$. Consider another example of this kind. From Theorem 12.8, it follows that every surjection $\underline{n} \to \underline{n}$ is a bijection (and vice versa, of course). As the number of such bijections equals $n!$ (by Lemma 13.3), we see that

$$\sum_{s=0}^{n} (-1)^s C_n^s (n-s)^n = n!,$$

which is another interesting arithmetical fact we have proved in a combinatorial manner (i. e., by counting the same quantity in two ways).

> Sometimes, mathematical modeling of a "real-world" problem is non-trivial (and much more so for genuine *real-world*) and the "answer" may depend on the model essentially. While the example below is straightforward in any respect, we urge the students to pay attention to modeling details. This will be especially important for probabilistic problems in sections below.

Example 13.18. How many ways are there to place six distinct balls into five distinct boxes so that no box remains empty?

First, we need a mathematical model for this 'real-world' problem. We can assign consecutive naturals to balls and boxes (separately) and then identify balls and boxes with their respective numbers. This way we obtain the sets $\underline{6}$ and $\underline{5}$ as formal models for the said collections. What is a 'way to place' a ball into a box? Essentially, it is an assignment of boxes to balls such that every ball has exactly one box assigned. Clearly, functions capture this idea exactly. So, a 'way to place' is just a function from $\underline{6}$ to $\underline{5}$. Finally, we require no box to be empty, that is, every box must be assigned to a certain ball. In other words, our function must be surjective.

The question is thus reduced to the following: how many surjections from $\underline{6}$ to $\underline{5}$ exist? We know the answer—it is

$$|\mathrm{Sur}(\underline{6}, \underline{5})| = \sum_{s=0}^{5} (-1)^s C_5^s (5-s)^6 = C_5^0 \cdot 5^6 - C_5^1 \cdot 4^6 + C_5^2 \cdot 3^6 - C_5^3 \cdot 2^6 + C_5^4 \cdot 1^6 - C_5^5 \cdot 0^6 = 1800.$$

How many ways are there to order numbers $1, 2, \ldots, n$ in such a way that no k takes the k-th place? Clearly, this is just the number of bijections $f \colon \underline{n} \to \underline{n}$ with $f(k) \neq k$ for each $k \in \underline{n}$. Such bijections are called *derangements* of the set $\underline{n}$.

Example 13.19. Let us count all the derangements of the set $\underline{n}$.

As in the above, we count *non-derangement* permutations first. Let

$$A_{k+1} = \{f \colon \underline{n} \to \underline{n} \mid f \text{ is bijective and } f(k) = k\}$$

for every $k \in \underline{n}$. We need to calculate $|A_1 \cup A_2 \cup \ldots \cup A_n|$. By the Inclusion–Exclusion Principle,

$$|A_1 \cup A_2 \cup \ldots \cup A_n| = \sum_{s=1}^{n} (-1)^{s-1} \sum_{0 \leqslant i_1 < i_2 < \ldots < i_s \leq n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}|.$$

Clearly, for any distinct $i_1, i_2, \ldots, i_s$, the set $A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}$ is the set of all bijections $\underline{n} \to \underline{n}$ with their values at $i_1, i_2, \ldots, i_s$ fixed. These bijections are in bijective correspondence with permutaions of the set $\underline{n} \setminus \{m_1, m_2, \ldots, m_s\}$. There are just $(n-s)!$ such permutations. Hence, the overall number of derangements equals

$$n! - \sum_{s=1}^{n} (-1)^{s-1} C_n^s (n-s)! = n! + \sum_{s=1}^{n} (-1)^s \frac{n!}{s!} = n! \sum_{s=0}^{n} (-1)^s \frac{1}{s!}.$$

The share of derangements in the total number of bijections thus equals $\sum\limits_{s=0}^{n} (-1)^s \frac{1}{s!}$. From Calculus, we know that for $n \to +\infty$, this sum converges to e^{-1}. In probabilistic terms (see Section **??**), this fact can be interpreted the following way: the more cards one has in an ordered deck, the closer to $e^{-1} = 0.3678\ldots$ is the probability that each card will change its position after a random shuffle. It might be thought somewhat counter-intuitive that this probability does not tend to 1 nor to 0.

Example 13.20. Our last example here is about computing values for Euler's totient function φ. Let us recall that for every natural $m > 1$, $\varphi(m)$ equals the number of the set $\underline{m}$ elements coprime with m.

Assume that $m = p_1^{a_1} \ldots p_n^{a_n}$ for some pairwise distinct primes p_i, where each $a_i > 0$. Let $A_k = \{x \in \underline{m} \mid p_k \,|\, x\}$. Clearly, a number x is *not* coprime with m iff they share some prime divisor p_k. So, $A_1 \cup A_2 \cup \ldots \cup A_n$ is the set of numbers that are not coprime with m among the elements of $\underline{m}$. By the Inclusion–Exclusion Principle,

$$\begin{aligned}\varphi(m) = m - |A_1 \cup A_2 \cup \ldots \cup A_n| = \\ m - \sum_{1 \leqslant i_1 \leqslant n} |A_{i_1}| + \sum_{1 \leqslant i_1 < i_2 \leqslant n} |A_{i_1} \cap A_{i_2}| - \sum_{1 \leqslant i_1 < i_2 < i_3 \leqslant n} |A_{i_1} \cap A_{i_2} \cap A_{i_3}| + \ldots \\ + (-1)^n \sum_{1 \leqslant i_1 < i_2 < \ldots < i_n \leqslant n} |A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_n}|.\end{aligned}$$

For each $x \in \underline{m}$, we have $x \in A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}$ iff $p_{i_t} \,|\, x$ for all t, which is equivalent to $p_{i_1} p_{i_2} \ldots p_{i_s} \,|\, x$. The latter is equivalent to $x = p_{i_1} p_{i_2} \ldots p_{i_s} \cdot l$, where $0 \le l < \frac{m}{p_{i_1} p_{i_2} \cdots p_{i_s}}$. Thus, $|A_{i_1} \cap A_{i_2} \cap \ldots \cap A_{i_s}| = m p_{i_1}^{-1} p_{i_2}^{-1} \ldots p_{i_s}^{-1}$ and

$$\begin{aligned}\varphi(m) = m\big(1 - (p_1^{-1} + p_2^{-1} + \ldots + p_n^{-1}) + \sum_{1 \leqslant i_1 < i_2 \leqslant n} p_{i_1}^{-1} p_{i_2}^{-1} - \sum_{1 \leqslant i_1 < i_2 < i_3 \leqslant n} p_{i_1}^{-1} p_{i_2}^{-1} p_{i_3}^{-1} + \ldots \\ + (-1)^n p_1^{-1} p_2^{-1} \ldots p_n^{-1}\big) = m(1 - p_1^{-1})(1 - p_2^{-1}) \ldots (1 - p_n^{-1}).\end{aligned}$$

If you do not believe the last equation, you may first try it for small values of n like 3 and 4, then prove it by induction on n.

Example 13.21. As $12 = 2^2 \cdot 3$, we have $\varphi(12) = 12 \cdot (1 - \frac{1}{2}) \cdot (1 - \frac{1}{3}) = 12 \cdot \frac{1}{2} \cdot \frac{2}{3} = 4$. On the other hand, 1, 5, 7, 11 are the only numbers from $\underline{12}$ that are coprime with 12.

14 Orders

> The two following sections contain the most common concepts concerning orders, equivalence relations, and partitions. While these topics are abstract, their usefulness for most mathematical courses outweighs the students' likely frustration. To make things more bearable, the Instructor may widely use previous sections as a source of concrete examples and, on the other hand, apply combinatorics to finite orders and equivalences. To attain the latter goal, we count partitions of a finite set and prove Dilworth's Theorem.

Besides functions, there are two other important classes of special binary relations: those of *orders* and of *equivalences*. While being inherently abstract, these concepts admit natural set-theoretic definitions. A binary relation R is called:

1. *reflexive for a set* Z iff $\forall x \in Z\,(x,x) \in R$;

2. *irreflexive* iff $\forall x\,(x,x) \notin R$;

3. *symmetric* iff $\forall x \forall y\,(xRy \Longrightarrow yRx)$;

4. *antisymmetric* iff $\forall x \forall y\,\big((xRy \wedge yRx) \Longrightarrow x = y\big)$;

5. *transitive* iff $\forall x \forall y \forall z\,\big((xRy \wedge yRz) \Longrightarrow xRz\big)$.

Clearly, the reflexivity property is relative to a parameter Z, while all the others are inherent in R. A relation R on a set A is just called *reflexive* when it is reflexive for A.

In 'arrow' terms, reflexivity endows each point of A with a loop; irreflexivity means absence of any loops; symmetry secures a converse for every arrow; antisymmetry guarantees loops to be only possible arrows with a converse; finally, transitivity procures a one arrow 'bypass' for every two arrow path.

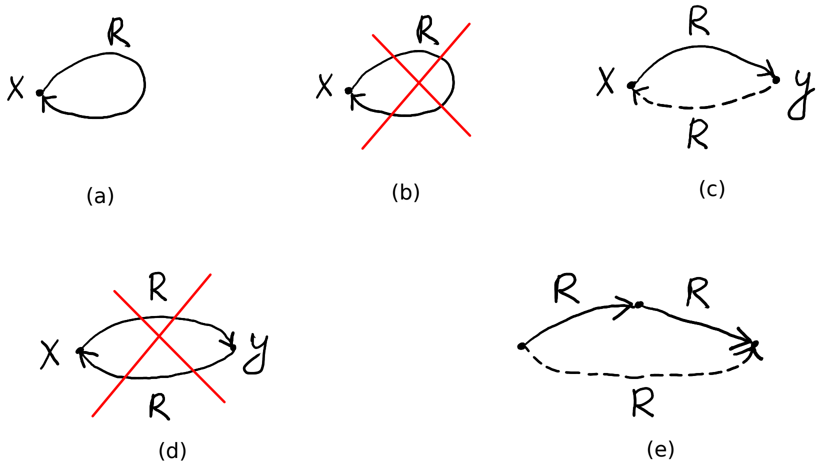

Figure 16: (a) Reflexivity: a loop at each point; (b) irreflexivity: no loops; (c) symmetry; (d) antisymmetry: no such cycle unless $x = y$; (e) transitivity.

Example 14.1. The relation id_A on a set A is reflexive, symmetric, antisymmetric, and transitive. The relation $\varnothing$ is irreflexive, symmetric, antisymmetric, and transitive. Indeed, say, the assumption $(x,y),(y,z) \in \varnothing$ is always false and implies thus anything—including the statement $(x,z) \in \varnothing$, whence transitivity of $\varnothing$ follows.

The relations $<$ and $\leq$ on the set $\mathbb{N}$ are transitive and antisymmetric (the assumptions $x < y$ and $y < x$ are inconsistent, so they imply $x = y$), while $<$ is irreflexive and $\leq$ is reflexive.

The relation $\subseteq$ on a set $\mathcal{P}(A)$ is reflexive, antisymmetric, and transitive. The relations $\sim$ and $\lesssim$ on $\mathcal{P}(A)$ are reflexive and transitive, whereas $\sim$ is symmetric. If the set A has at least two distinct elements a and b, then $\lesssim$ is neither symmetric ($\varnothing \lesssim \{a\}$ but $\{a\} \not\lesssim \varnothing$), nor antisymmetric ($\{a\} \lesssim \{b\}$ and $\{b\} \lesssim \{a\}$ but $\{a\} \neq \{b\}$).

It is convenient to describe our properties in terms of algebraic operations.

Lemma 14.2. *A relation $R \subseteq A^2$ is*

1. *reflexive* $\iff \mathrm{id}_A \subseteq R$;

2. *irreflexive* $\iff \mathrm{id}_A \cap R = \varnothing$;

3. *symmetric* $\iff R \subseteq R^{-1} \iff R = R^{-1} \iff R^{-1} \subseteq R$;

4. *antisymmetric* $\iff R \cap R^{-1} \subseteq \mathrm{id}_A$;

5. *transitive* $\iff R \circ R \subseteq R$.

Proof. Let us check the last three statements. If R is symmetric and $(x, y) \in R$, then, by definition, $(y, x) \in R$, whence $(x, y) \in R^{-1}$. So, $R \subseteq R^{-1}$. But this implies $R^{-1} \subseteq (R^{-1})^{-1} = R$ and $R = R^{-1}$, which, in turn, yields symmetry.

Given $R \cap R^{-1} \subseteq \mathrm{id}_A$, for each x and y, from xRy and $xR^{-1}y$, it follows that $x\,\mathrm{id}_A\,y$. Equivalently, from xRy and yRx, it follows that $x = y$. The latter statement but means R is antisymmetric.

Let R be transitive and $(x, y) \in R \circ R$. Then there exists z such that $(x, z) \in R$ and $(z, y) \in R$. By transitivity, $(x, y) \in R$. For the other direction, let $R \circ R \subseteq R$, xRz, and zRy. Then $(x, y) \in R \circ R$ and xRy. Hence, R is transitive. □

Example 14.3. When is a relation $R \subseteq A^2$ both symmetric and antisymmetric? For every such A, one has $R = R \cap R = R \cap R^{-1} \subseteq \mathrm{id}_A$. Hence, $R \subseteq \mathrm{id}_A$. Conversely, suppose $R \subseteq \mathrm{id}_A$. Then $R \cap R^{-1} \subseteq R \subseteq \mathrm{id}_A$, whence R is antisymmetric. Also, if xRy, then $x = y$, which gives yRx. Therefore R is symmetric.

Example 14.4. If relations P and Q are transitive, then $P \cap Q$ is transitive too. Indeed, by Exercise 9.22 and Example 9.26, we have

$$\begin{aligned}(P \cap Q) \circ (P \cap Q) &\subseteq ((P \cap Q) \circ P) \cap ((P \cap Q) \circ Q) \subseteq \\ &\subseteq (P \circ P) \cap (Q \circ P) \cap (P \circ Q) \cap (Q \circ Q) \subseteq \\ &\subseteq (P \circ P) \cap (Q \circ Q) \subseteq P \cap Q.\end{aligned}$$

We have applied transitivity of P and Q for the last step.

Exercise 14.5. Prove that a relation $R \circ R^{-1}$ is always symmetric.

Exercise 14.6. Let relations P and Q be symmetric. Prove that the relation $P \circ Q$ is symmetric iff $P \circ Q = Q \circ P$.

Order relations. A relation R on a set A is called a *strict partial order* (or simply a *strict order*) *on* A, if R is both irreflexive and transitive. Sometimes, they say *'ordering'* instead of *'order'*.

Example 14.7. For any set A, the relation $\varnothing$ is a strict order. The relations $<$ and $>$ on the set $\mathbb{N}$ are strict orderings, while $\leq$ is not for its being reflexive. The relation $\subsetneq$ on a set $\mathcal{P}(A)$ is a strict ordering as well.

Example 14.8. Let A be a set, $f\colon A \to \mathbb{N}$ and a relation $R \subseteq A^2$ be such that $xRy \iff f(x) < f(y)$ for every $x, y \in A$ (think of $f(x)$ as the 'price' of a 'product' $x \in A$). Then the relation R is a strict partial ordering. Evidently, one can substitute any strict ordering for the ordering $<$ on $\mathbb{N}$. The function $f\colon A \to B$ thus 'translates' an order from B onto A (or, as they usually say, f *induces* an order on A).

Remark 14.9. A strict order R is always antisymmetric. Indeed, if xRy and yRx, then xRx due to transitivity. As R is irreflexive, this leads us to a contradiction, which yields $x = y$. We have proved even more: every strict ordering is *asymmetric*, that is, if xRy, then yRx does not hold.

A relation R on a set A is called a *non-strict (partial) order(ing) on* A, if R is reflexive, transitive, and antisymmetric.

Example 14.10. On any set A, the relation id_A is a non-strict order. The relations $\leq$ and $\geq$ on $\mathbb{N}$ are non-strict orders as well, while the irreflexive relation $<$ is not. The relation $\subseteq$ on a set $\mathcal{P}(A)$ and the divisibility relation $\mid$ on $\mathbb{N}$ are non-strict orderings of the respective sets. Notice that the divisibility relation is *not* an ordering of $\mathbb{Z}$ for it is not antisymmetric: $1 \mid -1$ and $-1 \mid 1$ but $1 \neq -1$.

Exercise 14.11. In the context of Example 14.8, put $xQy \iff f(x) \leq f(y)$ for all $x, y \in A$. Is it necessary for Q to be a non-strict partial order on A?

We see that for every $n, m \in \mathbb{N}$ it holds either $n \leq m$ or $m \leq n$, yet this is not the case for the ordering $\mid$: indeed, $2 \nmid 3$ and $3 \nmid 2$. The elements 2 and 3 are thus called *incomparable* in the sense of $\mid$. It is this possibility which the expression '*partial* order' implies.

Exercise 14.12. If P and Q are both strict (or non-strict) orders on A, then the relations P^{-1} and $P \cap Q$ are such orderings as well.

If R is an ordering on A (either strict or not), the pair (A, R) is called a *partially ordered set* (or *poset*). If the relation R is clear from the context, the set A can be called a poset itself. Anyway, A is known as the *ground set* of the poset (A, R).

Strict and non-strict orders. Strict and non-strict orderings of A are closely interrelated. Namely, each strict order P has a natural non-strict 'counterpart' $\varphi(P)$, whereas every non-strict Q has a strict 'counterpart' $\psi(Q)$.

We put $S(A) = \{R \in \mathcal{P}(A^2) \mid R \text{ is a strict order}\}$ and similarly define the set $N(A)$ of all non-strict orders on A. Consider the functions $\varphi\colon S(A) \to \mathcal{P}(A^2)$ and $\psi\colon N(A) \to \mathcal{P}(A^2)$ such that

$$\varphi(P) = P \cup \mathrm{id}_A \quad \text{and} \quad \psi(Q) = Q \smallsetminus \mathrm{id}_A$$

for each $P \in S(A)$ and $Q \in N(A)$. In other words, we let

$$(x, y) \in \varphi(P) \iff xPy \vee x = y \qquad \text{and} \qquad (x, y) \in \psi(Q) \iff xQy \wedge x \neq y.$$

Theorem 14.13. *For every $P \in S(A)$ and $Q \in N(A)$, it holds that:*

1. *$\varphi(P) \in N(A)$ and $\psi(\varphi(P)) = P$;*

2. *$\psi(Q) \in S(A)$ and $\varphi(\psi(Q)) = Q$.*

The proof is straightforward but tedious a little.

As one can readily see, the functions $\psi\colon N(A) \to S(A)$ and $\varphi\colon S(A) \to N(A)$ are inverse to each other. This implies the following

Corollary 14.14. *The function $\varphi\colon S(A) \to N(A)$ is bijective and $\psi = \varphi^{-1}$.*

Exercise 14.15. Prove that $\varphi(P^{-1}) = (\varphi(P))^{-1}$ for every $P \in S(A)$.

We see that strict and non-strict orderings, despite being different things, enjoy a natural bijection between them (that is, each set A has 'as many' strict orderings as it has non-strict ones). We are already familiar with the pair of ordering $(<, \leq)$ on $\mathbb{N}$ or the pair $(\subsetneq, \subseteq)$ on an arbitrary set $\mathcal{P}(A)$.

When considering a strict order, this allows us to have the respective non-strict order at our disposal, and vice versa. Particularly, we can freely mention a *partial ordering* without specifying which version thereof we are speaking about. In our Course, all orderings are strict by default.

When denoting a strict order, they usually employ the symbol $<$ or the like. We shall suppose that in the pairs $(<, \leq)$, $(\prec, \preceq)$, etc., the first symbol stands for the strict version of an ordering, while the second one does for the non-strict one, i. e., $\leq\, = \varphi(<)$ and $<\, = \psi(\leq)$. Sometimes, they use symbols like $\subsetneq$ and $\lneq$ to denote a strict ordering, if things are to be clarified further.

It is likewise natural to identify our poset as a pair $(A, <)$ or $(A, \leq)$ with the triplet $(A, <, \leq)$.

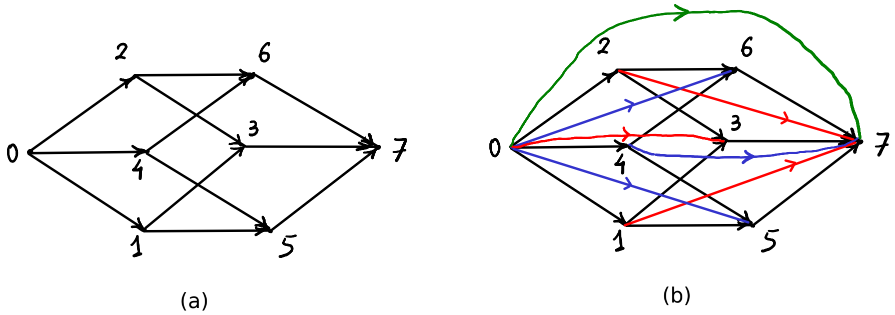

Figure 17: (a) This diagram does *not* depict a strict order for it lacks transitivity. (b) Nevertheless, one can easily restore the arrows required by transitivity, which are shown in color. In practice, they routinely use (a)-style diagrams to represent orders for the sake of clarity (so called *Hasse diagrams*). The 'colored' arrows are then implied (along with loops at each point in the case of a non-strict order).

Maxima and minima. For any poset $(A, <, \leq)$, an element $x \in A$ is called $(<\text{-})$*maximal* if

$$\forall y \in A\ \neg x < y.$$

In 'arrow' terms, a maximal element is such that no arrow starts from it. We similarly define a $(<)$-*minimal* element x, so that

$$\forall y \in A\ \neg y < x.$$

The set of $<$-maximal (which are exactly the same as $\leq$-maximal) elements of A (also called *maxima* (plural for *maximum*) thereof) is denoted by $\max_< A$ or just by $\max A$.

Example 14.16. As expected, one gets $\min_< \mathbb{N} = \{0\}$ and $\max_< \mathbb{N} = \varnothing$ but also $\max_> \mathbb{N} = \{0\}$ and $\min_> \mathbb{N} = \varnothing$. Exactly so! Indeed, there exists no $y \in \mathbb{N}$ such that $0 > y$.

Exercise 14.17. Let R be an ordering on a set A. Prove that $\min_R A = \max_{R^{-1}} A$ and $\max_R A = \min_{R^{-1}} A$.

Example 14.18. Consider a poset $(A, \varnothing)$. One has $\min_\varnothing A = A = \max_\varnothing A$. Indeed, each element of A is neither greater, nor lesser than any other. So, an ordering may have any number of minima or maxima.

Example 14.19. Consider the set $A = \mathcal{P}(\mathbb{N}) \setminus \{\varnothing, \mathbb{N}\}$ as ordered by $\subseteq$. What are the sets $\min A$ and $\max A$?

It is easy to see that $x \subsetneq \{n\}$ is possible for no $x \in A$ nor $n \in \mathbb{N}$. Hence, each singleton $\{n\}$ is a minimum in A. On the contrary, if a set $y \in A$ has at least two distinct elements n and m, then $\{m\} \subsetneq y$ and y is thus not minimal. So, $\min A = \{\{n\} \mid n \in \mathbb{N}\}$. Similarly, $\max A = \{\mathbb{N} \setminus \{n\} \mid n \in \mathbb{N}\}$.

Exercise 14.20. What are the sets $\min_| \mathbb{N}$ and $\max_| \mathbb{N}$? And what if one restricts this ordering to the set $\mathbb{N} \setminus \{0, 1\}$?

Let $(A, <)$ be a poset. It is natural to make the notions of maximum and minimum relative to a set $B \subseteq A$ by letting $\max_< B = \{x \in B \mid \forall y \in B\, x \not< y\}$ and similarly for $\min_< B$.

An element $x \in B$ is called a *greatest element* of the poset's $(A, <)$ subset B if $\forall y \in B\ y \leq x$; we likewise call $x \in B$ a *least element* of B if $\forall y \in B\ x \leq y$.

Lemma 14.21. *Let $(A, <)$ be a poset. If x is a greatest element in $B \subseteq A$, then* $\max_< B = \{x\}$. *Consequently, a greatest element of B is unique, so it is indeed* the *greatest element.*

Proof. Assume that $x \notin \max B$, so $x < y$ for some $y \in B$. On the other hand, we get $y \leq x$, which yields either $y = x$ or $y < x$ (since $\leq = \varphi(<)$). In the former case, it is immediate that $y < y$; in the latter case, the same holds by transitivity. This contradicts the fact $<$ is irreflexive. Thus, $x \in \max B$.

Now, suppose that $x' \in \max B$. Then $x' \not< x$ but $x' \leq x$. Hence, $x' = x$. □

Exercise 14.22. Suppose that $\max_< A = \{x\}$. Is x always a greatest element of $(A, <)$?

Example 14.23. Consider the set $A = \mathcal{P}(\mathbb{N}) \setminus \{\varnothing, \mathbb{N}\}$ as ordered by $\subseteq$. As we know, this set has multiple maxima and minima, hence it has neither greatest nor least element. Now consider $B = \{X \in A \mid \{1, 2, 3\} \subseteq X\}$. Clearly, the element $\{1, 2, 3\}$ is the least one in B. On the other hand, it is easy to see that

$$\max B = B \cap \max A = \{\mathbb{N} \setminus \{n\} \mid n \in \mathbb{N} \setminus \{1, 2, 3\}\}.$$

In particular, B does not have a greatest element.

Let $(A, <)$ be a poset and $B \subseteq A$. An element $x \in A$ is called an *upper bound* of the set B, if $y \leq x$ for every $y \in B$. A *lower bound* of B is defined similarly.

Example 14.24. The upper bounds of the set $\{2, 3, 7\}$ in the poset $(\mathbb{N}, |)$ are all the natural multiples of $42 = \operatorname{lcm}(2, 3, 7)$. The only lower bound of that set is $1 = \gcd(2, 3, 7)$.

Exercise 14.25. Let $(A, <)$ be a poset and $B, C \subseteq A$. Let $B^\triangle$ be the set of upper bounds of B and $B^\triangledown$ be the set of lower bounds of B. Prove that:

1. $(B \cup C)^{\vartriangle} = B^{\vartriangle} \cap C^{\vartriangle}$; $(B \cup C)^{\triangledown} = B^{\triangledown} \cap C^{\triangledown}$;

2. if $B \subseteq C$, then $C^{\vartriangle} \subseteq B^{\vartriangle}$ and $C^{\triangledown} \subseteq B^{\triangledown}$;

3. $B \subseteq B^{\vartriangle\triangledown} \cap B^{\triangledown\vartriangle}$;

4. $B^{\vartriangle} = B^{\vartriangle\triangledown\vartriangle}$; $B^{\triangledown} = B^{\triangledown\vartriangle\triangledown}$.

An element $x \in A$ is a *supremum* of the set B if x is the least upper bound of B (i.e., the least element of $B^{\vartriangle}$). Similarly, x is an *infimum* of B if it is the greatest lower bound thereof. As a least element (as well as a greatest) must be unique, the following notation does make sense: $x = \inf B$ ($x = \sup B$, respectively), provided the infimum (supremum) exists.

Remark 14.26. A set B has a greatest element iff $\sup B \in B$, and $\sup B$ is that very element. Things are similar for $\inf B$.

Exercise 14.27. How does $\sup B$ relate to $\inf B^{\vartriangle}$ (if both exist)?

Example 14.28. For the natural ordering of real numbers and the set $B = \{\frac{1}{n} \mid n \in \mathbb{N}_+\}$, one has $\sup B = 1 \in B$ and $\inf B = 0 \notin B$.

Example 14.29. Consider the order $P = \{(0,2),(0,3),(1,2),(1,3)\}$ on the set $A = \{0,1,2,3\}$. Letting $B = \{2,3\}$, we obtain $B^{\triangledown} = \{0,1\}$; yet these lower bounds are incomparable, hence B has no *greatest* lower bound, although each lower bound is a *maximal* one.

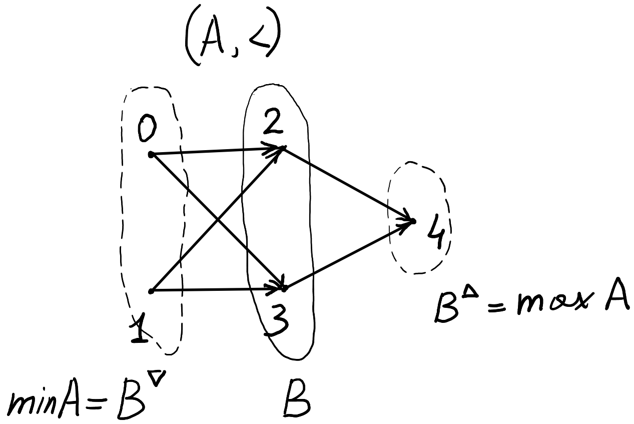

Figure 18: The order $(A,<)$ shown above has two minimal elements $0,1$ and one maximum 4, which is the greatest element. The order has no least element. For the set $B = \{2,3\}$, we have $B^{\triangledown} = \{0,1\}$ and $B^{\vartriangle} = \{4\}$.

Example 14.30. Let us consider the poset $(\mathcal{P}(A), \subseteq)$. It is easy to see that for $X = \{B_1, B_2\}$, one gets $\sup X = B_1 \cup B_2$ and $\inf X = B_1 \cap B_2$. Indeed, it is clear that $B_i \subseteq B_1 \cup B_2$ for either i. On the other hand, if $B_1 \subseteq C$ and $B_2 \subseteq C$, then one obtains $B_1 \cup B_2 \subseteq C$ by Example 4.23. The argument for $\inf X$ is similar.

The union of sets B_1 and B_2 is thus their *least* common *super*set (i. e., $\subseteq$-upper bound), and their intersection is their *greatest* common *sub*set ($\subseteq$-lower bound).

Exercise 14.31. In the setting of the above example, prove that $\sup X = \cup X$ for any $X \subseteq \mathcal{P}(A)$.

An important class of posets consists of so-called *lattices*, i. e., of such posets $(A, <)$ where for every $x, y \in A$, there exist both $\sup\{x, y\}$ and $\inf\{x, y\}$. For example, $(\mathcal{P}(A), \subseteq)$ is a lattice.

Exercise 14.32. Prove that the poset $(\mathbb{N}, |)$ is a lattice.

Linear orders. An order $<$ on a set A is called *linear* (or *total*[21]) if every two elements of A are comparable to each other, i. e.,

$$\forall x, y \in A \;\; x \leq y \vee y \leq x.$$

The poset $(A, <)$ is said to be a *linearly ordered set* if the order $<$ is linear.

Example 14.33. The natural orderings on $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, and $\mathbb{R}$ are linear, unlike $\subseteq$ on $\mathcal{P}(A)$ (if the set A has at least two distinct elements) or $|$ on $\mathbb{N}$.

Exercise 14.34. Each linear order is a lattice.

Remark 14.35. If an order $<$ on a set A is linear, then

$$x \not< y \iff y \leq x$$

for every $x, y \in A$. This implies that an element x is greatest (least) in the set $B \subseteq A$ iff x is maximal (minimal) in B. In particular, a linearly ordered set can have no more than one maximum (minimum). It is thus justified to write $x = \max B$ when x is maximal in some $B \subseteq A$.

> It is important to show the students that 'our' supremum is just a general version of what they may have come across in their Calculus course.

Furthermore, for a linearly ordered set, suprema (and infima) can be defined as minimal upper (maximal lower) bounds:

$$x = \sup B \iff x \in B^{\triangle} \wedge \forall z \in B^{\triangle} \; z \not< x \iff$$
$$x \in B^{\triangle} \wedge \forall z < x \; z \notin B^{\triangle} \iff \forall y \in B \; y \leq x \wedge \forall z < x \, \exists y \in B \; z < y.$$

This form of the definition is routinely used in Calculus courses for subsets of $\mathbb{R}$.

Let $(A, <)$ be a poset. A set $C \subseteq A$ is called a *chain* in A if

$$\forall x, y \in C \;\; x \leq y \vee y \leq x.$$

In other words, a chain is a subset whose every two elements are comparable or, equivalently, such that the order becomes linear when restricted to it. On the contrary, the set $D \subseteq A$ is called an *antichain*, if no two of its (distinct) elements are comparable, i. e.,

$$\forall x, y \in D \;\; x \not< y \wedge y \not< x.$$

Example 14.36. In the poset $(\mathbb{N}, |)$, the set $\{2^n \mid n \in \mathbb{N}\}$ is a chain, while any set of prime numbers is an antichain. The set $\{\mathbb{N}_{\geq k} \mid k \in \mathbb{N}\}$, where $\mathbb{N}_{\geq k} = \{n \in \mathbb{N} \mid n \geq k\}$, is a chain in $(\mathcal{P}(\mathbb{N}), \subseteq)$.

Exercise 14.37. When is a poset's subset both a chain and an antichain? Find all chains and all antichains in a linearly ordered set.

Exercise 14.38. In the poset $(\mathcal{P}(\mathbb{N}), \subseteq)$, find a non-empty chain that has no greatest element, nor least one.

[21] Clearly, a total order in this sense need *not* be a total binary relation, whereas that the latter would mean $\forall x \in A \, \exists y \, x < y$. For this reason, we prefer 'linear' to 'total' in this context.

Order isomorphism. Sometimes, two posets $\mathcal{A} = (A, <_A)$ and $\mathcal{B} = (B, <_B)$ are very similar to each other. For example, the natural numerical ordering $<$ aligns the sets $\{1, 2, 3\}$ and $\{2, 8, 15\}$ "in the same way": as $1 < 2 < 3$ and $2 < 8 < 15$, respectively, while this is not the case with infinite orders $(\mathbb{N}, <)$ and $(\mathbb{Z}, <)$, as the former is the only one that has a minimum: $0 < 1 < 2 < \ldots$, yet $\ldots < -2 < -1 < 0 < 1 < 2 < \ldots$.

In mathematics, one is usually interested in but the 'form' of an ordering, not in its very elements, whose nature may be irrelevant given their order. Say, the orderings $1 < 2 < 3$ and Moon $<$ Earth $<$ Sun are 'essentially' the same. So, a 'pure' order gets abstracted from a particular poset.

This is a very general situation. The notion of *isomorphism* (literally, "equality in form") is employed to treat it formally. Let us restrict ourselves to the case of posets. Two posets $\mathcal{A} = (A, <_A)$ and $\mathcal{B} = (B, <_B)$ are called *isomorphic* if there exists a bijection $\alpha\colon A \to B$ such that $x <_A y$ iff $\alpha(x) <_B \alpha(y)$ for every $x, y \in A$ (so, this α 'respects' the order). Such a mapping α is called an *isomorphism* from $\mathcal{A}$ to $\mathcal{B}$. We write $\mathcal{A} \overset{\alpha}{\cong} \mathcal{B}$ or, less explicitly, $\mathcal{A} \cong \mathcal{B}$ in this case.

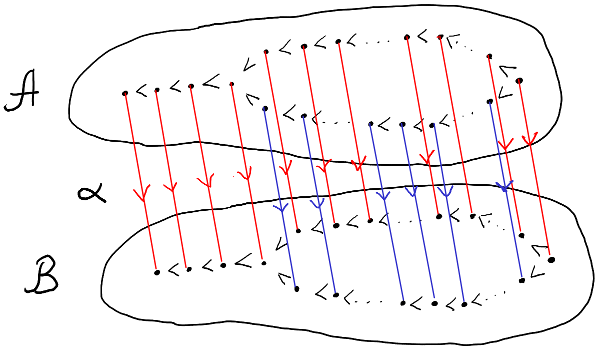

Figure 19: An order isomorphism: $\mathcal{A} \overset{\alpha}{\cong} \mathcal{B}$.

Remark 14.39. In general, it is natural to define *any* two pairs (A, P) and (B, Q), where $P \subseteq A^2$ and $Q \subseteq B^2$ (such pairs—not necessarily posets—are called *structures*) to be *isomorphic* if there exists a bijection $\alpha\colon A \to B$ such that xPy iff $\alpha(x)Q\alpha(y)$ for every $x, y \in A$. A structure on the set A may contain multiple relations $P_i \in A^{n_i}$ (not necessarily binary) and functions $f_i\colon A^{m_i} \to A$. It makes sense to think that diverse structures are *the* object of mathematics.

Exercise 14.40. Prove that if two structures (A, P) and (B, Q) are isomorphic and (A, P) is a poset, then (B, Q) is a poset as well.

Lemma 14.41. *For every posets $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$,*

1. $\mathcal{A} \overset{\mathrm{id}_A}{\cong} \mathcal{A}$;
2. *if* $\mathcal{A} \overset{\alpha}{\cong} \mathcal{B}$, *then* $\mathcal{B} \overset{\alpha^{-1}}{\cong} \mathcal{A}$;
3. *if* $\mathcal{A} \overset{\alpha}{\cong} \mathcal{B}$ *and* $\mathcal{B} \overset{\beta}{\cong} \mathcal{C}$, *then* $\mathcal{A} \overset{\beta\circ\alpha}{\cong} \mathcal{C}$.

Proof. Let us check the second statement. Clearly, α^{-1} is a bijection from B to A. Assume that $u <_B v$ for arbitrary $u, v \in B$. As α is surjective, we have $u = \alpha(x)$ and $v = \alpha(y)$ for some $x, y \in A$. But $\alpha(x) <_B \alpha(y)$ implies $x <_A y$ since α is an isomorphism. On the other hand, $\alpha^{-1}(u) = \alpha^{-1}(\alpha(x)) = x$ and $\alpha^{-1}(v) = \alpha^{-1}(\alpha(y)) = y$, whence $\alpha^{-1}(u) <_A \alpha^{-1}(u)$ as required. □

Example 14.42. One has $(\mathbb{Z}, <) \cong (\mathbb{Z}, >)$. Indeed, $x < y$ is equivalent to $-x > -y$ and the mapping $x \mapsto -x$ is a bijection $\mathbb{Z} \to \mathbb{Z}$. Hence, this mapping is an isomorphism required. On the other hand, $(\mathbb{N}, <) \ncong (\mathbb{Z}, <)$ [22]. Otherwise, there is an isomorphism $\alpha \colon \mathbb{N} \to \mathbb{Z}$. There exist elements $u \in \mathbb{Z}$ with $u < \alpha(0) \in \mathbb{Z}$ and $x \in \mathbb{N}$ such that $\alpha(x) = u$. From $\alpha(x) < \alpha(0)$, it follows that $x < 0$, which is impossible for any natural x. A contradiction.

Exercise 14.43. Prove that $(\mathbb{Q}, <) \ncong (\mathbb{Z}, <)$ and $(\mathbb{Q}, <) \ncong (\mathbb{R}, <)$.

Remark 14.44. It is straightforward to check that

$$(A, <_A) \overset{\alpha}{\cong} (B, <_B) \iff (A, \leq_A) \overset{\alpha}{\cong} (B, \leq_B).$$

Each isomorphism thus 'respects' the relation between strict and non-strict versions of an order.

Two isomorphic posets are *essentially* the same, i. e. they satisfy the same *order* properties. We have already seen that two isomorphic posets both contain a least element or both do not, while isomorphisms map those least elements to each other. This is also applicable to *every* property that is defined in order terms (the exact form of this statement is beyond the scope of our Course). Let us but take at look at some examples.

Lemma 14.45. *Suppose that $(A, <_A) \overset{\alpha}{\cong} (B, <_B)$. If $(A, <_A)$ is linearly ordered, then $(B, <_B)$ is linear as well. For every $X \subseteq A$, it holds that $\max_{<_B} \alpha[X] = \alpha[\max_{<_A} X]$; furthermore, $\sup_{<_B} \alpha[X] = \alpha(\sup_{<_A} X)$ if $\sup_{<_A} X$ exists. Similar statements are true for minima and infima.*

Proof. We check just the claim about suprema. Let $\sup_{<_A} X$ exist. If $u \in \alpha[X]$, then $u = \alpha(x)$ for some $x \in X$. From $x \leq_A \sup_{<_A} X$, it follows that $u \leq_B \alpha(\sup_{<_A} X)$. On the other hand, let $v = \alpha(y)$ be an arbitrary upper bound of the set $\alpha[X]$. Then for every $x \in X$, we have $\alpha(x) \leq_B \alpha(y)$, whence $x \leq_A y$. Thus, $\sup_{<_A} X \leq_A y$ and $\alpha(\sup_{<_A} X) \leq_B v$. □

Exercise 14.46. Prove that for every poset $(A, \leq)$ there exists a set $S \subseteq \mathcal{P}(A)$ such that $(A, \leq) \cong (S, \subseteq)$. In other words, each ordering looks like the inclusion order on a suitable subset family.

[22] Clearly, we abuse the notation here since the symbol $<$ in the left-hand side stands for not the same order as it does in the right-hand side. Otherwise, we could use $<_{\mathbb{N}}$ and $<_{\mathbb{Z}}$, respectively, where $<_{\mathbb{N}} = <_{\mathbb{Z}} \cap \mathbb{N}^2$.

15 Equivalences and Partitions

A relation $R \subseteq A^2$ is called an *equivalence relation* (or just, an *equivalence*) on the set A if R is reflexive, symmetric, and transitive.

Example 15.1. The relations A^2 and id_A are equivalences. Moreover, $\mathrm{id}_A \subseteq E \subseteq A^2$ for every equivalence E on A.

The relation $\equiv_m = \{(x, y) \in \mathbb{Z} \mid x \equiv y \pmod m\}$ is an equivalence. The relation "x is parallel to y" is an equivalence on the set of lines in the plane if one defines each line to be parallel to itself. The relation $\sim$ of set equivalence is an equivalence indeed for any set (of sets).

Example 15.2. Let $f\colon A \to B$. Then the relation

$$\ker f = \{(x, y) \in A^2 \mid f(x) = f(y)\},$$

which is called the *equivalence kernel* (or just the *kernel*) of the function f, is indeed an equivalence on A. Clearly, $\ker f = \mathrm{id}_A$ iff f is injective.

Example 15.3. Let us show that a relation $R \subseteq A^2$ is an equivalence iff $(R \circ R^{-1}) \cup \mathrm{id}_A = R$.

Suppose R is an equivalence. Then $R = \mathrm{id}_A \circ R \subseteq R \circ R \subseteq R$, whence $R = R \circ R = R \circ R^{-1}$. So, $R = R \cup \mathrm{id}_A = (R \circ R^{-1}) \cup \mathrm{id}_A$.

For the opposite direction, assume our equation. Then, clearly, $\mathrm{id}_A \subseteq R$. One also gets $R^{-1} = (R \circ R^{-1})^{-1} \cup \mathrm{id}_A^{-1} = ((R^{-1})^{-1} \circ R^{-1}) \cup \mathrm{id}_A = R$. And finally, $R \circ R = R \circ R^{-1} \subseteq R$.

Quotient set. Intuitively, given an equivalence on A, one can naturally 'identify' equivalent elements thereof neglecting their 'insignificant' differences. This procedure results in a new set of 'classes' of elements from A. If one, say, identifies every two integers of the same parity (thus employing the equivalence $\equiv_2$), he gets exactly two classes: those of even and of odd numbers. If one identifies every two objects of the same color, he obtains a set of classes, which can rightly be called 'colors'. Such constructions are widespread in mathematics.

So, let E be an equivalence on a set A and $x \in A$. We call the set

$$[x]_E = \{z \in A \mid xEz\}$$

the *equivalence class* of x w.r.t. (with respect to) E. Every element $y \in [x]_E$ is called a *representative* of the class $[x]_E$. The set

$$A/E = \{\sigma \in \mathcal{P}(A) \mid \exists x \in A \; [x]_E = \sigma\} = \{[x]_E \mid x \in A\}$$

is called the *quotient set* of the set A by E.

Remark 15.4. As a matter of fact, equivalence classes are just set images under E: namely, $[x]_E = E[\{x\}]$. This observation justifies the widespread notation xE for $[x]_E$.

Example 15.5. The set A/A^2 is just $\{A\}$ as all elements of A are pairwise A^2-equivalent and go to the same class. The set A/id_A is the set of the singletons for all elements of A. Therefore $A/\mathrm{id}_A \sim A$.

As we know, $x \equiv_m y$ iff the numbers x and y leave identical remainders after dividing by m. Hence, the class $[x]_{\equiv_m}$ consists of all integers with the same remainder as x has. After dividing by m, the possible remainders are $0, 1, \ldots, m-1$. Then $\mathbb{Z}/{\equiv_m} \sim \{0, 1, \ldots, m-1\}$.

Observe by the way that $\equiv_m$ equals $\ker r_m$, where $r_m\colon x \mapsto$ the remainder after dividing x by m. An exercise below shows this is a general situation.

Lemma 15.6. *Let E be an equivalence on A. Then for every $x, y \in A$ the following hold:*

1. $x \in [x]_E$;

2. $[x]_E \cap [y]_E \neq \varnothing \iff xEy \iff [x]_E = [y]_E$.

Proof. The first statement follows from xEx. For the second one, let us assume $z \in [x]_E \cap [y]_E$. Then xEz and yEz, whence zEy, and xEy by symmetry and transitivity. Let, in turn, xEy and $z \in [x]_E$, i. e., xEz. Likewise we get yEz. So, $[x]_E \subseteq [y]_E$. The converse inclusion is similar. Finally, suppose that $[x]_E = [y]_E$. By the first statement, obtain $x \in [x]_E \cap [y]_E \neq \varnothing$. □

Exercise 15.7. For every set A and equivalence $E \subseteq A^2$, there exist a set B and a surjection $f : A \to B$ such that $E = \ker f$.

Partitions. As one can see, equivalence classes 'cover' the whole set A, and every two distinct classes are disjoint. This way, each equivalence E 'partitions' the set A into pairwise disjoint 'pieces'. More formally, we call a set $\Sigma \subseteq \mathcal{P}(A)$ a *partition* of the set A if

$$\varnothing \notin \Sigma, \quad \cup\Sigma = A \quad \text{and} \quad \forall \sigma, \tau \in \Sigma\,(\sigma \cap \tau \neq \varnothing \Longrightarrow \sigma = \tau).$$

Example 15.8. Lemma 15.6 shows that each quotient set A/E is a partition of A. Let $\mathbb{R}_-$ be the set of all negative reals. Then $\{\mathbb{R}_-, \{0\}, \mathbb{R}_+\}$ is a partition of $\mathbb{R}$. The set of all circles centered at 0 of every possible radius $r \geq 0$ (the circle of radius 0 equals its single center point) is a partition of the plane.

Exercise 15.9. Describe all the partitions of the set $\varnothing$.

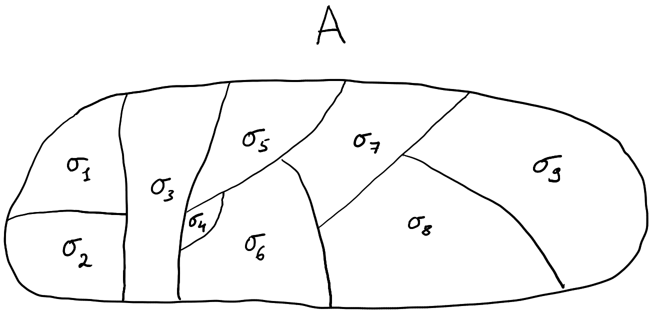

Figure 20: A partition $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_9\}$ of a set A. This Σ is finite as it has just nine elements.

It turns out that not only each quotient set is a partition, but each partition is the quotient by a suitable equivalence, so that there exists quite a natural bijection between objects of these two kinds.

So, let $Eq(A)$ be the set of all equivalences on a set A, and $\Pi(A)$ be the set of all partitions of A. Consider the functions $\pi : Eq(A) \to \mathcal{P}(\mathcal{P}(A))$ and $\varepsilon : \Pi(A) \to \mathcal{P}(A^2)$ such that

$$\pi(E) = A/E \quad \text{and} \quad \varepsilon(\Sigma) = \{(x, y) \in A^2 \mid \exists \sigma \in \Sigma\,(x \in \sigma \wedge y \in \sigma)\}$$

for every $E \in Eq(A)$ and every $\Sigma \in \Pi(A)$. In other words, the relation $\varepsilon(\Sigma)$ is composed of all such pairs whose both coordinates belong to the same element ('piece') of the partition Σ.

Theorem 15.10. *For every $E \in Eq(A)$ and $\Sigma \in \Pi(A)$ the following hold:*

1. $\pi(E) \in \Pi(A)$ *and* $\varepsilon(\pi(E)) = E$;

2. $\varepsilon(\Sigma) \in Eq(A)$ *and* $\pi(\varepsilon(\Sigma)) = \Sigma$.

Corollary 15.11. *The function $\pi : Eq(A) \to \Pi(A)$ is bijective and $\varepsilon = \pi^{-1}$.*

Counting equivalences. Let A be a finite set of size n. How many distinct equivalence relations on A are possible? Due to Corollary 15.11, it suffices to count all the partitions of A. The number $|Eq(A)| = |\Pi(A)|$ is denoted by B_n and called the n-th *Bell number*. In analogy to our "just the size matters" principle, it is clear that this number depends on n but not on the exact nature of the set A (as $A \sim A'$ implies $Eq(A) \sim Eq(A')$—check this!). So, formally one might define $B_n = |\Pi(\underline{n})|$.

Theorem 15.12. *$B_0 = 1$ and $\forall n \in \mathbb{N}\ B_{n+1} = \sum_{k=0}^{n} C_n^k B_k$.*

Proof. From Exercise 15.9, we know that $\varnothing$ is the only possible partition of $\underline{0} = \varnothing$ (the empty collection of non-empty 'pieces'). Hence the first statement.

Clearly, $n \in \underline{n+1}$. In order to specify a partition of $\underline{n+1}$, it is necessary and sufficient to determine the 'piece' $X \subseteq \underline{n+1}$ where n belongs to and fix some partition of $\underline{n+1} \setminus X$. As $n \in X$, it remains to determine the subset $X' = X \setminus \{n\}$ of $\underline{n}$ and fix a partition of $\underline{n} \setminus X'$. The cardinality of X' can be anything from 0 to n. If it is fixed to be k, one has as many partitions of $\underline{n+1}$ as elements in $\mathcal{P}_k(\underline{n}) \times \Pi(\underline{n} \setminus X')$ (this bijection is easy), which results in the number $C_n^k B_{n-k}$.

For distinct values of k, our partitions must be distinct. So, making k run over its range, we obtain $B_{n+1} = \sum_{k=0}^{n} C_n^k B_{n-k}$ by the rule of sum. Since $C_n^k = C_n^{n-k}$, we have $B_{n+1} = \sum_{k=0}^{n} C_n^{n-k} B_{n-k} = \sum_{k=0}^{n} C_n^k B_k$ after renaming $n-k$ into k. □

Example 15.13. We obtain one by one: $B_1 = C_0^0 B_0 = 1 \cdot 1 = 1$; $B_2 = C_1^0 B_0 + C_1^1 B_1 = 1 + 1 = 2$; $B_3 = C_2^0 B_0 + C_2^1 B_1 + C_2^2 B_2 = 1 + 2 + 2 = 5$. Here are all the five possible partitions of $\underline{3}$: $\{\{0\},\{1\},\{2\}\}$; $\{\{0\},\{1,2\}\}$; $\{\{0,1\},\{2\}\}$; $\{\{0,2\},\{1\}\}$; $\{\{0,1,2\}\}$.

Many combinatorial applications and interesting properties of Bell numbers are known today.

Dilworth's Theorem. Now, let us explore some combinatorial properties of orderings. This time, they will not be about counting but rather will reflect some 'extreme' possibilities for a finite poset structure.

Let $(A, <)$ be a poset. A partition Σ of A is called a *partition into chains* if each set $C \in \Sigma$ is a chain in the poset.

Example 15.14. Let the set $A = \{0, 1, 2, 3, 4\}$ be ordered by the relation $< = \{(0,1), (0,2), (0,3), (0,4), (1,4), (2,4), (3,4)\}$. Then $\Sigma_1 = \{\{2,4\},\{0\},\{1\},\{3\}\}$ and $\Sigma_2 = \{\{0,1,4\},\{2\},\{3\}\}$ are two possible partitions of A into chains.

What is the minimal possible size of such a partition? Clearly, the elements 1, 2, and 3 form an antichain in the poset, so no two of them can share a 'piece' in a partition into chains. Hence, no partition into less than three chains is possible.

On the other hand, Σ_2 is a partition of the least size allowed. Thus, it is 'optimal'.

Lemma 15.15. *Let $(A, <)$ be a finite poset, Σ be a partition thereof into chains, and D be an antichain in this poset. Then $|D| \leq |\Sigma|$.*

Proof. Let us map each element $x \in D$ to the (only) 'piece' C_x of the partition Σ that contains x. If $C_x = C_y$ for two distinct $x, y \in D$, then $x, y \in C_x$, so x and y are comparable to each other, which is not the case. Hence, the mapping $x \mapsto C_x$ from D to Σ is injective. By the Pigeonhole Principle, $|D| \leq |\Sigma|$. □

But given each antichain is of length r or less, is it always possible to partition the poset into no more than r chains? As a matter of fact, it is. At first, let us establish a simple auxiliary statement.

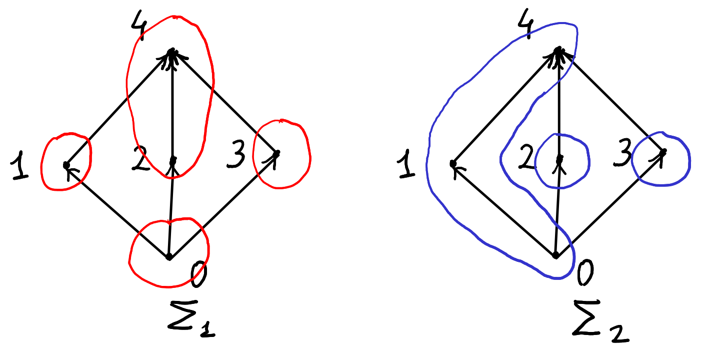

Figure 21: Example 15.14.

Lemma 15.16. *If $(A, <)$ is a finite poset, then for each $x \in A$, there exist $u \in \min A$ and $v \in \max A$ such that $u \leq x \leq v$.*

Proof. It suffices to prove this only for maxima, as the argument is otherwise similar. Let $n = |A|$ and assume the contrary. Since $x \notin \max A$, there exists $x_1 > x$. But $x_1 \notin \max A$ by our assumption. Hence, there is some $x_2 > x_1$. Repeating this argument n times, one gets a sequence $x = x_0 < x_1 < \ldots < x_n$. This can be viewed as a function $f\colon \underline{n+1} \to A$, where $f(k) = x_k$. If $k \neq m$, then, w. l. o. g., $k < m$, whence $f(k) < f(m)$ by transitivity. Thus, $f(k) \neq f(m)$ and f is an injection. We see that $\underline{n+1} \lesssim A \sim \underline{n}$, which contradicts the Pigeonhole Principle. □

Theorem 15.17 (Dilworth)**.** *Let $(A, <)$ be a finite poset. If D is an antichain in $(A, <)$ of the greatest possible length $|D|$, then there exists a partition Σ of that poset into $|D|$ chains.*

Proof. Let $n = |A|$. We prove the claim by (strong) induction on n. If $n = 0$, the only antichain is empty, which corresponds to the empty partition. Assume that $n > 0$ and the claim holds for every $n' < n$.

As A is non-empty, Lemma 15.16 provides us with two elements $m, M \in A$ such that $m \leq M$, $m \in \min A$, and $M \in \max A$ (these may coincide). Let r be the greatest length of an antichain in A and $A' = A \setminus \{m, M\}$. Suppose that $D \subseteq A'$ is a longest antichain in A'.

Clearly, it is an antichain in A as well, so $|D| \leq r$. If $|D| \leq r - 1$, there is a partition Σ' of the set A' into $|D|$ chains by the inductive hypothesis (for $|A'| < n = |A|$). Then $\Sigma = \Sigma' \cup \{\{m, M\}\}$ is a partition of A into exactly $|D| + 1 \leq r$ chains. On the other hand, $|\Sigma| \geq r$ by Lemma 15.15. So, Σ is just as desired.

Otherwise, $|D| = r$. In this case, let us define two sets

$$A_- = \{x \in A \mid \exists y \in D\ x \leq y\} \quad \text{and} \quad A_+ = \{x \in A \mid \exists y \in D\ y \leq x\}.$$

It is easy to see that $A_- \cap A_+ = D$. Clearly, $D \subseteq A_- \cap A_+$. Conversely, let $x \in A_- \cap A_+$. Then there are elements $y_1, y_2 \in D$ such that $y_1 \leq x$ and $x \leq y_2$, which implies $y_1 \leq y_2$. As D is an antichain, it must be that $y_1 = y_2$, whence $x = y_1 = y_2$ by antisymmetry. So, $x \in D$.

On the other hand, let us check $A_- \cup A_+ = A$. Clearly, $A_- \cup A_+ \subseteq A$. Now, assume that $x \in A$ but $x \notin A_-$ and $x \notin A_+$. This means that x is incomparable with any element of D and that $x \notin D$, so $D \cup \{x\}$ is an antichain of length $r + 1$ in A, which is not possible.

If $M \in A_-$, then $M \leq y$ for a certain $y \in D$. As M is maximal, we get $M = y \in D \subseteq A'$. But $M \notin A'$. So, $M \notin A_-$. Likewise we obtain $m \notin A_+$.

We see that $|A_-| < n = |A|$, while $D \subseteq A_- \subseteq A$ is a longest possible antichain in A_-. By the inductive hypothesis, this gives us a partition Σ_- of the set A_- into r chains. For every $y \in D$, we let C_y^- be the only chain from Σ_- containing y. As D is an antichain, one has $C_y^- \cap D = \{y\}$ and $C_y^- \neq C_z^-$ if $y \neq z$. So, the mapping $y \mapsto C_y^-$ from D to Σ_- is injective. That is, the set $\{C_y^- \mid y \in D\} \subseteq \Sigma_-$ has cardinality $|D| = r$ (cf. Remark 11.2). As Σ_- has the same cardinality r, it should be that $\{C_y^- \mid y \in D\} = \Sigma_-$ by the Rule of Sum. (Alternatively, one can refer to Theorem 12.8, which forces $y \mapsto C_y^-$ to be a surjection.)

By a similar argument, we obtain a partition $\Sigma_+ = \{C_y^+ \mid y \in D\}$ of the set A_+ whose cardinality is r. Now, consider the set $\Sigma = \{C_y^- \cup C_y^+ \mid y \in D\}$. Its elements are disjoint (and distinct) for distinct y's: indeed, if $y \neq z$,

$$(C_y^- \cup C_y^+) \cap (C_z^- \cup C_z^+) = (C_y^- \cap C_z^-) \cup (C_y^- \cap C_z^+) \cup (C_y^+ \cap C_z^-) \cup (C_y^+ \cap C_z^+).$$

Clearly, $C_y^- \cap C_z^- = \varnothing$ as Σ_- is a partition, and $C_y^- \cap C_z^+ = \varnothing$ since $x \in C_y^- \cap C_z^+$ implies $x \in A_- \cap A_+ \subseteq D$, whence $x = y$ and $x = z$, despite $y \neq z$. On the other hand, each $x \in A$ is covered by some element of Σ for $A = A_- \cup A_+$ and Σ_-, Σ_+ are partitions of A_- and A_+, respectively. Finally, each set $C_y^- \cup C_y^+$ is a chain, since from $x \in C_y^-$ and $z \in C_y^+$, it follows that $x \leq y \leq z$. As $\Sigma \sim D$, the set Σ is a required partition of A. □

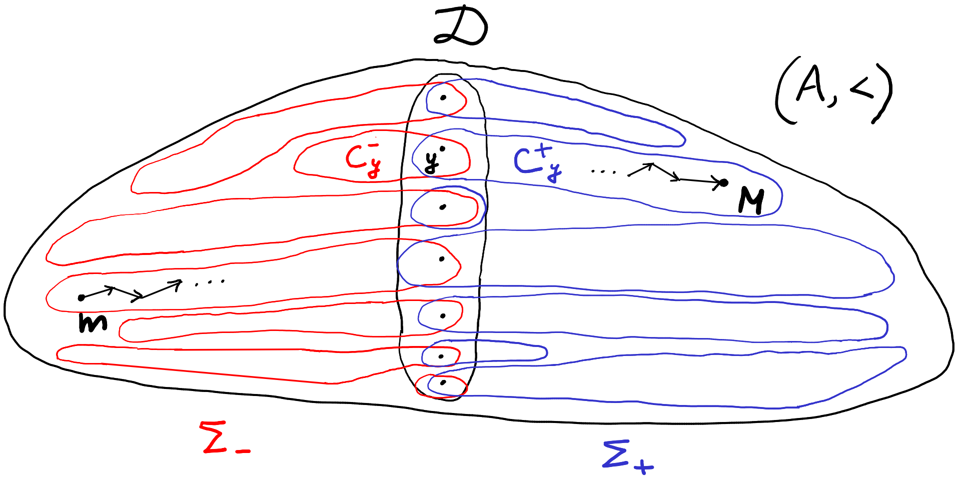

Figure 22: Proving Dilworth's Theorem.

In view of Lemma 15.15, the partition Σ just constructed is of the least possible size. So, combining the lemma and theorem, we obtain

Theorem 15.18 (Dilworth's Theorem, another form). *Suppose that $(A, <)$ is a finite poset. Then the greatest possible length of an antichain in $(A, <)$ equals the least possible size of a partition of A into chains.*

From Dilworth's Theorem, it is possible to extract some information about the *lengths* of chains in the poset.

Corollary 15.19. *Suppose that $(A,<)$ is a finite poset. If $|A| = nm+1$, then there is either an antichain of length $n+1$ or a chain of length $m+1$ in $(A,<)$.*

Proof. Assume the contrary. Then the longest antichain has at most n elements (otherwise, any longer antichain could be cut to $n+1$ elements). By Dilworth's Theorem, there exists a partition of A into no more than n chains. These chains are pairwise disjoint while each being of cardinality m or less. By the Rule of Sum, the union of all those chains has at most nm elements. On the other hand, that union equals A (for a *partition* it is) with $|A| > nm$. A contradiction. □

This result has the following concrete interpretation, which is interesting for its own sake.

Theorem 15.20 (Erdős–Szekeres)**.** *A sequence of $nm+1$ pairwise distinct integers has either an increasing subsequence of length $m+1$ or a decreasing subsequence of length $n+1$.*

Example 15.21. Among $7 = 2 \cdot 3 + 1 = 3 \cdot 2 + 1$ ordered integers: $-3, 1, -5, 8, 11, 0, 2$, one can spot both an increasing subsequence $-3, 1, 8, 11$ and a decreasing one $11, 0, 2$. Notice the lengths 4 and 3 are respectively greatest.

By the way, this result does not seem to be that exciting when nm is close to n (or m). Say, in a sequence of $8 = 7 \cdot 1 + 1$ there should be either a decreasing subsequence of length 8 (which would necessarily coincide with the whole sequence) or an increasing one of length 2, which is always the case unless the whole sequence is decreasing.

Proof. Assume we are given a sequence $a_1, a_2, \ldots, a_s$ where $s = nm+1$ and $i \neq j$ implies $a_i \neq a_j$. This can readily be considered as a set of pairs of the form (i, a_i). We routinely identify a_i with such a pair. Let us define an ordering $\prec$ on the set of pairs:

$$a_i \prec a_j \iff i < j \wedge a_i < a_j.$$

Clearly, this is indeed a strict ordering. Consider a set $X = \{(i_1, a_{i_1}), \ldots, (i_k, a_{i_k})\}$ and assume it is a chain in this ordering. W. l. o. g., we may suppose that $i_1 < i_2 < \ldots < i_k$ (i. e., we may sort the pairs by their first coordinates, which are clearly pairwise distinct). By the assumption, if $u < v$, then a_{i_u} and a_{i_v} are comparable, that is, either $a_{i_u} \prec a_{i_v}$, which means $a_{i_u} < a_{i_v}$, or $a_{i_v} \prec a_{i_v}$ which implies the false statement $i_v < i_u$. So, X is a chain iff $a_{i_1} < a_{i_2} < \ldots < a_{i_k}$.

Likewise, X is an antichain if for every $u < v$ the elements a_{i_u} and a_{i_v} are incomparable. As $i_u < i_v$, this is equivalent to $a_{i_u} \not< a_{i_v}$, that is, to $a_{i_u} > a_{i_v}$ since the numbers a_{i_u} and a_{i_v} are distinct. So, X is an antichain iff $a_{i_1} > a_{i_2} > \ldots > a_{i_k}$.

We see that $\prec$-chains correspond to increasing subsequences of the same length, whereas $\prec$-antichains correspond to decreasing subsequences of the same length. An application of Corollary 15.19 finishes the proof. □

It is easy to see that Corollary 15.19 (hence, the Erdős–Szekeres Theorem) may be also proved from the following dual of Dilworth's Theorem, which is but easier to prove.

Theorem 15.22 (Mirsky)**.** *Suppose that $(A,<)$ is a finite poset. Then the greatest possible length m of a chain in $(A,<)$ equals the least possible size n of a partition of A into antichains.*

Proof. It is obvious that $m \leq n$ (no two distinct comparable elements can share an antichain). The converse equality is obtained by induction on $|A|$. Assume that C is a chain of length $m > 0$. By Lemma 15.16, there exists a least element x in C. Notice that $\min A$ is an antichain. Clearly, $C \cap \min A = \{x\}$, for if $x \notin \min A$, there exists some $y < x$ so that $C \cup \{y\}$ is a chain longer than C. Let $A' = A \setminus \min A$. Clearly, $C' = C \setminus \{x\}$ is a longest possible chain in A' (since *each* chain of length m in A intersects $\min A$). As $|A'| < |A|$, one gets a partition Σ' of A' into $m-1$ antichains by the IH. Then $\Sigma' \cup \{\min A\}$ is a partition of A into m antichains, whence $n \leq m$. □

16 Graphs: the Basics

> Our treatment of graphs is brief and mostly traditional. We try to employ the previously developed formalism as much as it is reasonably possible.

Graphs form a simple and popular formalism yet expressive enough for a great many areas in mathematics and applications. The basic idea is as follows: any two distinct 'individuals' from some fixed set are either 'connected by a link' or not. This may refer to, say, friendship (friends are connected), a railway network (some stations are connected), workers and skills (a worker is connected to each of his skills), etc. One should discern a pair of objects 'immediately' connected by one 'link' (we shall call them *adjacent*) from a pair of objects connected by a sequence of 'links' (like when there are some intermediate 'stations').

There are two main features here: (1) if x is adjacent to y, then y is adjacent to x; (2) no x is adjacent to itself. This gives rise to the following formal definition. A *graph* G is a pair (V, E), where V is an arbitrary non-empty set, whose elements are called *vertices*, and $E \subseteq V^2$ is a symmetric irreflexive binary relation on V, which is called the *adjacency* relation ('a vertex x is *adjacent* to a vertex y'). This Course (as well as most of the applications) is restricted to *finite graphs*, that is, we assume the set V to be finite.

As xEy is equivalent to yEx, it is reasonable to identify these two 'links'. So, a 2-element set $\{x, y\} \subseteq V$ is called an *edge* of the graph G iff xEy (equivalently, yEx). Abusing the notation, we will denote an edge $\{x, y\}$ as xy or yx. The number $|V|$ is called the *order* of the graph G and the number of edges in G is called the *size* of G. Easily, the size of G equals $|E|/2$.

A graph $G = (V, E)$ is called an (n, m)*-graph* if it is of order n and of size m. For each n, there is a maximal possible number of edges m but any combination of the two is otherwise possible.

Lemma 16.1. *For every natural $n > 0$ and m, there exists an (n, m)-graph iff $0 \le m \le C_n^2$.*

Proof. Consider an arbitrary (n, m)-graph $G = (V, E)$. As each edge is a 2-element subset of the set V with $|V| = n$, the number m of edges cannot exceed $|\mathcal{P}_2(\underline{n})| = C_n^2$.

For the other direction, let us consider the *complete graph* $K_n = (\underline{n}, \underline{n}^2 \smallsetminus \mathrm{id}_{\underline{n}})$ on n vertices. In K_n, $E = \underline{n}^2 \smallsetminus \mathrm{id}_{\underline{n}}$, so every two distinct vertices are adjacent. This means that the edge set of K_n is just $\mathcal{P}_2(\underline{n})$ and K_n is an (n, C_n^2)-graph. If $0 \le m < C_n^2$, one can easily obtain an (n, m)-graph by removing $C_n^2 - m$ edges from K_n. Removing an edge uv from a graph $G = (V, E)$ means formally that we let $E' = E \smallsetminus \{(u, v), (v, u)\}$ and get the graph $G' = (V, E')$. Clearly, E' is still symmetric and irreflexive, hence G' is a graph indeed. □

The set $N_G(x) = \{y \in V \mid xEy\}$ of all vertices adjacent to x in a graph $G = (V, E)$ is called the *neighborhood* of the vertex x in G (hence, y is a *neighbor* of x). The number $d_G(x) = |N_G(x)|$ is called the *degree* of x in G. As usual, the subscript 'G' is omitted when the graph is clear from the context. Since $N_G(x) \subseteq V \smallsetminus \{x\}$, we have $0 \le d(x) \le n - 1$ for an (n, m)-graph G. Both bounds are tight (the upper one is reached, say, for K_n).

If one sort the degrees of all the vertices from G in descending order, one obtains the *degree sequence* $(d(v_1), d(v_2), \ldots, d(v_n))$ of the graph G, where $d(v_i) \ge d(v_{i+1})$.

The overall number of handshakes made among a group of people equals half the sum of handshake numbers every person has partaken in.

Lemma 16.2 (Handshake Lemma)**.** *For every (n, m)-graph $G = (V, E)$, it holds that*

$$\sum_{v \in V} d_G(v) = 2m.$$

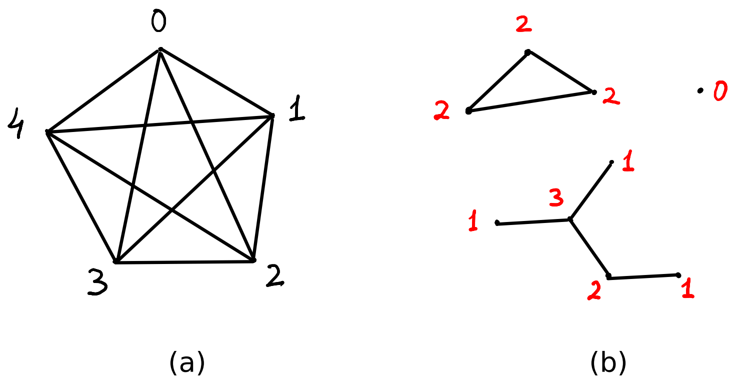

Figure 23: (a) The complete graph K_5. (b) A graph with the degree sequence $(3, 2, 2, 2, 2, 1, 1, 1, 0)$. Each vertex' degree is shown. According to Lemma 16.2, we have 7 edges here and $3+2+2+2+2+1+1+1+0 = 2 \cdot 7$.

Proof. By induction on m. If $m = 0$, then there are no edges in G and $d_G(v) = 0$ for each $v \in V$, whence the required statement follows. Assume that $m > 0$ and the statement holds for *every* (n, m')-graph G' with $m' < m$. Let us remove an arbitrary edge xy from G obtaining thus some new graph $G' = (V', E')$. Clearly, $V' = V$, $d_{G'}(x) = d_G(x) - 1$, $d_{G'}(x) = d_G(x) - 1$, $d_{G'}(v) = d_G(v)$ for every $v \in V \smallsetminus \{x, y\}$, and G' is a $(n, m-1)$-graph. By the IH, we get

$$\sum_{v \in V} d_G(v) = \sum_{v \in V' \smallsetminus \{x,y\}} d_{G'}(v) + (d_{G'}(x) + 1) + (d_{G'}(y) + 1) =$$

$$\sum_{v \in V} d_{G'}(v) + 2 = 2(m-1) + 2 = 2m.$$

□

Example 16.3. The lemma above provides one of the simplest necessary conditions for graph existence. Indeed, there is no graph with the degree sequence $(4, 3, 2, 2, 1, 1)$ as the sum of those degrees is odd.

Graph isomorphism. Like most of the other mathematical *structures* (orders, groups, etc.), graphs are usually considered *up to isomorphism.* This means that the nature of vertices is irrelevant provided the 'form' of the graph is fixed. Say, two triangles $\{0, 1, 2\}$ and $\{\pi, e, \mathrm{HSE}\}$ with edges $\{01, 12, 20\}$ and $\{\pi e, e\,\mathrm{HSE}, \mathrm{HSE}\,\pi\}$, respectively, are essentially the same.

Let us formalize this crucial idea. Two graphs $G = (V, E)$ and $G = (V', E')$ are said to be *isomorphic* if there exists a function $\varphi\colon V \to V'$ (which is then called an *isomorhism*) such that: (1) φ is a bijection $V \to V'$; (2) for every $x, y \in V$, one has $xEy \iff \varphi(x)E'\varphi(y)$, that is, φ 'respects' or 'copies' the adjacency structure of the graphs. In such a case, we write $G \cong G'$ or, to be more specific, $G \overset{\varphi}{\cong} G'$.

Loosely speaking, isomorphism is an equivalence relation. Indeed, it is straightforward to check the following

Lemma 16.4. *For every graphs G, H, F and functions φ, ψ, it holds that:*

1. *$G \overset{\mathrm{id}_{V_G}}{\cong} G$;*

2. *if $G \overset{\varphi}{\cong} H$, then $H \overset{\varphi^{-1}}{\cong} G$;*

3. *if $G \overset{\varphi}{\cong} H$ and $H \overset{\psi}{\cong} F$, then $G \overset{\psi\circ\varphi}{\cong} F$.*

An isomorphism preserves every notion defined for a graph in terms of its adjacency relation. Let us give a few examples for this phenomenon.

Lemma 16.5. *If $G \cong H$, the graphs G and H have the same:*

1. *order;*

2. *size;*

3. *degree sequence.*

Proof. Let us check the last statement. Suppose that $G \overset{\varphi}{\cong} H$. It suffices to prove that $d_G(v) = |N_G(v)| = |N_H(\varphi(v))| = d_H(\varphi(v))$ for every $v \in V_G$. As vE_Gu is equivalent to $\varphi(v)E_H\varphi(u)$, it is easy to see that $N_H(\varphi(v)) = \varphi[N_G(v)] \sim N_G(v)$, whence the required identity follows. □

Example 16.6. For the vertex set $\underline{4}$, let us consider two graphs: G with edges $\{02, 13\}$ and H with $\{01, 13\}$. These graphs do agree in both order and size but not in degree sequence, as it is $(1,1,1,1)$ for G but $(2,1,1,0)$ for H. Hence, $G \not\cong H$.

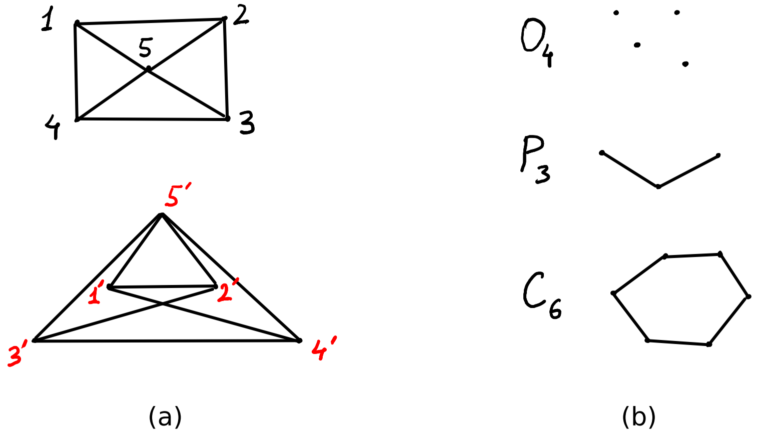

Figure 24: (a) Two isomorphic graphs. An isomorphism is shown as $n \mapsto n'$. (b) The graphs O_4, P_3, and C_6. These are defined up to isomorphism.

Exercise 16.7. Construct a pair of non-isomorphic graphs with the same degree sequence.

As we have already said, graphs are usually considered 'up to isomorphism', that is, ignoring their differences in vertex 'names'. Let us see some important examples. We have already introduced the complete graph K_n on n vertices. The 'complement' of this is the *empty graph* $O_n \cong (\underline{n}, \varnothing)$ that has no edges. (Notice that *every* graph isomorphic to $(\underline{n}, \varnothing)$ is denoted by O_n. We do not discern them from each other.)

The *path* (or *chain*) P_n on $n > 0$ vertices $1, 2, \ldots, n$ has all the edges from $\{12, 23, 34, \ldots, (n-1)n\}$. The *cycle* C_n, $n > 1$, on the same vertices is defined by the edge set $\{12, 23, 34, \ldots, (n-1)n, n1\}$.

Subgraphs and complements. It is sometimes obvious that one graph 'contains' (an isomorphic copy of) another one. For example, in K_4 one can find C_4^3 copies of K_3. Let us formalize this idea. A *graph* $G' = (V', E')$ is a *subgraph* of a graph $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$. So, a subgraph of G contains *some* of the vertices and *some* of the edges connecting those vertices in G.

A subgraph $G' = (V', E')$ of a graph $G = (V, E)$ is *induced* (*by the set* V') if G' contains *all* the edges between the vertices of V' in G, that is, $E' = E \cap (V' \times V')$. For example, in the K_4-graph with the edge set $\{12, 23, 34, 41, 13, 24\}$, the K_3-subgraph on the set $V' = \{1, 2, 3\}$ with the edges $12, 23, 31$ is induced (by the set V'), while the C_4-subgraph on the set $\{1, 2, 3, 4\}$ with the edges $12, 23, 34, 41$ is not.

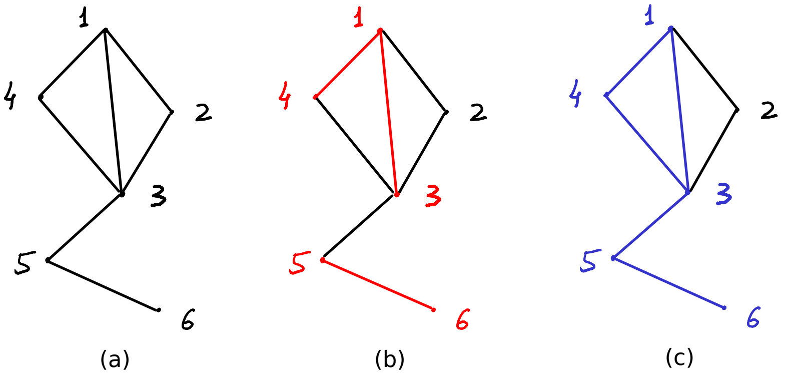

Figure 25: (a) A graph $G = (V, E)$. (b) Its subgraph $G' = (V', E')$ with vertices $V' = \{1, 3, 4, 5, 6\}$ is shown in red. (c) The subgraph $G'' = (V', E'')$ which is *induced* by the set V' is shown in blue.

Another useful construction is *complement*. The idea is as follows: keeping the vertex set unchanged, one removes every present edge from a graph but adds every edge which is possible but absent. As we have already mentioned, the complement of the complete graph K_n is the empty graph O_n, since K_n already contains all possible edges (i. e., every two *distinct* vertices are adjacent). On the other hand, the complement of the C_4-graph with the edges $12, 23, 34, 41$ has just the edges 13 and 24.

Formally, the complement $\bar{G} = (V', E')$ of a graph $G = (V, E)$ is defined by $V' = V$ and $E' = (V^2 \smallsetminus \mathrm{id}_V) \smallsetminus E$, that is, $xE'y \iff x \neq y \wedge \neg xEy$ for every $x, y \in V$. Observe that the complement $\bar{G}$ of an (n, m)-graph G is an $(n, C_n^2 - m)$-graph for $|E'| = |(V^2 \smallsetminus \mathrm{id}_V) \smallsetminus E| = n^2 - n - 2m = 2(C_n^2 - m)$.

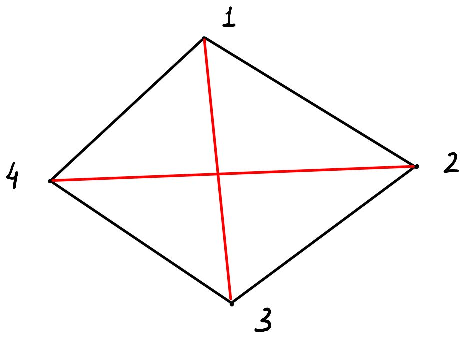

Figure 26: The graph C_4 (in black) with its complement $P_2 \sqcup P_2$ (in red). The latter symbol denotes the union two disjoint 'copies' of P_2 (these are disjoint indeed for they have no common vertex).

Paths and connectivity. Let us fix a graph $G = (V, E)$. A vertex sequence $p = v_1 v_2 \dots v_n$, $n > 0$, is called a *path* in G if $v_i E v_{i+1}$ for each i (formally, this *sequence* may be interpreted as either an n-tuple or a function $\underline{n} \to V$). The path p is *simple* if $v_i \neq v_j$ when $i \neq j$. By definition, the *length* $|p|$ of the path p is $n - 1$ (that is, the number of *edges* that link the vertices v_i together). We say that vertices x and y are *connected by the path* p if $v_1 = x$ and $v_n = y$. In particular, each vertex x is connected to itself by the single-vertex path x of length 0. We say that the path p *contains* an edge xy if $v_i = x$ and $v_{i+1} = y$ or $v_i = y$ and $v_{i+1} = x$ for some i. We write $x \overset{q}{\relbar\relbar} y$, where $q = u_1 \dots u_m$, for a path $x u_1 \dots u_m y$. Permitting $m = 0$, we will use the same symbol when $x = y$, so $x \overset{q}{\relbar\relbar} x$ may stand for the path x of length 0.

Lemma 16.8. *If vertices x and y are connected by a path, then there is a* simple *path that connects those vertices. Furthermore, a shortest such path is always simple.*

Proof. By the Least Number Principle, there exists a path p of the *least* possible length leading from x to y. Assume that p is not simple. Then there is a vertex z with at least two distinct occurences in p, so that $p = x u_1 \dots u_k z v_1 \dots v_l z w_1 \dots w_m y$, where $l \geq 1$. Clearly, the path $p' = x u_1 \dots u_k z w_1 \dots w_m y$ also connects x to y, while $|p'| < |p|$. A contradiction. □

A path $c = v_1 \dots v_n v_{n+1}$ is called a *cycle* in G if $v_{n+1} = v_1$ and c has all distinct edges, that is, $\{v_i, v_{i+1}\} \neq \{v_j, v_{j+1}\}$ whenever $i \neq j$. The latter condition means that we do not admit a path $u_1 u_2 u_3 u_2 u_1$ as a cycle (because otherwise every edge xy would give rise to a cycle of arbitrarily great length: $xyxy \dots xyx$, which would make the notion of cycle trivial). The cycle c is *simple* if all vertices $v_1, \dots, v_n$ are pairwise distinct.

Let us write $x \sim_G y$ when the vertices x and y are connected in $G = (V, E)$ by some path. Clearly, $\sim_G$ is a binary relation on V.

Lemma 16.9. *The relation $\sim_G$ is an equivalence relation on the set V.*

Proof. One has $x \sim_G x$ as x is connected to itself by the zero length path x. If $x v_1 \dots v_n y$ is a path, then $y v_n \dots v_1 x$ is a path as well, hence $x \sim_G y$ implies $y \sim_G x$. Given $x v_1 \dots v_n y$ and $y u_1 \dots u_m z$ are paths, one gets the path $x v_1 \dots v_n y u_1 \dots u_m z$ connecting x to z, so from $x \sim_G y$ and $y \sim_G z$, it follows that $x \sim_G z$. □

Consider the quotient set $V/{\sim_G} = \{[x]_{\sim_G} \mid x \in V\}$, where $[x]_{\sim_G} = \{y \in V \mid x \sim_G y\}$. Each equivalence class $[x]_{\sim_G}$ consists of all the vertices connected to x. Notice that $N_G(x) \subseteq [x]_{\sim_G}$ but not vice versa, generally. Such classes are called *connected components* of the graph G. A graph is called *connected* if it has exactly one connected component (i. e., $|V_G/{\sim_G}| = 1$ for our graph G) or, equivalently, every two vertices thereof are connected by a path (i. e., $\sim_G$-equivalent). Otherwise, a graph is called *disconnected.*

Exercise 16.10. If $G \cong H$, then $|V_G/{\sim_G}| = |V_H/{\sim_H}|$. In particular, G is connected iff H is connected.

Lemma 16.11 (connected graph edge removal)**.** *Suppose that $G = (V, E)$ is a connected graph and xy is an edge in G. We let G' be the result of removing the edge xy from G. Then the following hold:*

1. *if xy belongs to a cycle in G, then G' is still connected;*

2. *if xy belongs to no cycle in G, then G' has exactly 2 connected components; namely, $G'/{\sim_{G'}} = \{[x]_{\sim_{G'}}, [y]_{\sim_{G'}}\}$.*

Proof. For the first statement, consider some cycle $a \overset{q}{\text{---}} xy \overset{q'}{\text{---}} a$ and an arbitrary path $p = u \overset{r}{\text{---}} v$ in G. By the definition of cycle, the path $y \overset{q'}{\text{---}} a \overset{q}{\text{---}} x$ does not contain xy. If p does not contain the edge xy, it is still a path in G'. Suppose it does. By Lemma 16.8 and symmetry of $\sim_G$, we may assume that p is simple and $p = u \overset{r'}{\text{---}} yx \overset{r''}{\text{---}} v$, where neither $u \overset{r'}{\text{---}} y$ nor $x \overset{r''}{\text{---}} v$ contains xy. Then we have a path $u \overset{r'}{\text{---}} y \overset{q'}{\text{---}} a \overset{q}{\text{---}} x \overset{r''}{\text{---}} v$ in G'. Hence $u \sim_G v$ implies $u \sim_{G'} v$ as required.

Now assume that xy does not belong to a cycle. Let $A = [x]_{\sim_{G'}}$ and $B = [y]_{\sim_{G'}}$. It suffices to prove that A, B are the only connected components; that is, $A \cup B = V$ and $A \cap B = \varnothing$ (cf. Lemma 15.6). For the first equation, consider a vertex $v \in V$. Since G is connected, there are some paths connecting x to v and y to v. Let us take *shortest* such paths p and p', respectively, which must be simple by Lemma 16.8. W.l.o.g., $|p| \leq |p'|$.

We claim that the path p does not contain the edge xy. Otherwise, $p = xy \overset{q}{\text{---}} v$, where $p'' = y \overset{q}{\text{---}} v$ does not contain xy. Clearly, $|p''| < |p| \leq |p'|$, hence p is *not* the shortest path that connects y to v. A contradiction. So, p is a path in G' and $v \in A$.

Now, consider the second equation and assume, for the contrary, that $A \cap B \neq \varnothing$. Then $x \sim_{G'} y$ by Lemma 15.6. Thus, there is a (simple) path $x \overset{p}{\text{---}} y$ in G' (hence, in G) that does not contain xy. Adding xy to this path, we get a cycle $x \overset{p}{\text{---}} yx$ in G, which is not possible. □

Corollary 16.12. *Suppose that $G = (V, E)$ is a graph with exactly k connected components and xy is an edge in G. We let G' be the result of removing the edge xy from G. Then the following hold:*

1. *if xy belongs to a cycle in G, then G' has exactly k connected components;*

2. *if xy belongs to no cycle in G, then G' has exactly $k + 1$ connected components.*

Proof. Clearly, x and y belong to the same component of G (which is a connected graph). Applying the Lemma above to this component, we see that it is either still connected or split into two connected components. Hence, we either have k or $k + 1$ connected components in G'. □

It is natural to expect a connected graph to have not too few edges: the more cities you have, the more roads you need to connect them with one another.

Lemma 16.13. *If an (n, m)-graph G is connected, then $m \geq n - 1$.*

Proof. By induction on m. Let $m = 0$. If G has at least two distinct vertices, these cannot not be connected by a path, hence, it should be $n = 1$, as required.

Assume that $m > 0$ and that $m' \geq n-1$ for every $m' < m$, every n' and each connected (n', m')-graph (IH). Consider an arbitrary connected (n, m)-graph G and remove an edge therefrom. By Lemma 16.11, this shall result in an $(n, m - 1)$-graph H which is either connected or consists of two connected components that are an (n', m')-graph G' and an (n'', m'')-graph G''.

In the former case, we have $m - 1 \geq n - 1$ by the IH, whence $m \geq n - 1$. In the latter one, we get $n' + n'' = n$ and $m' + m'' = m - 1$. As $m', m'' < m$, obtain $m' \geqslant n' - 1$ and $m'' \geqslant n'' - 1$ by the IH. This implies $m - 1 \geq n - 2$, whence $m \geq n - 1$ again. □

Exercise 16.14. Prove that for every $n \in \mathbb{N}_+$, there exists a connected graph with exactly $n - 1$ edges; it is not thus possible to improve the bound set by Lemma 16.13 in general.

A connected graph G is called *minimally connected* if removing *any* edge makes it disconnected. So, a minimally connected road system has no redundancy: block one road and some cities get unreachable from each other. The following statement will be of use below.

Lemma 16.15. *If an (n, m)-graph G is minimally connected, then $m = n - 1$.*

Proof. By Lemma 16.11, we get two connected components if remove any edge from G. Clearly, each of these two is also minimally connected (otherwise some edge would be safe to remove from G). So, we may iteratively apply the same procedure to each of the components, while it still has at least one edge. After every removal of an edge, there is one more connected component than before it. At the procedure termination, we have m removals done (each edge has been removed), which should result in a graph with $1 + m$ connected components.

On the other hand, this graph has no edges, so every vertex is connected but to itself. Hence, there are exactly n connected components. Therefore, $1 + m = n$ as desired. □

Exercise 16.16. Is it the case that an $(n, n - 1)$-graph is always connected?

17 Special Form Graphs

Trees. One of the most common graph classes in applications (say, in computer programming) is *trees*. By definition, a graph is a *tree* if it is both connected and *acyclic* (that is, contains no cycle as a subgraph). There are other natural ways to define a tree as the following theorem shows.

Theorem 17.1. *Let G be an (n, m)-graph. Then the following statements are equivalent:*

1. *G is a tree;*
2. *G is minimally connected;*
3. *G is connected and $m = n - 1$;*
4. *every two vertices of G are connected by a unique simple path.*

Proof. Assume that G is a tree. Since G has no cycle, by Lemma 16.11, it becomes disconnected if one removes any edge therefrom. As G is connected, G must be minimally connected.

If G is minimally connected, then $m = n - 1$ by Lemma 16.15.

Assume that G is connected and $m = n - 1$ but, for a contradiction, G is not a tree. Then G must have a cycle. Let us remove an arbitrary edge from that cycle. By Lemma 16.11, we obtain a connected $(n, m - 1)$ graph G', for which it holds that $m - 1 \geq n - 1$ by Lemma 16.13. Then $m - 1 \geq m$. A contradiction.

Thus, the first three statements are pairwise equivalent. Now, assume that every two vertices of G are connected by a unique simple path, yet G is not minimally connected. Let xy be an edge which can be removed from G so that the resulting graph G' is still connected. By Lemma 16.8, there is a *simple* path $x \,\underline{\;p\;}\, y$ in G' (hence, in G). This path cannot contain xy, so $x \,\underline{\;p\;}\, y$ and xy are two distinct simple paths between x and y in G. A contradiction.

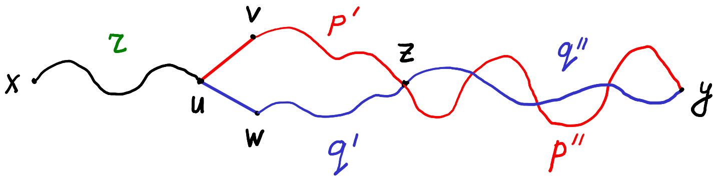

Figure 27: Proving Theorem 17.1.

Finally, we suppose that G is a tree. Then every two vertices of G are, clearly, connected by a simple path (in view of Lemma 16.8). For a contradiction, assume that for some vertices x and y, there are two such distinct paths $x \,\underline{\;p\;}\, y$ and $x \,\underline{\;q\;}\, y$. As $p \neq q$, one can find the leftmost vertices that differ these paths from each other and then, the leftmost common vertex to the right of these. So, we get $x \,\underline{\;r\;}\, uv \,\underline{\;p'\;}\, z \,\underline{\;p''\;}\, y$ and $x \,\underline{\;r\;}\, uw \,\underline{\;q'\;}\, z \,\underline{\;q''\;}\, y$, where $v \neq w$ and the sequences p' and q' have no common vertex. It is clear that $uv \,\underline{\;p'\;}\, z \,\underline{\;s'\;}\, wu$ is a cycle in G, where $s' = a_m \dots a_1$ if $q' = a_1 \dots a_m$. A contradiction. □

Let us see an important example of a tree. Consider the set of all binary words of lengths $0, 1, \ldots, n$, that is, the set $\underline{2}^{\leq n} = \underline{2}^{\underline{0}} \cup \underline{2}^{\underline{1}} \cup \ldots \cup \underline{2}^{\underline{n}}$. (Recall that we identify finite sequences from $A^{\underline{n}}$ with n-tuples from A^n, and have that $A^1 = A$ and $A^0 = \{\varnothing\}$; in the context of words, the set $\varnothing$ is called *the empty word* and denoted by ε.) It is natural to write a word $w \in \underline{2}^{m+1}$ as $w_0 \ldots w_m$, where $w_i \in \underline{2}$. We say that the word uv is the *concatenation* of words $u = u_1 \ldots u_m$ and $v = v_1 \ldots v_n$ if $uv = u_1 \ldots u_m v_1 \ldots v_n$, while $u\varepsilon = \varepsilon u = u$.

Let us say that words u and v are *adjacent* iff $u = vx$ or $v = ux$, where $x \in \underline{2}$. For example, the words 1101 and 110, as well as 110 and 1100 are adjacent. Clearly, this adjacency relation is irreflexive and symmetric, hence we have a graph on the set $\underline{2}^{\leq n}$. This graph is called the *perfect binary tree* $T_n = (V, E)$. But is it indeed a tree? Let us check it!

Assume that $n > 0$, for T_0 is otherwise a sure tree. How many vertices does the graph T_n have? Clearly, $|V| = |\underline{2}^{\leq n}| = 2^0 + 2^1 + \ldots + 2^n$ by the Rules of Sum and Product. Applying a well-known fact about the sum of a geometric progression (easily provable by induction), we obtain $|V| = 2^{n+1} - 1$. How many edges does T_n have?

We see that there is just one vertex ε (the *root* of the tree) of degree 2 (as it is adjacent to 0 and 1), exactly 2^n vertices of degree 1, which are the words of the maximal length n (*leafs* of the tree), each being adjacent to its longest proper prefix only: $v_1 \ldots v_{n-1} E v_1 \ldots v_{n-1} v_n$. Every other vertex $u_1 \ldots u_k$ is of degree 3 since $N(u_1 \ldots u_k) = \{u_1 \ldots u_{k-1}, u_1 \ldots u_k 0, u_1 \ldots u_k 1\}$ when $1 \leq k < n$.

If m is the size of T_n, we get $2m = 1 \cdot 2 + 2^n \cdot 1 + (2^{n+1} - 1 - 1 - 2^n) \cdot 3$ by the Handshake Lemma 16.2. This results in $m = 2^{n+1} - 2 = |V| - 1$. On the other hand, the graph T_n is connected since every vertex $v_1 \ldots v_k$ thereof is connected to ε by the path $\varepsilon,\ v_1,\ v_1v_2,\ v_1v_2v_3,\ \ldots, v_1v_2 \ldots v_{k-1},\ v_1v_2 \ldots v_{k-1}v_k$.

From Theorem 17.1, it follows now that T_n is indeed a tree.

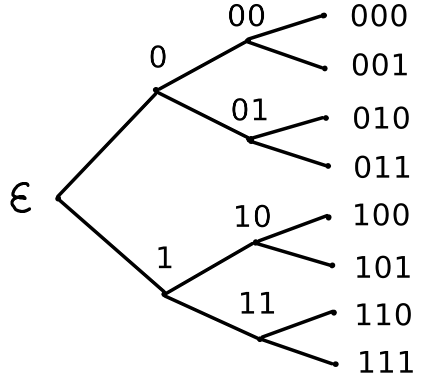

Figure 28: The perfect binary tree T_3.

In fact, every connected graph is "something more than a minimally connected graph", that is, it contains a minimally connected subgraph (i. e., a tree). For a graph $G = (V, E)$, a tree $T = (V, E')$ is called a *spanning tree* of G if $E' \subseteq E$. So, a spanning tree links all the vertices of G together without any redundant edge.

Theorem 17.2. *Every connected (n, m)-graph $G = (V, E)$ has a spanning tree T.*

Proof. By induction on m. If $m = 0$, as a vacuous truth, the graph is minimally connected for it is still connected after any edge removal (it does not change this way). So, put $T = G$.

Suppose that $m > 0$. If G is minimally connected, then it suffices to put $T = G$ again. Otherwise, there exists an edge xy which can be safely removed, so that the resulting $(n, m-1)$-graph $G' =$

$(V, E \setminus \{(x,y),(y,x)\})$ is connected. By the IH, there is a spanning tree $T' = (V, E')$ in G'. As $E' \subseteq E \setminus \{(x,y),(y,x)\} \subseteq E$, this T' is a spanning tree in G as well. □

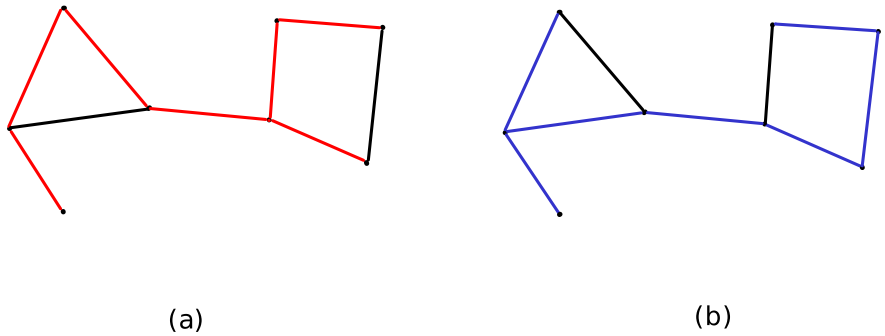

Figure 29: Two non-isomorphic spanning trees of a graph a shown in (a) red and (b) blue.

Bipartite graphs and colorings. Sometimes, graphs are employed to model a situation where 'vertices' represent objects of different kinds. For example, one can connect each worker to each of his skills (two kinds of vertices: workers and skills); or it is possible to link every point (from some finite geometric configuration) to every line it belongs to. In these natural examples, no two vertices of the same kind are adjacent. This restriction gives rise to a special class of graphs.

A graph $G = (V, E)$ is called *bipartite* if there exist two non-empty sets V_1 and V_2 such that $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \varnothing$, and $xy \notin E$ for every $x, y \in V_i$ for each i. That is, there is a partition of the vertex set into two pieces neither of which has an 'internal' edge.

Example 17.3. The cycle C_4 with edges $01, 12, 23, 30$ is bipartite as one may take $V_1 = \{0, 2\}$ and $V_2 = \{1, 3\}$. The cycle C_3 (as well as every graph that contains it as a subgraph, e. g. K_n for $n \geq 3$) is not bipartite since every two distinct vertices of C_3 are adjacent, so cannot belong to the same 'piece' V_i. Hence, $|V_i| \leq 1$ and $|V_1 \cup V_2| \leq 2 < 3$.

More generally, a graph $G = (V, E)$ is called *k-partite* if there exist non-empty sets $V_1, \ldots, V_k$ such that $V_1 \cup \ldots \cup V_k = V$, $V_i \cap V_j = \varnothing$ if $1 \leq i < j \leq k$, and $xy \notin E$ for every $x, y \in V_i$ for each i. Clearly, every (n, m)-graph is n-partite; empty graphs O_n are only 1-partite; and if a graph G is k-partite with $|V_i| \geq 2$ for some i, then G is $(k+1)$-partite as well (it is possible to split V_i into two non-empty sets without any 'internal' edges).

Another way to speak about vertex partitions is *colorings*. A *(proper) coloring* of a graph $G = (V, E)$ in k colors is a function $c\colon V \to \underline{k}$ such that $c(x) = c(y)$ implies $(x, y) \notin E$ for every $x, y \in V$ (i. e., no two vertices of one color are adjacent). Clearly, for every k, a graph is l-partite for some $l \leq k$ iff there is a coloring thereof in k colors. Indeed, put $V_{i+1} = c^{-1}[\{i\}]$; notice that $c^{-1}[\{i\}] = \varnothing$ may hold for some i (when color i is not used), hence we have just $l \leq k$ non-empty 'pieces' in general.

Let us obtain a simple criterion for a graph to be bipartite.

Theorem 17.4. *For every graph $G = (V, E)$ such that $|V| \geq 2$, the following statements are equivalent:*

1. *G is bipartite;*

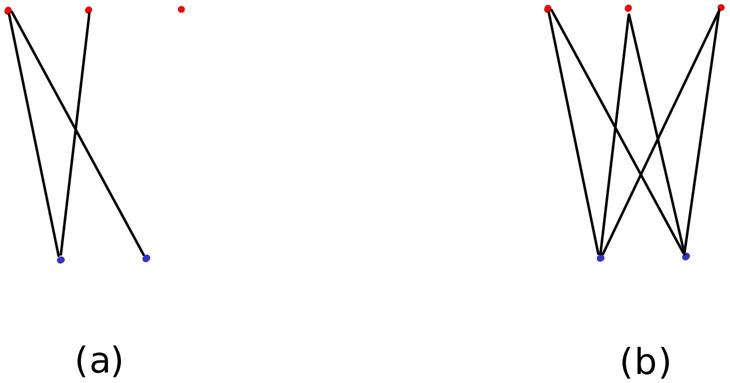

Figure 30: (a) A bipartite graph on five vertices. The two 'pieces' are shown in color. (b) The *complete bipartite* graph $K_{3,2}$. Each vertex of one 'piece' is adjacent to every vertex of the other one.

2. G has no cycle of odd length;

3. G has no simple cycle of odd length.

Proof. Suppose that G is bipartite with a partition $\{V_1, V_2\}$. Assume that G has an odd length cycle $x_1x_2 \dots x_{2n}x_{2n+1}x_1$. W. l. o. g., $x_1 \in V_1$. As x_1Ex_2, we get $x_2 \in V_2$. By an easy induction, we obtain $x_{2n+1} \in V_1$. But $x_{2n+1}Ex_1$. A contradiction.

The third statement follows immediately from the second one.

Suppose G has no simple cycle of odd length. Let us define a partition $\{V_1, V_2\}$ of the set V which makes G bipartite. We claim that it suffices to obtain such a partition for every *connected* graph G. Indeed, let G consist of connected components $G_1, G_2, \dots, G_k$, each of which contains no odd length cycle, hence, is bipartite with a partition $\{V_1^i, V_2^i\}$. Then we may put $V_1 = V_1^1 \cup \ldots \cup V_1^k$ and $V_2 = V_2^1 \cup \ldots \cup V_2^k$. The case when a few components are of order 1 is somewhat special. Our theorem is not directly applicable to those. If there is at least one component G_i of greater order, we may just add all the one-vertex components' vertices to V_1^i. If all the components are one-vertex, let us put one of them to V_1 and all the others to V_2. We still have $V_2 \neq \varnothing$ then for $|V| \geq 2$.

So, assume that G is connected. For every $x, y \in G$, let $d(x, y)$ denote the length of a shortest path connecting x to y in G (any such path is necessarily simple by Lemma 16.8). This quantity is well-defined since G is connected; it is called the *distance* between x an y. Let us fix some vertex $z \in V$ and put $V_1 = \{x \in V \mid d(z, x) \equiv 1 \pmod 2\}$, $V_2 = \{x \in V \mid d(z, x) \equiv 0 \pmod 2\}$. Clearly, $V_1 \cup V_2 = V$ and $V_1 \cap V_2 = \varnothing$. Let us check that each V_i is non-empty. Indeed, z is connected to z by a zero-length path, hence $z \in V_2$. There must be a vertex y which is adjacent to z for $|V| \geq 2$ and G is connected. Clearly, $d(z, y) = 1$ and $y \in V_1$. Now, we need to prove $\neg xEy$ when $x, y \in V_i$.

For the contrary, assume that xEy for certain $x, y \in V_i$, $x \neq y$. Consider some simple paths $z \xrightarrow{p} x$ and $z \xrightarrow{q} y$ of lengths $d(z, x)$ and $d(z, y)$, respectively. Clearly, $d(z, x) \equiv d(z, y) \pmod 2$. Let w be the rightmost common vertex of these paths, so we have $z \xrightarrow{p'} w \xrightarrow{p''} x$ and $z \xrightarrow{q'} w \xrightarrow{q''} y$, where $p = \xrightarrow{p'} w \xrightarrow{p''}$ and $q = \xrightarrow{q'} w \xrightarrow{q''}$, while the paths $\xrightarrow{p''} x$ and $\xrightarrow{q''} y$ have no common vertex. We clam that $|z \xrightarrow{p'} w| = |z \xrightarrow{q'} w|$. Otherwise, w. l. o. g., $|z \xrightarrow{p'} w| < |z \xrightarrow{q'} w|$. Then the path $z \xrightarrow{p'} w \xrightarrow{q''} y$ is shorter than $z \xrightarrow{q} y$, which is supposedly a shortest path between z and y. A contradiction.

Then, we have

$$|w \xrightarrow{p''} x| \equiv d(z,x) - |z \xrightarrow{p'} w| \equiv d(z,y) - |z \xrightarrow{q'} w| \equiv |w \xrightarrow{q''} y| \pmod 2.$$

It is easy to see that $w \xrightarrow{p''} xy \xrightarrow{s''} w$, where $s'' = a_m \dots a_1$ if $q'' = a_1 \dots a_m$, is a simple cycle in G. The length of this cycle is congruent to $2 \cdot |w \xrightarrow{p''} x| + |xy| \equiv 0 + 1 \pmod 2$. But we have assumed no odd-length simple cycles in G. A contradiction. □

Remark 17.5. In fact, our proof for Theorem 17.4 suggests an algorithm to check whether a given graph is bipartite. First of all, we need to know distances between some of the vertices. Let us take an arbitrary vertex z and label it with 0. Clearly, $d(z,z) = 0$. Then we label each neighbor of z with 1. This comprises Step 1. At Step $l+1$, we label each yet unlabeled neighbor of every l-labeled vertex with the number $l+1$. This procedure terminates when every labeled vertex has all its neighbors got labelled (which is inevitable). We claim that a vertex x has label l iff $d(z,x) = l$. By induction on l. The base case of $l = 0$ is evident. Assume our claim holds for all $l' < l$, $l > 0$, and consider an arbitrary vertex x. If x is labelled by l, this has been done at Step l, that is, x is a neighbor of a certain y with label $l-1$. By the IH, we have $d(z,y) = l-1$ and $d(z,x) \geq l$ (otherwise, x would have been labelled at an earlier step). As yEx, $d(z,x) \leq d(z,y)+1$, hence $d(z,x) = l$. For the other direction, let $d(z,x) = l$. There is a path $z \dots yx$ of length $l > 0$. Clearly, $d(z,y) = l-1$. By the IH, y must bear label $l-1$ and x must thus have been labeled with l at Step l.

Is it the case that every vertex of G has a label on this procedure's termination? Not necessarily so for G may be disconnected. But then we can take an arbitrary vertex z' without a label (hence, from another connected component) and repeat similar steps. Then take a vertex z'' yet unlabeled (if any), etc. Clearly, these iterations shall terminate with no vertex unlabeled.

From Theorem 17.4, we know that it suffices to check each of the connected components (now indexed by $z, z', z'', \dots$) for an odd-length cycle. Now, scan each component's edges for whether their respective ends bear labels of identical parity (that is, both even or both odd). If there exists any such edge xy, there is an odd-length cycle in G as our proof shows, hence G is not bipartite. Otherwise, the component we a looking at has a valid partition $(\{x \in V \mid d(z,x) \equiv 1 \pmod 2\}, \{x \in V \mid d(z,x) \equiv 0 \pmod 2\})$ (or is of order 1). If no odd-length cycle is found in any component, the whole graph is bipartite; moreover, extracting a respective partition is easy and has been already discussed in the theorem's proof.

Notice that labeling each vertex x with just the *remainder* after dividing $d(x,y)$ by 2 suffices for our purpose.

Example 17.6. Every tree that has at least 2 vertices is bipartite. Hence, *every* tree is 2-colorable, i. e., has a coloring in *at most* 2 colors.

Let us compute how many ways to color a tree exist. By bipartiteness, there is at least one way to do so. As there are just two colors 0 and 1, we can *invert* a coloring c by putting $c'(x) = 1 - c(x)$ for each vertex x. Clearly, $c(x) = c(y)$ iff $c'(x) = c'(y)$, so c' is a (proper) coloring as c is. Hence, there are at least two colorings for any bipartite graph.

Now, let us show that there are no more than two distinct colorings for a tree. It will be convenient to prove the following: *every (n,m)-tree T has exactly two coloring, each of which is the inversion of the other*—by induction on m.

If $m = 0$, then $n = 1$ as T is a tree. Clearly, there are just two colorings in this case. Assume that $m > 0$. Remove some edge xy from T to get a graph T' with two connected components $T_1 = [x]_{\sim_{T'}}$ and $T_2 = [y]_{\sim_{T'}}$. Since these two are trees, by the IH, we have exactly two colorings c_1, c_1' for T_1 and exactly two colorings c_2, c_2' for T_2, where $c_1(x) = 0 = c_2(y)$ and $c_1'(x) = 1 = c_2'(y)$. On the other hand,

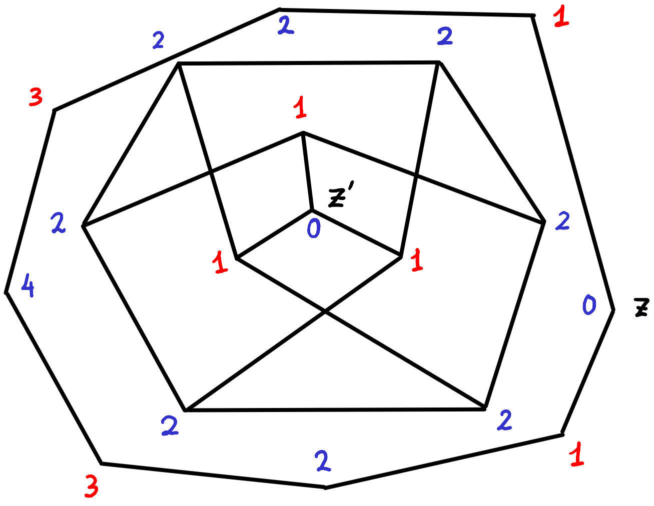

Figure 31: Applying the algorithm from Remark 17.5 to a graph G. This graph appears to have two connected components, of which just $[z']_{\sim_G}$ contains an odd-length cycle. This fact is witnessed by an edge whose ends bear labels of identical parity. Hence, G is not bipartite, whereas its connected component $[z]_{\sim_G}$ is (the latter's vertex set may be partitioned into the 'blue' and 'red' parts).

every coloring c of T induces a pair of colorings for T_1 and for T_2 (distinct c's yield distinct pairs). As $c(x) \neq c(y)$, these may be but c_1 and c_2' or c_1' and c_2, respectively. So, there are no more than two distinct colorings of T. As we have already noticed, at least two colorings for T are guaranteed, each of which is the inversion of the other.

Thus, we have exactly two distinct colorings for a tree. What about an arbitrary bipartite graph G? First, assume G is connected. As we have already seen, there are at least two colorings of G. By Theorem 17.2, there exists a spanning tree T in G. Every coloring of G is a coloring of the tree T. By the above, there are no more than two such colorings.

Finally, if a bipartite graph G has k connected components, one can color them independently, since there is no edge between two distinct components. Clearly, there are exactly 2^k distinct colorings of G in this case.

There is an interesting analogue of the Handshake Lemma for bipartite graphs.

Lemma 17.7. *For every bipartite (n, m)-graph $G = (V, E)$ with 'parts' V_1 and V_2, the following holds:*

$$\sum_{v \in V_1} d_G(v) = \sum_{v \in V_2} d_G(v) = m.$$

Proof. By induction on m. If $m = 0$, both sums equal $0 = m$. Assume that $m > 0$. Consider an edge x_1x_2 of G. W. l. o. g., we have $x_i \in V_i$. Let G' be an $(n, m-1)$-graph that one obtains on removing x_1x_2 from G. Clearly,

$$\sum_{v \in V_i} d_G(v) = \sum_{v \in V_i \smallsetminus \{x_i\}} d_G(v) + d_G(x_i) = \sum_{v \in V_i \smallsetminus \{x_i\}} d_{G'}(v) + d_{G'}(x_i) + 1 = \sum_{v \in V_i} d_{G'}(v) + 1$$

for each $i \in \{1, 2\}$. By the IH, $\sum_{v \in V_i} d_{G'}(v) = m - 1$, whence $\sum_{v \in V_i} d_G(v) = m$. $\square$

Example 17.8. Group A has 22 members and Group B has 21 members (the groups are disjoint). Each member of one group makes a handshake with some members of the other one (at most one per person). Everybody from Group A has made exactly 6 handshakes. Is it possible that all the members of Group B have made an identical number of handshakes?

Assume this is possible. An obvious model for this problem is the handshake graph $G = (V, E)$ where persons are treated as vertices, any two of which are adjacent iff they have made a handshake. This way, groups A and B form a partition of V that makes G bipartite. By our assumptions, $\sum_{v \in A} d_G(v) = 22 \cdot 6$ and $\sum_{v \in B} d_G(v) = 21 \cdot n$ for some n. By the above Lemma, $22 \cdot 6 = 21 \cdot n$, whence $7 \,|\, 22 \cdot 6$, which is not the case. A contradiction.

Digraphs. Sometimes, the irreflexivity and symmetry assumptions we made in the definition of graph are too restrictive. In general, lifting them leads to arbitrary binary relations. In Section 9, we have seen binary relations depicted as *diagrams*. Those diagrams consist of 'vertices' and 'arrows' and are much like graphs except that each link between two vertices has a 'direction' (or 'orientation') now. Two arrows between x and y are distinct iff they differ in their directions; 'loops' from x to x are allowed. Still, we do not have multiple arrows from x to y (as the Reader remembers, an arrow from x to y in the diagram of a relation R means that $(x, y) \in R$). Many ideas from the graph realm are fruitful in this more general case.

Formally, a *directed graph* (or *digraph*) is a pair $G = (V, E)$ where V is a non-empty finite set (we keep this restriction) and $E \subseteq V^2$ (i. e., E is an arbitrary binary relation on V).

Let us transfer some graph notions to digraphs. We call $|V|$ the *order* of a digraph $G = (V, E)$ and call $|E|$ the *size* of G (there is no need to identify 'symmetric' pairs (x, y) and (y, x) when counting edges as those are directed). The neighborhood of a vertex x is naturally split into two subsets $N^+(x) = \{y \in V \mid xEy\} = E[\{x\}]$ and $N^-(x) = \{y \in V \mid yEx\} = E^{-1}[\{x\}]$. The number $d^+(x) = |N^+(x)|$ is called the *outdegree* of the vertex x, while $d^-(x) = |N^-(x)|$ is the *indegree* of x.

Lemma 17.9. *For every digraph $G = (V, E)$,*

$$\sum_{v \in V} d^+_G(v) = \sum_{v \in V} d^-_G(v) = |E|.$$

This basically means that each edge has just one starting point (or 'source') and just one ending point (or 'target'). A formal proof (most naturally, by induction on $|E|$) is left to the reader.

18 Boolean Functions and Circuits

The two following sections treat mostly traditional questions of closed sets (clones), their bases and Post's functional completeness theorem. In our view, the most complicated issue here is to define what expressing one function via some others means. Two popular approaches are: (1) to define 'formulas' or 'terms' and their values, or (2) to use $\subseteq$-least clones. We have found both of them unsatisfactory for our audience. First, the students tend to mix functions and formulas since their intuition of functions is still heavily influenced by high-school 'functions-as-algebraic-expressions' (despite all our set-theoretical efforts). Second, applying clones requires some machinery for structural induction proofs. The latter is far from obvious to the students and is not very intuitive without a direct reduction to induction on naturals (on 'formula size', 'construction length', etc.). In this Course, we do not however target teaching a general inductive definition formalism (a modicum of which is required anyway) to our students.

With this in view, we have taken a compromise approach. Instead of formulas, we use circuits. It is much harder to identify a circuit (something unfamiliar and fancy) with a function. In particular, we emphasize that circuits comprise a "programming language"; most students know well that two distinct programs may compute identical functions. (The last but not the least, circuit formalism is an important prerequisite for many Computer Science courses.) Structural induction may be then reduced to that on circuit size.

Nevertheless, we omit some formalities like discerning a variable from its value. And we make a natural concession to 'formulas' when discussing DNF/CNF and Zhegalkin polynomials.

In Section 1, we saw *compound statements* like $2 = 3$ *and* $4 < 5$, whose truth value *depends* on the truth values of their parts. So, a statement of the form A *and* B, built with the logical connective *and*, is true iff both A and B are true. On the other hand, we can now express diverse dependencies formally as *functions*. Studying logical connectives via functions is interesting from the algebraic point of view and is very important for both the computer science and practical computing (the latter stems from the fact that all objects computers work with are routinely encoded by binary words).

Let us recall the truth tables for popular logical connectives:

A	B	not A	A and B	A or B	if A then B	A if and only if B
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

As usual, 1 stands for *true* and 0 stands for *false*. We will use this table to formally define a few functions from $\underline{2}^2$ and from $\underline{2}$ to $\underline{2}$. We will denote these *Boolean* functions by the symbols we used for logical connectives (and use the same names for both kinds of objects), while it is crucial to see the difference: a (binary) *connective* (if seen as a function) takes two *statements* and returns a *statement*, whereas a (binary) *Boolean function* takes two *truth values* (i. e., elements of $\underline{2}$) and returns a *truth value*.[23]

x	y	$\neg x$	$x \wedge y$	$x \vee y$	$x \to y$	$x \leftrightarrow y$	$x + y$
0	0	1	0	0	1	1	0
0	1	1	0	1	1	0	1
1	0	0	0	1	0	0	1
1	1	0	1	1	1	1	0

[23] It is generally unwise to identify statements with their truth values as Example 1.2 shows: both the statements $0 = 0$ and $\pi < 3.14159265358979323847$ are true but proving this may require a good deal of work.

Now, it makes perfect sense to state that $x = x \wedge x$ for each $x \in \underline{2}$ etc., since we mean functions and their values.

Formally, every function $f\colon \underline{2}^n \to \underline{2}$, where $n \in \mathbb{N}$, is called a *Boolean function of n arguments.* How many such functions exist? As a degenerate case, for $n = 0$ one has $\underline{2}^0 = \{\varnothing\}$, so the sets $\{(\varnothing, 0)\}$ and $\{(\varnothing, 1)\}$ are the only Boolean functions of 0 arguments. For technical reasons, we will ignore these functions and assume $n > 0$ in what follows.

In general, there are $|\underline{2}|^{|\underline{2}^n|} = 2^{2^n}$ distinct Boolean functions of n arguments by Corollary 12.20. Let us denote by $\top$ the set of all Boolean functions of one or more arguments.

Exercise 18.1. Build a truth table for every Boolean function of one and of two arguments.

The following useful equations can be easily checked by by truth tables for each function involved.

Lemma 18.2. *For every $x, y, z \in \underline{2}$,*

1. $x \wedge y = y \wedge x$*;* $x \vee y = y \vee x$*;* $x + y = y + x$*;*
2. $(x \wedge y) \wedge z = x \wedge (y \wedge z)$*;* $(x \vee y) \vee z = x \vee (y \vee z)$*;* $(x + y) + z = x + (y + z)$*;*
3. $x \wedge x = x$*;* $x \vee x = x$*;* $x + x = 0$*;*
4. $x \wedge (x \vee y) = x$*;* $x \vee (x \wedge y) = x$*;*
5. $\neg\neg x = x$*;*
6. $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$*;* $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$*;* $x \wedge (y + z) = (x \wedge y) + (x \wedge z)$*;*
7. $\neg(x \wedge y) = \neg x \vee \neg y$*;* $\neg(x \vee y) = \neg x \wedge \neg y$*;*
8. $x \wedge 0 = 0$*;* $x \wedge 1 = x$*;* $x \vee 0 = x$*;* $x \vee 1 = 1$*;* $x + 0 = x$*;* $x + 1 = \neg x$*;* $\neg 0 = 1$*;* $\neg 1 = 0$*;*
9. $x \wedge \neg x = 0$*;* $x \vee \neg x = 1$*;*
10. $x \to y = \neg x \vee y$*;* $\neg(x \to y) = x \wedge \neg y$*;* $0 \to x = 1$*;* $1 \to x = x$*;* $x \to 0 = \neg x$*;* $x \to 1 = 1$*;*
11. $x \leftrightarrow y = (x \to y) \wedge (y \to x)$*;* $x + y = \neg(x \leftrightarrow y)$*.*

Remark 18.3. The function $+$ returns the remainder after dividing its arguments' sum by 2. (The respective logical connective is *exclusive or* (xor): *either A or B but not both.*) It is also clear that the function $\wedge$ returns the product of its arguments (taking remainder afterwards is logical but does not change the result). We will sometimes denote conjunction by $\cdot$ for this reason. From Lemma 18.2, one can see that the functions $+$ and $\cdot$ satisfy many properties of real number addition and multiplication respectively. E. g., for every x, there exist a number y with $x + y = 0$ (take $y = x$) and, if $x \neq 0$, a number z with $x \cdot z = 1$ (take $z = x$). In general, these functions over the set $\underline{2}$ comprise a *field.* This particular field of two elements is denoted by GF(2) (*Galois field* of two elements). This fact makes many results for real, rational, or complex numbers (which form fields in their turn) valid for the arithmetic of Boolean values 0 and 1 as well. In particular, basic results concerning systems of linear equations are still true.

Another interpretation of Boolean functions stems from the natural ordering of $\underline{2}$ where $0 < 1$. W. r. t. this ordering, $x \wedge y = \inf\{x, y\}$ and $x \vee y = \sup\{x, y\}$ (in fact, $x \wedge y$ and $x \vee y$ are the least and the greatest elements of $\{x, y\}$, respectively), while $x \to y$ returns the truth value of the predicate $\leq$ itself: $x \to y = 1$ iff $x \leq y$.

Boolean circuits. As Lemma 18.2 suggests, one can express (or define) implication in terms of disjunction and negation: $x \to y = \neg x \vee y$. A natural question arises: when is one function definable via some others? This question is of practical importance since we might want to compute a multitude of Boolean functions on hardware capable of computing just some of them directly.

The first step is to rectify what 'to define a function via others' means. To this end, we introduce a simple "programming language" where a function is 'computed' via a few 'primitive' functions. The most important operator of that language is a primitive function application with assigning the result to a variable. The program

$$\begin{aligned} t_1 &= \neg x \\ t_2 &= t_1 \vee y \end{aligned}$$

is intended to compute $x \to y$ using just $\neg$ and $\vee$ as primitive functions. Here x and y are considered input variables, whereas t_1 and t_2 are 'local' or 'temporary' variables intended to store the results of assignments. The last of them (t_2) is interpreted as the value our program 'returns'. Notice the main idea: no more than one primitive function is applied at each step.[24] This is to make our program easy for analysis. And, of course, no temporary variable may be used before assigning some value to it!

In the assignment $t_2 = t_1 \vee y$, we have mixed input and temporary variables, which is not convenient technically. Otherwise, we might assign x and y to new temporary variables first and apply primitive functions to those exclusively:

$$\begin{aligned} t_1 &= x \\ t_2 &= \neg t_1 \\ t_3 &= y \\ t_4 &= t_2 \vee t_3. \end{aligned}$$

For technical reasons again, we want to allow assignments of one temporary variable to another one as well, like $t_i = t_j$, where $j < i$, of course.

Let us give a formal definition for our programs, which are known as *Boolean circuits*. Let P be as set of Boolean functions. We write $f^{(k)}$ for a Boolean function f of k arguments. Let $\vec{x} = (x_1, \ldots, x_n)$ be a tuple of pairwise distinct *input variables*[25]. A *(Boolean) circuit C over P of $\vec{x}$* is a non-empty finite sequence (that is, a tuple, up to bijection) of assignments:

$$\begin{aligned} t_1 &= R_1 \\ t_2 &= R_2 \\ &\cdots \\ t_m &= R_m, \end{aligned}$$

where the right-hand side R_i of each assignment $t_i = R_i$ is either (a) an input variable x_j or (b) a temporary variable t_{i_1} with $i_1 < i$ or (c) an expression $f(t_{i_1}, \ldots, t_{i_k})$ where $f^{(k)} \in P$ and $i_1, \ldots, i_k < i$. The variables $t_1, \ldots, t_m$ are assumed to be pairwise distinct. The number m (that of assignments in C) is called the *size* of the circuit C.

The Pedantic Reader might ask what an 'assignment' and an 'expression' mean. It is not hard to make these notions formal if one interprets $t_i = R_i$ as the pair (t_i, R_i) and $f(t_{i_1}, \ldots, t_{i_k})$ as the pair $(f, (t_{i_1}, \ldots, t_{i_k}))$. But the intended meaning of Boolean circuits renders such formalities unnecessary.

Indeed, we want to put a Boolean function into correspondence with a circuit—the function the circuit 'computes'. We do it recursively, that is, assuming that a function already corresponds to every circuit 'simpler' than the given one. The Reader might want to revise Section 3 for diverse recursive

[24] That is, each function must be applied to variables solely. Such a constraint is known as *A-normal form.*

[25] Formally, these variables may be whatever objects (sets), but a natural intuition of 'letters' as variables is adequate.

definitions, yet here we can use size as the measure of 'simplicity' to make things easier. A formal theory of recursive definitions is still beyond the scope of this Course.

At first, we notice that for every circuit C over P of $\vec{x}$, its every prefix C_i:

$$\begin{array}{rcl} t_1 & = & R_1 \\ t_2 & = & R_2 \\ & \dots & \\ t_i & = & R_i, \end{array}$$

where $i < m$, is again a circuit over P of $\vec{x}$—yet of the smaller size i. Let $C_m = C$ as well.

Consider a circuit C over P of $\vec{x}$ and assume that for each circuit C' over P of $\vec{x}$ of any size $m' < m$, we have already defined a corresponding function $g_{C'} \colon \underline{2}^n \to \underline{2}$. Let us define a function $g_C \colon \underline{2}^n \to \underline{2}$ for C. Consider the last assignment $t_m = R_m$ in C.

- If $R_m = x_j$, put[26] $g_C(\vec{x}) = x_j$ for all $\vec{x} \in \underline{2}^n$. In other words, g_C is a projector function in this case.
- If $R_m = t_{i_1}$, put $g_C = g_{C_{i_1}}$, where the latter function is already defined by assumption since the size i_1 of the prefix circuit C_{i_1} is less than m.
- If $R_m = f(t_{i_1}, \dots, t_{i_k})$, put $g_C(\vec{x}) = f(g_{C_{i_1}}(\vec{x}), \dots, g_{C_{i_k}}(\vec{x}))$ for all $\vec{x} \in \underline{2}^n$, which is well-defined as $i_1, \dots, i_k < m$.

We have thus defined g_C for every circuit C over P of $\vec{x}$ by recursion on its size. We may say that the circuit C *computes* the function g_C.

Example 18.4. Consider the following circuit C over $\{\neg, \vee, \wedge\}$ of (x_1, x_2):

$$\begin{array}{rclrclclcl} t_1 & = & x_1, & g_1(x_1,x_2) & = & x_1, & & & & \\ t_2 & = & x_2, & g_2(x_1,x_2) & = & x_2, & & & & \\ t_3 & = & \neg t_1, & g_3(x_1,x_2) & = & \neg g_1(x_1,x_2) & = & \neg x_1, & & \\ t_4 & = & \neg t_2, & g_4(x_1,x_2) & = & \neg g_2(x_1,x_2) & = & \neg x_2, & & \\ t_5 & = & t_1 \wedge t_4, & g_5(x_1,x_2) & = & g_1(x_1,x_2) \wedge g_4(x_1,x_2) & = & x_1 \wedge \neg x_2, & & \\ t_6 & = & t_3 \wedge t_2, & g_6(x_1,x_2) & = & g_3(x_1,x_2) \wedge g_2(x_1,x_2) & = & \neg x_1 \wedge x_2, & & \\ t_7 & = & t_5 \vee t_6, & g_7(x_1,x_2) & = & g_5(x_1,x_2) \vee g_6(x_1,x_2) & = & (x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) & = & x_1 + x_2. \end{array}$$

At the end of each i-th line, we have placed an equation for the function $g_i = g_{C_i}$ which the respective prefix circuit C_i computes. Clearly, $g_C = g_7 = +$.

The circuit C for $+$ has size 7. It is however possible to compute $+$ over the same set $\{\neg, \vee, \wedge\}$ of primitive functions in a more economical way. Indeed, consider another circuit D:

$$\begin{array}{rclrclcl} t_1 & = & x_1, & g_1(x_1,x_2) & = & x_1, & & \\ t_2 & = & x_2, & g_2(x_1,x_2) & = & x_2, & & \\ t_3 & = & t_1 \vee t_2, & g_3(x_1,x_2) & = & x_1 \vee x_2, & & \\ t_4 & = & t_1 \wedge t_2, & g_4(x_1,x_2) & = & x_1 \wedge x_2, & & \\ t_5 & = & \neg t_4, & g_5(x_1,x_2) & = & \neg(x_1 \wedge x_2), & & \\ t_6 & = & t_3 \wedge t_5, & g_6(x_1,x_2) & = & (x_1 \vee x_2) \wedge \neg(x_1 \wedge x_2) & = & x_1 + x_2. \end{array}$$

We still have $g_D = +$ but $\text{size}(D) = 6 < \text{size}(C)$.

[26] Here we have identified 'variables' $(x_1, \dots, x_n)$ with their possible values from $\underline{2}$. We are going to do so in what follows without any further comment.

Another important measure of circuit complexity is *depth*. Unlike size, one needs recursion (on size) in order to define it. Consider a circuit C over P of $\vec{x}$ and assume that for each circuit C' over P of $\vec{x}$ of any size $m' < m$, we have already defined a number $\mathrm{depth}(C') \in \mathbb{N}$. Consider the last assignment $t_m = R_m$ in C.

- If $R_m = x_j$, put $\mathrm{depth}(C) = 1$.
- If $R_m = t_{i_1}$, put $\mathrm{depth}(C) = 1 + \mathrm{depth}(C_{i_1})$.
- If $R_m = f(t_{i_1}, \ldots, t_{i_k})$, put $\mathrm{depth}(C) = 1 + \sup_< \{\mathrm{depth}(C_{i_1}), \ldots, \mathrm{depth}(C_{i_k})\}$.

Example 18.5. Let us compute the depths of circuits C and D from Example 18.4. We obtain:

$$\begin{array}{rclrcll}
t_1 & = & x_1, & \mathrm{depth}(C_1) & = & 1, & \\
t_2 & = & x_2, & \mathrm{depth}(C_2) & = & 1, & \\
t_3 & = & \neg t_1, & \mathrm{depth}(C_3) & = & 1 + \sup\{1\} & = \ 2, \\
t_4 & = & \neg t_2, & \mathrm{depth}(C_4) & = & 1 + \sup\{1\} & = \ 2, \\
t_5 & = & t_1 \wedge t_4, & \mathrm{depth}(C_5) & = & 1 + \sup\{1, 2\} & = \ 3, \\
t_6 & = & t_3 \wedge t_2, & \mathrm{depth}(C_6) & = & 1 + \sup\{1, 2\} & = \ 3, \\
t_7 & = & t_5 \vee t_6, & \mathrm{depth}(C_7) & = & 1 + \sup\{3, 3\} & = \ 4,
\end{array}$$

so $\mathrm{depth}(C) = 4$, and

$$\begin{array}{rclrcll}
t_1 & = & x_1, & \mathrm{depth}(D_1) & = & 1, & \\
t_2 & = & x_2, & \mathrm{depth}(D_2) & = & 1, & \\
t_3 & = & t_1 \vee t_2, & \mathrm{depth}(D_3) & = & 1 + \sup\{1, 1\} & = \ 2, \\
t_4 & = & t_1 \wedge t_2, & \mathrm{depth}(D_4) & = & 1 + \sup\{1, 1\} & = \ 2, \\
t_5 & = & \neg t_4, & \mathrm{depth}(D_5) & = & 1 + \sup\{2\} & = \ 3, \\
t_6 & = & t_3 \wedge t_5, & \mathrm{depth}(D_6) & = & 1 + \sup\{2, 3\} & = \ 4,
\end{array}$$

whence $\mathrm{depth}(D) = 4 = \mathrm{depth}(C)$.

There is another representation for Boolean circuits in terms of digraphs. This is especially vivid and useful when every function $f^{(k)}$ in P is *symmetric*, that is, $f(x_{\sigma(1)}, \ldots, x_{\sigma(k)}) = f(x_1, \ldots, x_k)$ for every permutation σ of the set $\{1, \ldots, k\}$ and every $\vec{x} \in \underline{2}^k$. For example, $\wedge$ is symmetric as $x_1 \wedge x_2 = x_2 \wedge x_1$ always holds, while $\rightarrow$ is not symmetric since $0 \rightarrow 1 \neq 1 \rightarrow 0$.

We will not give any formal definition for this graphical representation but be content with analyzing the two circuits from Example 18.4.

This graphical representation may be also seen as a scheme of a real electric circuit, as a logical primitive of today's sophisticated electronics. Assume that each arrow represents a conductor and at each input and each node's output two states are discernible: 'current' (identified with $1 \in \underline{2}$) or 'no current' (identified with 0). Assume that every primitive function node is a device cleverly engineered to transform its input states according to the respective function (a *logic gate*). Say, a negation gate outputs 'current' iff there is 'no current' at its input, thus doing a rudimentary computation. Assume we have 'current' at some of the inputs according to a tuple $\vec{x} \in \underline{2}^n$. Then there is 'current' at the circuit's C output iff $g_C(\vec{x}) = 1$.

As circuits comprise a "programming language", one can easily see that size and depth correspond to the well-known program performance measures: *space* and *time*, respectively. The first of these analogies seems quite clear. For the second one, imagine that each logic gate requires some time T to do its current-transforming job (as a matter of fact, there are such delays in real-world devices).

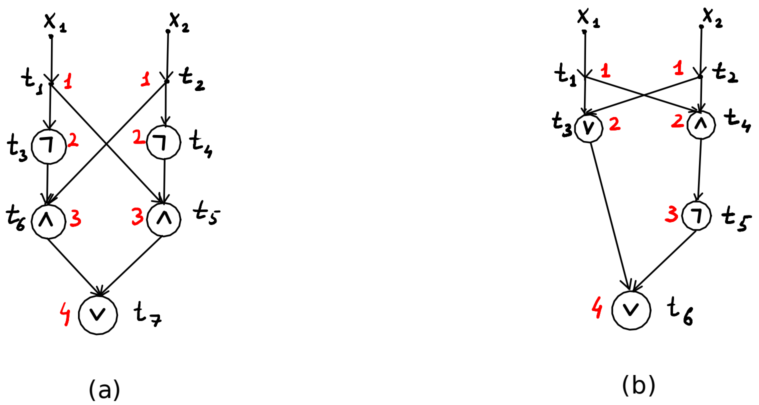

Figure 32: The circuits (a) C and (b) D from Example 18.4. For each node (or prefix circuit), its depth is shown in red. Clearly, the depth is the length of a *longest* path from an input to that node. In general, labeling nodes with temporary variables is optional.

How long does one have to wait until the correct state of the output is established? Clearly, this time is determined by the longest path for the current to flow from the inputs to the output and can be estimated as $T \cdot \mathrm{depth}(C)$.

Programming with circuits. In many programming languages, one can call a subroutine (also know as 'function', 'procedure' or 'method') from a program to do some specific task. Typically, each subroutine has its own local variables which are not visible from the calling program nor other subroutines. This way, one can write a subroutine independently of the caller and reuse it freely; in particular, one can safely reuse variables in different subroutines.

Yet our circuit formalism lacks anything like that. What can we do then? Assume that a circuit C of $(x_1, \dots, x_n)$ is given:

$$\begin{array}{rcl} t'_1 & = & R_1 \\ & \dots & \\ t'_m & = & R_m, \end{array}$$

and we want to 'call' it from another circuit D in order to apply the function g_C to some temporary variables in D:

$$t = g_C(t_1, \dots t_n).$$

Unless g_C is a primitive function, we have to somehow insert C into D:

$$\begin{array}{rcl} & \dots & \\ t'_1 & = & R_1 \\ & \dots & \\ t'_m & = & R_m \\ t & = & t'_m \\ & \dots & \end{array}$$

Two problems arise here. First, the temporary variables t'_i may occur elsewhere in D, which makes this sequence of assignments formally incorrect (what if, say, t is the same variable as t'_1?). Second, the terms R_i depend on $x_1, \dots, x_n$ as inputs whereas we want to feed $t_1, \dots, t_n$ to C in their stead.

Clearly, both these problems are purely formal. It is well possible to solve them in a perfectly formal manner, but this is too tedious. We shall describe a solution informally.

For the first problem, one can replace all occurrences of t'_i in C with a new variable t''_i which is 'fresh', i. e., has no occurrence in D (and surely t''_i differs from t''_j when $i \neq j$). For the second problem, we replace all occurrences of x_j in the terms $R_1, \ldots, R_m$ with t_j. (Notice that an assignment $t'_i = x_j$ will be replaced by $t''_i = t_j$. In the definition of circuit, we have provisioned an assignment with a temporary variable as its right-hand side for this very reason.) In what follows, we shall hide all these formalities behind assignments like

$$t = C(t_1, \ldots, t_n).$$

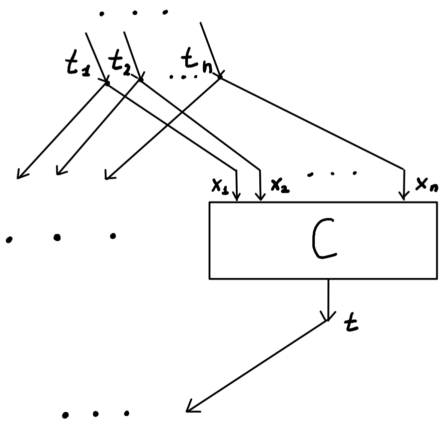

Figure 33: Everything looks simpler in this diagram of the 'caller' circuit D. The 'subroutine' circuit C is depicted here as a 'black box'. Nodes $t_1, \ldots, t_n$ are connected to its respective inputs $x_1, \ldots, x_n$. The output of C is connected to t for any subsequent usage.

Now, let us do some practical programming with circuits. With this in view, we shall generalize circuit computations slightly. Since every prefix circuit computes a function, it is possible to label more temporary variables as outputs (not just the last one)—this way, a circuit may compute a tuple of Boolean functions or equivalently, a function of the form $\underline{2}^n \to \underline{2}^m$ (cf. the identity $A^C \times B^C \sim (A \times B)^C$ from Theorem 11.6).

Let us build a circuit for adding two one-digit binary numbers. A binary (notation for a) natural number is a tuple of zeroes and ones, but adding two one-digit numbers may result in a two-digit number for $add(1, 1) = 10$. Assuming $0 = 00$ and $1 = 01$ (that is, introducing a 'leading zero'), one can always consider the result as a two-digit number. So, we may let $add\colon \underline{2}^1 \times \underline{2}^1 \to \underline{2}^2$ or $add\colon \underline{2}^2 \to \underline{2}^2$, and $add(x, y) = z_2z_1$ for $x, y, z_i \in \underline{2}$.

Our circuit C_{add} shall thus have two inputs x, y and two outputs z_1, z_2. It is convenient to make it a circuit over the set $\{+, \wedge\}$. Here it is:

$$\begin{aligned} t_1 &= x \\ t_2 &= y \\ z_1 &= t_1 + t_2 \\ z_2 &= t_1 \wedge t_2. \end{aligned}$$

Now, we will implement addition for two-digit (or longer) numbers. The main idea is to modify the circuit C_{add} slightly: we may rightly call the bit z_2 the *overflow* for it is 1 iff one digit is not enough to represent the sum. When adding the *second* bits of two-digit numbers, we must take the overflow resulting from the first bits into account; on the other hand, we may produce a new overflow. So, the modified bit addition add_1 is a function from $\underline{2}^3$ to $\underline{2}^2$ such that, say, $add_1(1,1,1) = 11$. The respective *bit adder* circuit A will take three inputs x, y, and o, where o is the incoming overflow, and produce still two outputs z_2 (the outgoing overflow) and z_1:

$$\begin{aligned} t_1 &= x \\ t_2 &= y \\ t_3 &= o \\ z_2 &= \mathrm{maj}(t_1, t_2, t_3) \\ t_4 &= t_1 + t_2 \\ z_1 &= t_4 + t_3. \end{aligned}$$

For the sake of clarity, we have constructed a circuit over the set $\{+, \mathrm{maj}\}$, where the important *majority function* maj returns 1 iff there are more '1's than '0's among its argument values, e. g., $\mathrm{maj}(1,0,1) = 1$ but $\mathrm{maj}(0,1,0) = 0$. Indeed, we have an overflow iff there are at least two input '1's. We shall see later that $\mathrm{maj}(x,y,z) = (x \wedge y) + (x \wedge z) + (y \wedge z)$ and the bit adder can be easily built over the set $\{+, \wedge\}$ as well.

Given the bit adder A, we can construct adders Add_n for n-digit numbers *recursively.* For $n = 1$, we just have to fix $o = 0$ obtaining the circuit Add_1 of (x_1, y_1):

$$\begin{aligned} t_1 &= x_1 \\ t_2 &= x_2 \\ o &= t_2 + t_2 \\ (z_2, z_1) &= A(t_1, t_2, o). \end{aligned}$$

The last line is similar to 'calling' one circuit from another which we have already discussed. Here we have but two outputs of A assigned to z_2 and z_1.

Assume that a circuit Add_n of $(x_1, \ldots, x_n, y_1, \ldots, y_n)$ for adding two n-digit numbers $x_n \ldots x_1$ and $y_n \ldots y_1$ with outputs $(z_{n+1}, z_n, \ldots, z_1)$ has been already built (z_{n+1} is the outgoing overflow or, equivalently, the most significant bit of the result). One can then obtain Add_{n+1} this way:

$$\begin{aligned} t_1 &= x_1 \\ s_1 &= y_1 \\ &\ldots \\ t_n &= x_n \\ s_n &= y_n \\ (o, z_n, \ldots, z_1) &= Add_n(t_1, \ldots, t_n, s_1, \ldots, s_n) \\ t &= x_{n+1} \\ s &= y_{n+1} \\ (z_{n+2}, z_{n+1}) &= A(t, s, o), \end{aligned}$$

where $(z_{n+2}, z_{n+1}, z_n, \ldots, z_1)$ are outputs.

This construction's recursive structure allows simple inductive proofs for diverse properties of the circuit. In particular, one can easily prove that Add_n indeed computes the sum of two binary numbers by induction on n.

Closures. Now, we can make it precise how one function may be expressed via some others. Let Q be a set of Boolean functions. The *closure* of Q is the set

$$[Q] = \{f \in \top \mid f = g_C \text{ for some circuit } C \text{ over } Q\}.$$

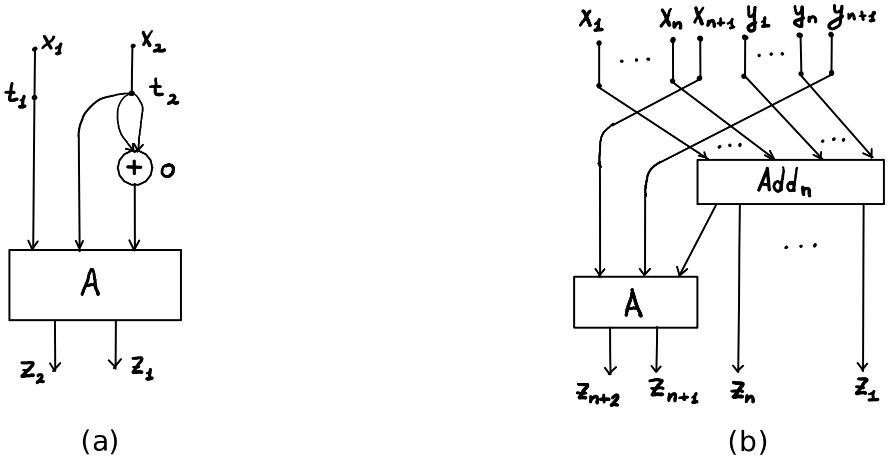

Figure 34: Constructing adders (a) Add_1 and (b) Add_{n+1} recursively.

The set $[Q]$ thus consists of all such functions that one can compute with a circuit over Q. From Example 18.4, it follows that $+ \in [\{\neg, \vee, \wedge\}]$. To make our notation prettier, we will write $[f_1, \ldots, f_n]$ instead of $[\{f_1, \ldots, f_n\}]$, so $+ \in [\neg, \vee, \wedge]$.

Example 18.6. We know that $\rightarrow \in [\neg, \vee]$, but we have $\rightarrow \in [\neg, \wedge]$ as well, since $x \rightarrow y = \neg(x \wedge \neg y)$ (it is easy to make a suitable circuit at this point; we shall skip further explanations in similar cases).

On the other hand, $x \wedge y = \neg(\neg x \vee \neg y)$ and $x \vee y = \neg(\neg x \wedge \neg y)$. Does not that mean we can always 'emulate' $\wedge$ using just $\vee$ (and vice versa) if $\neg$ is also available? As we shall see, it does. In particular, we shall have $[\neg, \wedge] = [\neg, \vee]$.

Let us consider extreme cases. Clearly, $[\top] = \top$, for $[\top] \subseteq \top$ by definition, and every function $f^{(n)}$ can be computed via itself with a trivial circuit:

$$\begin{array}{rcl} t_1 & = & x_1 \\ t_2 & = & x_2 \\ & \cdots & \\ t_n & = & x_n \\ t_{n+1} & = & f(t_1, \ldots, t_n). \end{array} \tag{9}$$

But what is $[\varnothing]$? Can we compute a function using no primitives at all? Revisiting the definition, we see that every circuit C over $\varnothing$ has only variables in its assignments' right-hand sides. If the last assignment is of the form $t_i = x_j$, we have $g_C(\vec{x}) = x_j$ by definition, that is, g_C is a *projector function*, which returns one of its inputs. If the last assignment is $t_i = t_{i_1}$ with $i_1 < i$, we need induction on size to prove that g_C is still a projector. Indeed, $g_{C_{i_1}}$ is a projector by the IH, while $g_C = g_{C_{i_1}}$ by definition. On the other hand, every projector is easily computable via a trivial circuit of the form $t_1 = x_j$.

Thus, $[\varnothing]$ equals the set of projectors $\perp = \{f \in \top \mid \exists j \forall \vec{x}\, f(\vec{x}) = x_j\}$.

Now, we are going to establish a few important properties of the closure operation. In order to prove them and many other interesting statements alike, we will apply the following 'structural induction' method.

Lemma 18.7 (Structural induction for Boolean functions)**.** *Let Q be as set of Boolean functions and φ be a unary predicate over Boolean functions. Suppose all of the following hold:*

- *for each $f \in Q$, $\varphi(f)$;*
- *for each projector function $p \in \bot$, $\varphi(p)$;*
- *if $\varphi(h), \varphi(g_1), \ldots, \varphi(g_m)$ and $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$, then $\varphi(f)$.*

Then we have $\varphi(f)$ for every $f \in [Q]$.

This statement asserts some properties that each function from Q enjoys to hold for all functions in $[Q]$ as well. Such a property must be "nice enough", i. e., it must hold for projectors and respect the *superposition* (sometimes called 'composition') operation: when f is obtained from some $h, g_1, \ldots, g_m$ by the equation $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$. The lemma does not explicitly mention circuits but its proof shall in fact mimic the way we define the function g_C for a circuit C. (It is very instructive to try to find a superposition operation there.)

Proof. For every function $f \in [Q]$, the exists a circuit over Q such that $f = g_C$. There may be many such circuits, but their *sizes* form a non-empty set of naturals. By the Least Number Principle, there is the *least* such size (and a respective circuit). We shall call this number the *size* of f (w. r. t. Q) and denote it by $\mathrm{size}(f)$.

We will prove $\varphi(f)$ by induction on $\mathrm{size}(f)$. That is, we shall assume as the IH that for each function $f' \in [Q]$ with $\mathrm{size}(f') < \mathrm{size}(f)$ it holds that $\varphi(f')$, and try to derive $\varphi(f)$ therefrom.

Let $s = \mathrm{size}(f)$ and let C be a circuit over Q computing f. What is the last assignment in C? If it is of the form $t_s = x_j$, then $f = g_C$ is a projector, whence $\varphi(f)$ by assumption. If it is of the form $t_s = t_i$ with $i < s$, we have $f = g_C = g_{C_i}$. The function g_{C_i} is computed by the prefix circuit C_i, where $\mathrm{size}(C_i) < s$. The circuit C_i is not necessarily a shortest among those computing g_{C_i}, but surely $\mathrm{size}(g_{C_i}) \le \mathrm{size}(C_i) < s$. By the IH, we get $\varphi(g_{C_i})$ and $\varphi(f)$.

The only remaining case is $t_s = h(t_{i_1}, \ldots, t_{i_k})$, where $h \in Q$ and $i_1, \ldots, i_k < s$. Then $f(\vec{x}) = g_C(\vec{x}) = h(g_{C_{i_1}}(\vec{x}), \ldots, g_{C_{i_k}}(\vec{x}))$ for all $\vec{x}$. As in the above, we have $\varphi(g_{C_{i_1}}), \ldots, \varphi(g_{C_{i_k}})$ by the IH. From the assumptions, we obtain $\varphi(h)$ and $\varphi(f)$ finally. □

> This lemma reduces the inductive definition for clones to our chosen function 'representation', be it a circuit, a formula or whatever. It states that $[Q]$ (as defined in representation terms) is included into every clone containing Q. On the other hand, the following theorem implies that $[Q]$ is indeed a clone (that is, $\bot \subseteq Q$ and $[Q]$ is closed under superposition) and contains Q. So, $[Q]$ is the $\subseteq$-least clone containing Q; this provides an abstraction layer allowing to mention no circuits in most of the subsequent arguments. We do not do so, nevertheless. If the Instructor likes this approach, he might want to add an explicit lemma stating $[Q]$ to be a clone.

Theorem 18.8 (Closure properties)**.** *For every $P, Q \subseteq \top$, it holds that:*

1. *$Q \subseteq [Q]$;*
2. *if $P \subseteq Q$, then $[P] \subseteq [Q]$;*
3. *$[[Q]] = [Q]$.*

Proof. Given $f \in Q$, the trivial circuit (9) over Q computes f, so $f \in [Q]$.

Since $g \in [P]$, there exists a circuit C over P with $g_C = g$. Obviously, C is a circuit over Q as well when $P \subseteq Q$. Hence, $f \in [Q]$.

From the first claim, it clearly follows that $[Q] \subseteq [[Q]]$. It remains to prove that $[[Q]] \subseteq [Q]$. We may apply Lemma 18.7 here. The predicate $\varphi(f)$ in question is just $f \in [Q]$. Clearly, this property holds for every function in $[Q]$ and every projector (consider a circuit of the form $t_1 = x_j$).

Finally, we have to check that superposition preserves φ. Suppose that $h, g_1, \dots, g_m \in [Q]$ and $f(\vec{x}) = h(g_1(\vec{x}), \dots, g_m(\vec{x}))$ for all $\vec{x} \in \underline{2}^n$. We need $f \in [Q]$. Let $D, C_1, \dots, C_m$ be some circuits over Q which compute the functions $h, g_1, \dots, g_m$, respectively. Consider the circuit C over Q:

$$\begin{array}{rcl} t_1 & = & x_1 \\ t_2 & = & x_2 \\ & \dots & \\ t_n & = & x_n \\ s_1 & = & C_1(t_1, \dots, t_n) \\ & \dots & \\ s_m & = & C_m(t_1, \dots, t_n) \\ r & = & D(s_1, \dots, s_m). \end{array}$$

It is easy to see that C computes just the function f. Hence, $f \in [Q]$. □

Example 18.9. Now, we can easily prove $[\neg, \wedge] = [\neg, \vee]$ in a perfectly formal manner. As we have already seen, $\wedge \in [\neg, \vee]$. Then $\{\neg, \wedge\} \subseteq [\neg, \vee]$, whence $[\neg, \wedge] \subseteq [[\neg, \vee]]$. But $[[\neg, \vee]] = [\neg, \vee]$, so $[\neg, \wedge] \subseteq [\neg, \vee]$. The other inclusion is similar. Likewise obtain $[\neg, \wedge, \vee] = [\neg, \wedge] = [\neg, \vee]$.

A set $Q \subseteq \top$ is called *closed* if $[Q] = Q$ (which is equivalent to $[Q] \subseteq Q$ in view of Theorem 18.8).[27] This essentially means that Q is large enough to contain everything one can compute over Q with a Boolean circuit. We already know that $\top$ is closed. The set of projectors $\bot = [\varnothing]$ is closed as well, for $[\bot] = [[\varnothing]] = [\varnothing] = \bot$. In fact, every closure set $[P]$ is closed for $[[P]] = [P]$ by Theorem 18.8. Thus, sets of the form $[P]$ and closed sets are exactly the same.

Example 18.10. If P and Q are closed, then $P \cap Q$ is closed as well. Indeed, $P \cap Q \subseteq P$ implies $[P \cap Q] \subseteq [P] = P$ and similarly for Q, whence $[P \cap Q] \subseteq P \cap Q$.

So, the set $[\neg, \vee]$ is closed, while the set $\{\neg, \vee\}$ is not since the latter does not contain the function $\to$ yet the former does. But can we have a better description for the set $[\neg, \vee]$? Which functions does it contain? (Like we know that $[\varnothing]$ consists of all projectors.) Which closed sets exist? On the other hand, given a closed set Q, can we find some 'minimal' P with $[P] = Q$? Clearly, $P = Q$ shall work, but we may have something more frugal, like in $[\varnothing] = \bot = [\bot]$.

By the way, we say that P is *complete in* Q iff $[P] = Q$. Clearly, this implies $P \subseteq Q$. A $\subseteq$-minimal complete subset P of Q is called a *basis* of Q, that is, P is a basis of Q iff $[P] = Q$ but $[P'] \neq Q$ for any $P' \subsetneq P$. E. g., $\bot$ is complete in $\bot$ but is not a basis thereof, while $\varnothing$ is.

Exercise 18.11. Can a finite set be closed?

Questions like those we presented had been thoroughly studied by the middle of the 20th century. Below, we shall present some of the most important results in this area.

Remark 18.12. The three closure properties from Theorem 18.8 are widespread in mathematics and are found in many contexts that have nothing in common with Boolean functions (at first glance, at least). To give but one example, consider *linear span*. For a subset S of a vector space V (say, over the field

[27] Such closed sets are also known as *clones* on the set $\underline{2}$. The definition may vary slightly.

$\mathbb{R}$), its linear span $\langle S\rangle$ may be defined as the set of all possible finite sums of the form $\alpha_1 v_1 + \ldots + \alpha_k v_k$, where $v_i \in S$ and $\alpha_i \in \mathbb{R}$, including the 'empty' sum with $k = 0$ which is identified with $\vec{0} \in V$.[28] (Such sums are known as *linear combinations.*)

It is easy to check that $S \subseteq \langle S\rangle$, $S \subseteq T$ implies $\langle S\rangle \subseteq \langle T\rangle$, and $\langle\langle S\rangle\rangle = \langle S\rangle$. Closed sets $S \subseteq V$, with $\langle S\rangle = S$, are exactly subspaces of the vector space V, and a basis of a subspace S may be equivalently defined as its $\subseteq$-minimal complete subset.

One can also prove that $\langle S\rangle$ is just the intersection of all subspaces of V that include S or, equivalently, the $\subseteq$-least such subspace of V.

> The Instructor might wish to give more examples of this kind like topological or deductive closures.

Normal forms for Boolean functions. As a matter of fact, one needs very few well-known functions in Q in order to obtain $[Q] = \top$. Moreover, every Boolean function can be computed by a circuit of a very special form. We shall however step aside from circuit formalism in favor of slightly more popular 'normal form expressions'.

We do not want to define 'expressions' formally; for every thing formal, we will still be able to emulate those with circuits. Given a variable x_i, the expressions x_i and $\neg x_i$ are called *literals.* Every expression of the form $l_1 \wedge l_2 \wedge \ldots \wedge l_k$, where each l_i is a literal and $k \geq 1$, is known as an *elementary conjunction.* Notice the lack of parentheses here; they do not matter due to the associativity property from Lemma 18.2. Similarly, every expression of the form $l_1 \vee l_2 \vee \ldots \vee l_k$, where $k \geq 1$, is called an *elementary disjunction.*[29] A *disjunctive normal form* (DNF) is a disjunction $c_1 \vee c_2 \vee \ldots \vee c_n$, $n \geq 1$, of elementary conjunctions c_i. Similarly, a *conjunctive normal form* (CNF) is a conjunction $d_1 \wedge d_2 \wedge \ldots \wedge d_n$, $n \geq 1$, of elementary disjunctions d_i.

Example 18.13. Every literal is an elementary conjunction and disjunction as well. Every elementary conjunction or disjunction is both a DNF and a CNF. So, $\neg x_1 \wedge x_2 \wedge x_1$ is both a CNF (of three elementary disjunctions) and a DNF (of one elementary conjunction). The expression $(\neg x_1 \wedge x_2 \wedge x_3) \vee x_3 \vee (x_5 \vee \neg x_2)$ is a DNF but not a CNF.

It is not hard to formally define, say, a DNF as a circuit of the form

$$\begin{array}{rcl} t_1 & = & R_1 \\ & \ldots & \\ t_m & = & R_m \end{array}$$

such that in the R_i terms, every occurrence of $\neg$ precedes every occurrence of $\wedge$ and every occurrence of $\wedge$ precedes every occurrence of $\vee$ (multiple negations are not a problem, of course, for $\neg\neg x = x$). Clearly, a function 'equals' a DNF-expression iff it is computable by such a circuit.

The most noteworthy fact about DNF is that *every* Boolean function equals one of such expressions. The same is true for CNF. Thus, for each $f \in \top$, an equation of the form $f(\vec{x}) = D(\vec{x}) = C(\vec{x})$ holds for a suitable DNF D and CNF C for all $\vec{x}$. In circuit terms, every function f is computable by a suitable DNF (or CNF) circuit.

[28] This empty sum is only important when $S = \varnothing$. In this case, $\langle\varnothing\rangle = \{\vec{0}\}$. Otherwise, one has $\vec{0} = v - v \in \langle S\rangle$ for whatever $v \in S$.

[29] Elementary conjunctions are also known as *minterms* (especially, when every variable x_i, $1 \leq i \leq n$, occurs just once therein). Elementary disjunctions are called *maxterms* then.

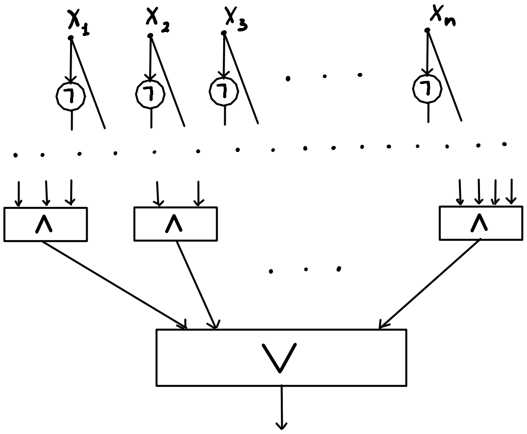

Figure 35: A DNF as a Boolean circuit. Each conjunction-labeled box computes the conjunction of its inputs (it is quite clear how to construct such a circuit). Similarly for disjunctions. The long dotted line hides a connection of literals to conjunction-boxes, which specifies a particular DNF and may be tricky.

Theorem 18.14. *For every Boolean function $f^{(n)}$, there exists a DNF D of variables $x_1, \ldots x_n$ such that $f(\vec{x}) = D(\vec{x})$ for all $\vec{x} \in \underline{2}^n$.*

Proof. For an arbitrary value $\sigma \in \underline{2}$ and a variable x, let

$$x^\sigma \quad = \quad \begin{cases} x & \text{if } \sigma = 1; \\ \neg x & \text{if } \sigma = 0. \end{cases}$$

Clearly, x^σ is just a shorthand notation for the literal x or $\neg x$. Considering all possible values for $x, \sigma \in \underline{2}$, it is easy to notice that $x^\sigma = 1$ iff $x = \sigma$.

For every $\vec{\sigma} = (\sigma_1, \ldots, \sigma_n) \in \underline{2}^n$, consider the elementary conjunction $\Phi_{\vec{\sigma}} = x_1^{\sigma_1} \wedge \ldots \wedge x_n^{\sigma_n}$. E. g., one has $\Phi_{1001} = x_1^1 \wedge x_2^0 \wedge x_3^0 \wedge x_4^1 = x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4$.

Let us show that $\Phi_{\vec{\sigma}}(\vec{x}) = 1$ iff $\vec{x} = \vec{\sigma}$. Indeed, the former equation means $x_i^{\sigma_i} = 1$ for each i, that is, $x_i = \sigma_i$ and $\vec{x} = \vec{\sigma}$ finally. A tuple $\vec{\sigma} \in \underline{2}^n$ is thus 'encoded' by the elementary conjunction $\Phi_{\vec{\sigma}}$.

Let U be the set $\{\vec{\sigma} \in \underline{2}^n \mid f(\vec{\sigma}) = 1\}$ of all tuples where the function $f^{(n)}$ takes the value 1. If $U = \varnothing$, we have $f(\vec{x}) = x_1 \wedge \neg x_1$, which is a DNF we need. Otherwise, let $U = \{\vec{\sigma}^1, \ldots, \vec{\sigma}^k\}$. Consider the expression $D = \Phi_{\vec{\sigma}^1} \vee \ldots \vee \Phi_{\vec{\sigma}^k}$. Clearly, D is a DNF.

Furthermore,

$$\begin{aligned} D(\vec{x}) = 1 \quad &\Longleftrightarrow \quad \exists j \; \Phi_{\vec{\sigma}^j}(\vec{x}) = 1 \\ &\Longleftrightarrow \quad \exists j \; \vec{x} = \vec{\sigma}^j \\ &\Longleftrightarrow \quad \vec{x} \in U \\ &\Longleftrightarrow \quad f(\vec{x}) = 1. \end{aligned}$$

So, D is a required DNF. □

Remark 18.15. A similar theorem holds for CNF. One can prove it by taking a 'dual' of the argument above. First of all, consider literals $x^{\neg\sigma}$. Clearly, $x^{\neg\sigma} = 0$ iff $x \neq \neg\sigma$ iff $x = \sigma$. Then take the elementary disjunction $\Psi_{\vec{\sigma}} = x_1^{\neg\sigma_1} \vee \ldots \vee x_n^{\neg\sigma_n}$ for each $\sigma \in \underline{2}^n$. This disjunction equals 0 iff $\vec{x} = \vec{\sigma}$; so, we can 'encode' a tuple with an elementary disjunction as well. Say, we have $\Psi_{1001} = x_1^0 \vee x_2^1 \vee x_3^1 \vee x_4^0 = \neg x_1 \vee x_2 \vee x_3 \vee \neg x_4$.

Finally, consider the set $Z = \{\vec{\sigma} \in \underline{2}^n \mid f(\vec{\sigma}) = 0\}$. If $Z = \{\vec{\sigma}^1, \ldots, \vec{\sigma}^k\} \neq \varnothing$, the CNF $C = \Psi_{\vec{\sigma}^1} \wedge \ldots \wedge \Psi_{\vec{\sigma}^k}$ is an expression we are interested in. If $Z = \varnothing$, just take $C = x_1 \vee \neg x_1$.

Example 18.16. Our proof for Theorem 18.14 provides an *algorithm* to obtain a DNF for any given function f (provided one can somehow compute its values).

For example, consider a Boolean function $f^{(3)}$ with $U = \{000, 010, 101\}$ and, respectively, $Z = \{001, 011, 100, 110, 111\}$. Applying the algorithm gives the DNF $D = (\neg x_1 \wedge \neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_2 \wedge \neg x_3) \vee (x_1 \wedge \neg x_2 \wedge x_3)$.

The 'dual' algorithm returns the CNF $C = (x_1 \vee x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee \neg x_2 \vee \neg x_3)$.

Neither D nor C is of the least possible length among the expressions for f of its kind. Say, $D' = (\neg x_1 \wedge \neg x_2) \vee (x_1 \wedge \neg x_2 \wedge x_3)$ is a shorter DNF for f.

This algorithm is not particularly effective since it generally requires computing all the 2^n values of the function $f^{(n)}$. No essentially better algorithm for this task is currently known.

As each DNF or CNF can be computed with a circuit over $\{\neg, \wedge, \vee\}$, so every Boolean function can, and we obtain

Corollary 18.17. $\top = [\neg, \wedge, \vee] = [\neg, \wedge] = [\neg, \vee]$.

This way, we have found very simple function sets being complete in $\top$. In fact, just one function is enough! Consider the function $|$ that may be defined by the equation $x \,|\, y = \neg(x \wedge y)$. This function is known as *Sheffer stroke*. As $\neg x = \neg(x \wedge x) = x \,|\, x$ and $x \wedge y = \neg(x \,|\, y) = (x \,|\, y) \,|(x \,|\, y)$, we get $\{\neg, \wedge\} \subseteq [\,|\,]$, whence $\top = [\wedge, \neg] \subseteq [[\,|\,]] = [\,|\,] \subseteq \top$ by Theorem 18.8. So, $\top = [\,|\,]$.

Regarding a Boolean circuit as a real electric circuit, this means that $|$-function gates alone suffice to compute any Boolean function. In electronics, they are called *NAND gates* ("not-and").

Remark 18.18. There is another instructive proof for $\top \subseteq [\neg, \wedge, \vee]$. First of all, we prove the equation

$$f(y, \vec{x}) = (y \wedge f(1, \vec{x})) \vee (\neg y \wedge f(0, \vec{x}))$$

for every function $f^{(n+1)}$ and $y \in \underline{2}$, $\vec{x} \in \underline{2}^n$ (this equation is known as an *expansion of f in the first argument* (there are other expansions of diverse types)). Indeed, if $y = 1$, it turns into $f(1, \vec{x}) = (1 \wedge f(1, \vec{x})) \vee (\neg 1 \wedge f(0, \vec{x})) = (1 \wedge f(1, \vec{x})) \vee 0$, which is clearly true. The case when $y = 0$ results in $f(0, \vec{x}) = 0 \vee (1 \wedge f(0, \vec{x}))$, that surely holds.

Now, we are to employ induction on the number n of a function's arguments. If $n = 1$, there are just four Boolean functions: $0^{(1)}$, $1^{(1)}$, $\neg$, $\mathrm{id}_{\underline{2}}$ (where *constants* $0^{(1)}$ and $1^{(1)}$ are defined by the equations $0(x) = 0$ and $1(x) = 1$ for each $x \in \underline{2}$, respectively), each of which belongs to $[\neg, \wedge, \vee]$ as $0(x) = x \wedge \neg x$, $1(x) = x \vee \neg x$, while $\mathrm{id}_{\underline{2}}$ is a projector.

Assume that every Boolean function of n arguments belongs to $[\neg, \wedge, \vee]$. Consider a function $f^{(n+1)}$. By the above formula, get

$$f(y, \vec{x}) = (y \wedge g_1(\vec{x})) \vee (\neg y \wedge g_2(\vec{x})),$$

where g_1 and g_2 with $g_1(\vec{x}) = f(1, \vec{x})$ and $g_2(\vec{x}) = f(0, \vec{x})$ are functions of n arguments and, hence, belong to $[\neg, \wedge, \vee]$. It is now obvious how one can construct a circuit for f over $[\neg, \wedge, \vee]$ given such circuits for g_1, g_2.

Actually, the above expansion formula suggests a direct proof for Theorem 18.14. Indeed, one can expand a function in more than one argument, say, in the first and second ones:

$$\begin{aligned} f(\vec{x}) = (x_1 \wedge x_2 \wedge f(1, 1, x_3, \ldots x_n)) \vee (x_1 \wedge \neg x_2 \wedge f(1, 0, x_3, \ldots x_n)) \vee \\ (\neg x_1 \wedge x_2 \wedge f(0, 1, x_3, \ldots x_n)) \vee (\neg x_1 \wedge \neg x_2 \wedge f(0, 0, x_3, \ldots x_n)). \end{aligned}$$

Continuing this way, one can make f-terms in the right-hand side constants, which effectively turns the right-hand side into a DNF.

Exercise 18.19. Prove Theorem 18.14 and the similar statement for CNF applying suitable expansion formulas.

Another important function set is $\{\wedge, +\}$, which we have already come across when building a binary adder. Is this set complete in $\top$? In fact, it is not but it lacks not much to be such. Moreover, there exists a very natural 'normal form' related to these functions. According to Remark 18.3, we will identify $\wedge$ with multiplication operation $\cdot$. Let us introduce more special form 'expressions' (easily formalizable as circuits, of course).

Let $\vec{x} = (x_1, x_2, \ldots, x_n)$ be a fixed tuple of *pairwise distinct* variables. A product of the form $x_{i_1}x_{i_2}\ldots x_{i_k} = x_{i_1} \cdot x_{i_2} \cdot \ldots \cdot x_{i_k} = x_{i_1} \wedge x_{i_2} \wedge \ldots \wedge x_{i_k}$ where $i_1 < i_2 < \ldots < i_k$ is called a *monomial of degree k*. If $k = 0$, we identify the resulting 'empty' product with $1 \in \underline{2}$. Consider a product $x_1x_2x_1x_1x_2$. Formally, it is not a monomial but it 'equals' one (in value) for $x_2x_1 = x_1x_2$ and $x_ix_i = x_i$ by Lemma 18.2. That is why we may require all the variables in a monomial to be arranged in ascending order. Let $a \in \underline{2}$ and $x_{i_1}x_{i_2}\ldots x_{i_k}$ be a monomial. By $ax_{i_1}x_{i_2}\ldots x_{i_k}$ we denote $x_{i_1}x_{i_2}\ldots x_{i_k}$ if $a = 1$, and 0 otherwise. We shall call a the *coefficient for* $x_{i_1}x_{i_2}\ldots x_{i_k}$ then.

How many monomials over $\vec{x}$ exist? Clearly, one can encode each monomial with the set $\{i_1, i_2, \ldots, i_k\}$, $i_1 < i_2 < \ldots < i_k$, and encode each set in its turn with the binary word

$$\underbrace{00\ldots0}_{i_1-1}1\underbrace{00\ldots0}_{i_2-i_1-1}1\ \underbrace{00\ldots0}_{i_3-i_2-i_1-1}\ 1\ldots\ \underbrace{00\ldots0}_{i_k-i_{k-1}-\ldots-i_1-1}\ 1\underbrace{00\ldots0}_{n-i_k},$$

that is, the word of length n with '1's at positions number $i_1, i_2, \ldots,$ and i_k exactly. It is easy to see that this encoding is indeed a bijection between the set of monomials and $\underline{2}^n$. For $n = 5$, e. g., the word 01101 encodes the monomial $x_2x_3x_5$ while 00000 encodes the 'empty' monomial 1. So, we have 2^n monomials overall. Furthermore, this encoding makes it convenient to name coefficients according to their respective monomials. Say, the notation a_{01101} stands for the coefficient for $x_2x_3x_5$. One more natural step in this direction is to identify a binary word $\vec{\sigma}$ with its meaning $b(\vec{\sigma})$ as a natural number. Then we may use a_{13} instead of a_{01101}, a_{31} for a_{11111}, and a_0 for a_{00000}.

With all these notations fixed, we may introduce our normal form finally. It is a *Zhegalkin polynomial*, that is, a sum of the form

$$P(\vec{x}) = a_{00\ldots0}\,1 + a_{10\ldots0}\,x_1 + a_{010\ldots0}\,x_2 + \ldots + a_{0\ldots01}\,x_n + a_{110\ldots0}\,x_1x_2 + \ldots + a_{11\ldots1}\,x_1x_2\ldots x_n.$$

The order of summands does not matter for computing the value $P(\vec{x})$ at $\vec{x} \in \underline{2}^n$, so we do not want to fix it formally. It is easy to see that this value can be computed by a natural circuit of $\vec{x}$ over the set $\{\cdot, +, 1^{(1)}\}$, where $1^{(1)}$ is the *constant* 1 unary function defined by the equation $1(x) = 1$ for each $x \in \underline{2}$. Formally, one might *define* a polynomial as such a circuit, but this would make coefficients $a_{\vec{\sigma}}$ less highlighted, though they are quite important here as we shall see in a moment.

Theorem 18.20. *For every Boolean function $f^{(n)}$, there exists a unique tuple* $(a_{00\ldots0}, a_{010\ldots0}, \ldots, a_{11\ldots1}) \in \underline{2}^{2^n}$ *such that*

$$f(\vec{x}) = a_{00\ldots0}\,1 + a_{10\ldots0}\,x_1 + a_{010\ldots0}\,x_2 + \ldots + a_{0\ldots01}\,x_n + a_{110\ldots0}\,x_1x_2 + \ldots + a_{11\ldots1}\,x_1x_2\ldots x_n \qquad (10)$$

for all $\vec{x} \in \underline{2}^n$.

Thus, unlike DNF and CNF, the polynomial representation of a Boolean function is essentially unique.

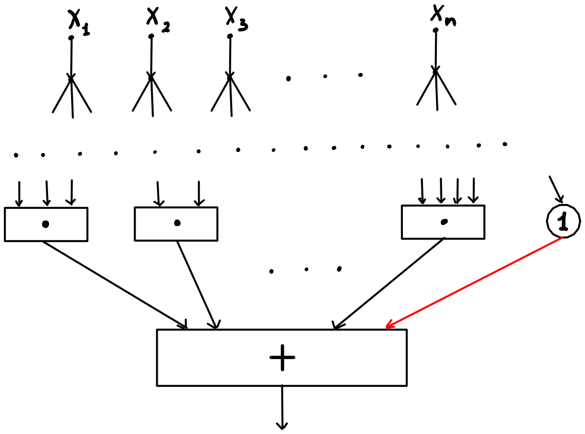

Figure 36: A Zhegalkin polynomial as a Boolean circuit. The red arrow is present iff $a_{00\ldots0} = 1$. Otherwise, a circuit over $\{\cdot, +\}$ would suffice.

Proof. There are finitely many tuples $\vec{\sigma} \in \underline{2}^n$, so by substituting all possible values $\vec{\sigma}$ for $\vec{x}$, we see that equation (10) is equivalent to the following system of simultaneous equations:

$$\begin{cases} f(000\ldots0) & = a_{00\ldots0} \\ f(100\ldots0) & = a_{00\ldots0} + a_{10\ldots0} \\ f(010\ldots0) & = a_{00\ldots0} + a_{010\ldots0} \\ & \ldots \\ f(110\ldots0) & = a_{00\ldots0} + a_{10\ldots0} + a_{010\ldots0} + a_{110\ldots0} \\ & \ldots \\ f(111\ldots1) & = a_{00\ldots0} + a_{10\ldots0} + a_{010\ldots0} + \ldots + a_{0\ldots01} + a_{110\ldots0} + \ldots + a_{11\ldots1}. \end{cases} \tag{11}$$

What is the main idea behind this? Assume that a binary tuple $\vec{\sigma} \in \underline{2}^n$ has '1's at positions $\{i_1, i_2, \ldots, i_k\}$ exactly—in other words, $\vec{\sigma}$ encodes the monomial $x_{i_1}x_{i_2}\ldots x_{i_k}$. This monomial takes 1 at $\vec{\sigma}$ (so we have $a_{\vec{\sigma}}$ in the right-hand side) as well as every monomial encoded by a subset of $\{i_1, i_2, \ldots, i_k\}$ does. That is, if $\vec{\tau} \in \underline{2}^n$ has '1's at *some* positions from $\{i_1, i_2, \ldots, i_k\}$ but nowhere else, the summand $a_{\vec{\tau}}$ is present in the right-hand side, and vice versa. Let us denote this set of tuples $\vec{\tau}$ by $I(\vec{\sigma})$. So, we have $2^k = |I(\vec{\sigma})|$ summands in the right-hand side when $\vec{\sigma}$ contains just k unities. For example, if $\vec{\sigma} = 01101$, we get $a_{\vec{\tau}}$ in the right-hand side iff $\vec{\tau} \in I(\vec{\sigma}) = \{01101, 00101, 01001, 01100, 01000, 00100, 00001, 00000\}$. It is clear that $b(\vec{\tau}) \leq b(\vec{\sigma})$ for each $\vec{\tau} \in I(\vec{\sigma})$ since $b(\vec{\sigma}) = 2^{n-i_1} + 2^{n-i_2} + \ldots 2^{n-i_k}$ while $b(\vec{\tau})$ includes just some of these summands.

Now, let us have a look at system (11). It is a system of 2^n linear equations in 2^n variables $a_{\vec{\sigma}}$ with coefficients from $\underline{2}$. A tuple $(a_{00\ldots0}, a_{010\ldots0}, \ldots, a_{11\ldots1}) \in \underline{2}^{2^n}$ is a solution to the system iff this tuple satisfies our required constraint (10).

We know from Remark 18.3 that the theory of solving such systems is basically the same as what the Reader should know for real coefficients from his Linear Algebra course. According to that theory, a system of linear equations has a unique solution iff the determinant of its matrix is not zero. Let us prove this is the case.

Reordering equations or variables preserves the absolute value of the determinant, so let us arrange the equations from (11), which are of the form $f(\vec{\sigma}) = \sum_{\vec{\tau} \in I(\vec{\sigma})} a_{\vec{\tau}} = a_{00\ldots0} + \ldots + a_{\vec{\sigma}}$, as $b(\vec{\sigma})$ ascends.

Identifying $\vec{\sigma}$ and $b(\vec{\sigma})$, we obtain the system

$$\begin{cases} f(0) & = a_0 \\ f(1) & = a_0 + a_1 \\ f(2) & = a_0 + a_2 \\ f(3) & = a_0 + a_1 + a_2 + a_3 \\ f(4) & = a_0 + a_4 \\ & \ldots \\ f(2^n - 1) & = a_0 + a_1 + a_2 + \ldots + a_{2^n - 1}. \end{cases}$$

The key observation here is that the equation for $f(b(\vec{\sigma}))$ clearly takes the $b(\vec{\sigma})$-th place and, in the right-hand side, it contains a sum of certain numbers a_i with $i \leq b(\vec{\sigma})$ whereas $a_{b(\vec{\sigma})}$ is always present. So, the matrix of the latter system is a lower triangular one with all unities at its main diagonal. The determinant of the matrix is thus 1, so it is non-zero for the original system as well.[30] □

Example 18.21. The proof above effectively reduces finding a polynomial for a function to solving a system of linear equations over the field GF(2). Given the values of f, this procedure may be highly efficient. Even without much optimization (but see Remark 18.24), solving a small-sized system is really easy.

Consider the Boolean function $f^{(3)}$ defined by the table

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Let us find such a tuple $\vec{a} = (a_{000}, a_{100}, \ldots, a_{111}) \in \underline{2}^8$ that

$$f(x,y,z) = a_{000} + a_{100}x + a_{010}y + a_{001}z + a_{110}xy + a_{101}xz + a_{011}yz + a_{111}xyz$$

for each $(x,y,z) \in \underline{2}^3$. Applying this equation to every possible value of (x,y,z), obtain the system

$$\begin{cases} a_{000} & = & 1 & = & f(0,0,0) \\ a_{000} + a_{100} & = & 0 & = & f(1,0,0) \\ a_{000} + a_{010} & = & 0 & = & f(0,1,0) \\ a_{000} + a_{001} & = & 1 & = & f(0,0,1) \\ a_{000} + a_{100} + a_{010} + a_{110} & = & 0 & = & f(1,1,0) \\ a_{000} + a_{100} + a_{001} + a_{101} & = & 0 & = & f(1,0,1) \\ a_{000} + a_{010} + a_{001} + a_{011} & = & 1 & = & f(0,1,1) \\ a_{000} + a_{100} + a_{010} + a_{001} + a_{110} + a_{101} + a_{011} + a_{111} & = & 1 & = & f(1,1,1). \end{cases}$$

[30] In fact, the determinant of the original system (11) is 1 for $-1 = 1$ in the field GF(2) (as $1 + 1 = 0$).

This is easily solvable by standard Gaussian elimination. Say, one can add rows 1, 2, 3, and 5 together to obtain $a_{110} = a_{000} + a_{000} + a_{100} + a_{000} + a_{010} + a_{000} + a_{100} + a_{010} + a_{110} = 1 + 0 + 0 + 0 = 1$. Finally, $\vec{a} = (1, 1, 1, 0, 1, 0, 1, 0)$ and

$$f(x, y, z) = 1 + x + y + xy + yz.$$

You may want to check this polynomial values to match the above table.

Corollary 18.22. $\top = [\,\cdot, +, 1^{(1)}]$.

Indeed, given the function $1^{(1)}$, one can compute the value $1 \in \underline{2}$ for the polynomial as $1(x_1)$. Notice that the constant $0^{(1)}$ is not needed for Zhegalkin polynomials since it can be obtained by letting all the coefficients equal zero (so the polynomial is 'empty' and does not contain any term).

Example 18.23. It is worth memorizing Zhegalkin polynomials for these important functions:

$$\begin{array}{rcl} \neg x & = & 1 + x; \\ x \vee y & = & x + y + xy; \\ x \to y & = & 1 + x + xy; \\ x \leftrightarrow y & = & 1 + x + y; \\ \mathrm{maj}(x, y, z) & = & xy + xz + yz. \end{array}$$

The first two equations here suffice to prove Corollary 18.22. Indeed, we have $\top = [\neg, \vee]$ by Corollary 18.17; then, clearly, $\neg \in [+, 1^{(1)}]$ and $\vee \in [\,\cdot, +, 1^{(1)}]$, whence $\top = [\neg, \vee] \subseteq [[\,\cdot, +, 1^{(1)}]] = [\,\cdot, +, 1^{(1)}]$ by Theorem 18.8. Yet this does not make Zhegalkin polynomials redundant because they are still important for their unicity property.

Remark 18.24. It is not hard to solve system (11) explicitly. Recall that $I(\vec{\sigma})$ is the set of all tuples $\vec{\tau}$ that may be obtained from $\vec{\sigma}$ by changing some '1's to '0's. If $\vec{\sigma}$ has '1's at positions $i_1, \ldots, i_k$ exactly, we get $|I(\vec{\sigma})| = 2^k$. System (11) contains the equations $f(\vec{\sigma}) = \sum_{\vec{\tau} \in I(\vec{\sigma})} a_{\vec{\tau}}$ for each $\vec{\sigma} \in \underline{2}^n$.

Now, let us consider the sum

$$\sum_{\vec{\tau} \in I(\vec{\sigma})} f(\vec{\tau}) = \sum_{\vec{\tau} \in I(\vec{\sigma})} \sum_{\vec{\rho} \in I(\vec{\tau})} a_{\vec{\rho}} = \sum_{\vec{\rho} \in I(\vec{\sigma})} c_{\vec{\rho}} \cdot a_{\vec{\rho}},$$

where the coefficient $c_{\vec{\rho}} \in \underline{2}$ shows how many occurrences $a_{\vec{\rho}}$ has in the sum ($c_{\vec{\rho}} = 0$ if this number $c'_{\vec{\rho}} \in \mathbb{N}$ is even, and $c_{\vec{\rho}} = 1$ otherwise). To get this formula, we have applied the fact that $I(\vec{\tau}) \subseteq I(\vec{\sigma})$ when $\vec{\tau} \in I(\vec{\sigma})$.

What is $c'_{\vec{\rho}}$ for a fixed $\vec{\rho} \in I(\vec{\sigma})$? Obviously, it is the number of such tuples $\vec{\tau} \in I(\vec{\sigma})$ that $\vec{\rho} \in I(\vec{\tau})$. As $\vec{\sigma}$ has '1's just at the positions from $I_1 = \{i_1, \ldots, i_k\}$, the 'position set' for $\vec{\rho}$ is some $I_2 = \{i_{t_1}, \ldots, i_{t_s}\} \subseteq I_1$. If I_3 is the 'position set' for $\vec{\tau}$, our requirement on $\vec{\tau}$ is clearly equivalent to $I_2 \subseteq I_3 \subseteq I_1$, that is, $I_3 = I_2 \cup X$ with $X \subseteq I_1 \smallsetminus I_2$. Hence, $c'_{\vec{\rho}} = |\mathcal{P}(I_1 \smallsetminus I_2)| = 2^{k-s}$. This number is even whenever $s < k$, so $c_{\vec{\rho}} = 0$ then. If $s = k$, we have $I_2 = I_1$, i. e., $\vec{\rho} = \vec{\sigma}$, and $c'_{\vec{\rho}} = 2^0 = 1 = c_{\vec{\rho}}$.

Finally,

$$\sum_{\vec{\tau} \in I(\vec{\sigma})} f(\vec{\tau}) = a_{\vec{\sigma}},$$

which is the explicit solution to (11). For example, $a_{00101} = f(00000) + f(00001) + f(00100) + f(00101)$. Such sums share some parts for distinct $a_{\vec{\sigma}}$. Therefore, further computational optimizations are possible.

Remark 18.25. One can prove Theorem 18.20 without any reference to linear algebra. Indeed, for each $\vec{a} = (a_{00\ldots0}, a_{010\ldots0}, \ldots, a_{11\ldots1}) \in \underline{2}^{2^n}$, the right-hand side polynomial $P_{\vec{a}}(\vec{x})$ of equation (10) represents a Boolean function we may denote by $\varphi(\vec{a})$. Hence, φ is a function from $\underline{2}^{2^n}$ to $\underline{2}^{\underline{2}^n}$, between two finite sets of equal size.

Let us show that φ is injective. Assume that $\varphi(\vec{a}) = \varphi(\vec{b})$ but $\vec{a} \neq \vec{b}$, whence $P_{\vec{a}}(\vec{x})$ and $P_{\vec{b}}(\vec{x})$ differ in at least one coefficient, so $Q(\vec{x}) = P_{\vec{a}}(\vec{x}) + P_{\vec{b}}(\vec{x}) = P_{\vec{c}}(\vec{x}) = P_{\vec{a}+\vec{b}}(\vec{x})$ has at least one non-zero coefficient (as one might expect, $\vec{a} + \vec{b} = (a_0 + b_0, \ldots, a_{2^n-1} + b_{2^n-1})$). On the other hand, the value of $P_{\vec{a}}(\vec{x})$ equals $\varphi(\vec{a})(\vec{x})$ at each $\vec{x} \in \underline{2}^n$, so the values of $P_{\vec{a}}(\vec{x})$ and $P_{\vec{b}}(\vec{x})$ are always identical; hence $Q(\vec{x}) = 0$ for each $\vec{x} \in \underline{2}^n$.

At least one coefficient $c_{\vec{\sigma}}$ in Q is 1. Let us take a tuple $\vec{\sigma}$ with $c_{\vec{\sigma}} = 1$ that contains the least possible number k of '1's (which is know as the *weight* $||\vec{\sigma}|| \in \mathbb{N}$ of $\vec{\sigma}$). This tuple has '1's at positions $i_1, i_2, \ldots, i_k$ exactly and encodes the monomial $x_{i_1} x_{i_2} \ldots x_{i_k}$. What is the value $Q(\vec{\sigma})$? The monomial $x_{i_1} x_{i_2} \ldots x_{i_k}$ (in particular, 1 if $k = 0$) contributes 1 to the sum; any monomial of lesser weight contributes 0 as its coefficient is zero; any monomial of greater or equal weight (yet other than $\vec{\sigma}$) contains a variable x_j with $j \notin \{i_1, i_2, \ldots, i_k\}$, so it contributes 0 as well. Thus, $Q(\vec{\sigma}) = 1$, which is impossible. A contradiction.

By Theorem 12.8, φ is surjective, that is, a tuple $\vec{a}$ with $P_{\vec{a}}(\vec{x}) = \varphi(\vec{a})(\vec{x}) = f(\vec{x})$ exists for every Boolean function f. Such $\vec{a}$ is unique by injectivity of φ.

The main drawback of this proof is that it gives no explicit algorithm for computing $\vec{a}$.

19 Closed sets

Up to now, we have seen $\top$ and $\bot$ as our only closed set examples with an explicit characterization (that is, other than "$[Q]$ for a given Q"). It is known that there are countably many closed sets of Boolean functions; for each set, a finite basis and quite a neat characterization are known as well. Let us consider a few particularly important closed sets.

Constant-preserving functions. The set $P_0 = \{f \in \top \mid f(\vec{0}) = 0\}$, where $\vec{0} = 00\ldots0$ (the length is just how many arguments f has), consists of all functions that *preserve constant* 0. Similarly, $P_1 = \{f \in \top \mid f(\vec{1}) = 1\}$, where $\vec{1} = 11\ldots1$, is the set of functions *preserving constant* 1. The set $P = P_0 \cap P_1$ is called that of *constant-preserving functions*. If P_0 and P_1 are closed, then P must be closed as well by Example 18.10.

Example 19.1. We have $\wedge, \vee \in P_0 \cap P_1$; $+ \in P_0 \setminus P_1$; $\rightarrow \in P_1 \setminus P_0$; and $\neg \notin P_0 \cup P_1$.

The set P_0 (and similarly P_1) is closed indeed. Let us prove $[P_0] \subseteq P_0$, i. e., every function $f \in [P_0]$ satisfies the property $f \in P_0$. By Lemma 18.7, it suffice to check that $P_0 \subseteq P_0$ (trivial); $\bot \subseteq P_0$, which is also clear as $p(00\ldots0) = 0$ for every projector p; and that superposition preserves this property. The latter means that from $h, g_1, \ldots, g_m \in P_0$ and $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$, it follows that $f \in P_0$. Indeed,

$$f(\vec{0}) = h(g_1(\vec{0}), \ldots, g_m(\vec{0})) = h(0, \ldots, 0) = 0.$$

Example 19.2. It is not possible to express $\rightarrow$ via $\wedge$ and $\vee$, i. e., $\rightarrow \notin [\wedge, \vee]$. Indeed, one gets $\wedge, \vee \in P_0$, whence $[\wedge, \vee] \subseteq [P_0] = P_0$ but $\rightarrow \notin P_0$. One may say that the set P_0 (or the property to preserve 0) is an *obstacle* (or *invariant*) preventing $\rightarrow$ from being computable over $\{\wedge, \vee\}$: every such function must preserve 0, whereas $\rightarrow$ does not. So, when checking a function for being expressible via some others, one either finds a circuit to prove it is, or finds an obstacle to prove it is not.

Exercise 19.3. Prove that there are exactly 2^{2^n-1} functions of n arguments in P_i, $i \in \underline{2}$.

Monotonicity. Another interesting set is that of *monotonic* functions. Let $\vec{\sigma}, \vec{\tau} \in \underline{2}^n$. We put $\vec{\sigma} \leq \vec{\tau}$ iff $\sigma_i \leq \tau_i$ for each i from 1 to n (x_i and y_i are compared w. r. t. the 'natural' order where $0 \leq 1$). For example, $(1,0,1,0,0) \leq (1,1,1,0,1)$, whereas $(1,0,1,0,0)$ and $(1,1,0,0,0)$ are incomparable. It is easy to see that $\leq$ is a non-strict partial order on the set $\underline{2}^n$. (Formally, one may define such a relation separately for each n.) From Section 14, we know that the relation $<$ with $\vec{\sigma} < \vec{\tau} \iff \vec{\sigma} \leq \vec{\tau} \wedge \sigma \neq \tau$ is the strict partial 'counterpart' order for $\leq$. Clearly, $\vec{\sigma} < \vec{\tau}$ iff there is one or more indices $i_1, \ldots, i_k$ such that $\sigma_j < \tau_j$ when $j = i_s$ for some $s \leq k$ and $\sigma_j = \tau_j$ otherwise. In the case of $(1,0,1,0,0) < (1,1,1,0,1)$, one has $\{i_1, i_2\} = \{2, 5\}$. See Figure 17 for a diagram of the poset $(\underline{2}^3, \leq)$ (each tuple $\vec{\sigma}$ is identified with its natural number value $b(\vec{\sigma})$ there). A Boolean function f is called *monotonic* iff $f(\vec{\sigma}) \leq f(\vec{\tau})$ whenever $\vec{\sigma} \leq \vec{\tau}$. The set of all monotonic functions is denoted by M.

Example 19.4. We have $\wedge, \vee, 0^{(1)}, 1^{(1)} \in M$ but $+, \neg \notin M$. In particular, $1+1 = 0 < 1 = 0+1$ despite $(1,1) > (0,1)$.

Why is the set M closed? Let us apply Lemma 18.7 again. Clearly, $M \subseteq M$ and $\bot \subseteq M$. Assume that $h, g_1, \ldots, g_m \in M$ and $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$. Let $\vec{\sigma} \leq \vec{\tau}$. Then we have $g_j(\vec{\sigma}) \leq g_j(\vec{\tau})$ for each j, whence $(g_1(\vec{\sigma}), \ldots, g_m(\vec{\sigma})) \leq (g_1(\vec{\tau}), \ldots, g_m(\vec{\tau}))$ and $f(\vec{\sigma}) = h(g_1(\vec{\sigma}), \ldots, g_m(\vec{\sigma})) \leq h(g_1(\vec{\tau}), \ldots, g_m(\vec{\tau})) = f(\vec{\tau})$ finally.

Example 19.5. Let $f(x, y, z) = x + y + z$. This function f is not computable via $\{\wedge, \vee, 0^{(1)}, 1^{(1)}\}$. Indeed, the latter functions are monotonic, while f is not: $f(001) = 1 > 0 = f(011)$ but $001 < 011$.

Remark 19.6. In the above, we have defined the set $I(\vec{\sigma})$ for each tuple $\vec{\sigma} \in \underline{2}^n$ so that $\vec{\tau} \in I(\vec{\sigma})$ iff $\vec{\tau}$ may be obtained from $\vec{\sigma}$ by changing some '1's to '0's. This set has been important for finding a Zhegalkin polynomial for a function. It is clear now that $\vec{\tau} \in I(\vec{\sigma})$ iff $\vec{\tau} \leq \vec{\sigma}$.[31] Hence, the formula for Zhegalkin coefficients from Remark 18.24 turns into $a_{\vec{\sigma}} = \sum_{\vec{\tau} \leq \vec{\sigma}} f(\vec{\tau})$, while $f(\vec{\sigma}) = \sum_{\vec{\tau} \leq \vec{\sigma}} a_{\vec{\tau}}$ by system (11). This nice symmetry is an example of the so-called *Möbius Inversion* for finite posets.

Exercise 19.7. Prove that $(\underline{2}^n, \leq) \cong (\mathcal{P}(\underline{n}), \subseteq)$.

Duality. One more closed set we need is the set S of *self-dual* functions. For $\vec{\sigma} = (\sigma_1, \ldots, \sigma_n) \in \underline{2}^n$, we put $\neg\vec{\sigma} = (\neg\sigma_1, \ldots, \neg\sigma_n)$. For example, $\neg(1, 0, 1) = (0, 1, 0)$. The function f^* defined by the equality $\neg f^*(\vec{\sigma}) = f(\neg\vec{\sigma})$ for each $\vec{\sigma} \in \underline{2}^n$ is called the *dual* of a function $f^{(n)}$. Equivalently, one has $f^*(\vec{\sigma}) = \neg f(\neg\vec{\sigma})$ for each $\vec{\sigma}$. A function f is *self-dual* iff $f^* = f$ or, equivalently, $\neg f(\vec{\sigma}) = f(\neg\vec{\sigma})$ for all $\vec{\sigma}$. In other words, f is self-dual iff its value is always inverted when all its arguments are. Notice that a function f is *not* self-dual iff there exists a tuple $\vec{\sigma}$ with $f(\vec{\sigma}) = f(\neg\vec{\sigma})$.

Exercise 19.8. Prove that $(f^*)^* = f$.

Example 19.9. One has $x \wedge y = \neg(\neg x \vee \neg y)$, so $\wedge = \vee^*$ (whence $\wedge^* = (\vee^*)^* = \vee$). As $\wedge \neq \vee$, neither function is self-dual. Nor $+$ is self-dual for $0+0 = 1+1$. In fact, $x +^* y = \neg(\neg x + \neg y) = 1 + x + 1 + y + 1 = 1 + x + y = x \leftrightarrow y$, so $+^* = \leftrightarrow$. On the other hand, the functions $\neg$ and maj are self-dual: $\neg\neg x = \neg\neg x$ and $\text{maj}(\neg x, \neg y, \neg z) = \text{maj}(x+1, y+1, z+1) = (x+1)\cdot(y+1) + (x+1)\cdot(z+1) + (y+1)\cdot(z+1) = xy + xz + yz + x + x + y + y + z + z + 1 + 1 + 1 = xy + xz + yz + 1 = \neg\text{maj}(x, y, z)$.

Exercise 19.10. Prove that there exist exactly $2^{2^{n-1}}$ self-dual functions of n arguments.

Let us check that $[S] = S$. At first, we need a useful

Lemma 19.11. *Suppose that* $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ *for all* $\vec{x}$. *Then* $f^*(\vec{x}) = h^*(g_1^*(\vec{x}), \ldots, g_m^*(\vec{x}))$ *for all* $\vec{x}$, *i. e., the superposition of some functions' duals equals the dual of their superposition.*

Proof.

$$f^*(\vec{x}) = \neg f(\neg\vec{x}) = \neg h(g_1(\neg\vec{x}), \ldots, g_m(\neg\vec{x})) = \neg h(\neg g_1^*(\vec{x}), \ldots, \neg g_m^*(\vec{x})) = h^*(g_1^*(\vec{x}), \ldots, g_m^*(\vec{x})).$$

□

Now, apply Lemma 18.7. If $p \in \perp$, we get $\neg p(\vec{\sigma}) = \neg\sigma_j = p(\neg\vec{\sigma})$, whence $p \in S$. Assume that $h, g_1, \ldots, g_m \in S$ and $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$. Then $f^*(\vec{x}) = h^*(g_1^*(\vec{x}), \ldots, g_m^*(\vec{x})) = h(g_1(\vec{x}), \ldots, g_m(\vec{x})) = f(\vec{x})$ by Lemma 19.11. So, $f \in S$.

Example 19.12. We have $\rightarrow \notin [\text{maj}, \neg]$. Indeed, $[\text{maj}, \neg] \subseteq [S] = S$, though $\rightarrow \notin S$ for $0 \rightarrow 0 = 1 \rightarrow 1$.

Let C be a circuit over a set Q computing a function f. What if we change every gate in C to its dual to obtain a circuit C^*? It seems clear that C^* computes f^*. But let us elaborate on this point. For a set Q of Boolean functions, we put $Q^* = \{f^* \in \top \mid f \in Q\}$. The set Q^* is thus *dual* to Q.

Lemma 19.13. *For any Boolean function sets* P *and* Q,

[31] The set $I(\vec{\sigma})$ is then called the *principal ideal generated by* $\vec{\sigma}$ in the poset $(\underline{2}^n, \leq)$.

1. $(Q^*)^* = Q$*;*

2. if $P \subseteq Q$*, then* $P^* \subseteq Q^*$*;*

3. $[Q]^* = [Q^*]$*;*

4. if Q *is a basis of* P*, then* Q^* *is a basis of* P^**.*

Proof. The first and second claims are obvious. For the third one, let us first prove $[Q] \subseteq [Q^*]^*$ for an arbitrary Q.

We can do it by applying Lemma 18.7. One has $Q^* \subseteq [Q^*]$ by Theorem 18.8, whence $Q = (Q^*)^* \subseteq [Q^*]^*$ by the previous claims. For projectors, we get $\bot = [\varnothing] \subseteq [Q^*]$ by Theorem 18.8. As we know, projectors are self-dual; hence $\bot = \bot^* \subseteq [Q^*]^*$ by the second claim. Assume now that $h, g_1, \ldots, g_m \in [Q^*]^*$ and $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$. By Lemma 19.11, $f^*(\vec{x}) = h^*(g_1^*(\vec{x}), \ldots, g_m^*(\vec{x}))$ for each $\vec{x}$. On the other hand, $h^*, g_1^*, \ldots, g_m^* \in ([Q^*]^*)^* = [Q^*]$. Then clearly $f^* \in [[Q^*]] = [Q^*]$ (see the proof of Theorem 18.8 if in doubt), whence $f = (f^*)^* \in [Q^*]^*$.

Finally, $[Q]^* \subseteq ([Q^*]^*)^* = [Q^*]$ by the previous claims, and $[Q^*] \subseteq [(Q^*)^*]^* = [Q]^*$ (as Q is arbitrary, we may substitute Q^* for Q).

For the last claim, assume that $[Q] = P$ but $[R] \subsetneq P$ for any $R \subsetneq Q$. Then $[Q^*] = [Q]^* = P^*$, so Q^* is complete in P^*. Suppose that $P^* = [R]$ for some $R \subsetneq Q^*$. Hence $[R^*] = [R]^* = (P^*)^* = P$, whereas $R^* \subsetneq (Q^*)^* = Q$ clearly. The contradiction shows that Q^* is a basis of P^*. □

Exercise 19.14. Prove that $S^* = S$.

Example 19.15. Consider the sets P_0^* and P_1^*. If $f \in P_0$, one has $f^*(\vec{1}) = \neg f(\neg\vec{1}) = \neg f(\vec{0}) = \neg 0 = 1$, that is, $f^* \in P_1$. So, $P_0^* \subseteq P_1$. One can likewise prove $P_1^* \subseteq P_0$. Then $P_0 = (P_0^*)^* \subseteq P_1^* \subseteq P_0$, whence $P_1^* = P_0$, and similarly, $P_0^* = P_1$.

Linear functions. The set L of *linear* functions may be defined as $[+, 1^{(1)}]$. On the one hand, L is clearly closed by Theorem 18.8. On the other hand, this characterization is not nice as we see no interesting property of the *functions* from L themselves (but rather of their circuits). It is therefore easy to prove a function belongs to L—just give a suitable circuit, while it seems somewhat hard to prove it does not: *why* cannot we have a circuit?

But things are much better here in view of Theorem 18.20. Indeed, for each function $f^{(n)}$, its Zhegalkin coefficients $a_{\vec{\sigma}}$ are unique. Let us prove that $f \in L$ iff $a_{\vec{\sigma}} = 0$ for each tuple $\vec{\sigma}$ with $||\vec{\sigma}|| \geq 2$ (that is, with two or more '1's) or, equivalently, the equation

$$f(\vec{x}) = a_{00\ldots0}\,1 + a_{10\ldots0}\,x_1 + a_{010\ldots0}\,x_2 + \ldots + a_{0\ldots01}\,x_n \tag{12}$$

holds for each $\vec{x}$. If it indeed holds, one can easily construct a circuit for f over $\{+, 1^{(1)}\}$, so $f \in [+, 1^{(1)}] = L$. For the other direction, we will prove that for each $f^{(n)} \in [+, 1^{(1)}]$, there exists a tuple $(b_0, b_1, \ldots, b_n) \in \underline{2}^{n+1}$ such that

$$f(\vec{x}) = b_0\,1 + b_1\,x_1 + b_2\,x_2 + \ldots + b_n\,x_n. \tag{13}$$

By structural induction on f (Lemma 18.7). When $f \in \{+, 1^{(1)}\} \cup \bot$, this is obvious. Assume that $f(\vec{x}) = h(g_1(\vec{x}), \ldots, g_m(\vec{x}))$ for all $\vec{x}$ and

$$\begin{array}{rcl} h(\vec{y}) & = & c_0\,1 + c_1\,y_1 + \ldots + c_m\,y_m \\ g_1(\vec{x}) & = & d_0^1\,1 + d_1^1\,x_1 + \ldots + d_n^1\,x_n \\ & \ldots & \\ g_m(\vec{x}) & = & d_0^m\,1 + d_1^m\,x_1 + \ldots + d_n^m\,x_n. \end{array}$$

Then

$$f(\vec{x}) = c_0\,1 + c_1\,g_1(\vec{x}) + \ldots + c_m\,g_m(\vec{x}) = (c_0 + c_1 d_0^1 + \ldots + c_m d_0^m) \cdot 1 + \\ (c_1 d_1^1 + \ldots + c_m d_1^m) \cdot x_1 + \ldots + (c_1 d_n^1 + \ldots + c_m d_n^m) \cdot x_n,$$

whence the coefficients b_i are clear.

Equation (13) gives a Zhegalkin polynomial for f. Since its coefficients are uniquely determined, there must be $b_0 = a_{00\ldots0}$, $b_1 = a_{10\ldots0}$, $b_2 = a_{010\ldots0}$, $\ldots$, $b_n = a_{0\ldots01}$. Hence, the required equation (12) holds for f.

Example 19.16. In practice, this means that checking $f \in L$ boils down to computing Zhegalkin coefficients for f. The functions $\neg$ and $\leftrightarrow$ are linear as $\neg x = 1 + x$ and $x \leftrightarrow y = 1 + x + y$, whereas $\wedge$ and maj are not: $x \wedge y = xy$ and $\mathrm{maj}(x, y, z) = xy + xz + yz$. Notice that just one coefficient $a_{\vec{\sigma}} = 1$ with $||\vec{\sigma}|| \geq 2$ is sufficient to prove $f \notin L$.

Example 19.17. The function $\wedge$ is not expressible via $\{+, 1^{(1)}\}$ for it is not linear.

Exercise 19.18. Prove that $L^* = L$.

Exercise 19.19. There is a description of L which does not mention circuits, Zhegalkin polynomials nor any other Boolean function representation. Let $(\vec{x}; a/i)$ stand of the tuple $(x_1, \ldots, x_{i-1}, a, x_i, \ldots, x_n)$. Then a function $f^{(n)} \in L$ iff for each $i \leq n$, from $\exists \vec{a}\; f(\vec{a}; 0/i) = f(\vec{a}; 1/i)$, it follows that $\forall \vec{b}\; f(\vec{b}; 0/i) = f(\vec{b}; 1/i)$. In other words, linearity means that if f is *not always dependent* on its i-th argument, then f is *always independent* of it. In view of this description, functions from L are also known as *affine* functions.

Example 19.20. The following table summarizes our results on whether important functions belong to P_0, P_1, M, S, or L. We put '+' into the respective cell if a function belongs to a set, and put '–' otherwise.

	P_0	P_1	M	S	L
$0^{(1)}$	$+$	$-$	$+$	$-$	$+$
$1^{(1)}$	$-$	$+$	$+$	$-$	$+$
$\neg$	$-$	$-$	$-$	$+$	$+$
$\wedge$	$+$	$+$	$+$	$-$	$-$
$\vee$	$+$	$+$	$+$	$-$	$-$
$+$	$+$	$-$	$-$	$-$	$+$
$\rightarrow$	$-$	$+$	$-$	$-$	$-$
$\leftrightarrow$	$-$	$+$	$-$	$-$	$+$
maj	$+$	$+$	$+$	$+$	$-$
$\mid$	$-$	$-$	$-$	$-$	$-$

Notice that the Sheffer stroke $\mid$ belongs to none of P_0, P_1, M, S, L. It is no coincidence that $[\,|\,] = \top$, as we shall learn from Post's Criterion below.

Finding bases. As we have already mentioned, every closed set of Boolean functions has a finite basis. Let us find these for some of the sets.

Example 19.21. For every function f, we have $f(00\dots0) = a_{00\dots0}$ by equation (10). That is, $f \in P_0$ iff $a_{00\dots0} = 0$. The latter implies that the polynomial for f is a circuit over $\{\wedge, +\}$, so $P_0 \subseteq [\wedge, +]$. As $\{\wedge, +\} \subseteq P_0$, we have $[\wedge, +] = P_0$ finally. Furthermore, $\wedge \in P_1$ but $+ \notin P_1$; hence $+ \notin [\wedge]$ and $P_0 \nsubseteq [\wedge]$. Likewise, $+ \in L$ but $\wedge \notin L$, whence $P_0 \nsubseteq [+]$. Therefore, the set $\{\wedge, +\}$ is a basis for P_0.

By Lemma 18.17 and Example 19.15, the set $\{\wedge^*, +^*\} = \{\vee, \leftrightarrow\}$ is a basis for $P_1 = P_0^*$.

A function $f^{(n)}$ is called *conjunctive* iff $f(\vec{\sigma} \wedge \vec{\tau}) = f(\vec{\sigma}) \wedge f(\vec{\tau})$, where $\vec{\sigma} \wedge \vec{\tau} = (\sigma_1 \wedge \tau_1, \dots, \sigma_n \wedge \tau_n)$, for every $\vec{\sigma}, \vec{\tau} \in \underline{2}^n$. Let us denote the set of all conjunctive functions by $\bigwedge$. Clearly, $\wedge, 0^{(1)}, 1^{(1)} \in \bigwedge$, while $\vee \notin \bigwedge$. Indeed, for $\vec{\sigma} = (1, 0)$ and $\vec{\tau} = (0, 1)$, we get

$$\begin{array}{rcccl} (\sigma_1 \wedge \tau_1) \vee (\sigma_2 \wedge \tau_2) & = & (1 \wedge 0) \vee (0 \wedge 1) & = & 0; \\ (\sigma_1 \vee \sigma_2) \wedge (\tau_1 \vee \tau_2) & = & (1 \vee 0) \wedge (0 \vee 1) & = & 1. \end{array}$$

The set $\bigwedge$ is closed. We routinely apply structural induction (Lemma 18.7) to show this. Clearly, $\perp \subseteq \bigwedge$. Assume that $h, g_1, \dots, g_m \in \bigwedge$ and $f(\vec{x}) = h(g_1(\vec{x}), \dots, g_m(\vec{x}))$ for all $\vec{x}$. Then

$$f(\vec{\sigma} \wedge \vec{\tau}) = h(g_1(\vec{\sigma} \wedge \vec{\tau}), \dots, g_m(\vec{\sigma} \wedge \vec{\tau})) = h(g_1(\vec{\sigma}) \wedge g_1(\vec{\tau}), \dots, g_m(\vec{\sigma}) \wedge g_m(\vec{\tau})) =$$
$$h(g_1(\vec{\sigma}), \dots, g_m(\vec{\sigma})) \wedge h(g_1(\vec{\tau}), \dots, g_m(\vec{\tau})) = f(\vec{\sigma}) \wedge f(\vec{\tau}).$$

Every conjunctive function f is monotonic. Indeed, let $\vec{\sigma} \le \vec{\tau}$. Then $\sigma_i \wedge \tau_i = \sigma_i$ for each i (cf. Remark 18.3), whence $f(\vec{\sigma}) = f(\vec{\sigma} \wedge \vec{\tau}) = f(\vec{\sigma}) \wedge f(\vec{\tau}) \le f(\vec{\tau})$. So, $\bigwedge \subseteq M$.

For each Boolean function $f^{(n)}$, consider the set $N_f = \min_{\le} \{\vec{\sigma} \in \underline{2}^n \mid f(\vec{\sigma}) = 1\}$, which is called the set of *lower units* for f. For example, $N_\wedge = \{11\}$, $N_\vee = \{01, 10\}$, $N_{1^{(1)}} = \{0\}$, and $N_{0^{(1)}} = \varnothing$. Clearly, N_f is an antichain in $(\underline{2}^n, \le)$. This set is mainly interesting when $f \in M$.

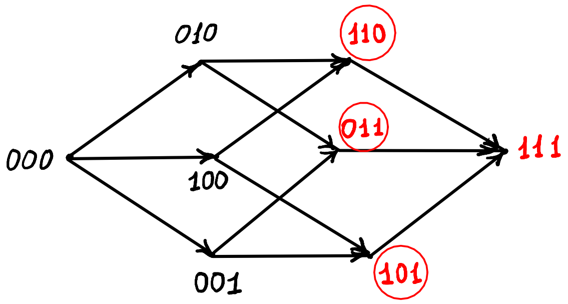

Figure 37: For the function maj, its *units* (i. e., tuples $\vec{\sigma}$ with $\mathrm{maj}(\vec{\sigma}) = 1$) are shown in red. The *lower units* are encircled.

Lemma 19.22. *Let $f \in M$. Then for each $\vec{\tau}$, $f(\vec{\tau}) = 1$ iff there exists $\vec{\sigma} \in N_f$ such that $\vec{\sigma} \le \vec{\tau}$.*

Proof. If there is such a tuple $\vec{\sigma}$, obtain $f(\vec{\tau}) = 1$ by monotonicity. For the other direction, it suffices to prove a general fact: every element of a finite poset (of f-units, in this case) is greater or equal than some *minimal* element thereof (a lower unit). Assume that a finite poset $\mathcal{A} = (A, <)$ has just m elements and $x_0 \in A$. If x_0 is *not* greater nor equal than a minimum, then $x_0 \notin \min \mathcal{A}$, so there exists $x_1 < x_0$. Nor x_1 may be minimal, whence $x_2 < x_1 < x_0$ for some x_2. Iterating this argument m times, we obtain $x_m < x_{m-1} < \ldots < x_1 < x_0$. By transitivity and irreflexivity, all x_i are pairwise distinct, whence $|A| \geq m + 1$. A contradiction. □

Corollary 19.23. *For every $f, g \in M$, it holds that $N_f = N_g$ iff $f = g$. That is, a monotonic function is uniquely determined by its lower units. Moreover, let $N_f = \{\vec{\sigma}^1, \ldots, \vec{\sigma}^k\}$, $k > 0$, and let $\mu_{\vec{\sigma}}$ be the monomial encoded by $\vec{\sigma}$ (as* 101 *encodes x_1x_3, etc.). Then $f(\vec{x}) = \mu_{\vec{\sigma}^1} \vee \ldots \vee \mu_{\vec{\sigma}^k}$ for each $\vec{x}$. Of course, $f(\vec{x}) = 0$ if $N_f = \varnothing$.*

Proof. It is clear that $\mu_{\vec{\sigma}}(\vec{\tau}) = 1$ iff $\vec{\sigma} \leq \vec{\tau}$ (cf. the proof of Theorem 18.20). Denote the expression $\mu_{\vec{\sigma}^1} \vee \ldots \vee \mu_{\vec{\sigma}^k}$ by μ. Applying Lemma 19.22, we get

$$\begin{aligned} \mu(\vec{x}) = 1 &\iff \exists j\ \mu_{\vec{\sigma}^j}(\vec{x}) = 1 \\ &\iff \exists j\ \vec{\sigma}^j \leq \vec{x} \\ &\iff \exists \vec{\sigma} \in N_f\ \vec{\sigma} \leq \vec{x} \\ &\iff f(\vec{x}) = 1. \end{aligned}$$

□

Example 19.24. As $N_{\mathrm{maj}} = \{011, 101, 110\}$, we have $\mathrm{maj}(x, y, z) = xy \vee xz \vee yz$.

Example 19.25. Every expression of the form $\mu_{\vec{\sigma}}$ is a disjunction of monomials; hence, it is computable over $\{1^{(1)}, \wedge, \vee\}$. Allowing N_f to be empty as well, we may conclude that $M = [0^{(1)}, 1^{(1)}, \wedge, \vee]$ and, of course, $\bigwedge \subseteq [0^{(1)}, 1^{(1)}, \wedge, \vee]$.

It is possible strengthen this result for the set $\bigwedge$. First, we claim that a monotonic function f is conjunctive iff $|N_f| \leq 1$. Indeed, assume that $|N_f| > 1$. There are two distinct tuples $\vec{\sigma}, \vec{\tau}$ in N_f; they must be $\leq$-incomparable, whence $\sigma_i = 1$ and $\tau_i = 0$ for a certain i (otherwise, $\vec{\sigma} \leq \vec{\tau}$). Let $\vec{\rho} = \vec{\sigma} \wedge \vec{\tau}$. Clearly, $\vec{\rho} \leq \vec{\sigma}$. We have $\rho_i = 0$, whence $\vec{\rho} \neq \vec{\sigma}$. By Lemma 19.22, there should be $\vec{\xi} \in N_f$ with $\xi \leq \vec{\rho}$ if $f(\vec{\rho}) = 1$, that is, $\vec{\xi} \leq \vec{\rho} < \vec{\sigma}$ and $\vec{\sigma}$ is not a lower unit. This contradiction proves $f(\vec{\sigma} \wedge \vec{\tau}) = f(\vec{\rho}) = 0$, but $f(\vec{\sigma}) \wedge f(\vec{\tau}) = 1 \wedge 1 = 1$. Hence, $f \notin \bigwedge$.

For the other direction, suppose that $|N_f| \leq 1$. If $|N_f| = 0$, then $N_f = \varnothing$ and $f = 0^{(1)} \in \bigwedge$. Let $N_f = \{\vec{\rho}\}$. Consider arbitrary tuples $\vec{\sigma}$ and $\vec{\tau}$. Applying Lemma 19.22 and the fact that conjunction returns an $\leq$-infimum (see Remark 18.3), we get

$$\begin{aligned} f(\vec{\sigma}) \wedge f(\vec{\tau}) = 1 &\iff f(\vec{\sigma}) = f(\vec{\tau}) = 1 \\ &\iff \vec{\rho} \leq \vec{\sigma} \text{ and } \vec{\rho} \leq \vec{\tau} \\ &\iff \vec{\rho} \leq \vec{\sigma} \wedge \vec{\tau} \\ &\iff f(\vec{\sigma} \wedge \vec{\tau}) = 1. \end{aligned}$$

Thus, $f \in \bigwedge$.

Finally, given f is conjunctive, from $|N_f| \leq 1$ and Corollary 19.23, it follows that $f(\vec{x})$ equals either 0 or *one* monomial $\mu_{\vec{\sigma}}$ (when $N_f = \{\vec{\sigma}\}$). So, $f \in [0^{(1)}, 1^{(1)}, \wedge]$ and $\bigwedge = [0^{(1)}, 1^{(1)}, \wedge]$.

Example 19.26. Is the set $\{0^{(1)}, 1^{(1)}, \wedge, \vee\}$ a basis for M? In fact, it is since no lesser set suffices: we have $0^{(1)} \notin [1^{(1)}, \wedge, \vee]$ as $0^{(1)} \notin P_1$ but $[1^{(1)}, \wedge, \vee] \subseteq P_1$; $1^{(1)} \notin [0^{(1)}, \wedge, \vee]$ as $1^{(1)} \notin P_0$ but $[0^{(1)}, \wedge, \vee] \subseteq P_0$; $\vee \notin [0^{(1)}, 1^{(1)}, \wedge]$ as $\vee \notin \bigwedge$ but $[0^{(1)}, 1^{(1)}, \wedge] \subseteq \bigwedge$.

And what about $\wedge$? Is there any obstacle preventing it from being expressible via $\{0^{(1)}, 1^{(1)}, \vee\}$? As the Reader could expect, this obstacle should be the set $\bigvee$ of *disjunctive* functions $\{f \in \top \mid \forall \vec{\sigma} \forall \vec{\tau}\ f(\vec{\sigma} \vee \vec{\tau}) = f(\vec{\sigma}) \vee f(\vec{\tau})\}$. We leave finishing this argument to the Reader.

From all these considerations, it should be clear now that $\{0^{(1)}, 1^{(1)}, \wedge\}$ is a basis in $\bigwedge$. In particular, $\wedge \notin [0^{(1)}, 1^{(1)}]$ as the functions $0^{(1)}$ and $1^{(1)}$ are linear, whereas $\wedge$ is not.

Exercise 19.27. Prove that $\bigwedge^* = \bigvee$ and $M^* = M$.

Exercise 19.28. Prove that the set $\bigvee$ is closed and $\{0^{(1)}, 1^{(1)}, \vee\}$ is one of its bases.

Exercise 19.29. For any natural $k \leq n$, there are at least $2^{C_n^k}$ monotonic functions of n arguments. In order to prove this, think of how large the set N_f may be.

Exercise 19.30. Prove the identity $f(y, \vec{x}) = (y \wedge f(1, \vec{x})) \vee f(0, \vec{x})$ for each $f^{(n)} \in M$. Then use it to establish $M \subseteq [0^{(1)}, 1^{(1)}, \wedge, \vee]$ by induction on n, similarly to Remark 18.18.

Post's Criterion. Now, we are ready to provide a nice characterization for all such sets Q that $[Q] = \top$.

Theorem 19.31 (Post's Criterion of Functional Completeness)**.** *For every $Q \subseteq \top$, it holds that $[Q] = \top$ iff Q is* not included *to any of the sets P_0, P_1, M, S, L.*

Proof. Assume that $Q \subseteq R$ for some $R \in \{P_0, P_1, M, D, L\}$. Then $[Q] \subseteq [R] = R$, whereas $R \subsetneq \top$ for each R as Example 19.20 shows. Hence, $[Q] \neq \top$.

For the other direction, assume that $Q \not\subseteq R$ for any $R \in \{P_0, P_1, M, D, L\}$. It follows that Q contains functions $f_0 \notin P_0$, $f_1 \notin P_1$, $f_M \notin M$, $f_S \notin S$, and $f_L \notin L$. (Some of these may coincide.) It suffices to prove that $\{\neg, \wedge\} \subseteq [Q]$ for $[\neg, \wedge] = \top$.

Our first goal is to establish $\{\neg, 0^{(1)}, 1^{(1)}\} \subseteq [Q]$. Consider the function g_0 such that $g_0(x) = f_0(x, \ldots, x)$. We clearly have $g_0 \in [Q]$. Furthermore, $g_0(0) = f_0(\vec{0}) = 1$ as $f_0 \notin P_0$. If $g_0(1) = 1$, then $g_0 = 1^{(0)}$. Otherwise, $g_0(1) = 0$ and $g_0 = \neg$. A similar argument for the function $g_1 \in [Q]$ with $g_1(x) = f_1(x, \ldots, x)$ yields $g_1 = 0^{(1)}$ or $g_1 = \neg$.

Two cases are possible now. We either have $\{1^{(1)}, 0^{(1)}\} = \{g_0, g_1\} \subseteq [Q]$ or $\neg \in \{g_0, g_1\} \subseteq [Q]$.

Suppose that $\{1^{(1)}, 0^{(1)}\} \subseteq [Q]$. Consider the function $f_M^{(m)} \in Q \setminus M$. There exist tuples $\vec{\sigma}, \vec{\tau}$ such that $\vec{\sigma} \leq \vec{\tau} \in \underline{2}^m$ but $f_M(\vec{\sigma}) > f_M(\vec{\tau})$ (that is, $f_M(\vec{\sigma}) = 1$ and $f_M(\vec{\tau}) = 0$). As $\vec{\sigma} \neq \vec{\tau}$, obtain $\vec{\sigma} < \vec{\tau}$, that is, there exist indices $i_1 < i_2 < \ldots < i_k$, $k > 0$, such that $\sigma_j < \tau_j$ (whence $\sigma_j = 0$ and $\tau_j = 1$) when $j = i_s$ for some $s \leq k$ and $\sigma_j = \tau_j$ otherwise. Now, let

$$g_M(x) = f_M(\sigma_1, \ldots, \sigma_{i_1 - 1}, x, \sigma_{i_1+1}, \ldots, \sigma_{i_2 - 1}, x, \sigma_{i_2+1}, \ldots, \sigma_{i_k - 1}, x, \sigma_{i_2+1}, \ldots, \sigma_m).$$

For example, if $\vec{\sigma} = (1, 0, 0, 1, 1, 0, 0)$ and $\vec{\tau} = (1, 0, 1, 1, 1, 1, 1)$, we shall get $g_M(x) = f_M(1, 0, x, 1, 1, x, x)$. Clearly, $g_M \in [f_M, 0^{(1)}, 1^{(1)}] \subseteq [Q]$. On the other hand, $g_M(0) = f_M(\vec{\sigma}) = 1$ and $g_M(1) = f_M(\vec{\tau}) = 0$. Thus, $g_M = \neg$ and $\{\neg, 1^{(1)}, 0^{(1)}\} \subseteq [Q]$.

Now, suppose that $\neg \in [Q]$. Consider the function $f_S^{(s)} \in Q \setminus S$. We have $f(\vec{\sigma}) = f(\neg\vec{\sigma})$ for a certain $\vec{\sigma} \in \underline{2}^s$. Let

$$g_S(x) = f_S(x^{\sigma_1}, \ldots, x^{\sigma_s}).$$

Recall that x^1 stands for x and x^0 for $\neg x$. E. g., one has $g_S(x) = f_S(x, x, \neg x, x, \neg x)$ if $\sigma = (1, 1, 0, 1, 0)$. It is obvious that $g_S \in [f_S, \neg] \subseteq [Q]$. Moreover, $g_S(1) = f_S(\vec{\sigma}) = f_S(\neg\vec{\sigma}) = g_S(0)$. Thus, $g_S \in \{0^{(1)}, 1^{(1)}\}$. Given one constant, we may obtain the other one by applying negation. Therefore, $\{\neg, 1^{(1)}, 0^{(1)}\} \subseteq [Q]$ in this case as well.

Now, we shall prove $\wedge \in [f_L, \neg, 1^{(1)}, 0^{(1)}] \subseteq [Q]$. Since the function $f_L^{(l)}$ is not linear, its Zhegalkin coefficient $a_{\vec{\sigma}} = 1$ for at least one tuple $\vec{\sigma}$ with $||\vec{\sigma}|| \geq 2$. W. l. o. g., let $\sigma_1 = 1 = \sigma_2$ and $\vec{\sigma} = 11\vec{\rho}$ for some $\vec{\rho}$. By grouping the monomials which contain x_1x_2 in the equation (10) together, we then obtain

$$\begin{aligned} f_L(\vec{x}) = x_1x_2(&a_{110\ldots0}\,1 + a_{1110\ldots0}\,x_3 + a_{11010\ldots0}\,x_4 + \ldots + a_{111\ldots1}\,x_3x_4\ldots x_l) + \\ &a_{00\ldots0}\,1 + a_{10\ldots0}\,x_1 + a_{010\ldots0}\,x_2 + \ldots + a_{0\ldots01}\,x_l + a_{101\ldots0}\,x_1x_3 + \ldots + a_{001\ldots1}\,x_3\ldots x_l = \\ &= x_1x_2 \cdot P(x_3, \ldots, x_l) + Q(\vec{x}). \end{aligned}$$

The value of the polynomial P is *not always* zero, for otherwise one could change every $a_{11\vec{\tau}}$ to 0 and obtain a *different* polynomial for f_L (since $a_{11\vec{\rho}} = 1$), contrary to Theorem 18.20. Also notice that the polynomial Q has no monomial containing both x_1 and x_2. So, there exists a tuple $\vec{\tau}$ such that $P(\vec{\tau}) = 1$; by grouping together the terms with x_1, x_2, and none of those variables, respectively, in Q we obtain

$$f_L(x_1, x_2, \vec{\tau}) = x_1x_2 + Q(x_1, x_2, \vec{\tau}) = x_1x_2 + ax_1 + bx_2 + c$$

for some $a, b, c \in \underline{2}$. Let $g_L(x_1, x_2) = f_L(x_1, x_2, \vec{\tau})$. We clearly have $g_L \in [f_L, 1^{(1)}, 0^{(1)}] \subseteq [Q]$. Our last goal is to obtain conjunction from g_L. Let

$$\begin{aligned} g(x_1, x_2) = g_L(&x_1 + b, x_2 + a) + ab + c = \\ &(x_1 + b) \cdot (x_2 + a) + a(x_1 + b) + b(x_2 + a) + c + ab + c = \\ &x_1x_2 + ax_1 + bx_2 + ab + ax_1 + ab + bx_2 + ab + ab = x_1x_2 = x_1 \wedge x_2. \end{aligned}$$

So, $g = \wedge$. On the other hand, computing g requires adding a constant to a variable; this boils down to applying negation for $x + 0 = x$ and $x + 1 = \neg x$. For example, we have $g(x_1, x_2) = \neg g_L(x_1, \neg x_2)$ when $(a, b, c) = (1, 0, 1)$. Therefore, $\wedge = g \in [g_L, \neg] \subseteq [Q]$. Finally, $\{\neg, \wedge\} \subseteq [Q]$, as it was required. □

Example 19.32. Let $Q = \{1^{(1)}, \neg, +, \mathrm{even}^{(3)}, \mathrm{maj}\}$, where the function even takes 1 iff the number of '1's among its arguments is *even*. In particular, $\mathrm{even}(0, 0, 0) = \mathrm{even}(1, 0, 1) = 1$. It is easy to see that $\mathrm{even}(x_1, x_2, x_3) = x_1 + x_2 + x_3 + 1$. Let us check whether Q is complete in $\top$ and express $\vee$ over Q if it is possible.

Let us make a table similar to that of Example 19.20. By Theorem 19.31, the set Q is complete in $\top$ iff there is a '–' in each column. So we may spare our effort and leave some cells blank.

	P_0	P_1	M	S	L
$1^{(1)}$	−	+	+	−	+
$\neg$		−	−		+
$+$					+
even					+
maj					−

It appears that Q is complete in $\top$. Hence, $\vee \in [Q]$. How can we make this explicit? As $\vee$ is non-linear, we need $\mathrm{maj} \in Q \smallsetminus L$ in order to compute it. We have $\mathrm{maj}(x_1, x_2, x_3) = x_1x_2 + x_1x_3 + x_2x_3$ and $x_1 \vee x_2 = x_1x_2 + x_1 + x_2$, whence $x_1 \vee x_2 = \mathrm{maj}(x_1, x_2, 1(x_3))$.

Exercise 19.33. It is clear that neither $+$ nor even are necessary for the set Q to be complete in $\top$. Fill all the cells in and find all bases of $\top$ among the subsets of Q.

Exercise 19.34. No basis of $\top$ has more than four elements. (Suppose a function does not preserve a constant. May it be both monotonic and self-dual?)

The sets P_0, P_1, M, S, L are indeed unique with respect to completeness in $\top$. We say that a set Q is *precomplete* (in $\top$) iff $[Q] \neq \top$ but $[Q \cup \{f\}] = \top$ for every $f \notin Q$.

Exercise 19.35. None of the sets P_0, P_1, M, S, L is included into another one. Each of these sets is precomplete in $\top$.

Exercise 19.36. Every precomplete set is closed. Every closed set except $\top$ itself is a subset of a precomplete set. The sets P_0, P_1, M, S, L are only precomplete in $\top$.

Post's Criterion is sometimes instrumental in finding bases of various closed sets. The main idea is typically as follows. Assume we want to prove $Q = [R]$. We may then add more functions to R to entail $[R'] = \top$ where $R \subsetneq R'$. Finally, we look at a circuit over R' for $f \in Q$ (or a related function) and try to eliminate every element of $R' \setminus R$ therefrom—in such a way that the circuit will still compute f.

Example 19.37. Let us prove that $R = \{\neg, \mathrm{maj}\}$ is a basis of S. It is clear that $[R] \subseteq S$. Moreover, $\neg \in L$ but $\mathrm{maj} \notin L$, whence $\mathrm{maj} \notin [\neg]$; $\mathrm{maj} \in P_0$ but $\neg \notin P_0$, whence $\neg \notin [\mathrm{maj}]$. It remains to prove that $S \subseteq [R]$.

From Theorem 19.31, it follows that the set $R' = \{\neg, \mathrm{maj}, 0^{(1)}\}$ is complete in $\top$. Consider an arbitrary $f^{(n+1)} \in S$. If $n = 0$, we have either $f = \neg$ or $f = \mathrm{id}_{\underline{2}} \in \bot$, whence $f \in [R]$. Assume $n > 0$. Let $g(\vec{x}) = f(0, \vec{x})$ for each $\vec{x} \in \underline{2}^n$ and C be a circuit of $(x_1, \dots, x_n)$ over R' computing $g^{(n)}$.

We change C in the following way: replace every assignment of the form $t_i = 0(t_j)$ with $t_i = y$, where y is a new input variable. This procedure results in a circuit D of $(y, x_1, \dots, x_n)$ over R. The latter computes a certain function $h^{(n+1)} = g_D \in [R]$, so $h \in S$.

It is easy to see that $h(0, \vec{x}) = g(\vec{x}) = f(0, \vec{x})$ for each $\vec{x}$ since the circuits D and C compute identical functions at inputs $(0, x_1, \dots, x_n)$ and $(x_1, \dots, x_n)$, respectively. (For a formal argument, one has to define D by recursion on $\mathrm{size}(C)$ and prove the claim by induction on $\mathrm{size}(C)$.)

On the other hand, $h(1, \vec{x}) = \neg h(\neg 1, \neg \vec{x}) = \neg h(0, \neg \vec{x}) = \neg f(0, \neg \vec{x}) = \neg f(\neg 1, \neg \vec{x}) = f(1, \vec{x})$, by the previous equation and the fact that $f, h \in S$. Hence, $f(y, \vec{x}) = h(y, \vec{x})$ for all $y, \vec{x}$ and $f = h \in [R]$.

Exercise 19.38. Find a basis of size 1 for the set S.

Exercise 19.39. Consider the following *conditional operator* ?: (also known as *conditioned disjunction*) such that

$$x \mathbin{?} y : z \quad = \quad \begin{cases} y \text{ if } x = 1; \\ z \text{ if } x = 0 \end{cases}$$

for all $x, y, z \in \underline{2}$. Prove that $[\,?\!:\,] = P_0 \cap P_1$. (Try adding constants to this operator.)

Let $T_1^\infty = \{f \in \top \mid \exists i \, \forall \vec{\sigma} \; f(\vec{\sigma}) \geq \sigma_i\}$, that is, the set T_1^∞ consists of functions bounded below by an argument. For example, one has $\vee, \to \in T_1^\infty$ as $x_1 \vee x_2 \geq x_i$ for each $i = \{1, 2\}$ and $x_1 \to x_2 \geq x_2$. On the contrary, $\wedge \notin T_1^\infty$ since $0 \wedge x = x \wedge 0 = 0$ for any x. It is easy to see that $T_1^\infty \subseteq P_1$.

Exercise 19.40. Prove that the set T_1^∞ is closed.

We can use this closed set as an obstacle to refute expressibility: say, $\wedge \notin [\vee, \to]$.

Exercise 19.41. Nevertheless, $\vee \in [\wedge, \to]$ and, in fact, $\vee \in [\to]$.

Exercise 19.42. Prove that $T_1^\infty = [\to]$. (You might want to establish the identity $f(y, \vec{x}) = y \vee f(0, \vec{x})$ for the case $f(y, \vec{x}) \geq y$ first and notice that $[\neg, \to] = \top$ then.)