Alexander Knop

Introduction to Discrete Mathematics

APRIL 13, 2020

Contents

List of Symbols

The letters A, B, X, Y, and Z denote sets, the letters x, y, and z denote the elements of X, Y, and Z respectively, P and Q denote propositions and predicates, the lower case latin letters f and g denote functions from X to Y and from Y to Z respectively, the letters a, b, c, n,and k denote integer numbers, and the greek letter α and β denote real numbers.

Counting

$(m)_n$	denotes the number of ways to choose a subset of n elements from a fixed set of m elements, page 108
$\binom{m}{n}$	denotes the number of ways to choose an unordered subset of n elements from a fixed set of m elements, page 109
$n!$	denotes $n \cdot (n-1) \cdot (n-2) \cdot \ldots \cdot 1$, page 16
$\prod_{i=1}^{k} \alpha_i$	denotes $\alpha_1 \cdot \ldots \cdot \alpha_k$, page 16
$\sum_{i=1}^{k} \alpha_i$	denotes $\alpha_1 + \cdots + \alpha_k$, page 15
$\sum_{i \in S \,:\, P(i)} \alpha_i$	denotes $\alpha_{i_1} + \cdots + \alpha_{i_k}$, where $\{i \in S \;:\; P(i)\} = \{i_1, \ldots, i_k\}$, page 81
$B(n)$	denotes the nth Bell number; i.e. the number of partitions of $[n]$ into nonempty blocks, page 117
$I(h)$	denotes the number of inversions in h, page 138
$p(n)$	denotes the number of all the partitions of n, page 119
$p_k(n)$	denotes the number of all the partitions of n into k blocks, page 119
$S(n,k)$	denotes the Stirling number of the second kind; i.e. the number of partitions of $[n]$ into k nonempty blocks, page 115

Functions

I_A	denotes the identity function on the set A, page 40
$\operatorname{Im} f$	denotes the image of f, page 41
$\tau_{i,j}$	denotes the transposition of i and j, page 139
$f \circ g$	denotes the composition of functions f and g; i.e, it denotes the function $h(x) = f(g(x))$., page 40
$f\|_A$	denotes the restriction of f to the set A, page 39
f^{-1}	denotes the inverse of the function f (it's defined only when f is a bijection), page 94
$f^{-1}(y)$	depend on the context if f is not a bijection it denotes the set $\{x \in X \ : \ f(x) = y\}$ and it denotes the value of f^{-1} at y if f is a bijection, page 94

Graphs

$G + e$	denotes the graph $(V, E \cup \{e\})$, page 189
$G - e$	denotes the graph $(V, E \setminus \{e\})$, page 182
$G - v$	denotes the graph $(V \setminus \{v\}, E \cap (V \setminus \{v\})^2)$, page 182
$G[F]$	denotes the induced subgraph of G on the edges F (i.e. (V, F)), page 182
$G[U]$	denotes the induced subgraph of G on the vertices U (i.e. $(U, \{e \in E \ : \ e \in U^2\})$), page 182
K_n	denotes the complete graph on n vertices, page 182

Logical Notation

$\exists x \in X \ P(x)$	denotes the statement saying that P is true for some element of X, page 35
$\forall x \in X \ P(x)$	denotes the statement saying that P is true for all elements of X, page 35
$\neg P$	denotes the statement saying that P is false, page 25
$P \implies Q$	denotes the statement saying that if P is true, then Q is true as well, page 3
$P \wedge Q$	denotes the statement saying that P and Q are both true, page 25
$P \vee Q$	denotes the statement saying that at least one of P and Q is true, page 24

Combinatorial Games

$\oplus$	denotes the Nim sum, page 65

Relations

$a \mid b$	says that a divides b, page 46
$a \equiv b \pmod{n}$	says that n divides $a - b$, page 44
$A \subseteq B$	says that A is a subset of B, page 28

Set Notation

$(B)_A$	denotes the set of injections from A to B, page 108
2^A	denotes the set of all the subsets of the set A, page 31
$[n]$	denotes the set of all the integers from 1 to n, page 29
$\bigcap_{i=1}^{k} A_i$	denotes $A_1 \cap \cdots \cap A_k$, page 32
$\bigcap_{i \in S \,:\, P(i)} A_i$	denotes $A_{i_1} \cap \cdots \cap A_{i_k}$, where $\{i \in S \;:\; P(i)\} = \{i_1, \ldots, i_k\}$, page 140
$\bigcup_{i=1}^{k} A_i$	denotes $A_1 \cup \cdots \cup A_k$, page 32
$\bigcup_{i \in S \,:\, P(i)} A_i$	denotes $A_{i_1} \cup \cdots \cup A_{i_k}$, where $\{i \in S \;:\; P(i)\} = \{i_1, \ldots, i_k\}$, page 140
$\binom{A}{k}$	denotes the set of subsets of A of cardinality k, page 109
$\mathbb{C}$	denotes the set of all complex numbers, page 27
$\emptyset$	denotes the set that does not have elements, page 28
$\mathbb{N}$	denotes the set of all integers greater than 0, page 27
$\mathbb{N}_0$	denotes the set of nonnegative integers, page 65
$\mathbb{Q}$	denotes the set of all rational numbers, page 27
$\mathbb{R}$	denotes the set of all real numbers, page 27
$\mathbb{R}[[x]]$	denotes the set all the power series in the variable x, page 225
$\mathbb{Z}$	denotes the set of all integers, page 27
$A \cap B$	denotes the intersection of two sets A and B, page 29
$A \cup B$	denotes the union of two sets A and B, page 29
$A \setminus B$	denotes the difference of two sets A and B, page 29

$A \times B$	denotes the set of all ordered pairs of elements of A and B, page 37
B^A	denotes the set of functions from A to B, page 108
S_n	denotes the set of all permutations of $[n]$, page 121

Numbers

$(c_0, \ldots, c_\ell)_b$	denotes the number $n = \sum_{i=0}^{\ell} b^i c_i$; i.e., $c_0, \ldots, c_\ell$ are the digits of n in the base-b digital representation, page 20
$\lceil \alpha \rceil$	denotes the smallest integer greater than or equal to α, page 102

Preface

- Why is a math book so sad?
- Because it's full of problems.

Anonymous, Unknown

If you are reading this book, you probably have never studied proofs before. So let me give you some advice: mathematical books are very different from fiction, and even books in other sciences. Quite often you may see that some steps are missing, and some steps are not really explained and just claimed as obvious. The main reason behind this is to make the ideas of the proof more visible and to allow grasping the essence of proofs quickly.

Since the steps are skipped, you cannot just read the book and believe that you studied the topic; the best way to actually study the topic is to try to prove every statement before you read the actual proof in the book. In addition to this, I recommend trying to solve all the exercises in the book (you may find exercises in the middle and at the end of every chapter).

Additionally, many topics in this book have a corresponding five-minute video explaining the material of the chapter, it is useful to watch them before you go into the topic.

Organization

Part I covers the basics of mathematics and provide the language we use in the next parts. We start from the explanation of what a mathematical proof is (in Chapter 1). Chapter 2 shows how to prove theorems indirectly using proof by contradiction. **??** explains the most powerful method in our disposal, proof by induction. Finally, Chapters 5 to 8 define several important objects such as sets, functions, and relations.

Part IV studies the basics of combinatorics, a branch of mathematics that answers the question "how many objects of this kind?". Chapter 17 gives a formal definition of "size" of a set and show how to compare sizes of two sets. Chapter 18 proves several simple principles that allow to find sizes of sets. In Chapter 19 we learn how to prove existence of an object with some properties using simple inequalities between sizes of sets. Chapters 20 to 22 prove several properties of standard combinatorial objects. Finally, Chapter 23 provides a framework that helps to find sizes of sets in many cases.

Part VII returns back to proofs; however, instead of studying *how* to prove something we study what can we prove and how to define "proof" so that we can use computer to generate proofs and verify them.

In Part VIII we study basics of graph theory. Chapter 30 gives the definition of a graph and prove the one of the simplest and at the same time most important theorems in graph theory. In Chapter 31 we define what it means beeing connected and how to use this notion in real-life applications. Finally, Chapter 32 defines a tree and show how to use these objects in computer networks.

Alexander Knop
San Diego, California, USA

Part I

Introduction to Mathematical Reasoning

1. Proofs

What is a Mathematical Proof:
Introduction to Mathematical Reasoning #1

https://youtu.be/eJD0gGqveIE

1.1 Direct Proofs

We start the discussion of the proofs in mathematics from an example of a proof in "everyday" life. Assume that we know that the following statements are true.

1. If a salmon has fins and scales it is kosher,
2. if a salmon has scales it has fins,
3. any salmon has scales.

Using these facts we may conclude that any salmon is kosher; indeed, any salmon has scales by the third statement, hence, by the second statement any salmon has fins, finally, by the first statement any salmon is kosher since it has fins and scales.

One may notice that this explanation is a sequence of conclusions such that each of them is true because the previous one is true. Mathematical proof is also a sequence of statements such that every statement is true if the previous statement is true. If P and Q are some statements and Q is always true when P is true, then we say that P implies Q. We denote the statement that P implies Q by $P \implies Q$.

In order to define the implication formally let us consider the following table.

P	Q	$P \implies Q$
T	T	T
T	F	F
F	T	T
F	F	T

Let P and Q be some statements. Then this table says that if P and Q are both false, then $P \implies Q$ is true etc.

Exercise 1.1. *Let n be an integer.*

1. *Is it always true that "n^2 is positive" implies "n is not equal to 0"?*

2. *Is it always true that "$n^2 - n - 2$ is equal to 0" implies "n is equal to 2"?*

In the example we gave at the beginning of the section we used some *known* facts. But what does it mean to know something? In math we typically say that we know a statement if we can prove it. But in order to prove this statement we need to know something again, which is a problem! In order to solve it, mathematicians introduced the notion of an *axiom*. An axiom is a statement that is believed to be true and when we prove a statement we prove it under the assumption that these axioms are true[1].

[1] Note that in different parts of math axioms may be different.

For example, we may consider axioms of inequalities for real numbers.

1. Let $a, b \in \mathbb{R}$. Only one of the following is true:
   - $a < b$,
   - $b < a$, or
   - $a = b$.
2. Let $a, b, c \in \mathbb{R}$. Then $a < b$ iff $a + c < b + c$ (iff is an abbreviation for "if and only if").
3. Let $a, b, c \in \mathbb{R}$. Then $a < b$ iff $ac < bc$ provided that $c > 0$ and $a < b$ iff $ac > bc$ if $c < 0$.
4. Let $a, b, c \in \mathbb{R}$. If $a < b$ and $b < c$, then $a < c$.

What We Know and How to Find a Proof: Introduction to Mathematical Reasoning #2

https://youtu.be/nBjJi6aTk2M

Let us now try to prove something using these axioms, we prove that if $a > 0$, then $a^2 > 0$. Note that $a > 0$, hence, by the third axiom $a^2 > 0$ (note that we also used an additional statement saying that $0 \cdot 0 = 0$).

Similarly, we may prove that if $a < 0$, then $a^2 > 0$. And combining these two statements together we may prove that if $a \neq 0$, then $a^2 > 0$.

Such a way of constructing proof is called direct proofs.

Exercise 1.2. *Axiomatic system for a four-point geometry.*

Undefined terms: point, line, is on.

Axioms:

- *For every pair of distinct points x and y, there is a unique line ℓ such that x is on l and y is on l.*
- *Given a line ℓ and a point x that is not on ℓ, there is a unique line m such that x is on m and no point on ℓ is also on m.*
- *There are exactly four points.*
- *It is impossible for three points to be on the same line.*

Prove that there are at least two distinct lines.

Let n and m be some integers. Using direct proofs we may prove the following two statements.

- if n is even, then nm is also even (a number ℓ is even if there is an integer k such that $\ell = 2k$),
- if n is even and m is even, then $n + m$ is also even.

We start from proving the first statement. There is an integer k such that $n = 2k$ since n is even. As a result, $nm = 2(nk)$ so nm is even.

Now we prove the second statement. Since n and m are even there are k and ℓ such that $n = 2k$ and $m = 2\ell$. Hence, $n + m = 2(k + \ell)$ so $n + m$ is even.

1.2 Constructing Proofs Backwards

However, sometimes it is not easy to find the proof. In this case one of the possible methods to deal with this problem is to try to prove starting from the end.

For example, we may consider the statement $(a + b)^2 = a^2 + 2ba + b^2$. Imagine, for a second, that you have not learned about axioms. In this case you would write something like this:

$$\begin{aligned}(a+b)^2 = (a+b)\cdot(a+b) = \\ a(a+b) + b(a+b) = \\ a^2 + ab + ba + b^2 = a^2 + 2ba + b^2.\end{aligned}$$

Let us try to prove it completely formally using the following axioms.

1. Let a, b, and c be reals. If $a = b$ and $b = c$, then $a = c$.
2. Let a, b, and c be reals. If $a = b$, then $a + c = b + c$ and $c + a = c + b$.
3. Let a, b, and c be reals. Then $a(b + c) = ab + ac$.
4. Let a and b be reals. Then $ab = ba$.
5. Let a and b be reals. Then $a + b = b + a$.
6. Let a be a real number. Then $a^2 = a \cdot a$ and $a \cdot a = a^2$.
7. Let a be a real number. Then $a + a = 2a$.

So the formal proof of the statement $(a + b)^2 = a^2 + 2ab + b^2$ is as follows. First note that $(a + b)^2 = (a + b) \cdot (a + b)$ (by axiom 6), hence, by axiom 1, it is enough to show that $(a + b) \cdot (a + b) = a^2 + 2ab + b^2$. By axiom 3, $(a + b) \cdot (a + b) = (a + b) \cdot a + (a + b) \cdot b$. Axiom 4 implies that $(a+b)\cdot a = a\cdot(a+b)$ and $(a+b)\cdot b = b\cdot(a+b)$ Hence, by axioms 1 and 2 applied twice

$$a\cdot(a+b)+b\cdot(a+b) = (a+b)\cdot a + b\cdot(a+b) = (a+b)\cdot a + (a+b)\cdot b.$$

As a result,

$$(a+b)\cdot(a+b) = (a+b)\cdot a + (a+b)\cdot b = \\ a\cdot(a+b)+b\cdot(a+b) = a\cdot a + a\cdot b + b\cdot a + b\cdot b;$$

so by axiom 1, it is enough to show that $a\cdot a + a\cdot b + b\cdot a + b\cdot b = a^2 + 2ab + b^2$. Additionally, by axiom 6, $a\cdot a = a^2$ and $b\cdot b = b^2$. Hence, by axiom 2, it is enough to show that $a^2 + a\cdot b + b\cdot a + b^2 = a^2 + 2ab + b^2$. By axiom 4, $a\cdot b = b\cdot a$, hence, by axiom 2, $a\cdot b + b\cdot a = b\cdot a + b\cdot a$. Therefore by axiom 7, $a\cdot b + b\cdot a = 2b\cdot a$. Finally, by axiom 2, $a\cdot b + b\cdot a + a^2 + b^2 = 2b\cdot a + a^2 + b^2$ and by axiom 5, $a\cdot b + b\cdot a + a^2 + b^2 = a^2 + a\cdot b + b\cdot a + b^2$ and $2b\cdot a + a^2 + b^2 = a^2 + 2b\cdot a + b^2$. Which finishes the proof by axiom 1.

1.3 *Analysis of Simple Algorithms*

We can use this knowledge to analyze simple algorithms. For example, let us consider the following algorithm. Let us prove that it is correct

Algorithm 1.1: The algorithm that finds the maximum element of a, b, c.

```
1:  function Max(a, b, c)
2:      r ← a
3:      if b > r then
4:          r ← b
5:      end if
6:      if c > r then
7:          r ← c
8:      end if
9:      return r
10: end function
```

i.e. it returns the maximum of a, b, and c. We need to consider the following cases.

- If the maximum is equal to a. In this case, at line 2, we set $r = a$, at line 3 the inequality $b > r$ is false (since $a = r$ is the maximum) and at line 6 the inequality $c > r$ is also false (since $a = r$ is the maximum). Hence, we do not change the value of r after line 2 and the returned value is a.
- If the maximum is equal to b. We set $r = a$ at line 2. The inequality $b > r$ at line 3 is true (since b is the maximum) and we set r to be

equal to b. So at line 6, the inequality $c > r$ is false (since $b = r$ is the maximum). Hence, the returned value is b.

- If the maximum is equal to c. We set $r = a$ at line 2. If the inequality $b > r$ is true at line 3 we set r to be equal to b. So at line 6 the inequality $c > r$ is true (since c is the maximum). Hence, we set r being equal to c and the returned value is c.

1.4 *Proofs in Real-life Mathematics*

In this chapter we explicitly used axioms to prove statements. However, it leads us to really long and hard to understand proofs (the last example in the previous section is a good example of this phenomenon). Because of this mathematicians tend to skip steps in the proofs when they believe that they are clear. It is worth to mention a nice quotation of Scott Aaronson about this problem

> When mathematicians say that a theorem has been "proved," they still mean, as they always have, something more like: "we've reached a social consensus that all the ideas are now in place for a strictly formal proof that could be verified by a machine . . . with the only task remaining being massive rote coding work that none of us has any intention of ever doing!"

This is the reason why it is arduous to read mathematical texts and it is very different from reading non-mathematical books. A problem that arises because of this tendency is that some mistakes may happen if we skip way too many steps. In the last two centuries there were several attempts to solve this issue, one approach to this we are going to discuss in Part VII.

Death of proof greatly exaggerated

`https://scottaaronson.com/blog/?p=4133`

End of The Chapter Exercises

1.3 Using the axioms of inequalities show that if a is a non-zero real number, then $a^2 > 0$.

1.4 Using the axioms of inequalities prove that for all real numbers a, b, and c,

$$bc + ac + ab \leq a^2 + b^2 + c^2.$$

1.5 *(recommended)* Prove that for all integers a, b, and c, If a divides b and b divides c, then a divides c. Recall that an integer m divides an integer n if there is an integer k such that $mk = n$.

1.6 *(recommended)* Show that square of an even integer is even.

1.7 Prove that 0 divides an integer a iff $a = 0$.

1.8 Using the axioms of inequalities, show that if $a > 0$, b, and c are real numbers, then $b \geq c$ implies that $ab \geq ac$.

1.9 Using the axioms of inequalities, show that if $a, b < 0$ are real numbers, then $a \leq b$ implies that $a^2 \geq b^2$.

2. *Proofs by Contradiction*

Proofs by Contradiction:
Introduction to Mathematical Reasoning #3

https://youtu.be/bWP0VYx75DI

2.1 *Proving Negative Statements*

The direct method is not very convenient when we need to prove a negation of some statement.

For example, we may try to prove that $78n + 102m = 11$ does not have integer solutions. It is not clear how to prove it directly since we can not consider all possible n and m. Hence, we need another approach. Let us assume that such a solution n, m exists. Note that $78n + 102m$ is even, but 11 is odd. In other words, an odd number is equal to an even number, which is impossible. Thus, the assumption was false.

Let us consider a more useful example, let us prove that if p^2 is even, then p is also even (p is an integer). Assume the opposite i.e. that p^2 is even but p is not. Let $p = 2b + 1$[1]. Note that $p^2 = (2b+1)^2 = 2(2b^2 + 2b) + 1$. Hence, p^2 is odd which contradicts to the assumption that p^2 is even.

[1] Note that we use here the statement that an integer n is not even iff it is odd, which, formally speaking, should be proven.

Using this idea we may prove much more complicated results e.g. one may show that $\sqrt{2}$ is irrational. For the sake of contradiction, let us assume that it is not true. In other words there are p and q such that $\sqrt{2} = \frac{p}{q}$ and $\frac{p}{q}$ is an irreducible fraction.

Note that $\sqrt{2}q = p$, so $2q^2 = p^2$. Which implies that p is even and 4 divides p^2. Therefore 4 divides $2q^2$ and q is also even. As a result, we get a contradiction with the assumption that $\frac{p}{q}$ is an irreducible fraction.

Template for proving a statement by contradiction.

Assume, for the sake of contradiction, that *the statement* is false. Then *present some argument that leads to a contradiction*. Hence, the assumption is false and *the statement* is true.

Exercise 2.1. *Show that $\sqrt{3}$ is irrational.*

2.2 Proving Implications by Contradiction

This method works especially well when we need to prove an implication. Since the implication $A \implies B$ is false only when A is true but B is false. Hence, you need to derive a contradiction from the fact that A is true and B is false.

We have already seen such examples in the previous section, we proved that p^2 is even implies p is even for any integer p. Let us consider another example. Let a and b be reals such that $a > b$. We need to show that $(ac < bc) \implies c < 0$. So we may assume that $ac < bc$ but $c \geq 0$. By the multiplicativity of the inequalities we know that if $(a > b)$ and $c > 0$, then $ac > bc$ which contradicts to $ac < bc$.

A special case of such a proof is when we need to prove the implication $A \implies B$, assume that B is false and derive that A is false which contradicts to A (such proofs are called proofs by contraposition); note that the previous proof is a proof of this form.

2.3 Proof of "OR" Statements

Another important case is when we need to prove that at least one of two statements is true. For example, let us prove that $ab = 0$ iff $a = 0$ or $b = 0$. We start from the implication from the right to the left. Since if $a = 0$, then $ab = 0$ and the same is true for $b = 0$ this implication is obvious.

The second part of the proof is the proof by contradiction. Assume $ab = 0$, $a \neq 0$, and $b \neq 0$. Note that $b = \frac{ab}{a} = 0$, hence $b = 0$ which is a contradiction to the assumption.

End of The Chapter Exercises

2.2 *(recommended)* Prove that if n^2 is odd, then n is odd.

2.3 In Euclidean (standard) geometry, prove: If two lines share a common perpendicular, then the lines are parallel.

2.4 *(recommended)* Let us consider four-lines geometry, it is a theory with undefined terms: point, line, is on, and axioms:

1. there exist exactly four lines,
2. any two distinct lines have exactly one point on both of them, and
3. each point is on exactly two lines.

Show that every line has exactly three points on it.

2.5 Let us consider group theory, it is a theory with undefined terms: group-element and times (if a and b are group elements, we denote a times b by $a \cdot b$), and axioms:

1. $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for every group-elements a, b, and c;
2. there is a unique group-element e such that $e \cdot a = a = a \cdot e$ for every group-element a (we say that such an element is the identity element);
3. for every group-element a there is a group-element b such that $a \cdot b = e$, where e is the identity element;
4. for every group-element a there is a group-element b such that $b \cdot a = e$, where e is the identity element.

Let e be the identity element. Show the following statements

- if $b_0 \cdot a = b_1 \cdot a = e$, then $b_0 = b_1$, for every group-elements a, b_0, and b_1.
- if $a \cdot b_0 = a \cdot b_1 = e$, then $b_0 = b_1$, for every group-elements a, b_0, and b_1.
- if $a \cdot b_0 = b_1 \cdot a = e$, then $b_0 = b_1$, for every group-elements a, b_0, and b_1.

2.6 Let us consider three-points geometry, it is a theory with undefined terms: point, line, is on, and axioms:

1. There exist exactly three points.
2. Two distinct points are on exactly one line.
3. Not all the three points are collinear i.e. they do not lay on the same line.
4. Two distinct lines are on at least one point i.e. there is at least one point such that it is on both lines.

Show that there are exactly three lines.

2.7 Show that there are irrational numbers a and b such that a^b is rational.

2.8 *(recommended)* Show that there does not exist the largest integer.

2.9 Let us consider Young's geometry, it is a theory with undefined terms: point, line, is on, and axioms:

1. there exists at least one line,
2. every line has exactly three points on it,
3. not all points are on the same line,

4. for two distinct points, there exists exactly one line on both of them,
5. if a point does not lie on a given line, then there exists exactly one line on that point that does not intersect the given line.

Show that for every point, there are exactly four lines on that point.

Solutions to The Exercises

2.6 Let us denote the points by p_1, p_2, and p_3 (they exist by Axiom 1). By Axiom 2, there are lines $l_{1,2}$, $l_{1,3}$, and $l_{2,3}$ such that p_i and p_j are on $l_{i,j}$ ($i \neq j$).

Note that the lines $l_{1,2}$, $l_{1,3}$, and $l_{2,3}$ are different. Indeed, assume the opposite, i.e., without loss of generality that $l_{1,2} = l_{1,3}$. Note that p_1, p_2, and p_3 are on $l_{1,2}$ which contradicts Axiom 3.

Let us now prove that there are no other lines. Assume the opposite i.e. that there is another line l. There is a point that is on l and $l_{1,2}$. Without loss of generality, this point is p_1. Additionally there is a point p_i ($i \neq 1$) that is on l and $l_{2,3}$. However, it means that p_1 and p_i are on l which contradicts Axiom 2.

3. Simple Induction

The Induction Principle:
Introduction to Mathematical Reasoning #4

`https://youtu.be/jOnZTWGpX_I`

3.1 Proofs by Induction

Let us consider a simple problem: what is bigger 2^n or n? In this chapter, we are going to study the simplest way to prove that $2^n > n$ for all positive integers n. First, let us check that it is true for small integers n.

	1	2	3	4	5	6	7	8
n	1	2	3	4	5	6	7	8
2^n	2	4	8	16	32	64	128	256

We may also note that 2^n is growing faster than n, so we expect that if $2^n > n$ for small integers n, then it is true for all positive integers n.

In order to prove this statement formally, we use the following principle.

Principle 3.1 (The Induction Principle)**.** *Let $P(n)$ be some statement about a positive integer n. Hence, $P(n)$ is true for every positive integer n iff*

(the base case) $P(1)$ *is true and*

(the induction step) $P(k) \implies P(k+1)$ *is true for all positive integers k.*

Let us prove now the statement using this principle. We define $P(n)$ be the statement that "$2^n > n$". $P(1)$ is true since $2^1 > 1$. Let us assume now that $2^n > n$. Note that $2^{n+1} = 2 \cdot 2^n > 2n \geq n + 1$. Hence, we proved the induction step.

Exercise 3.1. *Prove that $(1+x)^n \geq 1 + nx$ for all positive integers n and real numbers $x \geq -1$.*

3.2 Changing the Base Case

Let us consider functions n^2 and 2^n.

	1	2	3	4	5	6	7	8
n^2	1	4	9	16	25	36	49	64
2^n	2	4	8	16	32	64	128	256

Note that 2^n is greater than n^2 starting from 5. But without some trick we can not prove this using induction since for $n = 3$ it is not true!

The trick is to use the statement $P(n)$ stating that $(n+4)^2 < 2^{n+4}$. The base case when $n = 1$ is true. Let us now prove the induction step. Assume that $P(k)$ is true i.e. $(k+4)^2 < 2^{k+4}$. Note that $2(k+4)^2 < 2^{k+1+4}$ but $(k+5)^2 = k^2 + 10k + 25 \leq 2k^2 + 16k + 32 = 2(k+4)^2$. Which implies that $2^{k+1+4} > (k+5)^2$. So $P(k+1)$ is also true.

In order to avoid this strange $+4$ we may change the base case and use the following argument.

Theorem 3.1. *Let $P(n)$ be some statement about an integer n. Hence, $P(n)$ is true for every integer $n > n_0$ iff*

(the base case) $P(n_0 + 1)$ *is true and*

(the induction step) $P(k) \implies P(k+1)$ *is true for all integers $k > n_0$.*

Using this generalized induction principle we may prove that $2^n \geq n^2$ for $n \geq 4$. The base case for $n = 4$ is true. The induction step is also true; indeed let $P(k)$ be true i.e. $(k+4)^2 < 2^{k+4}$. Hence, $2(k+4)^2 < 2^{k+1+4}$ but $(k+5)^2 = k^2 + 10k + 25 \leq 2k^2 + 16k + 32 = 2(k+4)^2$.

Let us now prove the theorem. Note that the proof is based on an idea similar to the trick with $+4$, we just used.

Proof of Theorem 3.1. $\Rightarrow$ If $P(n)$ is true for any $n > n_0$ it is also true for $n = n_0 + 1$ which implies the base case. Additionally, it true for $n = k + 1$ so the induction step is also true.

$\Leftarrow$ In this direction the proof is a bit harder. Let us consider a statement $Q(n)$ saying that $P(n + n_0)$ is true. Note that by the base case for P, $Q(1)$ is true; by the induction step for P we know that $Q(n)$ implies $P(n+1)$. As a result, by the induction principle $Q(n)$ is true for all positive integers n. Which implies that $P(n)$ is true for all integers $n > n_0$.

$\square$

3.3 *Inductive Definitions*

We may also define objects inductively. Let us consider the sum $1 + 2 + \cdots + n$ a line of dots indicating "and so on" which indicates the definition by induction. In this case, a more precise notation is $\sum_{i=1}^{n} i$.

Definition 3.1. *Let $a(1), \ldots, a(n), \ldots$ be a sequence of integers. Then $\sum_{i=1}^{n} a(i)$ is defined inductively by the following statements:*

- $\sum_{i=1}^{1} a(i) = a(1)$, *and*
- $\sum_{i=1}^{k+1} a(i) = \sum_{i=1}^{k} a(i) + a(k+1)$.

Let us prove that $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$. Note that by definition $\sum_{i=1}^{1} i = 1$ and $\frac{1(1+1)}{2} = 1$; hence, the base case holds. Assume that $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$. Note that $\sum_{i=1}^{n+1} i = \sum_{i=1}^{n} i + (n+1)$ and by the induction hypothesis $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$. Hence, $\sum_{i=1}^{n+1} i = \frac{n(n+1)}{2} + (n+1) = \frac{(n+1)(n+2)}{2}$.

Exercise 3.2. *Prove that* $\sum_{i=1}^{n} 2^i = 2^{n+1} - 2$.

3.4 *Analysis of Algorithms with Cycles*

Induction is very useful for analysing algorithms using cycles. Let us extend the example we considered in Section 1.3.

Let us consider the following algorithm. We prove that it is working

Algorithm 3.1: The algorithm that finds the maximum element of $a_1, \ldots, a_n$.

```
function Max(a_1, ..., a_n)
    r ← a_1
    for i from 2 to n do
        if a_i > r then
            r ← a_i
        end if
    end for
    return r
end function
```

correctly. First, we need to define $r_1, \ldots, r_n$ the value of r during the execution of the algorithm. It is easy to see that $r_1 = a_1$ and

$$r_{i+1} = \begin{cases} r_i & \text{if } r_i > a_{i+1} \\ a_{i+1} & \text{otherwise} \end{cases}.$$

Secondly, we prove by induction that r_i is the maximum of $a_1, \ldots, a_i$. It is clear that the base case for $i = 1$ is true. Let us prove the induction step from k to $k+1$. By the induction hypothesis, r_k is the maximum of $a_1, \ldots, a_k$. We may consider two following cases.

- If $r_k > a_{k+1}$, then $r_{k+1} = r_k$ is the maximum of $a_1, \ldots, a_{k+1}$ since r_k is the maximum of $a_1, \ldots, a_k$.
- Otherwise, a_{k+1} is greater than or equal to $a_1, \ldots, a_k$, hence, $r_{k+1} = a_{k+1}$.

Exercise 3.3. *Show that line 6 in the following sorting algorithm executes* $\frac{n(n+1)}{2}$ *times.*

End of The Chapter Exercises

3.4 Show that there does not exist the largest integer.

3.5 *(recommended)* Show that for any positive integer n, $n^2 + n$ is even.

3.6 Show that for any positive integer n, 3 divides $n^3 + 2n$.

3.7 Show that for any integer $n \geq 10$, $n^3 \leq 2^n$.

3.8 Show that for any positive integer n, $\sum_{i=0}^{n} x^i = \frac{1-x^{n+1}}{1-x}$.

3.9 *(recommended)* Show that $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$ for all integers $n \geq 1$.

3.10 Show that $\sum_{i=1}^{n} \frac{1}{i(i+1)} = \frac{n}{n+1}$ for all integers $n \geq 1$.

3.11 Show that $\sum_{i=1}^{n} \frac{1}{i^2} \leq 2 - \frac{1}{n}$ for all integers $n \geq 1$.

3.12 Show that $\sum_{i=1}^{n}(2i-1) = n^2$ for any positive integer n.

3.13 Prove that $\sum_{i=1}^{n} \frac{1}{i(i+1)} = \frac{n}{n+1}$ for any positive integer n.

3.14 Prove that $\sum_{i=1}^{n}(i+1)2^i = n2^{n+1}$ for all integers $n > 2$.

3.15 Let $a_1, \ldots, a_n$ be a sequence of real numbers. We define inductively $\prod_{i=k}^{n} a_i$ as follows:

- $\prod_{i=1}^{1} a_i = a_1$ and
- $\prod_{i=1}^{k+1} a_i = \left(\prod_{i=1}^{k} a_i\right) \cdot a_{k+1}$.

Prove that $\prod_{i=1}^{n-1}\left(1 - \frac{1}{(i+1)^2}\right) = \frac{n+1}{2n}$ for all integers $n > 1$.

3.16 Let us define $n!$ as follows: $1! = 1$ and $n! = (n-1)! \cdot n$. Show that $n! \geq 2^n$ for any $n \geq 4$.

3.17 *(open)* Find all the natural numbers n such that $n! = m^2$ for some integer m.

3.18 Show that $\int\limits_0^{+\infty} x^n e^{-x}\,\mathrm{d}x = n!$ for all $n \geq 0$.

3.19 Show that $\sum_{k=1}^{n} k \cdot k! = (n+1)! - 1$.

3.20 Show that Algorithm 3.2 executes line 6 exactly $\frac{n(n+1)}{2}$ times.

3.21 Show that Algorithm 3.2 sorts the array.

function SELECTIONSORT($a_1, \dots, a_n$)
    for i from 1 to n **do**
        $r \leftarrow a_i$
        $\ell \leftarrow i$
        for j from i to n **do**
            if $a_j > r$ **then**
                $r \leftarrow a_j$
                $\ell \leftarrow j$
            end if
        end for
        Swap a_i and a_ℓ.
    end for
end function

Algorithm 3.2: The algorithm is selection sort, it sorts $a_1, \dots, a_n$.

4. Strong Induction

Sometimes $P(k)$ is not enough to prove $P(k+1)$ and we need all the statements $P(1), \ldots, P(k)$. In this case we may use the following induction principle.

Theorem 4.1 (The Strong Induction Principle)**.** *Let $P(n)$ be some statement about positive integer n. Hence, $P(n)$ is true for every integer $n > n_0$ iff*

(the base case) $P(n_0 + 1)$ *is true and*

(the induction step) *If $P(n_0 + 1), \ldots, P(n_0 + k)$ are true, then $P(n_0 + k + 1)$ is also true for all positive integers k.*

Before we prove this theorem let us present some applications of this principle.

The Fibonacci numbers are defined as follows: $f_0 = 0$, $f_1 = 1$, and $f_k = f_{k-1} + f_{k-2}$ for $k \geq 2$ (note that they are also defined using strong induction since we use not only f_{k-1} to define f_k).[1]

[1] Fibonacci numbers are named after Italian mathematician Leonardo of Pisa, later known as Fibonacci. In his 1202 book "Liber Abaci", Fibonacci introduced the sequence to Western European mathematics. However, the sequence had been described as early as 200 BC in work by Indian mathematician Pingala on enumerating possible patterns of Sanskrit poetry formed from syllables of two lengths.

Fibonacci numbers appear unexpectedly often in mathematics, so much so that there is an entire journal dedicated to their study, the "Fibonacci Quarterly". Applications of Fibonacci numbers include computer algorithms such as the Fibonacci search technique and the Fibonacci heap data structure, and graphs called Fibonacci cubes used for interconnecting parallel and distributed systems.

Theorem 4.2 (The Binet formula)**.** *The Fibonacci numbers are given by the following formula*

$$f_n = \frac{\alpha^n - \beta^n}{\sqrt{5}},$$

where $\alpha = \frac{1+\sqrt{5}}{2}$ and $\beta = \frac{1-\sqrt{5}}{2}$.

Proof. We use the strong induction principle to prove this statement with $n_0 = -1$. Let us first prove the base case, $\frac{(\alpha^0 - \beta^0)}{\sqrt{5}} = 0 = f_0$. We also need to prove the induction step.

- If $k = 1$, then $\frac{(\alpha^1 - \beta^1)}{\sqrt{5}} = 1 = f_1$.

- Otherwise, by the induction hypothesis, $f_k = \frac{\alpha^k - \beta^k}{\sqrt{5}}$ and $f_{k-1} = \frac{\alpha^{k-1} - \beta^{k-1}}{\sqrt{5}}$. By the definition of the Fibonacci numbers $f_{k+1} = f_k + f_{k-1}$. Hence,

$$f_{k+1} = \frac{\alpha^k - \beta^k}{\sqrt{5}} + \frac{\alpha^{k-1} - \beta^{k-1}}{\sqrt{5}}.$$

Note that it is enough to show that

$$\frac{\alpha^{k+1}-\beta^{k+1}}{\sqrt{5}} = \frac{\alpha^{k}-\beta^{k}}{\sqrt{5}} + \frac{\alpha^{k-1}-\beta^{k-1}}{\sqrt{5}}. \tag{4.1}$$

Note that it is the same as

$$\frac{\alpha^{k+1}-\alpha^{k}-\alpha^{k-1}}{\sqrt{5}} = \frac{\beta^{k+1}-\beta^{k}-\beta^{k-1}}{\sqrt{5}}.$$

Additionally, note that α and β are roots of the equation $x^2-x-1=0$. Hence, $\alpha^{k+1}-\alpha^k-\alpha^{k-1}=\alpha^{k-1}(\alpha^2-\alpha-1)=0$ and $\beta^{k+1}-\beta^k-\beta^{k-1}=\beta^{k-1}(\beta^2-\beta-1)=0$. Which implies equality (4.1).

□

Another example of an application of the strong induction is the proof that any number can be written in digital numeral systems with any base.

Theorem 4.3. *Let $b>1$ be an integer. Then there is a unique representation $(c_0,\ldots,c_\ell)_b$ of any positive number n in the base-b digital numeral system. In other words, for any positive integer n, there are unique $0 \le c_0,\ldots,c_\ell < b$ such that $n=\sum_{i=0}^{\ell} b^i c_i$.*

Proof. We prove the statement using strong induction by n. The base case for $n<b$ is clear (we can choose $\ell=0$ and $c_0=n$). Let us now prove the induction step. Assume the statement is true for all $k<n$. Let n divided by b be equal to q with the remainder c_0. Note that $(n-c_0)/b<n$ is a positive integer. Hence, by the induction hypothesis, there are $0\le c_1,\ldots,c_\ell<b$ such that $(n-c_0)/b=\sum_{i=1}^{\ell} b^{i-1}c_i$. Hence, $n=\sum_{i=0}^{\ell} b^i c_i$. □

Now we are ready to prove the strong induction principle.

Proof of Theorem 4.1. It is easy to see that if $P(n)$ is true for all $n>n_0$, then the base case and the induction steps are true. Let us prove that if the base case and the induction step are true, then $P(n)$ is true for all $n>n_0$.

Let $Q(k)$ be the statement that $P(n_0+1)$, …, $P(n_0+k)$ are true. Note that $Q(1)$ is true by the base case for P. Additionally, note that if $Q(k)$ is true, then $Q(k+1)$ is also true, by the induction step for P. Hence, by the induction principle, $Q(k)$ is true for all positive integers k. Which implies that $P(n_0+k)$ is true for all positive integers k. □

4.1 *Analysis of Recursive Algorithms*

To illustrate the power of recursive definitions and strong induction, consider following game: Alice have chosen a number from 1 to 1000.

Bob wants to guess the number so he is asking Alice "yes" or "no" questions. How many questions does Bob need to ask to determine the number in the worst-case scenario?

The following simple algorithm allows Bob to learn the number using 10 questions.

1. Bob start with two numbers $\ell = 0$ and $u = 1000$.
2. If there is only one integer x such that $\ell < x \leq u$, Bob says that Alice's number is x and terminates the algorithm.
3. Bob asks whether the Alice's number is at most $(\ell + u)/2$. If the answer is yes, then Bob replaces u by $(\ell + u)/2$; otherwise Bob replaces ℓ by $(\ell + u)/2$. Bob goes to step 3.

We need to prove that Bob's algorithm is correct and that it makes at most 10 questions. We prove a bit stronger statement. If on step 3 $u - \ell < n$ and Alice's number is between ℓ and u (u and ℓ are some reals), then the algorithm returns Alice's number using at most $\log_2 n$ questions ($\log_2 1000 \leq 10$). We are going to use strong induction by n.

The base case is clear since if $u - \ell < 1$ there is at most one integer between u and ℓ and by the assumption there is Alice's number between ℓ and u. Hence, Bob is going to guess Alice's number correctly without asking any questions.

Assume that the statement is true for all $m < n$. Note that $\ell < (\ell + u)/2 < u$ since $n \geq 2$. Therefore Alice's number is either between ℓ and $(\ell + u)/2$ or between $(\ell + u)/2$ and u. Hence, we go to step 3 with new ℓ' and u' such that $u' - \ell' < n/2$ and Alice's number is between ℓ' and u'. As a result, by the induction hypothesis, Bob is going to guess Alice's number correctly using at most $1 + \log_2(n/2) = \log_2 n$ questions.

End of The Chapter Exercises

4.1 Let $a_0 = 2$, $a_1 = 5$, and $a_n = 5a_{n-1} - 6a_{n-2}$ for all integers $n \geq 2$. Show that $a_n = 3^n + 2^n$ for all integers $n \geq 0$.

4.2 Let $f_0 = 1$, $f_1 = 1$, and $f_{n+2} = f_{n+1} + f_n$ for all integers $n \geq 0$. Show that $f_n \geq \left(\frac{3}{2}\right)^{n-2}$.

4.3 Show that $f_{n+m} = f_{n-1}f_{m-1} + f_n f_m$.

4.4 Give a nonadaptive algorithm for Bob that allows him to guess the number using 10 queries. In other words, write 10 questions such that answers to these questions allow Bob to guess the number.

4.5 Show that Algorithm 4.1 makes at most $6 + 2\log_2(n)$ comparisons.

Algorithm 4.1: The binary search algorithm that finds an element e in the sorted list $a_1, \dots, a_n$.

```
function BinarySearch(e, a_1, ..., a_n)
    if n ≤ 5 then
        for i from 1 to n do
            if a_i = e then
                return i
            end if
        end for
    else
        ℓ ← ⌊n/2⌋
        if a_ℓ ≤ e then
            BinarySearch(e, a_1, ..., a_ℓ)
        else
            BinarySearch(e, a_{ℓ+1}, ..., a_n)
        end if
    end if
end function
```

Solutions to The Exercises

4.1 We prove this using induction by n. The base case for $n \le 1$ is clear since $3^0 + 2^0 = 2$ and $3^1 + 2^1 = 5$.

Let us prove the induction step. Assume that $a_n = 3^n + 2^n$ and $a_{n-1} = 3^{n-1} + 2^{n-1}$, we need to prove that $a_{n+1} = 3^{n+1} + 2^{n+1}$. Note that

$$a_{n+1} = 5a_n - 6a_{n-1} = 5 \cdot 3^n + 5 \cdot 2^n - 6 \cdot 3^{n-1} - 6 \cdot 2^{n-1} = \\ 3^{n-1} \cdot 9 + 2^{n-1} 4 = 3^{n+1} + 2^{n+1}.$$

5. Predicates and Connectives

Connectives and Propositions:
Introduction to Mathematical Reasoning #5

https://youtu.be/0unvlq2OTaE

5.1 Propositions and Predicates

In the previous chapters we used the word "statement" without any even relatively formal definition of what it means. In this chapter we are going to give a semi-formal definition and discuss how to create complicated statements from simple statements.

It is difficult to give a formal definition of what a mathematical statement is, hence, we are not going to do it in this book. The goal of this section is to enable the reader to recognize mathematical statements.

A *proposition* or a mathematical statement is a declarative sentence which is either true or false but not both. Consider the following list of sentences.

1. $2 \times 2 = 4$
2. $\pi = 4$
3. n is even
4. 32 is special
5. The square of any odd number is odd.
6. The sum of any even number and one is prime.

Of those, the first two are propositions; note that this says nothing about whether they are true or not. Actually, the first is true and the second is false. However, the third sentence becomes a proposition only when the value of n is fixed. The fourth is not a proposition. Finally, the last two are propositions (the fifth is true and the sixth is false).

The third statement is somewhat special, because there is a simple way to make it a proposition: one just needs to fix the value of the variables. Such sentences are called predicates and the variables that need to be specified are called free variables of these predicates.

Note that the fourth sentence is also interesting, since if we define what it means to be special, the phrase became a proposition. Mathematicians tend to do such things to give mathematical meanings to everyday words.

5.2 *Connectives*

Mathematicians often need to decide whether a given proposition is true or false. Many statements are complicated and constructed from simpler statements using *logical connectives*. For example we may consider the following statements:

1. $3 > 4$ and $1 < 1$;
2. $1 \times 2 = 5$ or $6 > 1$.

Logical connective "OR". The second statement is an example of usage of this connective. The statement "P or Q" is true if and only if at least one of P and Q is true. We may define the connective using the truth table of it.

P	Q	P or Q
T	T	T
T	F	T
F	T	T
F	F	F

The or connective is also called *disjunction* and the disjunction of P and Q is often dented as $P \vee Q$.

Warning: Note that in everyday speech "or" is often used in the exclusive case, like in the sentence "we need to decide whether it is an insect or a spider". In this case the precise meaning of "or" is made clear by the context. However, mathematical language should be formal, hence, we always use "or" inclusively.

Logical connective "AND". The first statement is an example of this connective. The statement "P and Q" is true if and only if both P and Q are true. We may define the connective using the truth table of it.

P	Q	P and Q
T	T	T
T	F	F
F	T	F
F	F	F

The and connective is also called *conjunction* and the conjunction of P and Q is often dented as $P \wedge Q$.

> **Warning:** Not all the properties of "and" from everyday speech are captured by logical conjunction. For example, "and" sometimes implies order. For example, "They got married and had a child" in common language means that the marriage came before the child. The word "and" can also imply a partition of a thing into parts, as "The American flag is red, white, and blue." Here it is not meant that the flag is at once red, white, and blue, but rather that it has a part of each color.

Logical connective "NOT". The last connective is called *negation* and examples of usage of it are the following:

1. 5 is not greater than 8;
2. Does not exist an integer n such that $n^2 = 2$.

Note that it is not straightforward where to put the negation in these sentences.

The negation of a statement P is denoted as $\neg P$ (sometimes it is also denoted as $\sim P$).

End of The Chapter Exercises

5.1 Construct truth tables for the statements

- not (P and Q);
- (not P) or (not Q);
- P and (not Q);
- (not P) or Q;

5.2 *(recommended)* Consider the statement "All gnomes like cookies". Which of the following statements is the negation of the above statement?

- All gnomes hate cookies.
- All gnomes do not like cookies.
- Some gnomes do not like cookies.
- Some gnomes hate cookies.
- All creatures who like cookies are gnomes.
- All creatures who do not like cookies are not gnomes.

5.3 Using truth tables show that the following statements are equivalent:

- $P \implies Q$,
- $(P \vee Q) \iff Q$ ($A \iff B$ is the same as $(A \implies B) \wedge (B \implies A)$),
- $(P \wedge Q) \iff P$

5.4 Prove that three connectives "or", "and", and "not" can all be written in terms of the single connective "notand" where "P notand Q" is interpreted as "not (P and Q)" (this operation is also known as Sheffer stroke or NAND).

5.5 Show the same statement about the connective "notor" where "P notor Q" is interpreted as "not (P or Q)" (this operation is also known as Peirce's arrow or NOR).

6. Sets

Sets:
Introduction to Mathematical Reasoning #6

https://youtu.be/bshBV2H4Sqo

6.1 The Intuitive Definition of a Set

A set is one of the two most important concepts in mathematics. Many mathematical statements involve "an integer n" or "a real number a". Set theory notation provides a simple way to express that a is a real number. However, this language is much more expressible and it is impossible to imagine modern mathematics without this notation.

As in the previous chapter it is difficult to define a set formally so we give a less formal definition which should be enough to use the notation. A *set* is a well-defined collection of objects. Important examples of sets are:

1. $\mathbb{R}$ a set of reals,
2. $\mathbb{Z}$ the set of integers[1],
3. $\mathbb{N}$ the set of natural numbers[2],
4. $\mathbb{Q}$ a set of rational numbers,
5. $\mathbb{C}$ a set of complex numbers.

[1] "Z" stands for the German word Zahlen ("numbers").

[2] Note that in the literature there are two different traditions: in one 0 is a natural number, in another it is not; in this book we are going to assume that 0 is not a natural number.

Usually, sets are denoted by a single letter.

Objects in a set are called *elements* of the set and we denote the statement "x is in the set E" by the formula $x \in E$ and the negation of this statement by $x \notin E$. For example, we proved that $\sqrt{2} \notin \mathbb{Q}$[3].

[3] The symbol $\in$ was first used by Giuseppe Peano 1889 in his work "Arithmetices principia, nova methodo exposita". Here he wrote on page X: "The symbol $\in$ means is. So $a \in b$ is read as a *is* a b; ..." The symbol itself is a stylized lowercase Greek letter epsilon ("ϵ"), the first letter of the word εστι, which means "is".

Exercise 6.1. *Which of the following sets are included in which? Recall that a number is prime iff it is an integer greater than* 1 *and divisible only by* 1 *and itself.*

1. *The set of all positive integers less than* 10.
2. *The set of all prime numbers less than* 11.
3. *The set of all odd numbers greater than* 1 *and less than* 6.
4. *The set of all positive integers less than* 10.
5. *The set whose only elements are* 1 *and* 2.

6. *The set whose only element is* 1.

7. *The set of all prime numbers less than* 11.

6.2 Basic Relations Between Sets

Many problems in mathematics are problems of determining whether two descriptions of sets are describing the same set or not. For example, when we learn how to solve quadratic equations of the form $ax^2 + bx + c = 0$ ($a, b, c \in \mathbb{R}$) we learn how to list the elements of the set $\{x \in \mathbb{R} : ax^2 + bx + c = 0\}$.

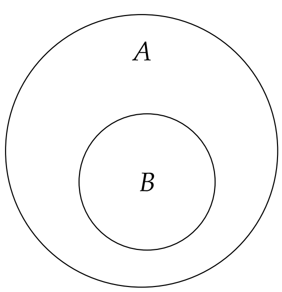
We say that two sets A and B are equal if they contain the same elements (we denote it by $A = B$). If all the elements of A belong to B we say that A is a subset of B and denote it by $A \subseteq B$[4].

For example, $\mathbb{Q} \subseteq \mathbb{R}$ since any rational number is also a real number. A special set is an empty set i.e. the set that does not have elements, we denote it $\emptyset$.

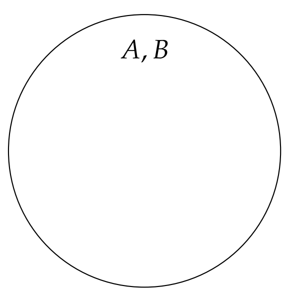
[4] In the literature there are three symbols for "subset": $\subseteq$, $\subsetneq$, and $\subset$. $A \subseteq B$ means that A is a subset of B and we allow $A = B$ and $A \subsetneq B$ means that A is a subset of B and we forbid $A = B$. However, there is a problem with the third symbol, some people use it as a synonym of $\subseteq$ and some use it as a synonym of $\subsetneq$. Due to this ambiguity we are going to avoid using it in this book.

Diagrams

If we think of a set A as represented by all the points within a circle or any other closed figure, then it is easy to represent the notion of A being a subset of another set B also represented by all the points within a circle. We just put a circle labeled by A inside of the circle labeled by B. We can also diagram an equality by drawing a circle labeled by both A and B. (see fig. 6.1). Such diagrams are called Euler diagrams and it is clear that one may draw Euler diagrams for more than two sets.



(a) $B \subseteq A$



(b) $A = B$

Figure 6.1: Euler diagrams for subset and equality relations

Descriptions of Sets

In this section we describe how to define new sets, this notation is also known as *set-builder notation*.

Listing elements. The simplest way to define a set is just to list the elements. For example

1. $\{1, 2, \pi\}$ is the set consisting of three elements 1, 2, and π, and

2. $\{1, 2, 3, \ldots\}$ is the set of all positive integers i.e. it is the set $\mathbb{N}$.

Conditional definitions. We may also describe a set using some constraint e.g we may list all the even numbers using the following formula $\{n \in \mathbb{Z} : n \text{ is even}\}$ (we read it as "the set of all integers n such that n is even").

Using this we may also define the set of all integers from 1 to m, we denote it $[m]$; i.e. $[m] = \{n \in \mathbb{N} : 0 < n \leq m\}$.

Constructive definitions. Another way to construct a set of all even numbers is to use the constructive definition of a set: $\{2k : k \in \mathbb{Z}\}$.

We may also describe a set of rational numbers using this description: $\mathbb{Q} = \{a/b : a \in \mathbb{Z}, b \in \mathbb{N}\}$ (note that we may also use a mix of a conditional and constructive definitions, $\mathbb{Q} = \{a/b : a, b \in \mathbb{Z}, b \neq 0\}$).

Exercise 6.2. *Describe a set of perfect squares using constructive type of definition.*

Disjoint Sets

Two sets are *disjoint* iff they do not have common elements. We also say that two sets are *overlapping* iff they are not disjoint i.e. they share at least one element.

More generally, $A_1, \ldots, A_\ell$ are pairwise disjoint iff A_i is disjoint with A_j for all $i \neq j \in [\ell]$

Exercise 6.3. *Of the sets in Exercise 6.1, which are disjoint from which?*

6.3 *Operations over Sets.*

Another way to describe a set is to apply operation to other sets. Let A and B be sets.

The first example of the operations on sets is the *union* operation. The union of A and B is the set containing all the elements of A and all the elements of B i.e. $A \cup B = \{x : x \in A \text{ or } x \in B\}$[5].

Another example of such an operation is *intersection*. The intersection of A and B is the set of all the elements belonging to both A and B i.e $A \cap B = \{x : x \in A \text{ and } x \in B\}$[6].

The third operation we are going to discuss this lecture is *set difference*. If A and B are some sets, then $A \setminus B = \{x : x \in A \text{ and } x \notin B\}$.

[5] Note that this definition is not correct since in the conditional definitions we have to specify the set x belongs to and we cannot do this here.

[6] You may notice that in the definition of the union we use disjunction and in the definition of intersection we use conjunction. Actually this is the reason the symbol of the conjunction is similar to the symbol of intersection and the symbol of the disjunction is similar to the symbol of union.

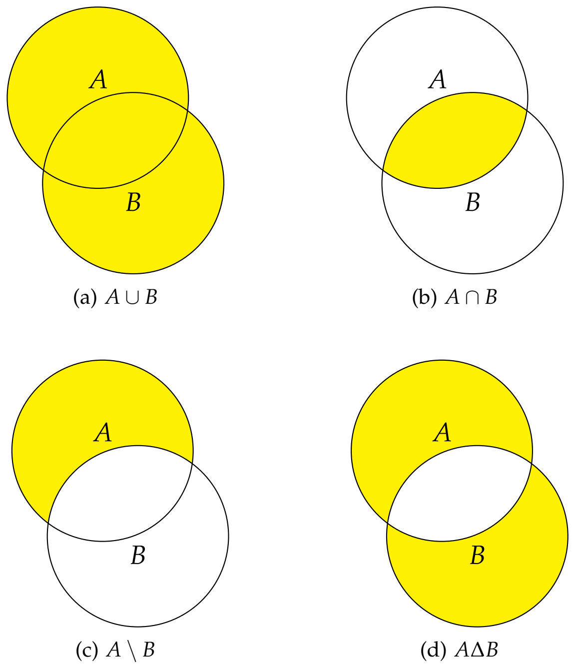


Figure 6.2: Euler diagrams for set operations

The last operation is *symmetric difference*. If A and B are some sets, then $A\Delta B = (A \setminus B) \cup (B \setminus A)$. Note that alternatively $A\Delta B = (A \cup B) \setminus (A \cap B)$

Exercise 6.4. *Describe the set* $\{n \in \mathbb{N} \ : \ n \text{ is even}\} \cap \{3n \ : \ n \in \mathbb{N}\}$.

Theorem 6.1. *Let A, B, and C be some sets. Then we have the following identities.*

(associativity) $A \cup (B \cup C) = (A \cup B) \cup C$ *and* $A \cap (B \cap C) = (A \cap B) \cap C$.

(commutativity) $A \cup B = B \cup A$ *and* $A \cap B = B \cap A$.

(distributivity) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ *and* $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$.

Proof. One may prove these properties using the Euler diagrams. Alternatively they can be proven by definitions. Let us prove only the first part of the distributivity, the rest is Exercise 6.5.

Our proof consists of two parts in the first part we prove that $A \cup (B \cap C) \subseteq (A \cup B) \cap (A \cup C)$. Suppose that $x \in A \cup (B \cap C)$. Then $x \in A$ or $x \in (B \cap C)$.

- If $x \in A$, then $x \in (A \cup B)$ and $x \in (A \cup C)$ i.e. $x \in ((A \cup B) \cap (A \cup C))$.
- If $x \in (B \cap C)$, then $x \in B$ and $x \in C$. Which implies that $x \in (A \cup B)$ and $x \in (A \cup C)$. As a result, $x \in ((A \cup B) \cap (A \cup C))$.

$\square$

Exercise 6.5. *Prove the rest of the equalities in Theorem 6.1.*

Probably the most difficult concept connected to sets is the concept of a power set. Let A be some set, then the set of all possible subsets of A is denoted by 2^A (sometimes this set is denoted by $\mathcal{P}(A)$) and called the power set of A. In other words $2^A = \{B \,:\, B \subseteq A\}$.

> **Warning:** Please do not forget about two extremal elements of the power set 2^A: the empty set and A itself.

For example if $A = \{1, 2, 3\}$, then

$$2^A = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}\}.$$

6.4 The Well-ordering Principle

Using the set notation we may finally justify the proof of the statement that $2^n > n$ for all positive integers n from the video about mathematical induction. In order to do this let us first formulate the following theorem.

Theorem 6.2. *Let $A \subseteq \mathbb{Z}$ be a non-empty set. We say that $b \in \mathbb{Z}$ is a lower bound for the set A iff $b \leq a$ for all $a \in A$. Additionally, we say that the set A is bounded if there is a lower bound for A.*

Given this, if A is bounded, then there is a lower bound $a \in A$ for the set A (we say that a is the minimum of the set A).

Note that this theorem also states that any subset of natural numbers have a minimum.

Recall that we wish to prove that $2^n > n$ for all positive n. Assume that it is not true, in this case the set $A = \{n \in \mathbb{N} \,:\, 2^n < n\}$ is non-empty. Denote by n_0 the minimum of the set A, n_0 exists by Theorem 6.2. We may consider the following two cases.

- If $n_0 = 1$, then it leads to a contradiction since $2 = 2^1 > 1$.
- Otherwise, note that $1 \leq n_0 - 1 < n_0$, hence, $2^{n_0 - 1} > n_0 - 1$. So $2^{n_0} > 2n_0 - 2 \geq n_0$. Which is a contradiction with the definition of n_0.

Finally, we prove Theorem 6.2.

Proof of Theorem 6.2. Let b be a lower bound for the set A. Assume that there is no minimum of the set A. Let $P(n)$ be the statement that $n \notin A$.

First, we are going to prove that $P(n)$ is true for all $n \geq b$. The base case is true since if $b \in A$, then b is the minimum of A which contradicts to the assumption that there is no minimum of A. The induction step is also clear, by the induction hypothesis we know that $P(b)$, $\ldots$, $P(k)$ are true, hence, $(k+1) \in A$ implies that $k+1$ is the minimum of A.

Now we prove that A is empty. Assume the opposite i.e. assume that there is $x \in A$. Note that $x \geq b$ since b is a lower bound of A. However, $P(x)$ is true which implies that $x \notin A$. Therefore the assumption was false and A is empty, but this contradicts to the fact that A is non- empty. $\square$

End of The Chapter Exercises

6.6 Find the power sets of $\emptyset$, $\{1\}$, $\{1,2\}$, $\{1,2,3,4\}$. How many elements in each of this sets?

6.7 *(recommended)* Prove that

- $A \subseteq B \iff A \cup B = B$,
- $A \subseteq B \iff A \cap B = A$.

6.8 Let A be a subset of a set U we call this set a universe. We say that the set $\overline{A} = U \setminus A$ is a complement of A in U. Show the following equalities

- $\overline{\overline{A}} = A$.
- $\overline{A \cup B} = \overline{A} \cap \overline{B}$.
- $\overline{A \cap B} = \overline{A} \cup \overline{B}$.

6.9 *(recommended)* Let us define an intersection of more than two sets as follows. Let $A_1, \ldots, A_n$ be some sets. Then

- $\bigcap_{i=1}^{1} A_i = A_1$ and
- $\bigcap_{i=1}^{k+1} A_i = \left(\bigcap_{i=1}^{k} A_i\right) \cap A_{k+1}$.

Show that $\bigcap_{i=1}^{n} \{x \in \mathbb{N} \,:\, i \leq x \leq n\} = \{n\}$ for all integers $n > 0$.

6.10 Let us define a union of more than two sets as follows. Let A_1, $\ldots$, A_n be some sets. Then

- $\bigcup_{i=1}^{1} A_i = A_1$ and
- $\bigcup_{i=1}^{k+1} A_i = \left(\bigcup_{i=1}^{k} A_i\right) \cup A_{k+1}$.

Show that $\bigcup_{i=1}^{n} [i] = [n]$ for all integers $n > 0$.

6.11 *(recommended)* Let Ω be some set and $A_1, \ldots, A_n \subseteq \Omega$. Show that $\bigcup_{i=1}^{n} A_i = \{x \in \Omega \,:\, \exists i \in [n]\; x \in A_i\}$.

6.12 Let $A_1, \dots, A_n$ be some sets. Show that $\bigcup_{i=1}^{n}(A_i \cap B) = (\bigcup_{i=1}^{n} A_i) \cap B$.

6.13 Show that $A\Delta(B\Delta C) = (A\Delta B)\Delta C$.

6.14 *(recommended)* Let $\mathbb{R}^{m\times n}$ be the set of all matrices $m \times n$ and $\mathbb{R}^n$ be the set of n dimensional vectors. Show that for any matrix $A \in \mathbb{R}^{m\times n}$ $(n > m)$ there is a nonzero vector $x \in \mathbb{R}^n$ such that $Ax = 0$.

7. Functions

Functions and Quantifiers:
Introduction to Mathematical Reasoning #7

https://youtu.be/VHJeUrCedTU

Another important type of objects in mathematics are functions. Function f from a set X to a set Y (written as $f : X \to Y$) is a unique assignment of elements of Y to the elements of X (note that it is not necessary that all the elements of Y are used). In other words, for each element $x \in X$ there is one assigned element $f(x) \in Y$. We call such an element the *value* of f at x, we also say that $f(x)$ is an *image* of x.

Unfortunately, the definition is not formal. Through this chapter we will provide a more formal definition.

7.1 Quantifiers.

The first ingredient is called quantifiers. Very often we use phrases like "all the people in the class have smartphones." However, we still do not know how to write it using symbols.

The Universal Quantifier. In order to say "all" or "every" we use the symbol $\forall$[1]: if $P(a)$ is a predicate about $a \in A$, then $\forall a \in A\ P(a)$ is a statement saying that all the elements of A satisfy the predicate P. In other words it is the same as the statement $\{a \in A\ :\ P(a)\} = A$. For example, $\forall x \in \mathbb{R}\ x \cdot 0 = 0$ says that product of every real number and zero is equal to zero.

[1] The symbol is a turned "A" symbol, the first letter of the word "all".

The Existential Quantifier. The second quantifier means "there is" and is denoted by the symbol $\exists$[2]: if $P(a)$ is a predicate about an element of A, then $\exists a \in A\ P(a)$ says that there is an element of A satisfying the predicate P i.e. $\{a \in A\ :\ P(a)\} \neq \emptyset$. For example, $\exists x \in \mathbb{R}\ x^2 - 1 = 0$ states that there is a real solution of the equation $x^2 - 1 = 0$.

[2] The symbol is a turned "E" symbol, the first letter of the word "exists". It is also interesting that the symbol for the universal quantifier was introduced by Gerhard Gentzen in 1935 but the symbol for the existential quantifier was introduced, 38 years earlier, by Giuseppe Peano in 1897.

> **Warning:** Note that the word "any" sometimes indicates a universal statement and sometimes an existential statement.
> Standard meaning of "any" is "every" like in the statement "$a^2 \geq 0$ for any real number", therefore this statement can be rewritten as $\forall a \in \mathbb{R}\ a^2 \geq 0$. Nonetheless, in the negative and interrogative statements "any" is used to mean "some". For example, "There is not any real number a such that $a^2 < 0$" is asserting that the statement $\exists a \in \mathbb{R}\ a^2 < 0$ is false. And "Is there any real number a such that $a^2 = 1$?" is asking whether the existential statement $\exists a \in \mathbb{R}\ a^2 = 1$ is true.
> Real care is required with questions involving "any": "Is there any integer a such that $a \geq 1$?" clearly is asking whether $\exists a \in \mathbb{R}\ a^2 \geq 1$ is true; however, "Is $a \geq 1$ for any integer a" is less clear and might be taken to asking about the same question as the first question, $\exists a \in \mathbb{Z}\ a \geq 1$ (which is true) but might also be taken to be asking about $\forall a \in \mathbb{Z}\ a \geq 1$ (which is false).

Proving Statements Involving Quantifiers

Most of the statements in mathematics involve quantifiers. This is one of the factors distinguishing advanced from elementary mathematics. In this section we give an overview of the main methods of proof. Though the whole book is about proving such results.

Proving statements of the form $\forall a \in A\ P(a)$. Such statements can be rewritten in the form $a \in A \implies P(a)$. For example, we proved earlier that $a^2 \geq 0$ for all real numbers a using this approach.

Proving statements of the form $\exists a \in A\ P(a)$. The easiest way to prove such a statement is by simply exhibiting an element a of A such that $P(a)$ is true. This method is called *proof by example*.

Let us prove the statement $\exists x \in \mathbb{N}\ x^2 = 4$ using this method. Observe that $2 \in N$ and $2^2 = 4$ so $x = 2$ provides an example proving this statement. There are, however, less direct methods such as use of the counting arguments.

Proving statements involving both quantifiers. To illustrate problems of this type let us prove that for any integer n, if n is even, then n^2 is also even.

This statement is a universal statement $\forall n \in \mathbb{Z}\ (n \text{ is even} \implies n^2 \text{ is even})$. However, the hypothesis that n is even is an existential statement $\exists q \in \mathbb{Z}\ n = 2q$. So we begin the proof as follows:

> Suppose that n is an even integer. Then $n = 2q$ for some integer q.

The conclusion we wish to prove is that n^2 is even, which may be written as $\exists q \in \mathbb{Z}\ n^2 = 2q$. Note that q here is a dummy variable used to express the statement n^2 is a doubled integer. We may replace it by any other letter not already in use, for example $\exists p \in \mathbb{Z}\ n^2 = 2p$. Hence, if we present p such that $n^2 = 2p$, we finish the proof. As a result, we can complete the proof as follows.

Therefore, $n^2 = (2q)^2 = 4q^2$ and so, since $2q^2$ is an integer n^2 is even.

Disproving Statements Involving Quantifiers

Disproving something seems a bit off from the first glance, but to some extent it is the same as proving the negation.

Disproving statements of the form $\forall a \in A\ P(a)$. We may note that the negation of such a statement is the statement $\exists a \in A\ \neg P(a)$. So we can disprove it by giving a single example for which it is false. This is called *disproof by counterexample* to $P(a)$.

For example, we may disprove the statement $\forall x \in \mathbb{R}\ x^2 > 2$ by giving a counterexample $x = 1$ since $1^2 = 1 < 2$.

Disproving statements of the form $\exists a \in A\ P(a)$. The negation of this statement is the statement $\forall a \in A\ \neg P(a)$. Which gives one way of disproving the statement.

Let us prove that there does not exist a real number x such that $x^2 = -1$. We know that, for all $x \in \mathbb{R}$, we have the inequality $x^2 \geq 0$ and so $x^2 \neq -1$. Hence, there does not exist $x \in \mathbb{R}$ such that $x^2 = -1$.

7.2 *Cartesian product*

Another ingredient is the notion of Cartesian product. If X and Y are two sets, then $X \times Y = \{(x, y) : x \in X \text{ and } y \in Y\}$. We also denote $\underbrace{X \times X \times \cdots \times X}_{k \text{ times}}$ by X^k.

Consider the following example. If $X = \{a, b, c\}$ and $Y = \{a, b\}$, then

$$X \times Y = \{(a, a), (a, b), (b, a), (b, b), (c, a), (c, b)\}.$$

Additionally, $\mathbb{R}^2 = \mathbb{R} \times \mathbb{R}$ is the familiar 2-dimensional Euclidean plane.

Exercise 7.1. *Find the set $\{a, b\} \times \{a, b\} \setminus \{(x, x) : x \in \{a, b\}\}$*

Theorem 7.1. *For all sets A, B, C, and D the following hold:*

- $A \times (B \cup C) = (A \times B) \cup (A \times C)$;
- $A \times (B \cap C) = (A \times B) \cap (A \times C)$;

- $(A \times B) \cup (C \times D) \subseteq (A \cup C) \times (B \cup D)$;
- $(A \times B) \cap (C \times D) = (A \cap C) \times (B \cap D)$.

Proof. It is easy to prove this statement by the definitions. Let us prove only the second equality, the rest is Exercise 7.2.

Note that $(x,y) \in A \times (B \cap C)$ iff $x \in A$ and $y \in (B \cap C)$. Hence, $(x,y) \in A \times (B \cap C)$ iff $x \in A$, $y \in B$, and $y \in C$. Thus $(x,y) \in A \times (B \cap C)$ iff $(x,y) \in (A \times B)$ and $(x,y) \in (A \times C)$. As a result, $(x,y) \in A \times (B \cap C)$ iff $(x,y) \in (A \times B) \cap (A \times C)$ as required. □

Exercise 7.2. *Prove the rest of the equalities in Theorem 7.1.*

7.3 *Graphs of Functions*

Now we have all the components to define a function. Mathematicians think about the functions in the way we defined them at the beginning of the chapter, however formally in order to define a function $f : X \to Y$ one need to define a set $D \subseteq X \times Y$ (such a set is called the *graph of the function* f) such that

- $\forall x \in X\ \exists y \in Y\ (x,y) \in D$ and
- $\forall x \in X, y_1, y_2 \in Y\ ((x,y_1) \in D \wedge (x,y_2) \in D \implies y_1 = y_2)$.

We say that $y \in Y$ is the value $f(x)$ of the function described by D at $x \in X$ iff $(x,y) \in D$.

The simplest way to think about the functions is in the terms of tables. Let us use this idea to list all the functions $\{a,b,c\}$ to $\{d,e\}$.

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$	$f_6(x)$	$f_7(x)$	$f_8(x)$
a	d	d	d	d	e	e	e	e
b	d	d	e	e	d	d	e	e
c	d	e	d	e	d	e	d	e

Exercise 7.3. *List all the functions from* $\{a,b\}$ *to* $\{a,b\}$.

However, listing all the values of a function is only possible when the domain of the function is finite. Thus the most common way to describe a function is using a formula which provides a way to find the value of a function. When the function is defined as a formula it is important to be clear which sets are the domain and the codomain of the function.

Let $\mathbb{R}_+ = \{x \in \mathbb{R} : x \geq 0\}$. Consider the following functions.

- $g_1 : \mathbb{R} \to \mathbb{R}$ such that $g_1(x) = x^2$;
- $g_2 : \mathbb{R}_+ \to \mathbb{R}$ such that $g_2(x) = x^2$;

- $g_3 : \mathbb{R} \to \mathbb{R}_+$ such that $g_3(x) = x^2$;
- $g_4 : \mathbb{R}_+ \to \mathbb{R}_+$ such that $g_4(x) = x^2$;

Nonetheless that all these functions are defined using the same formula x^2, we will see in the next chapters that these four functions have different properties.

Exercise 7.4. *Find the graph of the function $f : \mathbb{Z} \to \mathbb{Z}$ such that $f(x) = 3x$.*

Note that when you define the function you need to define it such that the definition makes sense for all the elements of the domain. For example, the formula $g(x) = \frac{x^2-3x+2}{x-1}$ does not define a function from $\mathbb{R}$ to $\mathbb{R}$ since it is not defined for $x = 1$. It is typical to define a function from real numbers to real numbers by a formula and the convention is that the domain is the set of all numbers for which the formula makes sense (unless the domain is specified explicitly). Using this convention the formula g defines a function from $\mathbb{R} \setminus \{1\}$ to $\mathbb{R}$.

If we really need a function from $\mathbb{R}$ there are two possible approaches for extending g.

Rewriting the formula. We can rewrite the formula such that it makes sense for all the real numbers. Note that for all $x \in \mathbb{R} \setminus \{1\}$,

$$\frac{x^2 - 3x + 2}{x - 1} = \frac{(x-2)(x-1)}{x-1} = x - 2.$$

Then $g_1(x) = x - 2$ defines a function on $\mathbb{R}$ extending the function g.

Explicit definition. Alternatively we can explicitly specify the value of g at 1. So

$$g_2(x) = \begin{cases} \frac{x^2-3x+2}{x-1} & \text{if } x \neq 1 \\ -1 & \text{if } x = 1 \end{cases}$$

defines a function from $\mathbb{R}$ to $\mathbb{R}$. Note that we can specify the values at individual points any way we want.

Similarly to sets we may define the equality between functions. We say that two functions $f, g : X \to Y$ are equal ($f = g$) iff $f(x) = g(x)$ for all $x \in X$ i.e. their graphs are equal. Note that two functions are equal only if they have the same domains and codomains. For example, g_1 and g_2 we just defined are equal to each other nonetheless that we defined them in two different ways.

We defined g_1 and g_2 to extend g to a bigger domain, similarly we can make a domain smaller.

Definition 7.1. *Let $f : X \to Y$ and $A \subseteq X$. Then $f|_A : A \to Y$ is a function such that $\forall x \in A$ $f|_A(x) = f(x)$ (we say that $f|_A$ is the restriction of f to the set A).*

7.4 Composition of Functions

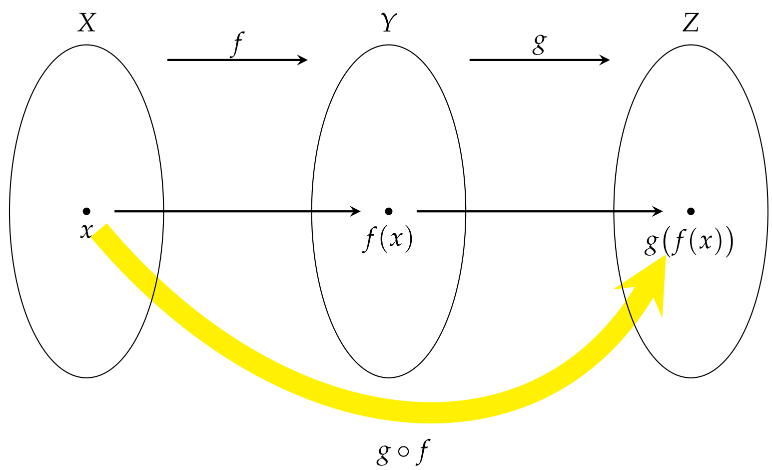


Figure 7.1: Composition of functions

Suppose $f : X \to Y$ and $g : Y \to Z$ be some function. Then, given an element $x \in X$, the function f assigns $y = f(x) \in Y$, and the function g assigns $z = g(y) = g(f(x)) \in Z$. Thus using f and g an element of Z can be assigned to x. This operation defines a function from X to Z and the result of this operation is called the *composition* of f and g.

Definition 7.2. *If $f : X \to Y$ and $g : Y \to Z$, then $h = g \circ f$ is a function from X to Z such that $g(f(x)) = h(x)$ for all $x \in X$.*

Let us consider an example. Let $f : \mathbb{R} \to \mathbb{R}$ such that $f(x) = x + 1$ and $g : \mathbb{R} \to \mathbb{R}$ such that $g(x) = x^2$. Then $(g \circ f) : \mathbb{R} \to \mathbb{R}$ and $(g \circ f)(x) = (x+1)^2$ for all $x \in \mathbb{R}$. Note that the order of f and g is important since $(f \circ g)(x) = x^2 + 1$. Thus composition is not *commutative*.

There are two special type functions.

- Let $A \subseteq X$, then $i : A \to X$ such that $i(a) = a$ for all $a \in A$ is called the *inclusion* function of A into X. Observe that $(f \circ i) : A \to Y$ and $(f \circ i) = f|_A$ for any function $f : X \to Y$.
- Another important function is called the *identity* function. Let X be some set. Then $I_X : X \to X$ is the identity function on X iff $I_X(x) = x$.

Theorem 7.2. *Let $f : X \to Y$, $g : Y \to Z$, and $h : Z \to W$. Then*

- $h \circ (g \circ f) = (h \circ g) \circ f$.
- $f \circ I_X = f = I_Y \circ f$.

Proof. These results can be proven simply by evaluating the functions. For example, both functions in the first equality assign $h(g(f(x)))$ for any $x \in X$ and so functions are equal. $\square$

Notice that this theorem states that we may write $f \circ g \circ h$ without ambiguity.

7.5 The Image of a Function

Given a function $f : X \to Y$, it is not necessary that every element of Y is an image of some $x \in X$. For example, the function $\mathbb{R} \to \mathbb{R}$ defined by the formula x^2 does not have -1 as a value.

Thus we may give the following definition.

Definition 7.3. *The image of the function f is defined as follows*

$$\operatorname{Im} f = \{y \in Y \,:\, \exists x \in X\ f(x) = y\} = \{f(x) \,:\, x \in X\}$$

(in other words it is the projection of the graph D of f on the second coordinate: $\operatorname{Im} f = \{y \,:\, (x, y) \in D\}$*).*

End of The Chapter Exercises

7.5 Is there $x, y, z \in \mathbb{N}$ such that $29x + 30y + 31z = 366$.

7.6 Find then image of the function $f : \mathbb{Z} \to \mathbb{Z}$ such that $f(x) = 3x$.

7.7 *(recommended)* Determine the following sets:

- $\{m \in \mathbb{N} \,:\, \exists n \in \mathbb{N}\ m \leq n\}$;
- $\{m \in \mathbb{N} \,:\, \forall n \in \mathbb{N}\ m \leq n\}$;
- $\{n \in \mathbb{N} \,:\, \exists m \in \mathbb{N}\ m \leq n\}$;
- $\{n \in \mathbb{N} \,:\, \forall m \in \mathbb{N}\ m \leq n\}$.

7.8 Prove or disprove the following statements.

- $\forall m, n \in \mathbb{N}\ m \leq n$.
- $\exists m, n \in \mathbb{N}\ m \leq n$.
- $\exists m \in \mathbb{N}\ \forall n \in \mathbb{N}\ m \leq n$.
- $\forall m \in \mathbb{N}\ \exists n \in \mathbb{N}\ m \leq n$.
- $\exists n \in \mathbb{N}\ \forall m \in \mathbb{N}\ m \leq n$.
- $\forall n \in \mathbb{N}\ \exists m \in \mathbb{N}\ m \leq n$.

7.9 *(recommended)* We call elements of the set $\{0,1\}^n$ Binary strings of length n. Moreover, instead of $(c_1, \ldots, c_n)$ we write $c_1, \ldots, c_n$ and we call c_is characters. Show that all Binary strings of length n may be ordered such that every successive strings in this order are different only in one character. (For example, for $n = 2$ the order may be 00, 01, 11, 10.)

8. Relations

Nonetheless that function are used almost everywhere in mathematics, many relations are not functional by their nature. For example, for any real a, there are two solutions of $x^2 = a$ and there are zero solutions for $a < 0$. To work with such situations, relations are used.

In order to define a relation we need to relax the definition of the graph of a function (Section 7.3) by allowing more than one "result" and by allowing zero "results". In other words we just say that any set $R \subseteq X_1 \times \cdots \times X_k$ is a k-ary *relation* on $X_1, \ldots, X_k$. We also say that $x_1 \in X_1, \ldots, x_k \in X_k$ are in the relation R iff $(x_1, \ldots, x_k) \in R$. If $k = 2$ such a relation is called a *binary relation* and we write xRy if x and y are in the relation R. If $X_1 = \cdots = X_k = X$, we say that R is a k-ary relation on X.

Note that $=$, $\leq$, $\geq$, $<$, and $>$ define relations on $\mathbb{R}$ (or any subset S of $\mathbb{R}$). For example, if $S = \{0, 1, 2\}$, then $<$ defines the relation $R = \{(0,1), (0,2), (1,2)\}$.

Another widely used family of relations on $\mathbb{Z}$ can be defined as follows. Let $n, a, b \in \mathbb{Z}$. If n divides $a - b$, we say that "a equivalent to b modulo n" and denote it as $a \equiv b \pmod{n}$. For example, 1 and 4 are equivalent modulo 3 since 3 divides $1 - 4 = -3$.

8.1 Equivalence Relations

The definition of a relation is way too broad. Hence, quite often we consider some types of relation. Probably the most interesting type of the relations is equivalence relations.

Definition 8.1. *Let R be a binary relation on a set X. We say that R is an* equivalence relation *if it satisfies the following conditions:*

(reflexivity) xRx *for any* $x \in X$*;*

(symmetry) xRy *iff* yRx *for any* $x, y \in X$*;*

(transitivity) *for any* $x, y, z \in X$*, if* xRy *and* yRZ*, then* xRz*.*

One may guess that the equivalence relation are mimicking $=$, so it is not a surprise that $=$ is an equivalence relation.

The definition seems quite bizarre, however, all of you are already familiar with another important example: you know that equivalent fractions represent the same number. For example, $\frac{2}{4}$ is the same as $\frac{1}{2}$. Let us consider this example more thorough, let S be a set of symbols of the form $\frac{x}{y}$ (note that it is not a set of numbers) where $x, y \in Z$ and $y \neq 0$. We define a binary relation R on S such that $\frac{x}{y}$ and $\frac{z}{w}$ are in the relation R iff $xw = zy$. It is easy to prove that this relation is an equivalence relation.

(reflexivity) Let $\frac{a}{b} \in S$. Since $ab = ab$, we have that $\frac{a}{b} R \frac{a}{b}$.

(symmetry) Let $\frac{a}{b}, \frac{c}{d} \in S$. Suppose that $\frac{a}{b} R \frac{c}{d}$, by the definition of R, it implies that $ac = db$. As a result, $\frac{c}{d} R \frac{a}{b}$.

(transitivity) Let $\frac{a}{b}, \frac{c}{d}, \frac{e}{f} \in S$ with $\frac{a}{b} R \frac{c}{d}$ and $\frac{c}{d} R \frac{e}{f}$. Then $ad = cb$ and $cf = ed$. The first equality can be rewritten as $c = ad/b$. Hence, $adf/b = ed$ and $af = eb$ since $d \neq 0$. So $\frac{a}{b} R \frac{e}{f}$.

Partitions

Let S be some set. We say that $\{P_1, \ldots, P_k\}$ form a partition of S iff P_1, $\ldots$, P_k are pairwise disjoint and $P_1 \cup \cdots \cup P_k = S$; in other words, a partition is a way of dividing a set into overlapping pieces.

Exercise 8.1. *Let $\{P_1, \ldots, P_k\}$ be a partition of a set S and R be a binary relation of S such that aRb iff $a, b \in P_i$ for some $i \in [k]$. Show that R is an equivalence relation.*

This exercise shows that one may transform a partition of the set S into an equivalence relation on S. However, it is possible to do the opposite.

Theorem 8.1. *Let R be a binary equivalence relation on a set S. For any element $x \in S$, define $R_x = \{y \in S : xRy\}$ (the set of all the elements of S related to x) we call such a set the equivalence class of x. Then $\{R_x : x \in S\}$ is a partition of S.*

Exercise 8.2. *Prove Theorem 8.1.*

Modular Arithmetic

The relation "$\equiv \pmod{n}$" is actively used in the number theory. One of the important properties of this relation is that it is an equivalence relation.

Theorem 8.2. *The relation $\equiv \pmod{n}$ is an equivalence relation.*

Proof. To prove this statement we need to prove all three properties: reflexivity, symmetry, and transitivity.

(reflexivity) Note that for any integer x, $x - x = 0$ is divisible by any integer including n. Hence, $x \equiv x \pmod{n}$.

(symmetry) Let us assume that $x \equiv y \pmod{n}$; i.e., $x - y = kn$ for some integer k. Note that $y - x = (-k)n$, so $y \equiv x \pmod{n}$.

(transitivity) finally, assume that $x \equiv y \pmod{n}$ and $y \equiv z \pmod{n}$; i.e. $x - y = kn$ and $y - z = \ell n$ for some integers k and ℓ. It is easy to note that $x - z = (x - y) + (y - z) = (k + \ell)n$. As a result, $x \equiv z$ $(\text{mod } n)$.

Thus, we proved that $\equiv \pmod{n}$ is an equivalence relation. $\square$

Let $x \in \mathbb{Z}$; we denote by $r_{x,n}$ the equivalence class of x with respect to the relation $\equiv \pmod{n}$, we also denote by $\mathbb{Z}/n\mathbb{Z}$ the set of all the equivalence classes with respect to the relation $\equiv \pmod{n}$.

Another important property of these relations is that they behave well with respect to the arithmetic operations.

Theorem 8.3. *Let $x, y \in Z$ and $n \in \mathbb{N}$. Suppose that $a \in r_{x,n}$ and $b \in r_{y,n}$, then $(a + b) \in r_{x+y,n}$ and $ab \in r_{xy,n}$.*

Using this theorem we may define arithmetic operations on the equivalence classes with respect to the relation $\equiv \pmod{n}$. Let $x, y \in \mathbb{Z}$ and $n \in \mathbb{N}$. Then $r_{x,n} + r_{y,n} = \{a + b \; : \; a \in r_{x,n}, b \in r_{y,n}\} = r_{x+y,n}$ and $r_{x,n}r_{y,n} = \{ab \; : \; a \in r_{x,n}, b \in r_{y,n}\} = r_{xy,n}$. Moreover, these operations have plenty of good properties.

Exercise 8.3. *Let $a, b, c \in \mathbb{Z}/n\mathbb{Z}$. Show that the following equalities are true:*

- $a + (b + c) = (a + b) + c$,
- $a + r_{0,n} = a$ *(thus we denote $r_{0,n}$ as 0)*,
- $ar_{1,n} = a$ *(thus we denote $r_{1,n}$ as 1)*,
- *there is a class $d \in \mathbb{Z}/n\mathbb{Z}$ such that $a + d = r_{0,n}$ (thus we denote this d as $-a$)*,
- $a + b = b + a$,
- $ab = ba$,
- $a(b + c) = ab + ac$,

8.2 *Partial Orderings*

In the previous section we discussed a generalization of "=". In this section we are going to give a way to analyze relations similar to "<".

Definition 8.2. *A binary relation R on S is a partial ordering if it satisfies the following constraints.*

(reflexivity) xRx *for any* $x \in S$*;*

(antisymmetry) *if* xRy *and* yRx*, then* $x = y$ *for all* $x, y \in S$*;*

(transitivity) *for any* $x, y, z \in S$*, if* xRy *and* yRZ*, then* xRz*;*

We say that a partial ordering R on a set S is total iff for any $x, y \in S$, either xRy or yRx.

Note that $\leq$ defines a partial ordering on any $S \subseteq \mathbb{R}$; moreover, it defines a total order.

Typically we use symbols similar to $\preceq$ to denote partial orderings and we write $a \prec b$ to express that $a \preceq b$ and $a \neq b$.

Let $|$ be the relation on $\mathbb{Z}$ such that $d \mid n$ iff d divides n.

Theorem 8.4. *The relation $|$ is a partial ordering of the set $\mathbb{N}$.*

Proof. To prove that this relation is a partial ordering we need to check all three properties.

(reflexivity) Note that $x = 1 \cdot x$ for any integer x; hence, $x \mid x$ for any integer x.

(antisymmetry) Assume that $x \mid y$ and $y \mid x$. Note that it means that $kx = y$ and $\ell y = x$ for some integers k and ℓ. Hence, $y = (k \cdot \ell)y$ which implies that $k \cdot \ell = 1$ and $k = \ell = 1$. Thus, $x = y$.

(transitivity) finally, assume that $x \mid y$ and $y \mid z$; i.e., $kx = y$ and $\ell y = z$. As a result, $(k \cdot \ell)x = z$ and $x \mid z$.

□

Exercise 8.4. *Let S be some set, show that $\subseteq$ defines a partial ordering on the set 2^S.*

Topological Sorting

Partial orderings are very useful for describing complex processes. Suppose that some process consists of several tasks, T denotes the set of these tasks. Some tasks can be done only after some others e.g. when you cooking a salad you need to wash vegetables before you chop them. If $x, y \in T$ be some tasks, $x \preceq y$ if x should be done before y and this is a partial ordering.

In the applications this order is not a total order because some steps do not depend on other steps being done first (you can chop tomatoes and chop cucumbers in any order). However, if we need to create a schedule in which the tasks should be done, we need to create a total ordering on T. Moreover, this order should be compatible with the partial ordering. In other words, if $x \preceq y$, then $x \preceq_t y$ for all $x, y \in T$, where $\preceq_t$ is the total order. The technique of finding such a total ordering is called *topological sorting*.

Theorem 8.5. *Let S be a finite set and $\preceq$ be a partial order on S. Then there is a total order $\preceq_t$ on S such that if $x \preceq y$, then $x \preceq_t y$ for all $x, y \in S$*

This sorting can be done using the following procedure.

- Initiate the set S being equal to T
- Choose the minimal element of the set S with respect to the ordering $\preceq$ (such an element exists since S is a finite set, see Chapter 17). Add this element to the list, remove it from the set S, and repeat this step if $S \neq \emptyset$.

Let us consider the following example. In the left column we list the classes and in the right column the prerequisite.

Courses	Prerequisite
Math 20A	
Math 20B	Math 20A
Math 20C	Math 20B
Math 18	
Math 109	Math 20C, Math 18
Math 184A	Math 109

We need to find an order to take the courses.

1. We start with

   $$S = \{\text{Math 20A}, \text{Math 20B}, \text{Math 20C}, \text{Math 18}, \text{Math 109}, \text{Math 184}\}.$$

   There are two minimal elements: Math 20A and Math 18. Let us remove Math 18 from S and add it to the resulting list R.

2. Now we have

   $$R = \text{Math 18}$$

   and

   $$S = \{\text{Math 20A}, \text{Math 20B}, \text{Math 20C}, \text{Math 109}, \text{Math 184}\}.$$

   There is only one minimal element Math 20A. We remove it and add it to the list R.

3. On this step
$$R = \text{Math 18}, \text{Math 20A}$$
and
$$S = \{\text{Math 20B}, \text{Math 20C}, \text{Math 109}, \text{Math 184}\}.$$
Again there is only one minimal element: Math 20B.

4. $$R = \text{Math 18}, \text{Math 20A}, \text{Math 20B}$$
and
$$S = \{\text{Math 20C}, \text{Math 109}, \text{Math 184}\}.$$
There is only one minimal element: Math 20C.

5. $$R = \text{Math 18}, \text{Math 20A}, \text{Math 20B}, \text{Math 20C}$$
and
$$S = \{\text{Math 109}, \text{Math 184}\}.$$
There is only one minimal element: Math 109.

6. Finally,
$$R = \text{Math 18}, \text{Math 20A}, \text{Math 20B}, \text{Math 20C}, \text{Math 109}$$
and
$$S = \{\text{Math 184}\}.$$
There is only one minimal element: Math 184A.

As a result, the final list is
$$R = \text{Math 18}, \text{Math 20A}, \text{Math 20B}, \text{Math 20C}, \text{Math 109}, \text{Math 184A}.$$

End of The Chapter Exercises

8.5 *(recommended)* Show that the relation $|$ does not define a partial ordering on $\mathbb{Z}$.

8.6 Let a relation R be defined on the set of real numbers as follows: xRy iff $2x + y = 3$. Show that it is antisymmetric.

8.7 Are there any minimal elements in $\mathbb{N}$ with respect to $|$? Are there any maximal elements?

9. Structural Induction

To illustrate the notions we introduce in this chapter we are going to prove that in the game from Section 4.1 it is impossible to guess the number correctly using less than 10 questions.

To prove such a statement we need a formal definition of an algorithm that Bob may use.

9.1 Recursive Definitions

First note that any question for Alice can be formulated as follows: "Is the value of a function f at your number equal to T?", where f is a function from $\mathbb{Z}$ to $\{0,1\}$ (here and in the sequel we interpret 1 as "yes" and 0 as "no").

Hence, there are two possible behaviours of any algorithm for Bob.

- The algorithm prescribes Bob to just say that the answer is some number $x \in \mathbb{Z}$, or
- The algorithm prescribes Bob to ask whether f at Alice's number is equal to T. If the answer is yes, then the algorithm prescribes Bob to proceed according to an algorithm A_1, otherwise the algorithm prescribe Bob to proceed according to an algorithm A_0.

Hence, any algorithm for Bob can be described using the following object.

Definition 9.1. *We say that T is a B-decision tree if*

(base case) either T is equal to **return** *y, where $y \in \mathbb{Z}$; or*

(recursion step) T is equal to **if** *f* **then** *T_0* **else** *T_1, where $f : \mathbb{Z} \to \{0,1\}$, and T_0 and T_1 are B-decision trees.*

Note that this definition is not quite formal since it is recursive and we usually do not allow recursive definitions. So we will need to give a more formal way to define B-decision trees. However, this definition allows us to prove that

$$T = \textbf{if } f_1 \textbf{ then if } f_2 \textbf{ then return } 1 \textbf{ else return } 2 \textbf{ else return } 3$$

is a B-decision tree, where

$$f_1(x) = \begin{cases} 1 & \text{if } x \leq 2 \\ 0 & \text{otherwise} \end{cases},$$

and

$$f_2(x) = \begin{cases} 1 & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}.$$

This can be explained as follows.

- It is clear that **return** 1, **return** 2, and **return** 3 are B-decision trees by the base case.

- Hence, by the recursion step case, **if** f_2 **then return** 1 **else return** 2 is a B-decision tree.

- Finally, by the recursion step case, T is a B-decision tree.

In other words we proved that T is a B-decision tree by providing $T_1, \ldots, T_5$ such that $T_5 = T$ and for each $i \in [5]$, T_i is either equal to **return** y (for $y \in \mathbb{Z}$) or T_i is equal to **if** f **then** T_j **else** T_k (for $j, k < i$ and $f : \mathbb{Z} \to \{0, 1\}$).

This idea leads to the framework that would allows us to give a formal definition of B-decision trees.

Definition 9.2. *Let U be a set, let $S_0 \subseteq U$, and let*

$$\mathcal{F} = \left\{ F_1 : U^{\ell_1} \to U, \ldots, F_n : U^{\ell_n} \to U, \ldots \right\}.$$

Then we say that the set S is generated by $\mathcal{F}$ from S_0 if it is the set of all $u \in U$ such that there is a sequence $u_1, \ldots, u_m$ satisfying the following constraints: $u_m = u$ and for each $i \in [m]$,

- *either $u_i \in S_0$, or*

- *$u_i = F(u_{k_1}, \ldots, u_{k_\ell})$ for $F \in \mathcal{F}$ and $k_1, \ldots, k_\ell < i$.*

In case of B-decision trees U is the set of all sequences of numbers, **if**, **then**, **else**, and functions from $\mathbb{Z}$ to $\{0, 1\}$, S_0 is the set of all sequences **return** y for $y \in \mathbb{Z}$, and

$$\mathcal{F} = \left\{ F_f \ : \ f \text{ is a function from } \mathbb{Z} \text{ to } \{0, 1\} \right\},$$

where $F_f(T_0, T_1)$ is equal to **if** f **then** T_0 **else** T_1.

Remark 9.1. *Let U be a set, let $S_0 \subseteq U$, and let*

$$\mathcal{F} = \left\{ F_1 : U^{\ell_1} \to U, \ldots, F_n : U^{\ell_n} \to U, \ldots \right\}.$$

Let us consider the sets $S_0, \ldots, S_n, \cdots \subseteq U$ such that

$$S_{i+1} = S_i \cup \{F(u_1, \ldots, u_\ell) \ : \ u_1, \ldots, u_\ell \in S_i, F \in \mathcal{F}\} .$$

It is clear that $\bigcup_{i \geq 0} S_i$ is the set generated by $\mathcal{F}$ from S_0.

Using similar ideas we may define some functions on the objects defined recursively.

Definition 9.3. *Let T be a B-decision tree. Then the height $\mathrm{h}(T)$ of T can be defined as follows.*

(base case) *If T is equal to* **return** *y for $y \in \mathbb{Z}$, then $\mathrm{h}(T) = 0$.*

(recursion step) *Let T be equal to* **if** *f* **then** *T_0* **else** *T_1, where T_0 and T_1 are B-decision trees. Then $\mathrm{h}(T) = \max(\mathrm{h}(T_0), \mathrm{h}(T_1)) + 1$.*

Note that $\mathrm{h}(T)$ correpsonds to the worst-case number of queries made by Bob if we interpret T as a description of Bob's algorithm.

However, before we explain how to formalize such a definition, we need to note that in the general case such definition may be contradictory. Consider $U = \mathbb{R}$, $S_0 = \{0\}$, and $\mathcal{F} = \{f, g\}$, where $f(x, y) = xy$ and $g(x) = x + 1$. We define $v : U \to \mathbb{R}$ as follows.

(base case) $v(0) = 0$.

(recursion step) $v(f(x, y)) = f(v(x), v(y))$ and $v(g(x)) = v(x) + 2$.

Note that $v(f(g(0), g(0))) = f(v(g(0)), v(g(0))) = \big(v(g(0))\big)^2 = 4$ and $v(g(0)) = v(0) + 2 = 2$. However, $g(0) = 1$ and $f(g(0), g(0)) = 1$.

Therefore handle such an issue, we consider S that is *freely* generated from S_0 by by $\mathcal{F}$.

Definition 9.4. *The set S is freely generated by $\mathcal{F}$ from S_0 iff it is generated by $\mathcal{F}$ from S_0, $S_0 \cap \operatorname{Im} F = \varnothing$, and $\operatorname{Im} F \cap \operatorname{Im} G = \varnothing$ for any $F, G \in \mathcal{F}$.*

The following theorem claims existence of functions defined recursively.

Theorem 9.1. *Let $S \subseteq U$ be the set freely generated from $S_0 \subseteq U$ by $\mathcal{F} = \Big\{F_1 : U^{\ell_1} \to U, \ldots, F_n : U^{\ell_n} \to U, \ldots\Big\}$. In addition, let $G_0 : S_0 \to V$ and $G_1 : V^{\ell_1} \to V, \ldots, G_n : V^{\ell_n} \to V, \ldots$ be some functions.*

Then there is a function $h : S \to V$ such that

(base case) *$h(u) = G_0(u)$ for any $u \in S_0$.*

(recursion step) *$h(F_i(u_1, \ldots, u_{\ell_i})) = G_i(h(u_1), \ldots, h(u_{\ell_i}))$ for any i and $u_1, \ldots, u_{\ell_i} \in S$.*

Exercise 9.1. *Prove Theorem 9.1.*

B-decision tree are used in this chapter to represent Bob's algorithms; hence, we need to define values of *B*-decision trees.

Definition 9.5. *The value* $\mathrm{val}(T,n)$ *of a B-decision tree T at an integer n can be defined as follows.*

(base case) If T is equal to **return** *y for* $y \in \mathbb{Z}$, *then* $\mathrm{val}(T,n) = y$.

(recursion step) Let T be equal to **if** f **then** T_0 **else** T_1, *where* T_0 *and* T_1 *are B-decision trees, and* $f : \mathbb{Z} \to \{0,1\}$. *Then*

$$\mathrm{val}(T,n) = \begin{cases} \mathrm{val}(T_0,n) & \text{if } f(x) = 0 \\ \mathrm{val}(T_1,n) & \text{otherwise} \end{cases}.$$

Using all these notions we can reformulate the results about Alice and Bob's game.

Theorem 9.2. *1. There is a B-decision tree T such that*

- $\mathrm{h}(()T) \leq 10$ *and*
- $\mathrm{val}(T,n) = n$ *for all* $n \in [1000]$.

2. *Let T be a B-decision tree such that* $\mathrm{val}(T,n) = n$ *for all* $n \in [1000]$. *Then* $\mathrm{h}(T) \geq 10$.

Exercise 9.2. *Prove the first part of Theorem 9.2.*

9.2 *Structural Induction Theorem*

To prove the second part of Theorem 9.2 we need to introduce the notion of structural induction.

Theorem 9.3 (The Structural Induction Principle)**.** *Let* $S \subseteq U$ *be the set freely generated from* $S_0 \subseteq U$ *by* $\mathcal{F} = \left\{F_1 : U^{\ell_1} \to U, \ldots, F_n : U^{\ell_n} \to U, \ldots\right\}$.

Assume that $S' \subseteq U$ *is a set such that the following constraints are true.*

(base case) $S_0 \subseteq S'$

(induction step) $F_i(u_1, \ldots, u_{\ell_i}) \in S'$ *for any* $u_1, \ldots, u_{\ell_i} \in S'$ *and* $i \in \mathbb{N}$.

Then $S \subseteq S'$.

Using this result, we may prove Theorem 9.2.

Proof of Theorem 9.2. We need to prove only the second part of the statement. Let $V(T) = \{\mathrm{val}(T,n) : n \in \mathbb{Z}\}$, where T is a *B*-decision tree. This proof is based on the following observation.

Claim 9.3.1. *For any B-decision tree T, size of* $V(T)$ *is at most* $2^{\mathrm{h}(T)}$.

Assume that T is a B-decision tree such that $\mathrm{val}(T, n) = n$ for any $n \in [1000]$. Whence $[1000] \subseteq V(T)$. Therefore $V(T)$ has at least 1000 elements. As a result, $2^{\mathrm{h}(T)} \geq 1000$; which implies that $\mathrm{h}(T) \geq 10$ since $\mathrm{h}(T)$ is an integer.

So we just need to prove Claim 9.3.1. We prove it using structural induction. Let S' be the set of decision trees T such that size of $V(T)$ is at most $2^{h(T)}$.

(base case) Let T be equal to **return** y, where $y \in Z$, then $\mathrm{val}(T, n) = n$ for any $n \in \mathbb{Z}$. Hence, the size of $V(T)$ is equal to $1 = 2^0 = 2^{\mathrm{h}(T)}$.

(induction step) Assume that T is equal to **if** f **then** T_0 **else** T_1 for some $T_0, T_1 \in S'$. We know that the size of $V(T_0)$ is at most $2^{\mathrm{h}(T_0)}$ and the size of $V(T_0)$ is at most $2^{\mathrm{h}(T_1)}$. In addtion, it is clear that $V(T) \subseteq V(T_0) \cup V(T_1)$. Therefore, the size of $V(T)$ is at most[1]

$$2^{\mathrm{h}(T_0)} + 2^{\mathrm{h}(T_1)} \leq 2^{\max(\mathrm{h}(T_0), \mathrm{h}(T_1))+1} = 2^{\mathrm{h}(T)}.$$

[1] Formally speaking, we use Theorem 18.1 to prove this; i.e. we use the fact that the size of a set $A \cup B$ is at most the size of A plus the size of B.

Hence, by Theorem 9.3, S' is equal to the set of all B-decision trees. □

Now we are ready to prove Theorem 9.3.

Proof of Theorem 9.3. We prove the statement using induction. More precisely, we prove using induction by m that if there is a sequence u_1, $\dots$, u_m such that for each $i \in [m]$, $u_i \in S_0$ or $u_i = F(u_{k_1}, \dots, u_{k_\ell})$ for $F \in \mathcal{F}$ and $k_1, \dots, k_\ell < i$, then $u_m \in S'$.

The case when $m = 1$ is clear since in this case $u_1 \in S_0$ which implies that it is in S'.

Let us now prove the induction step. Assume that the statement is true for any $k \leq m$. Consider a sequence u_1, $\dots$, u_{m+1} such that for each $i \in [m+1]$, $u_i \in S_0$ or $u_i = F(u_{k_1}, \dots, u_{k_\ell})$ for $F \in \mathcal{F}$ and $k_1, \dots, k_\ell < i$. Let us consider $F \in \mathcal{F}$ and $k_1, \dots, k_\ell < m+1$ such that $u_{m+1} = f(u_{k_1}, \dots, u_{k_\ell})$. By the induction hypothesis, $u_{k_1}, \dots, u_{k_\ell} \in S'$. Therefore, by the properties of S', $u_{m+1} \in S'$. □

End of The Chapter Exercises

9.3 *(recommended)* Let $S \subseteq \mathbb{Z}$ be a set of size at least 2^k, and let T be a decision tree such that $\mathrm{val}(T, n) = n$ for any $n \in S$. Show that $\mathrm{h}(T) \geq k$.

9.4 *(recommended)* Let S be a set of integers defined recursively such that

- $3 \in S$, and
- if $x, y \in S$, then $x + y \in S$.

Show that $S = \{3k \ : \ k \in \mathbb{N}\}$.

9.5 *(recommended)* Using recursive definition we can define an arithmetic formula on the variables $x_1, \dots, x_n$,

(base case) x_i is an arithmetic formula on the variables $x_1, \dots, x_n$ for all i; if c is a real number, then c is also an arithmetic formula on the variables $x_1, \dots, x_n$.

(recursion step) If P and Q are arithmetic formulas on the variables $x_1, \dots, x_n$, then $(P + Q)$ and $P \cdot Q$ are arithmetic formulas on the variables $x_1, \dots, x_n$.

We can also define recursively the value of such a formula. Let v_1, $\dots$, v_n be some integers.

(base cases) $x_i\big|_{x_1=v_1,\dots,x_n=v_n} = v_i$; in other words, the value of the arithmetic formula x_i is equal to v_i when $x_1 = v_1, \dots, x_n = v_n$; if c is a real number, then $c|_{x_1=v_1,\dots,x_n=v_n} = c$.

(recursion steps) If P and Q are arithmetic formulas on the variables $x_1, \dots, x_n$, then

$$(P+Q)\big|_{x_1=v_1,\dots,x_n=v_n} = P\big|_{x_1=v_1,\dots,x_n=v_n} + Q\big|_{x_1=v_1,\dots,x_n=v_n}$$

and

$$(P \cdot Q)\big|_{x_1=v_1,\dots,x_n=v_n} = P\big|_{x_1=v_1,\dots,x_n=v_n} \cdot Q\big|_{x_1=v_1,\dots,x_n=v_n}.$$

Prove that for any arithmetic formula A on x, there is a polynomial p such that $p(v) = A\big|_{x=v}$ for any $v \in \mathbb{R}$.

9.6 • Define arithmetic formulas with division and define their value (make sure that you handled divisions by 0).

- Show that for any arithmetic formula with division A on x, there are polynomials p and q such that $\frac{p(v)}{q(v)} = A\big|_{x=v}$ or $A\big|_{x=v}$ is not defined for any real value v.

9.7 *(recommended)* We say that L is a B-decision list

(base case) if either L is equal to **return** y, where $y \in \mathbb{Z}$; or

(recursion step) L is equal to **if** f **then return** v **else** L', where $f : \mathbb{Z} \to \{0, 1\}$, $v \in \mathbb{Z}$, and L is a B-decision list.

We can also define the value $\mathrm{val}(L, x)$ of a B-decision list L at $x \in \mathbb{Z}$.

(base case) Let L be equal to **return** y, where $y \in \mathbb{Z}$. Then $\mathrm{val}(L, x) = y$.

(recursion step) Let L be equal to **if** f **then return** v **else** L', then

$$\mathrm{val}(L, x) = \begin{cases} v & \text{if } f(x) = 1 \\ \mathrm{val}(L', x) & \text{otherwise} \end{cases}.$$

Similarly one may define the length $\ell(L)$ of a B-decition list L.

(base case) If L is equal to **return** y, then $\ell(L) = 1$; and

(recursion step) if L is equal to **if** f **then return** v **else** L', then $\ell(L) = \ell(L') + 1$.

Assume that $\mathrm{val}(L, x) = x$ for any $x \in [1000]$; show that $\ell(L) \geq 1000$.

9.8 Let us consider the following game modification of the game studied in this chapter. Alice have chosen a number from 1 to 1000. Bob wants to guess the number so he is asking Alice "yes" or "no" questions. However, he cannot get more than ℓ "yes" answers; i.e., as soon as Alice says ℓth "yes", Bob is supposed to be able to guess her number.

1. Show that if $\ell = 1$, then Bob needs at least 1000 quesitons.
2. Show that if $\ell = 2$, then Bob needs at least $\sqrt{1000}$ quesions.
3. Show that if Bob needs at least $\sqrt[\ell]{1000}$ questions.
4. Show that if $\ell = 2$, there is an algorithm for Bob such that he is able to guess the number using $2\left\lceil\sqrt{1000}\right\rceil$ questions.

9.9 You have a 100-story building and two eggs. When you drop an egg from any floor of the building, the egg will either break or survive the fall. If the egg survives, then it would have survived any lesser fall. If the egg breaks, then any greater fall would have broken it as well. The eggs are all identical and interchangeable. You'd like to find the minimum height that will break an egg. What is the fewest number of drops in which you are guaranteed to find the right floor?

Part II

Introduction to Combinatorial Game Theory

10. P-positions and N-positions

This part is based on Part I of "Game Theory" by Ferguson.

Combinatorial Games:
Introduction to Combinatorial Game Theory #1

https://youtu.be/DbCKHPlMN2c

In this part we use our knowledge about basics of mathematical reasoning to study games similar to checkers, chess, shogi, and tic tac toe. The games we are going to study are called combinatorial games. In these games there are two players, each know all the information, there are no chance moves, and when the game ends there is always a winner. (The last condition implies that among the aforementioned games only checkers are combinatorial since all of them allow draws; however, we may change the rules to disallow the draws and this change would make all of them combinatorial.) Such a game is determined by a set of positions, and possible moves from each position for each player. Usually, players are taking turns until they reach a position such that no moves are possible and one of the player is declared a winner.

10.1 Take-Away Game

Since chess, shogi and even tic tac toe are relatively complicated, we are going to start from much simpler example of combinatorial games.

Game 10.1 (Take-Away Game)**.** *In this game there are two players.*

- *They have a pile of* 21 *chips.*
- *They make moves in turns with player I starting, each move consists of moving one, two or three chips out of the pile.*
- *The player that removes the last chip wins.*

The question we would like to answer is whether there is a strategy for one of the players to always win. So in the rest of this part we assume that both players are playing optimally; i.e., if there is a winning strategy they follow the strategy.

To analyze this game we need the following two observations:

1. the game is symmetric and the only difference between the players is who makes the first move, and
2. if at some point the players have n chips it does not matter how they achieved this, it will not affect the rest of the game.

Using these remarks and induction (this style of induction is sometimes referred as *backward induction*) we are able to analyze the game.

Let us consider some certain states of the game. Assume that they have at most 3 chips left, in this case the player that make the move wins. However, if there are 4 chips, the player that makes the first move should always take at least 1 chip so she loses since after her turn there are at most 3 chip. Similarly, if there are 5 chips, the first player to move wins since she can take a chip and make the second player to start with 4.

So we can formulate the following conjecture. Assume that n chips left in the pile. Let r be the remainder of n modulo 4. Then if $r = 0$, the first player to move loses, otherwise, the other player loses.

Let us prove this using induction. We already proved the base case so we need to prove the induction step from n to $n + 1$.

- If $n \equiv 0 \pmod 4$, then the first player to move can remove one chip and the other player will start with n chips so by the induction hypothesis he/she loses.
- If $n \equiv 1 \pmod 4$, then the first player to move can remove two chips and the other player will start with n chips so by the induction hypothesis he/she loses.
- If $n \equiv 2 \pmod 4$, then the first player to move can remove three chips and the other player will start with n chips so by the induction hypothesis he/she loses.
- If $n \equiv 3 \pmod 4$, then after the current player moves the other player will start with either n, or $n - 1$, or $n - 2$ chips. But all these numbers have non-zero remainders modulo 4. So the other player can win in any case.

To study combinatorial games we need to give a formal definition of them.

Definition 10.1. *A game is combinatorial if*

- *there are two players,*
- *there is a set of possible positions in the game,*
- *for each position and each player, there is a fixed set of possible legal moves,*
- *players alternate moving,*
- *the game ends when no moves are possible for the player whose turn is to move.*

There are possible winning conditions, *normal play rule: the player that made the last move wins, and*

misère play rule: the player that made the last move loses.

If the game never ends, we declare a draw. If the game always ends, we say that the game satisfies the ending condition*.*

If the possible moves are the same for both players the game is called impartial *otherwise it is called* partisan*.*

Note that these games do not allow random moves, hidden information, simultaneous moves, and a draw in a finite number of steps so poker, battleships, rock-paper-scissors, and tic tac toe are not combinatorial games.

P-positions and N-positions: Introduction to Combinatorial Game Theory #2

`https://youtu.be/YV_oWBi1_ck`

Since we gave a formal definition of combinatorial games we can give a framework that allows to analyze these games.

Definition 10.2. *We say that a position in a combinatorial game is* terminal *if there are no legal moves.*

All terminal positions are P-positions*. Every position that allows for the current player to move to a P-position is an* N-position*. If all possible moves lead to N-positions, then the position is a P-position.*

For the game using the Misère rule, the definition is the same except the terminal positions are N-positions.

Using this definition, one may create the following procedure that would allow to determine which positions are P-positions and which are N-positions.

Steps necessary to find P- and N-positions

1. Label all terminal positions as P-positions.
2. If some position is not labeled but all the moves lead to labeled positions, then label the position using the definition; i.e., if there are moves leading to P-position, it is an N-position, otherwise it is a P-position.
3. If not all the positions are labeled, go to Step 2.

Note that P- and N-positions are defined recursively so in some games not all the positions are either P- or N-positions. (For example, if there are no terminal positions.) However, Theorem 12.2 proves that if the game satisfies the ending condition, then all the positions are either P- or N-positions.

0	1	2	3	4	5	6	7	8
P	N	N	N	P	N	N	N	P

Table 10.1: P-positions and N-positions for Game 10.1

So in Game 10.1 the only terminal position is 0; hence, 0 is a P-position. Similarly we can go to 0 from 1, 2, and 3 so they are N-positions. Hence, 4 is a P-position, since all the moves from 4 lead to N-positions.

Exercise 10.1. *Show that a position n is a P-position if 4 divides n, and it is an N-position otherwise.*

In other words, in this game, P-positions coincide with the positions where the current player loses. However, it is not a coincidence.

Theorem 10.1. *If some position in a combinatorial game is an N-position, then the player to move has a winning strategy if we start from this position. If the position is a P-position, then the other player has a winning strategy.*

Subtractraction Games

Let us define a big class of games that generalizes the take-away game we discussed at the beginning of the chapter.

Game 10.2. *Let $S \subseteq \mathbb{N}$ be some set. The subtraction game with the subtraction set S is the following combinatorial game. Two players start with a pile of n chips. On each move they remove $s \in S$ chips out of the pile.*

So Game 10.1 is the subtraction game with the subtraction set $\{1, 2, 3\}$.
Let us analyze the subtraction game with the subtraction set $\{1, 3, 4\}$.

0	1	2	3	4	5	6	7	8
P	N	P	N	N	N	N	P	N

Table 10.2: P-positions and N-positions for Game 10.1

Clearly 0 is a P-position since it is the only terminal position in the game. We can go to 0 from 1 so 1 is an N-position. The only possible move from 2 is to 1 so 2 is a P-position. From 3 and 4 we can go to 0 so they are N-positions. From 5 and 6 one may go to 2 so they are a N-positions as well. Hence, 7 is a P-position.

Now we may notice the pattern: n is a P-position iff $n \equiv 0 \pmod 7$ or $n \equiv 2 \pmod 7$. We prove is using induction. The base case for $n < 8$ we already proved. Let us now prove the induction step. Assume that the statement is true for all $k < n$. Consider the following cases.

1. If $n \equiv 0 \pmod 7$, the current player can move to $n - 1 \equiv 5 \pmod 7$, $n - 3 \equiv 4 \pmod 7$, or $n - 3 \equiv 5 \pmod 7$ which are all N-positions so n is a P-position.

2. If $n \equiv 1 \pmod 7$, the current player can move to $n-1$ which is a P-position so n is an N-position.

3. If $n \equiv 2 \pmod 7$, the current player can move to $n-1 \equiv 1 \pmod 7$, $n-3 \equiv 6 \pmod 7$, or $n-4 \equiv 5 \pmod 7$ which are all N-positions so n is a P-position.

4. If $n \equiv 3 \pmod 7$, the current player can move to $n-1$ which is a P-position so n is an N-position.

5. If $n \equiv 4 \pmod 7$, the current player can move to $n-4$ which is a P-position so n is an N-position.

6. If $n \equiv 5 \pmod 7$, the current player can move to $n-3$ which is a P-position so n is an N-position.

7. If $n \equiv 6 \pmod 7$, the current player can move to $n-4$ which is a P-position so n is an N-position.

End of The Chapter Exercises

10.2 Two players I and II are playing the following game.

- They start with a number 0 written on a blackboard.
- On each step one of the players replace a number n on the blackboard by either $n+1$ or by $n+2$.
- Player I makes the first move and players do moves one after another.
- The player who writes 20 wins.

Who has a winning strategy? (Note that the game is not a combinatorial game).

10.3 Two players I and II are playing the following game.

- Initially, there are 20 numbers written on a blackboard: 10 numbers 1 and 10 numbers 2.
- On each step one of the players select two numbers; and if they were the same, replace them by 2; otherwise, replace them by 1.
- Player I makes the first move and players do moves one after another.

Who is the winner? (Note that the game is not a combinatorial game).

10.4 Consider the subtraction game where players may subtract 2 and 3 chips on their turn, is 5 an N-position?

10.5 Consider the Misère subtraction game where players may subtract 1, 2 or 5 chips on their turn, identify N-positions and P-positions.

10.6 Consider the Misère subtraction game where players may subtract 1, 5 or 6 chips on their turn, identify N-positions and P-positions.

10.7 In the subtraction game where players may subtract 1, 2, or 5 chips on their turn, identify N-positions and P-positions.

10.8 Two players one by one put bishops on the chessboard such that none of the bishops attack each other. Determine the winning strategy.

10.9 Consider the following game: two players I and II are writing an 11-digit number from left to right, one digit after another. Player I wins if 7 divides the number and player II wins otherwise. Determine who is the winner if player I makes the first move.

11. The Game of Nim

The game of Nim:
Introduction to Combinatorial Game Theory #3

https://youtu.be/H-SyB0NK3H8

You can play Nim on this website

https://dotsphinx.com/games/nim/

This chapter discusses probably the most famous combinatorial game, the game of *Nim*. In this game there are several piles of chips on the table. On each turn the current player may remove some number of chips from *one* of the piles; however, the player should remove *at least one chip*. We say that a game of Nim is a k-pile game of Nim if there are k piles.

We start from analysis of the game when we have one pile of chips. It is clear that the first player to move wins since he/she may remove all the chips.

Consider a more complicated case when we have two piles of size n and m respectively. We need to consider two cases:

1. If $n = m$, then the second player to move wins. Indeed, we can use the symmetric strategy; i.e., if the first player removes s chips from one pile we also remove s chips from the other pile. It is clear that we can always make a move as long as the first player can.

2. Otherwise, the first player wins because it can move to the state with two equal piles.

The case of three piles is even more complicated. So we spend the rest of the chapter studying it.

11.1 Nim Sum

We start from a definition of the XOR operation $\oplus : \{0,1\} \times \{0,1\} \to \{0,1\}$, also known as "exclusive or"), this operation is defined as follows: $a \oplus b = 1$ iff $a \neq b$.

It is well-known that any number $n \in \mathbb{N}_0$ can be represented as a binary number ($\mathbb{N}_0$ denotes nonnegative integers); we write $n = (a_\ell, \dots, a_0)_2$ if $n = \sum_{i=0}^{\ell} a_i 2^i$. For example, $5 = 4 + 1 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = (1,0,1)_2$ and $6 = 4 + 2 = 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = (1,1,0)_2$. So we can define the Nim sum $\oplus : \mathbb{N}_0 \times \mathbb{N}_0 \to \mathbb{N}_0$, also known as bitwise xor, as follows: $(a_\ell, \dots, a_0)_2 \oplus (b_\ell, \dots, b_0)_2 = (a_\ell \oplus b_\ell, \dots, a_0 \oplus b_0)_2$. For example, $5 \oplus 6 = (1,0,1)_2 \oplus (1,1,0)_2 = (1 \oplus 1, 0 \oplus 1, 1 \oplus 0)_2 = (0,1,1)_2$.

Exercise 11.1. *Show that $a \oplus (b \oplus c) = (a \oplus b) \oplus c$ for any $a, b, c \in \mathbb{N}_0$.*

Hence, we are going to write $a \oplus b \oplus c$ instead of $a \oplus (b \oplus c)$ and $(a \oplus b) \oplus c$.

11.2 Bouton's Theorem

Now we may notice that $a \oplus b = 0$ iff $a = b$. So our result about 2-pile Nim can be rephrased: a position (a, b) in the 2-pile Nim is a P-position iff $a \oplus b = 0$. Which leads us to the next theorem.

Theorem 11.1 (Bouton)**.** *A position (a, b, c) in the 3-pile Nim is a P-position iff $a \oplus b \oplus c = 0$*

Proof. We prove the statement using structural induction. First note that the only terminal position the 3-pile Nim is $(0, 0, 0)$ and $(0, 0, 0)$ and $0 \oplus 0 \oplus 0 = 0$.

Let us consider some (a, b, c) such that $a \oplus b \oplus c \neq 0$. We need to show that there is a move from this position to a P-position. Let $a \oplus b \oplus c = (0, \ldots, 0, 1, r_{k-1}, \ldots, r_0)_2$. So among a, b, and c there is a number that has 1 in the kth position. Note that without loss of generality $a = (p_\ell, \ldots, p_{k+1}, 1, p_{k-1}, \ldots, p_0)_2$. Consider $a' = (p_\ell, \ldots, p_{k+1}, 0, p_{k-1} \oplus r_{k-1}, \ldots, r_0 \oplus p_0)_2$. It is clear that $a' < a$ and $a' \oplus b \oplus c = 0$. Hence, (a', b, c) is a P-position and therefore, (a, b, c) is an N-position.

Finally, let us consider (a, b, c) such that $a \oplus b \oplus c = 0$. Assume that there is a move to a position (a', b, c) such that $a' \oplus b \oplus c = 0$. This implies that $(a' \oplus b \oplus c) \oplus (a \oplus b \oplus c) = a \oplus a' = 0$, whence $a = a'$. $\square$

12. *Graph Games*

Graph Games:
Introduction to Combinatorial Game Theory #4

https://youtu.be/71fSkXp4DJc

This section gives an alternative definition of a combinatorial game. This definition allows us to study general combinatorial games.

Definition 12.1. *A* directed graph *G is a pair (V, N) such that V is a non-empty set and $N : V \to 2^V$.*[1]

We say that a game on G is the impartial game where elements of V are positions and each player can move from $x \in V$ to any $y \in N(x)$. (Elements of $N(x)$ are called followers of x.)

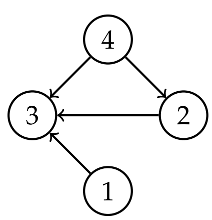
Remark 12.1. *It is also easy to see that any impartial game can be transformed into a graph G such that the game on G and the impartial game are equivalent.*

For example, the take-away game from Chapter 10 can be considered as a graph on a graph $G = (\mathbb{N}_0, N)$, where $N(0) = \varnothing$, $N(1) = \{0\}$, $N(2) = \{0, 1\}$, and $N(n+3) = \{n, n+1, n+2\}$ for any $n \in \mathbb{N}_0$.

The key ingredient for the analysis of games based on graphs was proposed by Sprague and Grundy. They proposed to consider the following function:

Definition 12.2. *Let $G = (V, N)$ be a directed graph. A function $g : V \to \mathbb{N}$ is a* Sprague–Grundy function *for G iff $g(x) = \text{mex } \{g(y) \; : \; y \in N(x)\}$, where* $\text{mex } S = \min \{n \in \mathbb{N}_0 \; : \; n \notin S\}$.

Consider the following graph (arrows depict possible moves).



Let us assume that g is a Sprague–Grundy function for this graph. Note that 3 is a terminal position so $g(3) = \text{mex } \varnothing = 0$. Since from 1 and 2 there are only moves to 3, it is clear that $g(1) = g(2) = \text{mex } \{0\} = 1$. Finally, $g(4) = \text{mex } \{0, 1\} = 2$.

Note that the Sprague–Grundy function is recursively defined so it may not exist or not to be unique if graph violates the ending condition. For example, the graph depicted on Figure 12.1a does not have

[1] We are going to have a more in-depth discussion of graphs in Part VIII.

a Sprague–Grundy function. Indeed, assume that such a function g exists. Consider two following cases.

- First case is when $g(3) = 0$. Note that $g(2) = \text{mex } \{0\} = 1$. Hence, $g(1) = \text{mex } \{1\} = 0$ which contradicts the assumption that $g(3) = 0$ since $g(3) = \text{mex } \{g(0)\} = 1$.
- Second case is when $g(3) \neq 0$. Note that $g(2) = \text{mex } \{g(3)\} = 0$. Therefore $g(1) = \text{mex } \{0\} = 1$ and $g(3) = \text{mex } \{1\} = 0$ which is a contradiction.

Note that the graph depicted on Figure 12.1b has several Sprague–Grundy functions. We may consider functions g_1 and g_2 such that $g_1(1) = g_1(3) = g_2(2) = g_2(4) = 0$ and $g_2(1) = g_2(3) = g_1(2) = g_1(4) = 1$. It is clear that they are Sprague–Grundy functions for the graph from Figure 12.1b.

(a) A graph without a Sprague–Grundy function

(b) A graph with several Sprague–Grundy functions

Figure 12.1: Graphs where Sprague–Grundy function is either not unique or does not exist.

Unfortunately, even if there are no cycles, a graph my not have a Sprague–Grundy function or have several Sprague–Grundy functions. Indeed, consider the graph $G = (\mathbb{Z}, N)$ such that $N(x) = \{x - 1\}$. It is clear that the functions g_1 and g_2 such that

$$g_1(x) = \begin{cases} 0 & \text{if } x \text{ is even} \\ 1 & \text{if } x \text{ is odd} \end{cases}$$

and

$$g_2(x) = \begin{cases} 0 & \text{if } x \text{ is even} \\ 1 & \text{if } x \text{ is odd} \end{cases}$$

are Sprague–Grundy functions for G.

Exercise 12.1. *Let $G = (\mathbb{N} \cup \{\infty\}, N)$ such that $N(x) = \{y \in \mathbb{N}_0 \ : \ y < x\}$, and $N(\infty) = \mathbb{N}$. Show that G does not have a Sprague–Grundy function.*

However, for all the combinatorial games we are going to consider the Sprague–Grundy function exists and is unique.

Theorem 12.1. *Let $G = (V, N)$ be a graph such that $N(v)$ is finite and G satisfies the ending condition. Then Sprague–Grundy function for G exists and it is unique.*

Before we prove this theorem, let us illustrate by proving that the Sprague–Grundy function for Game 10.1 is unique. Assume that a Sprague–Grundy function g for Game 10.1 exists. We are going to show that it is unique. Note that if x is a terminal position, then $g(x) = 0$. Hence, $g(0) = 0$. There is only one move from 1 so $g(1) = \text{mex } \{0\} = 1$. Similarly there are two moves from 2: one to 1 and one to 0 so $g(2) = \text{mex } \{0, 1\} = 2$. In the same way $g(3) = \text{mex } \{0, 1, 2\} = 3$ and $g(4) = \text{mex } \{1, 2, 3\} = 0$. One may notice that there is a pattern and conjecture that

$$g(x) = \begin{cases} 0 & \text{if } x \equiv 0 \pmod 4 \\ 1 & \text{if } x \equiv 1 \pmod 4 \\ 2 & \text{if } x \equiv 2 \pmod 4 \\ 3 & \text{if } x \equiv 3 \pmod 4 \end{cases}.$$

We already proved the base case, let us now prove the induction step. Assume the equality is true for all $y < x$ and consider the following cases.

- If $x \equiv 0 \pmod 4$, then $x - 1 \equiv 3 \pmod 4$, $x - 2 \equiv 2 \pmod 4$, and $x - 3 \equiv 1 \pmod 4$. Hence, $g(x) = \text{mex } \{1, 2, 3\} = 0$.

- If $x \equiv 1 \pmod 4$, similarly $g(x) = \text{mex } \{2, 3, 0\} = 1$.

- If $x \equiv 2 \pmod 4$, $g(x) = \text{mex } \{3, 0, 1\} = 2$.

- If $x \equiv 3 \pmod 4$, $g(x) = \text{mex } \{0, 1, 2\} = 3$.

It is also clear that the constructed function is indeed a Sprague–Grundy function for Game 10.1. Therefore, we proved its existence and uniqueness.

0	1	2	3	4	5	6	7	8
0	1	2	3	0	1	2	3	0

Table 12.1: The Sprague–Grundy function for Game 10.1

Note that in this proof we used a procedure very similar to the one we used to find P- and N-positions.

Steps necessary to find a Sprague–Grundy function

Assume we are trying to construct a a Sprague–Grundy function g.

1. Set $g(x) = 0$ for all terminal positions x.
2. If $g(x)$ is not defined but $g(y)$ is defined for all $y \in N(x)$, then set $g(x) = \text{mex } \{g(y) \,:\, y \in N(x)\}$.
3. If $g(x)$ is not defined for some x, go to Step 2.

Let V_0 be the set of terminal positions, and let V_i be the set of x such that we define $g(x)$ on the ith iteration of the procedure. The following lemma essentially says that this procedure defines g everywhere.

Lemma 12.1 (Kőnig). *Let $G = (V, N)$ be a graph such that $N(v)$ is finite for all $v \in V$ and G satisfies the ending condition. Let V_0 be the set of terminal positions, and let $V_{n+1} = \{v \in V \,:\, N(v) \subseteq V_n\}$. Then $V = \bigcup_{n\in\mathbb{N}_0} V_n$.*

Proof of Theorem 12.1. Let V_0 be the set of terminal positions, and let $V_{n+1} = \{v \in V \,:\, N(v) \subseteq V_n\}$. Lemma 12.1 claims that $V = \bigcup_{n\in\mathbb{N}_0} V_n$.

For each $n \in \mathbb{N}_0$, we define $g_n : V_n \to \mathbb{N}_0$ such that $g_{n+1}(v) = \text{mex } \{g_n(u) \,:\, u \in N(v)\}$ for $v \in V_{n+1}$ and $g_0(v) = 0$ for $v \in V_0$. It is easy to see that the function $g : V \to \mathbb{N}_0$ such that $g(v) = g_n(v)$ for $v \in V_n$ is a Sprague–Grundy function for G.

To finish the proof we need to prove that there are no other Sprague–Grundy functions for G. Assume that g' is a Sprague–Grundy function for G; we prove using induction by $n \in \mathbb{N}_0$ that $g(v) = g'(v)$ for $v \in V_n$. The base case for $v = 0$ is clear since $N(v) = \varnothing$ for $v \in V_0$. Let us consider the induction step from n to $n + 1$. By the induction hypothesis, $g(u) = g'(u)$ for $u \in V_n$. Let us consider some $v \in V_{n+1}$. Note that

$$g'(v) = \text{mex } \{g(u) \,:\, u \in N(v)\} = \text{mex } \{g(u) \,:\, u \in N(v)\} = g(v).$$

Therefore $g(v) = g'(v)$ for $v \in V_{n+1}$. As a result, $g(v) = g'(v)$ for $v \in V$; i.e., $g = g'$. $\square$

One may note that in Game 10.1 P-positions are the positions where the Sprague–Grundy function is zero. In fact, this is not a coincidence.

Theorem 12.2. *Let $G = (V, N)$ be a graph such that $N(v)$ is finite for all $v \in V$ and G satisfies the ending condition. Then all the vertices of G can be labeled as either P- or N-positions. Moreover, $v \in V$ is a P-position iff $g(v) = 0$, where g is the Sprague–Grundy function for G.*

End of The Chapter Exercises

12.2 Prove Theorem 12.2.

12.3 Show that there are only two Sprague–Grundy functions for the graph depicted on Figure 12.1b.

12.4 Prove that there is unique Sprague–Grundy function for the one pile Nim game.

12.5 Prove that there is unique Sprague–Grundy function for the subtraction game where players may subtract 2 and 3 chips on their turn.

12.6 Prove that there is unique Sprague–Grundy function for the subtraction game where players may subtract 1, 2, or 5 chips on their turn.

13. Sums of Combinatorial Games

Sum of Games:
Introduction to Combinatorial Game Theory #5

https://youtu.be/dRaqJKZh3y0

Let us consider the following game.

Game 13.1 (Take-Away Game)**.** *In this game there are two players.*

- *They have two piles of n and m chips, respectively.*
- *They make moves in turns with player I starting, each move consists of moving one or two chips out of the pile.*
- *The player that removes the last chip wins.*

It is not hard to draw the graph corresponding to the game for small n and m. Using this graph it is easy to see that all drawn positions,

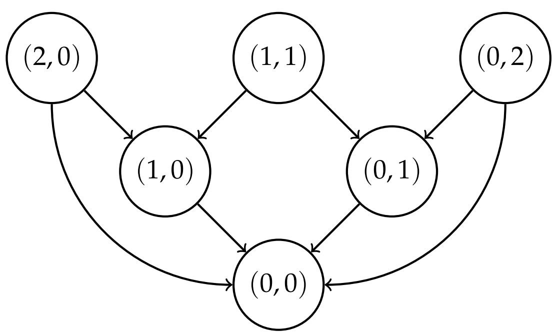


Figure 13.1: Part of the graph of Game 13.1

except $(1,1)$ and $(0,0)$ are N-positions.

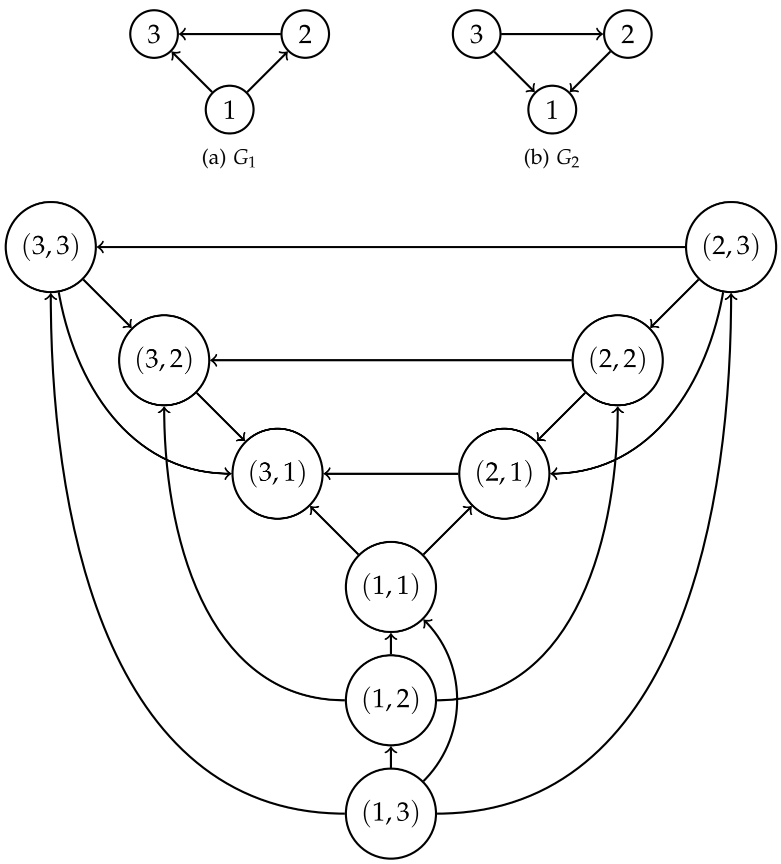
In the rest of the chapter we will discuss a method to study similar games. Assume we have two combinatorial games $\mathcal{G}_1$ and $\mathcal{G}_2$. One may form another game played as follows: the initial position of the new game consists of the pair of initial positions of $\mathcal{G}_1$ and $\mathcal{G}_2$, players alternate moves, and on each turn a player make a move in one of the game leaving the position in the second untouched. The new game is called *sum of $\mathcal{G}_1$ and $\mathcal{G}_2$*.

Let us give a formal definition.

Definition 13.1. *Let $G_1 = (V_1, F_1)$ and $G_2 = (V_1, F_2)$ be directed graphs. We say that G is the sum of G_1 and G_2, denoted $G_1 + G_2$, is a graph $(V_1 \times V_2, F)$ such that*

$$F(x_1, x_2) = \{(y_1, x_2) \ : \ y_1 \in F_1(x_1)\} \cup \{(x_1, y_2) \ : \ y_2 \in F_2(x_2)\}.$$

Figure 13.2c gives an example of this operation.



(a) G_1 (b) G_2

(c) $G_1 + G_2$

Figure 13.2: Figure 13.2c depicts sum of graphs from Figure 13.2a and Figure 13.2b.

Another example is given by the game of Nim; it is easy to see that 2-pile Nim is a sum of two 1-pile Nims. This observation leads to a generalization of Bouton's Theorem (Theorem 11.1).

Theorem 13.1 (The Sprague–Grundy Theorem)**.** *Let G_1 and G_2 be some graphs and g_1 and g_2 be corresponding Sprague–Grundy functions. Then the graph G_1 and G_2 has a Sprague–Grundy function g such that $g(x_1, x_2) = g_1(x_1) \oplus g_2(x_2)$.*

Proof. Let $G_1 = (V_1, F_1)$, $G_2 = (V_2, F_2)$, and $G = G_1 + G_2$. Consider some $x_1 \in V_1$ and $x_2 \in V_2$. Let $a = g_1(x_1) \oplus g_2(x_2)$. To prove the statement we need to show that

1. for any $0 \leq b < a$, there is $(y_1, y_2) \in F(x_1, x_2)$ such that $g(y_1, y_2) = b$;
2. for any $(y_1, y_2) \in F(x_1, x_2)$, $g(y_1, y_2) \neq a$.

We start from proving the first statement. Let us fix some $0 \leq b < a$ and let $c = a \oplus b$. Let $g_i(x_i) = (p_{i,\ell}, \ldots, p_{i,0})$ for each $i \{1, 2\}$

and $c = (1, q_{k-1}, \ldots, q_0)$ where $k \leq \ell$. For some $j \in \{1,2\}$, $p_{j,k} = 1$ since $a = g_1(x_1) \oplus g_2(x_2)$. Without loss of generality $j = 1$. Hence, $c \oplus g_1(x_1) < g_1(x_1)$, whence there is x_1' such that $g_1(x_1') = c \oplus g_1(x_1)$. As a result, there is a move in G from (x_1, x_2) to (x_1', x_2) and $g(x_1', x_2) = g_1(x_1') \oplus g_2(x_2) = c \oplus g_1(x_1) \oplus g_2(x_2) = c \oplus a = b$.

To prove the second statement, assume that there is $(y_1, y_2) \in F(x_1, x_2)$ so that $g(y_1, y_2) = a$. Without loss of generality we may assume that $x_2 = y_2$. Hence, $0 = g(y_1, x_2) \oplus g(x_1, x_2) = g_1(y_1) \oplus g_1(x_1)$. However, $g_1(y_1) \neq g_1(x_1)$ since there is a move from x_1 to y_1. Therefore $g_1(y_1) \oplus g_1(x_1) \neq 0$ which is a contradiction. □

It is also easy to see that if G_1 and G_2 satisfy the ending condition, then $G_1 + G_2$ also satisfies the ending condition. Therefore, if G_1 and G_2 satisfy the ending condition and g_1, g_2 are Sprague–Grundy functions of them, $G_1 + G_2$ has unique Sprague–Grundy function g such that $g(x_1, x_2) = g_1(x_1) \oplus g_2(x_2)$.

The simple example of an application of this theorem is the analysis of the following game.

Game 13.2. *Alice and Bob have two piles with* 10 *and* 11 *chips respectively. They take turns and remove* 1 *or* 2 *chips from one of the piles. If one of them cannot make a move he/she loses.*

To determine who is the winner in this game, we start with a subtraction game G with the subtraction set $\{1,2\}$. It is easy to see that $g : \mathbb{N}_0 \to \mathbb{N}_0$ such that

$$g(x) = \begin{cases} 0 & \text{if } x \equiv 0 \pmod 3 \\ 1 & \text{if } x \equiv 1 \pmod 3 \\ 2 & \text{if } x \equiv 2 \pmod 3 \end{cases}$$

is the Sprague–Grundy function for G. It is also clear that Game 13.2 is equal to $G + G$. Therefore the function $g' : \mathbb{N}_0^2 \to \mathbb{N}_0$ such that

$$g(x,y) = \begin{cases} 0 & \text{if } x \equiv 0 \pmod 3 \text{ and } y \equiv 0 \pmod 3 \\ 1 & \text{if } x \equiv 0 \pmod 3 \text{ and } y \equiv 1 \pmod 3 \\ 2 & \text{if } x \equiv 0 \pmod 3 \text{ and } y \equiv 2 \pmod 3 \\ 1 & \text{if } x \equiv 1 \pmod 3 \text{ and } y \equiv 0 \pmod 3 \\ 0 & \text{if } x \equiv 1 \pmod 3 \text{ and } y \equiv 1 \pmod 3 \\ 3 & \text{if } x \equiv 1 \pmod 3 \text{ and } y \equiv 2 \pmod 3 \\ 2 & \text{if } x \equiv 2 \pmod 3 \text{ and } y \equiv 0 \pmod 3 \\ 3 & \text{if } x \equiv 2 \pmod 3 \text{ and } y \equiv 1 \pmod 3 \\ 0 & \text{if } x \equiv 2 \pmod 3 \text{ and } y \equiv 2 \pmod 3 \end{cases}$$

Therefore, the position $(10, 11)$ is an N-position.

Applications of the Sprague-Grundy Theorem:
Introduction to Combinatorial Game Theory #6

https://youtu.be/V4yI_1P1Jcc

A surprising example of the application of this theorem is the following game.

Game 13.3. *In this game position is described by a polygon with several diagonals. The game starts with a polygon with n sides. On each turn a player draw a new diagonal so that it does not intersect with previously drawn diagonals. Players take turns and the one who cannot make a move loses.*

It is easy to see that we do not care about the shape of the polygon and diagonals, the only important information is the number of nodes and which nodes are connected by a diagonal.

Let $g(n)$ be the value of the Sprague–Grundy function at the polygon with n sides. It is easy to see that if we split the polygon, by a diagonal, into two parts with ℓ and m sides, then the resulting position is essentially a position in the sum of two games; hence, $g(n) = \text{mex } \{g(\ell) \oplus g(m) \ : \ \ell, m \geq 3 \text{ and } \ell + m = n + 2\}$. Using this observation it is easy to compute the value of $g(n)$ for small n. It is easy to see to make a conjecture that $g(n) = 0$ for odd n and $g(n) = 1$ for even n. Let us prove this using induction. The base case follows from the computation necessary to write the table. Let us prove the induction step from $1, \ldots, n$ to $n + 1$.

3	4	5	6	7	8
0	1	0	1	0	1

Table 13.1: Sprague–Grundy function for Game 13.3

- Let $n + 1$ be even. It is clear that if $\ell + m = (n + 1) + 2$, then ℓ and m have the same remainder modulo 2. Therefore, $g(m) = g(\ell)$ by the induction hypothesis. As a result,

  $$\{g(\ell) \oplus g(m) \ : \ \ell, m \geq 3 \text{ and } \ell + m = n + 2\} = \{0\}$$

  and $g(n + 1) = 1$.

- Let $n + 1$ be odd. It is clear that if $\ell + m = (n + 1) + 2$, then ℓ and m have different remainders modulo 2. Therefore, $g(m) \neq g(\ell)$ by the induction hypothesis. As a result,

  $$\{g(\ell) \oplus g(m) \ : \ \ell, m \geq 3 \text{ and } \ell + m = n + 2\} = \{1\}$$

  and $g(n + 1) = 0$.

End of The Chapter Exercises

13.1 Compute the Sprague–Grundy function for states of the subtraction game with two piles of chips where players and the subtraction set $\{1, 2, 5\}$.

13.2 Let G_1 be the subtraction game with the subtraction set $\{1, 2\}$ Let G_2 be the game of Nim with three piles. Find all the moves from $(11, (1, 6, 7))$ to P-positions in $G1 + G2$.

Part III

Introduction to Discrete Probability

14. Sample Spaces and Events

In the previous chapter we discussed games that do not involve move by chance. A theory that studies experiments, processes and interactions that are subject to chance is called *probability theory*; we are going to study this theory in this part.

The most important assumption of probability theory is that nonetheless the outcome of an experiment is not known in advance — the set of all possible outcomes is known. This set is called the *sample space* or the *probability space*.

For example, if our experiment consists of tossing a coin, then the sample space consists of two outcomes $\{H, T\}$, where H stands for heads and T stands for tails.

Exercise 14.1. *Write the sample space for the experiment consisting of tossing two coins.*

Each element of the sample space is called an *outcome* or an *elementary event*. Typically, we are interested in observing several outcomes; e.g., in the experiment consisting of tossing five coins, the set $\{HTTTT, THTTT, TTHTT, TTTHT, TTTTH\}$ describe all possible outcomes when exactly one coin shows heads. We say that a set of outcomes is an *event*.

Another assumption of probability theory is that every outcome ω of a sample space Ω is assigned some probability $\mathcal{D}(\omega)$; intuitively, $\mathcal{D}(\omega)$ is the likelihood that the outcome ω occurs in the experiment. It is convenient to normalize probabilities, so we require that $0 \leq \mathcal{D}(\omega) \leq 1$ for all $\omega \in \Omega$ and the sum of $\mathcal{D}(\omega)$ for $\omega \in \Omega$ is equal to 1. (In the sequel, we use $\sum_{x \in X} f(x)$ to describe the sum of $f(x)$ for $x \in X$; see Section 24.2 for the formal definition.) The function $\mathcal{D}$ is called a *probability distribution* on Ω.

The pair of a sample space and a probability distribution on the space is called a *finite discrete probability space*.

We can extend the notion of probability from elementary events to all events as follows. Let $E \subseteq \Omega$ be an event in the finite discrete probability space $(\Omega, \mathcal{D})$. Then $\Pr_{\mathcal{D}}(E)$ denotes $\sum_{\omega \in E} \mathcal{D}(\omega)$.

Exercise 14.2. *In many cases we consider* uniform distribution *on a set Ω, the distribution $\mathcal{U}_\Omega$ such that all the outcomes are equally likely. Let $\Omega =$*

$\{HH, HT, TH, TT\}$ and $\mathcal{U}_\Omega$ be the uniform distribution on Ω. Find the probability of the event $\{HT, TH, TT\}$, and give an informal interpretation of the answer.

14.1 Basic Principles

Events are subsets of the sample space; hence, they can be combined using the standard set operations. So it is natural to ask whether the probabilities $\Pr_{\mathcal{D}}(A \cup B)$ and $\Pr_{\mathcal{D}}(A \cap B)$ can be expressed in terms of $\Pr_{\mathcal{D}}(A)$ and $\Pr_{\mathcal{D}}(B)$.

Theorem 14.1 (The Additive Principle)**.** *Let $(\Omega, \mathcal{D})$ be a finite discrete probability space, and let $A, B \subseteq \Omega$ be two disjoint events. Then $\Pr_{\mathcal{D}}(A \cup B) = \Pr_{\mathcal{D}}(A) + \Pr_{\mathcal{D}}(B)$.*

Proof Sketch, see Theorem 18.1. Let $A = \{\omega_{1,1}, \ldots, \omega_{1,k}\}$ and let $B = \{\omega_{2,1}, \ldots, \omega_{2,\ell}\}$. Then $A \cup B = \{\omega_{1,1}, \ldots, \omega_{1,k}, \omega_{2,1}, \ldots, \omega_{2,\ell}\}$. Therefore $\Pr_{\mathcal{D}}(A \cup B) = \Pr_{\mathcal{D}}(\omega_{1,1}) + \cdots + \Pr_{\mathcal{D}}(\omega_{2,k}) + \Pr_{\mathcal{D}}(\omega_{2,1}) + \cdots + \Pr_{\mathcal{D}}(\omega_{2,\ell}) = \Pr_{\mathcal{D}}(A) + \Pr_{\mathcal{D}}(B)$. □

Exercise 14.3. *Let $(\Omega, \mathcal{D})$ be a finite discrete probability space, and let $A \subseteq \Omega$ be an event. Show that $\Pr_{\mathcal{D}}(\Omega \setminus A) = 1 - \Pr_{\mathcal{D}}(A)$.*

This result can be easily extended to the cases when A and B are not disjoint.

Corollary 14.1 (The Inclusion-exclusion Principle)**.** *Let $(\Omega, \Pr)$ be a finite discrete probability space, and let $A, B \subseteq \Omega$ be two events. Then $\Pr_{\mathcal{D}}(A \cup B) = \Pr_{\mathcal{D}}(A) + \Pr_{\mathcal{D}}(B) - \Pr_{\mathcal{D}}(A \cap B)$.*

Unfortunately, $\Pr_{\mathcal{D}}(A \cap B)$ cannot be expressed via $\Pr_{\mathcal{D}}(A)$ and $\Pr_{\mathcal{D}}(B)$. However, in many cases $\Pr_{\mathcal{D}}(A \cap B) = \Pr_{\mathcal{D}}(A) \Pr_{\mathcal{D}}(B)$; if this equality holds, we say that A and B are *independent*.

For example, let us consider an experiment where we toss two fair coins; i.e., let us consider $\Omega = \{HH, HT, TH, TT\}$ and let $\mathcal{U}_\Omega$ be the uniform distribution on Ω. It is easy to see that $\Pr_{\mathcal{U}_\Omega}(\{HH, HT\}) = 1/2$, $\Pr_{\mathcal{U}_\Omega}(\{HH, TH\}) = 1/2$, and $\Pr_{\mathcal{U}_\Omega}(\{HH, HT\} \cap \{HH, TH\}) = \Pr_{\mathcal{U}_\Omega}(\{HH\}) = 1/4$. Hence, these two events are independent.

To analyze experiments consisting of tossing several coins, we need to be able to study products of finite discrete probability spaces.

Theorem 14.2 (The Multiplicative Principle)**.** *Let $\Omega = \Omega_1 \times \Omega_2$ and let $(\Omega_1, \mathcal{D}_1)$ and $(\Omega_2, \mathcal{D}_2)$ be finite discrete probability spaces. Then $\mathcal{D} : \Omega \to \mathbb{R}$ such that $\mathcal{D}(\omega_1, \omega_2) = \Pr_{\mathcal{D}_1}(\omega_1) \cdot \Pr_{\mathcal{D}_2}(\omega_2)$ is a probability distribution on Ω. Moreover, $\Pr_{\mathcal{D}}(E_1 \times E_2) = \Pr_{\mathcal{D}_1}(E_1) \cdot \Pr_{\mathcal{D}_2}(E_2)$ for all $E_1 \subseteq \Omega_1$ and $E_2 \subseteq \Omega_2$.*

Using this principle we can show that in the experiment consisting of tossing five coins, the event where the first flip is H and the event where the second flip is H are independent. Indeed, let $\Omega = \{H, T\}^5$; Pr be the uniform distribution on Ω; and $E_1 = \left\{t \in \{H, T\}^5 \ : \ t_1 = H\right\}$ and $E_2 = \left\{t \in \{H, T\}^5 \ : \ t_2 = H\right\}$. By Theorem 14.2, $\Pr(E_1) = \Pr(E_2) = 1/2$. Moreover, $E_1 \cap E_2 = \left\{t \in \{H, T\}^5 \ : \ t_1 = H \text{ and } t_2 = H\right\}$; therefore, $\Pr_{\mathcal{D}}(E_1 \cap E_2) = \frac{1}{4}$.

14.2 Random Variables

Sometimes we are more interested in some function of the result of the experiment rather than the result itself. For example, Sasha may play Dungeons and Dragons and be interested in his chances to roll 7 on two dice together. Let us formalize the question. Let $\Omega = [6]^2$ and Pr be a the uniform distribution on Ω. Sasha is interested in the probability of the event $\left\{(x, y) \in [6]^2 \ : \ x + y = 7\right\}$.

More generally, let $(\Omega, \mathcal{D})$ be a finite discrete probability space. Then a function $\chi : \Omega \to \mathbb{R}$ is called a *random variable* and $\Pr_{\mathcal{D}}(\chi = a)$ denotes $\Pr_{\mathcal{D}}(\{\omega \in \Omega \ : \ \chi(\omega) = a\})$.

In the example about Dungeons and Dragons, $\chi(x, y) = x + y$ and we are interested in $\Pr_{\mathcal{D}}(\chi = 7) = \Pr_{\mathcal{D}}(\{(1,6), (2,5), \ldots, (6,1)\} = 1/6$.

Exercise 14.4. *Let $\Omega = [6]^2$ and $\mathcal{U}_\Omega$ be the uniform distribution on Ω. Let $\chi : \Omega \to \mathbb{R}$ be the random variable such that $\chi(x, y) = x + y$. Find $\Pr_{\mathcal{U}_\Omega}(\chi = 1), \ldots, \Pr_{\mathcal{U}_\Omega}(\chi = 12)$.*

We are going to adopt some simple additional notation, if $\chi_1, \chi_2 : \Omega \to \mathbb{R}$ are random variables then $(\chi_1 + \chi_2), (\chi_1 \cdot \chi_2) : \Omega \to \mathbb{R}$ are the random variables such that $(\chi_1 + \chi_2)(\omega) = \chi_1(\omega) + \chi_2(\omega)$ and $(\chi_1 \cdot \chi_2)(\omega) = \chi_1(\omega) \cdot \chi_2(\omega)$.

End of The Chapter Exercises

14.5 Prove Corollary 14.1.

14.6 Let $\Omega = \{HH, HT, TH, TT\}$ and let $\mathcal{U}_\Omega$ be the uniform distribution on Ω. Show that $\{HH\}$ and $\{TT\}$ are not independent.

14.7 Alice is rolling a dice n times, compute the probability that Alice sees 6, 6, and 6 in three consecutive rolls.

14.8 Alice is rolling a dice n times, compute the probability that Alice sees 4, 5, and 6 in three consecutive rolls.

15. Conditional Probability

In general, the occurrence of an event B changes the probability that another event A occurs, $\Pr_{\mathcal{D}}(A \mid B)$ denotes the latter probability. More formally, $\Pr(A \mid B) = \Pr(A \cap B) / \Pr(B)$, where $(\Omega, \mathcal{D})$ is a finite discrete probability space, $A, B \subseteq \Omega$, and $\Pr_{\mathcal{D}}(B) \neq 0$. We say that $\Pr_{\mathcal{D}}(A \mid B)$ is the conditional probability that A occurs given that B occurs.

For example, let us consider $\Omega = [6]^2$ and the uniform distribution $\mathcal{U}_\Omega$ on Ω; i.e., we consider an experiment consisting of rolling two dice. Let us compute the probability that the sum of numbers on the dice exceeds 6 given that the first dice's number is 3. In other words we need to compute $\Pr_{\mathcal{U}_\Omega}(A \mid B)$, where $A = \{(i,j) \in [6]^2 \; : \; i + j > 6\}$ and $B = \{(3,j) \; : \; j \in [6]\}$. It is clear that $\Pr_{\mathcal{U}_\Omega}(B) = 1/6$ and $\Pr_{\mathcal{U}_\Omega}(A \cap B) = \{(3,4), (3,5), (3,6)\} = 1/12$. Hence, $\Pr_{\mathcal{U}_\Omega}(A \mid B) = 1/2$.

Exercise 15.1. *A family has two children. What is the probability that both are boys, given at least one is a boy? What if it is given that* the first child *is a boy. (You assume that the probability distribution of families is uniform.)*

Let us consider another example known as "Monty Hall Problem". On the television game *Let's make a deal*, a contestant is presented with a choice of three closed doors. Behind exactly one door is a prize; the other doors conceal cheap items. First, the contestant is asked to choose a door. Then Monty Hall, the host of the show, shows the contestant one of the worthless prizes behind one of the other doors. At this point, there are two closed doors, and the contestant is given the opportunity to switch from his original choice to the other closed door. The question is, is it better for the contestant to stick to his original choice or to switch doors?

Let us analyze this question using the conditional probabilities. Without loss of generality, we may assume that the contestant chooses door 1. Note that the sample space is equal to $\{(1,2), (1,3), (2,3), (3,2)\}$, where the first number denotes the door with the prize and the second number denotes the door opened by the host. The probability distribution is equal to

$$\mathcal{D}(x) = \begin{cases} 1/6 & \text{if } x = (1,2) \\ 1/6 & \text{if } x = (1,3) \\ 1/3 & \text{if } x = (2,3) \\ 1/3 & \text{if } x = (3,2) \end{cases}$$

since in the first two cases Monty has two possible choices to show a door without the prize. Suppose the host revealed the door number 2 (the probability of this is $3/6$). Then the probability that we win the price if we stick to the original choice is $(1/6)/(3/6) = 1/3$. However, the probability to win the prize in case of us swithcing the door is $(1/3)/(3/6) = 2/3$. Which implies, paradoxically, that it is beneficial to switch the door!

Theorem 15.1 (Bayes' Rules)**.** *Let $(\Omega, \mathcal{D})$ be a finite discrete probability space.*

- *Let $A, B \subseteq \Omega$ be two events such that $\Pr_{\mathcal{D}}(A) > 0$ and $\Pr_{\mathcal{D}}(B) > 0$. Then $\Pr_{\mathcal{D}}(A \mid B) = \frac{\Pr_{\mathcal{D}}(B \mid A) \Pr_{\mathcal{D}}(B)}{\Pr_{\mathcal{D}}(A)}$.*
- *Let $A, B \subseteq \Omega$ be two events such that $\Pr_{\mathcal{D}}(A) < 1$ and $\Pr_{\mathcal{D}}(B) < 1$; i.e., $\Pr_{\mathcal{D}}(\overline{A}) > 0$ and $\Pr_{\mathcal{D}}(\overline{B}) > 0$, where $\overline{A} = \Omega \setminus A$ and $\overline{B} = \Omega \setminus B$. Then $\Pr_{\mathcal{D}}(A) = \Pr_{\mathcal{D}}(A \mid B) \Pr_{\mathcal{D}}(B) + \Pr_{\mathcal{D}}(A \mid \overline{B}) \Pr_{\mathcal{D}}(\overline{B})$.*

Usefulness of this result can be illustrated with the following example. Assume that there is a rare disease that has the property that if a patient is affected by the disease, then the test is positive in 99% of the cases. However, it happens in 2% of the cases that a healthy patient tests positive. Statistical data shows that one person out of 1000 has the disease. What is the probability for a patient with a positive test to be affected by the disease?

Let $(\Omega, \mathcal{D})$ be a finite discrete probability space from this problem. Let S be the event that the patient has the disease, and P and N the events that the test is positive or negative. We know that $\Pr_{\mathcal{D}}(S) = 0.001$, $\Pr_{\mathcal{D}}(P \mid S) = 0.99$, and $\Pr_{\mathcal{D}}(P \mid S) = 0.02$, where $\overline{S}$ is the event that the patient does not have the desease. Therefore $\Pr_{\mathcal{D}}(S \mid P) = \frac{\Pr_{\mathcal{D}}(P \mid S) \Pr_{\mathcal{D}}(S)}{\Pr_{\mathcal{D}}(P)}$ and $\Pr_{\mathcal{D}}(P) = \Pr_{\mathcal{D}}(P \mid S) \Pr_{\mathcal{D}}(S) + \Pr_{\mathcal{D}}(P \mid \overline{S}) \Pr_{\mathcal{D}}(\overline{S})$. As a result $\Pr_{\mathcal{D}}(S \mid P) = \frac{0.99 \cdot 0.001}{0.99 \cdot 0.001 + 0.02 \cdot 0.999} \approx \frac{1}{20}$.

16. *Expectation of a Random Variable*

To understand the behaviour of a random variable one may try to use the average value; there are several ways to define the average value, but we are going to concentrate on the definition based on the mean. Given a random variable X in the finite discrete probability space $(\Omega, \mathcal{D})$, the expected value $\mathbb{E}_{\mathcal{D}}[X]$ (or $\mathbb{E}_{\omega \leftarrow \mathcal{D}}[X(\omega)]$) of χ is equal to $\sum_{\omega \in \Omega} \Pr_{\mathcal{D}}(\omega) X(\omega)$.

Note that it is possible that there are several ω's with the same $X(\omega)$. Hence, one may give an alternative definition of the expectation.

Theorem 16.1. *Let $(\Omega, \mathcal{D})$ be a finite discrete probability space, and let X be a random variable in the probability space. Then $\mathbb{E}_{\mathcal{D}}[X] = \sum_{a \in \operatorname{Im} X} a \Pr_{\mathcal{D}}(X = a)$.*

An important property of the expectation that often allows simplifications in computing the expected value of a random variable is its linearity.

Theorem 16.2. *Let $(\Omega, \mathcal{D})$ be a finite discrete probability space; X and Y be random variables; and let $\lambda \in \mathbb{R}$. Then $\mathbb{E}_{\mathcal{D}}[X+Y] = \mathbb{E}_{\mathcal{D}}[X] + \mathbb{E}_{\mathcal{D}}[Y]$ and $\mathbb{E}_{\mathcal{D}}[\lambda X] = \lambda \mathbb{E}_{\mathcal{D}}[X]$.*

Let us give a simple example showing that this theorem can help to compute the expected value of a random variable. Consider an experiment consisting of tossing n standard coins; i.e. consider the finite discrete probability space $(\Omega, \Pr)$ such that $\Omega = \{H, T\}^n$ and $\mathcal{U}_\Omega$ is the uniform distribution on Ω. We would like to find the expected number X of heads in the experiment. Let X_i be the random variable that is equal to 1 if the ith flip yields heads, otherwise it is equal to 0. It is clear that $X = \sum_{i=1}^n X_i$. Hence, $\mathbb{E}_{\mathcal{U}_\Omega}[X] = \sum_{i=1}^n \mathbb{E}_{\mathcal{U}_\Omega}[X_i]$. However, $\mathbb{E}_{\mathcal{U}_\Omega}[X_i] = 1/2$ which implies that $\mathbb{E}_{\mathcal{U}_\Omega}[X] = n/2$.

Let us use this knowledge to study a game similar to the game discussed in Chapter 9. Alice selected 500 numbers from 1 to 1000 and Bob would like to guess at least one of them. How many questions Bob need to ask to do this?

Apparently, the situation is drastically different if Bob's algorithm can be randomized and if it cannot be randomized. To show this we need to extend the definition of B-decision trees so that they can operate not only with integers.

Definition 16.1. *Let X and Y be some sets. We say that T is a B-decision tree if*

(base case) either T is equal to **return** y *for $y \in Y$, or*

(recursion step) T is equal to **if** f **then** T_0 **else** T_1, *where $f : X \to \{0, 1\}$, and T_0 and T_1 are B-decision trees.*

The number of queries $\mathrm{h}(T, x)$ of T at $x \in X$ can be defined as follows.

(base case) Let T be equal to **return** y, *wherey $\in Y$. Then $\mathrm{h}(T, x) = 0$ for all $x \in X$.*

(recursion step) Let T be equal to **if** f **then** T_0 **else** T_1. *Then*

$$\mathrm{h}(T, x) = \begin{cases} \mathrm{h}(T_0, x) + 1 & \text{if } f(x) = 0 \\ \mathrm{h}(T_1, x) + 1 & \text{otherwise} \end{cases}$$

The value $\mathrm{val}(T, x)$ of a B-decision tree T at $x \in X$ can be defined as follows.

(base case) Let T be equal to **return** y, *where $y \in Y$. Then $\mathrm{val}(T, x) = y$.*

(recursion step) Let T be equal to **if** f **then** T_0 **else** T_1. *Then*

$$\mathrm{val}(T, x) = \begin{cases} \mathrm{val}(T_0, x) & \text{if } f(x) = 0 \\ \mathrm{val}(T_1, x) & \text{otherwise} \end{cases}.$$

Theorem 16.3. *Let $\binom{[1000]}{500}$ denote the set of subsets of $[1000]$ with 500 elements.*[1]

- *Let T be a B-decision tree such that $\mathrm{val}(T, S) \in S$ for all $S \in \binom{[1000]}{500}$. Then $\mathrm{h}(T) \geq 9$.*
- *There are a set Ω of B-decision trees and a probability distribution $\mathcal{D}$ on Ω such that $\mathrm{val}(T, S) \in S$ for all $S \in \binom{[1000]}{500}$ and $T \in \Omega$, but $\mathbb{E}_{\mathcal{D}}[\mathrm{val}(T, S)] \leq 2$ for all $S \in \binom{[1000]}{500}$.*

[1] We are going to discuss such sets in Chapter 20.

Proof. We prove only the second part of the statement, the first part can be proven similarly to Theorem 9.2. To prove this statement let us consider the following algorithm.

- Choose $x_1, \ldots, x_n \in [1000]$ uniformly at random;
- Set $i = 1$;
- Query whether $x_i \in S$ or not.

- If yes, the output is x_i, otherwise increase i.
- If $i \leq n$ go to step 3, otherwise go to step 6.
- Bruteforce all the numbers from [1000] and check whether they belong to S or not.

Let us compute the expected number of queries made by this algorithm. It is clear that the probability that the algorithm gets yes for $i = k$ is equal to $\left(\frac{1}{2}\right)^k$. Hence, the expected number of queries is equal to

$$\left(\frac{1}{2}\right)^n 1000 + \sum_{k=1}^{n} \left(\frac{1}{2}\right)^k k.$$

The following claim gives the formula that allows to compute this sum.[2]

[2] We are going to discuss a method to guess such formulas in Chapter 20.

Claim 16.3.1. *For any $k \in \mathbb{N}$, $\sum_{k=1}^{n} \left(\frac{1}{2}\right)^k k = 2 - \frac{n+2}{2^n}$.*

Therefore, the average number of queries is at most $2 - \frac{n+998}{2^n} \leq 2$ for $n \geq 10$.

It is also clear that the result of this algorithm is always correct.

To finish the proof we need to prove Claim 16.3.1. We prove it using induction by n. It is clear that the statement is true for $n = 0$. Assume the statement is true for n; i.e., $\sum_{k=1}^{n} \left(\frac{1}{2}\right)^k k = 2^{-n}(-n + 2^{n+1} - 2)$. This implies that

$$\sum_{k=1}^{n+1} \left(\frac{1}{2}\right)^k k = 2 - \frac{n+2}{2^n} + \frac{n+1}{2^{n+1}} =$$
$$2 - \frac{2n+4-n-1}{2^{n+1}} = 2 - \frac{(n+1)+2}{2^{n+1}}.$$

In other words, the statement is true for $n + 1$. Hence, by the induction principle, we proved the statement for all n. □

End of The Chapter Exercises

16.1 Prove the first part of Theorem 16.3.

Part IV

Introduction to Combinatorics

17. Bijections, Surjections, and Injections

Bijections, Surjections, and Injections: Introduction to Combinatorics #1

https://youtu.be/fW5Zxg0TMDc

Combinatorics is an area of mathematics primarily concerned with counting; hence, the questions studied in combinatorics are usually forulated as results about the sizes of sets. This chapter uses informal notion of size of a set, for the formal definition see Chapter 24.

17.1 Bijections

The simplest way to explain that one set has the same number of elements as another is to show a correspondence between elements of these sets. For example, in order to explain that the set $\{0, \pi, 1/4\}$ has the same number of elements as $\{1, 2, 3\}$ we may just say that 0 corresponds to 1, π corresponds to 2, and $1/4$ corresponds to 3. More formally such a correspondence is defined using the following definition.

Definition 17.1. *Let $f : X \to Y$ be a function. We say that f is a bijection iff the following properties are satisfied.*

- *Every element of Y is an image of some element of X. In other words,*

$$\forall y \in Y \; \exists x \in X \; f(x) = y.$$

- *Images of any two elements of X are different. In other words,*

$$\forall x_1, x_2 \in X \; f(x_1) \neq f(x_2).$$

Let us consider the following example. Let $f : \mathbb{R} \to \mathbb{R}$ be a function such that $f(x) = x + 1$; Note that it is a bijection:

- For any $y \in \mathbb{R}$, $f(y - 1) = (y - 1) + 1 = y$.
- If $f(x_1) = f(x_2)$, then $x_1 + 1 = x_2 + 1$ i.e. $x_1 = x_2$.

Exercise 17.1. *Show that x^3 is a bijection.*

One of the nicest properties of bijections is that composition of two bijections is a bijection.

Theorem 17.1. *Let X, Y, and Z be some sets and $f : X \to Y$ and $g : Y \to Z$ be bijections. Then $(g \circ f) : X \to Z$ is also a bijection.*

Proof. We need to check two properties.

- Let $x_1 \neq x_2 \in X$. Note that $f(x_1) \neq f(x_2)$ since f is a bijection. Hence, $g(f(x_1)) \neq g(f(x_2))$ since g is a bijection as well. As a result, $(g \circ f)(x_1) \neq (g \circ f)(x_2)$.
- Let $z \in Z$; we need to find $x \in X$ such that $(g \circ f)(x) = z$. Note that since g is a bijection there is $y \in Y$ such that $g(y) = z$. Additionally, there is $x \in X$ such that $f(x) = y$ since f is a bijection. Thus, $(g \circ f)(x) = g(f(x)) = z$.

□

Another important property of bijections is that they can be inverted.

Theorem 17.2. *Let $f : X \to Y$ be a function. f is invertible (i.e. there is a function $g : Y \to X$ such that $(f \circ g)(y) = y$ and $(g \circ f)(x) = x$ for all $x \in X$ and $y \in Y$) iff f is a bijection.*

Proof. $\Rightarrow$ Let's assume that f is invertible. We need to prove that f is a bijection.

- Let's assume that f does not satisfy the first property in the definitions of bijections i.e. there are $x_1, x_2 \in X$ such that $f(x_1) = f(x_2)$ but $x_1 = g(f(x_1)) = g(f(x_2)) = x_2$, which is a contradiction.
- Let $y \in Y$. Note that $f(g(y)) = y$, hence, $\operatorname{Im} f = Y$.

$\Leftarrow$ Let's assume that f is bijective. We need to define a function $g : Y \to X$ which is an inverse of f. Let $y \in Y$, note that there is a unique x such that $f(x) = y$, we define $g(y) = x$. Note that $f(g(y)) = y$ for every y by the construction of g. Additionally, $g(f(x)) = x$ since $f(g(f(x))) = f(x)$ and f is a bijection.

□

We denote g from this theorem as f^{-1} and in case when f is not a bijection $f^{-1}(y)$ denotes the set $\{x \in X \ : \ f(x) = y\}$.

Because of the following therem, bijections are very useful in combinatorics.

Theorem 17.3. *Let X and Y be two finite sets such that there is a bijection f from X to Y. Then $|X| = |Y|$.*

Using this result we can make prove the following equality.

Corollary 17.1. *Let X be a finite set of cardinality n. Then 2^X has the same cardinality as $\{0,1\}^{|X|}$.*

Proof. We are going to prove this statement for $X = [n]$ since for the full proof we would need a formal definition a finite set.

Now we need to construct a bijection g from $2^{[n]}$ to $\{0,1\}^n$ such that $g(Y) = (u_1, \ldots, u_n)$, where $u_i = 1$ iff $i \in Y$. It is clear that $g^{-1}(u_1, \ldots, u_n) = \{i \in [n] \ : \ u_i = 1\}$ is an inverse of g so g is indeed a bijection. □

17.2 *Surjections and Injections*

It is possible to note that the definition of the bijection consists of two part. Both of these parts are interesting in their own regard, so they have their own names.

Definition 17.2. *Let $f : X \to Y$ be a function.*

- *We say that f is a surjection iff every element of Y is an image of some element of X. In other words,*

$$\forall y \in Y \ \exists x \in X \ f(x) = y.$$

- *We say that f is an injection iff images of any two elements of X are different. In other words,*

$$\forall x_1, x_2 \in X \ f(x_1) \neq f(x_2).$$

Remark 17.1. *Let $f : X \to Y$ be an injection. Then $g : X \to \operatorname{Im} f$ such that $f(x) = g(x)$ is a bijection.*

Exercise 17.2. *Let $\mathbb{R}^+ = \{x \in \mathbb{R} \ : \ x > 0\}$. Is $f : \mathbb{R}^+ \to \mathbb{R}^+$ such that $f(x) = x + 1$ a surjection/injection?*

Like in the case of the bijection we may use surjections and injections to compare sizes of sets.

Theorem 17.4. *Let X and Y be finite sets.*

- *If there is an injection from X to Y, then $|X| \leq |Y|$.*
- *If there is a surjection from X to Y, then $|X| \geq |Y|$.*

End of The Chapter Exercises

17.3 Construct a bijection from $\{0,1,2\}^n$ to

$$\{(A, B) \ : \ A, B \subseteq [n] \text{ and } A, B \text{ are disjoint}\}.$$

17.4 *(recommended)* Construct a bijection from $\{0,1\} \times [n]$ to $[2n]$.

18. *Counting Principles*

Counting Principles:
Introduction to Combinatorics #2

https://youtu.be/dAoperLCjb8

18.1 *The Additive Principle*

The first principle is called *additive* principle and it states that if you have two disjoint sets, then their union have size equal to the sum of their sizes.

A simple illustration of this statement is the following. Assume you have three pencils and two pens; how many ways to choose a writing accessory. According to this principle the answer is $2+3=5$.

Theorem 18.1 (The Additive Principle)**.** *Let X and Y be finite sets. If $X \cap Y = \emptyset$, then $|X \cup Y| = |X| + |Y|$.*

Proof. Let $|X| = n$, $|Y| = m$ and $g : [n] \to X$ and $h : [m] \to Y$ be bijections. In order to prove it we just construct a bijection $f : [n+m] \to (X \cup Y)$.

$$f(i) = \begin{cases} g(i) & i < n \\ h(i-n) & i > n \end{cases}.$$

It's easy to see that f is an injection. Let us start by assuming the opposite i.e. that $i_0 \neq i_1 \in X \cup Y$ such that $f(i_0) = f(i_1)$. There are three cases.

- The first is when $i_0, i_1 \in [n]$. In this case $g(i_0) = g(i_1)$ which contradicts the assumption that g is a bijection.

- The second is when $i_0, i_1 \in \{n+1, n+2, \ldots, m\}$. In this case $h(i_0 - n) = h(i_1 - n)$ which contradicts the assumption that h is a bijection.

- Finally, the last case is when $i_0 \in [n]$ and $i_1 \in \{n+1, n+2, \ldots, m\}$. It is easy to see that this implies that $g(i_0) = h(i_1 - n)$. However, it means that $g(i_0) = h(i_1 - n) \in (X \cap Y)$, which contradicts the assumption that $X \cap Y = \emptyset$.

To finish the proof we need to show that f is a surjection. Let $w \in (X \cup Y)$. Consider the following two cases.

- Let $w \in X$. There is $i \in [n]$ such that $f(i) = g(i) = w$ since g is a bijection.

- Otherwise, $w \in Y$. In this case, there is $i \in [m]$ such that $f(i+n) = h(i) = w$ since h is a bijection.

$\square$

Corollary 18.1. *Let $X_1, \ldots, X_n$ be some pairwise disjoint sets. Then $|\bigcup_{i=1}^n X_i| = \sum_{i=1}^n |X_i|$.*

Exercise 18.1. *Prove Corollary 18.1.*

18.2 The Multiplicative Principle

The next principle is called the *multiplicative* principle and it can be illustrated as follows: imagine that you are given two postal stamps and three envelopes, how many ways are there to pack the letters? The answer is obviously $2 \cdot 3 = 6$.

Theorem 18.2 (The Multiplicative Principle)**.** *Let X and Y be finite sets. Then $|X \times Y| = |X| \times |Y|$.*

Proof. If one of the sets X and Y is empty, then $X \times Y$ is empty as well and the statement is as follows.

Assume that none of the sets are empty. Let $|X| = n$, $|Y| = m$, and $f : [n] \to X$ and $g : [m] \to Y$ be bijections. Note that

$$\bigcup_{i=1}^{n}(\{f(i)\} \times Y) = X \times Y.$$

Additionally, note that $(\{f(i)\} \times Y) \cap (\{f(j)\} \times Y) = \varnothing$ for $i \neq j$. Finally, it is easy to see that $g_i : [m] \to (\{f(i)\} \times Y)$ such that $g_i(j) = (f(i), g(j))$ is a bijection. Hence, $|X \times Y| = \sum_{i=1}^n |\{f(i)\} \times Y| = n \cdot m$. $\square$

Exercise 18.2. *Find the cardinality of the set*

$$\{(x, y) \ : \ x, y \in [9] \text{ and } x \neq y\}.$$

By analogy with unions and intersections of many sets we can define the cross product of many sets. Let $X_1, \ldots, X_n$ be some sets. Then $\bigtimes_{i=1}^1 X_i = A_1$ and $\bigtimes_{i=1}^{k+1} X_i = \left(\bigtimes_{i=1}^k X_i\right) \times X_{k+1}$[1].

[1] Note that cross product is not associative and different definitions of the product of several sets are not equivalent. However, the bijection constructed in the previous section allow us to think about these definitions as if they are equivalent.

Corollary 18.2. *Let $X_1, \ldots, X_n$ be some finite sets. Then $|\bigtimes_{i=1}^n X_i| = \prod_{i=1}^n |X_i|$.*

Exercise 18.3. *Prove Corollary 18.2.*

Theorem 18.3. *For any set X, $|2^X| = 2^{|X|}$.*

Proof. By Corollary 17.1, $|2^X| = \left|\{0,1\}^{|X|}\right|$, so it is enough to prove that $|\{0,1\}^{|X|}| = 2^{|X|}$. This statement is true by Corollary 18.2 since $|\{0,1\}^{|X|}| = \prod_{i=1}^{|X|} |\{0,1\}| = 2^{|X|}$. $\square$

18.3 The Inclusion-exclusion Principle

The last principle we are going to discuss in this chapter is the inclusion-exclusion principle which helps us to find the size of the union of sets when they are not disjoint.

Theorem 18.4 (The Inclusion-exclusion Principle)**.** *Let X and Y be finite sets. Then $|X \cup Y| = |X| + |Y| - |X \cap Y|$.*

Proof. Note that $X \cup Y = (X \setminus Y) \cup (Y \setminus X) \cup (X \cap Y)$. Hence, $|X \cup Y| = |X \setminus Y| + |Y \setminus X| + |X \cap Y|$. But it is possible to note that $|Y \setminus X| + |X \cap Y| = |Y|$ and $|X \setminus Y| + |X \cap Y| = |X|$. □

Corollary 18.3. *Let $X_1, \dots, X_n$ be some finite sets. Then*

$$\left|\bigcup_{i=1}^{n} X_i\right| = \sum_{S \subseteq [n]\,:\,S \neq \emptyset} (-1)^{|S|+1} \left|\bigcap_{i \in S} X_i\right|.$$

Proof. As always, we prove this statement using induction by n. The base case for $n = 2$ is true by Theorem 18.4.

By the induction hypothesis,

$$\left|\bigcup_{i=1}^{k} X_i\right| = \sum_{S \subseteq [k]\,:\,S \neq \emptyset} (-1)^{|S|+1} \left|\bigcap_{i \in S} X_i\right|.$$

In addition, by Theorem 18.4,

$$\left|\bigcup_{i=1}^{k+1} X_i\right| = \left|\bigcup_{i=1}^{k} X_i\right| + |X_{k+1}| - \left|\left(\bigcup_{i=1}^{k} X_i\right) \cap X_{k+1}\right|.$$

We need to simplify two elements of the sum on the right of the equality. By the induction hypothesis,

$$\left|\bigcup_{i=1}^{k} X_i\right| = \sum_{S \subseteq [k]\,:\,S \neq \emptyset} (-1)^{|S|+1} \left|\bigcap_{i \in S} X_i\right|.$$

In addition, it is easy to note that

$$\left|\left(\bigcup_{i=1}^{k} X_i\right) \cap X_{k+1}\right| = \left|\bigcup_{i=1}^{k} (X_i \cap X_{k+1})\right|.$$

Thus using the induction hypothesis,

$$\left|\left(\bigcup_{i=1}^{k} X_i\right) \cap X_{k+1}\right| = \sum_{S \subseteq [k]\,:\,S \neq \emptyset} (-1)^{|S|+1} \left|\bigcap_{i \in S} (X_i \cap X_{k+1})\right| = \sum_{S \subseteq [k+1]\,:\,(k+1) \in S \text{ and } S \neq \{k+1\}} (-1)^{|S|} \left|\bigcap_{i \in S} X_i\right|.$$

As a result,

$$|X_{k+1}| - \left| \left(\bigcup_{i=1}^{k} X_i \right) \cap X_{k+1} \right| = \sum_{S \subseteq [k+1] \,:\, (k+1) \in S} (-1)^{|S|+1} \left| \bigcap_{i \in S} X_i \right|.$$

Which implies that

$$\left| \bigcup_{i=1}^{k+1} X_i \right| = \sum_{S \subseteq [k] \,:\, S \neq \emptyset} (-1)^{|S|+1} \left| \bigcap_{i \in S} X_i \right| + \sum_{S \subseteq [k+1] \,:\, (k+1) \in S} (-1)^{|S|+1} \left| \bigcap_{i \in S} X_i \right| = \sum_{S \subseteq [k+1] \,:\, S \neq \emptyset} (-1)^{|S|+1} \left| \bigcap_{i \in S} X_i \right|.$$

□

End of The Chapter Exercises

18.4 *(recommended)* How many numbers from [999] are not divisible neither by 3, nor by 5, nor by 7.

18.5 How many numbers x from 1 to 999 such that at least one of the digits of x is 7?

18.6 Let A, B be some finite sets such that $A \subseteq B$. Show that $|B \setminus A| = |B| - |A|$.

18.7 *(recommended)* Let n be some positive integer. Find the cardinality of the set

$$\{(A, B) \;:\; A, B \subseteq [n] \text{ and } A \cap B \neq \emptyset\}?$$

18.8 Let X and Y be some finite sets, and $f : X \to Y$ be a function such that $|f^{-1}(y)| = k$ for all $y \in Y$. Prove that $|X| = k|Y|$.

18.9 *(recommended)* Show that if U and $X_1, \ldots, X_n \subseteq U$ are some finite sets, then

$$\left| \bigcap_{i=1}^{n} X_i \right| = \sum_{S \subseteq [n]} (-1)^{|S|} \left| \bigcap_{i \in S} \overline{X}_i \right|,$$

where $\overline{X}_i = U \setminus X_i$ and $\bigcap_{i \in \emptyset} \overline{X}_i = U$.

19. *The Pigeonhole Principle*

The Pigeonhole Principle: Introduction to Combinatorics #3

https://youtu.be/1D1Fa7WIUO8

The principle we are going to discuss in this chapter is very simple, it states that if you have more objects than boxes, then you cannot put all the objects into boxes without putting two objects into the same box.

More formally the principle can be formulated as follows: if $n > m$, then any function from $[n]$ to $[m]$ is not an injection. This simple statement is famous in mathematics and called *the pigeonhole principle*[1].

[1] The pigeonhole principle is also called the Dirichlet principle, after the German mathematician G. Lejeune Dirichlet, who demonstrated, using this principle, that there were at least two Parisians with the same number of hairs on their heads.

Theorem 19.1 (the pigeonhole principle)**.** *Let X and Y be some sets such that $|X| > |Y|$. Then for any function $f : X \to Y$ there are $x_0 \neq x_1 \in X$ such that $f(x_0) = f(x_1)$.*

Proof. The statement follows from Theorem 17.4. □

This simple statement is very handy in combinatorics. For example, using this statement one may prove that in any group of more than 12 people there are two people who were born in the same month.

Assume that there are n people in the group and $n > 12$. Consider the following function $f : [n] \to [12]$ such that $f(i) = j$ if the ith person was born in jth month. Note that f is not an injection since $n > 12$ i.e. there are $i_0 \neq i_1$ such that i_0th and i_1th person are born in the same month.

We may also prove that in any group of people there are two people who are friends with the same number of people in the group.

Assume the number of people is n. It is easy to see that every person may have at most $n-1$ friends. Hence, we may define a function $f : [n] \to \{0, \dots, n-1\}$ such that $f(i)$ is equal to the number of friends in this group of the ith person in this group. We need to consider two cases.

- If $\operatorname{Im} f \subseteq [n-1]$, then $|[n]| > |\operatorname{Im} f|$ and f is not an injection.
- Otherwise, note that it is not possible that $(n-1) \in \operatorname{Im} f$ because if there is a friend with no friends it is not possible that there is a friend who is friends with everyone. Hence, $f : [n] \to \{0, 1, \dots, n-2\}$ and f is not an injection.

Theorem 19.2 (Erdős—Szekeres)**.** *Every sequence of $(r-1)(s-1)+1$ distinct real numbers contains a subsequence of length r that is increasing or* *a subsequence of length s that is decreasing.*

Proof. Given a sequence of length $(r-1)(s-1)+1$, label each number x_i in the sequence with the pair (a_i, b_i), where a_i is the length of the longest increasing subsequence ending with x_i and b_i is the length of the longest decreasing subsequence ending with x_i. Each two numbers in the sequence are labeled with a different pair: if $i < j$ and $x_i < x_j$ then $a_i < a_j$, and on the other hand if $x_i > x_j$ then $b_i < b_j$. But there are only $(r-1)(s-1)$ possible labels if a_i is at most $r-1$ and b_i is at most $s-1$, so by the pigeonhole principle there must exist a value of i for which a_i or bi is outside this range. If a_i is out of range then x_i is part of an increasing sequence of length at least r, and if b_i is out of range then x_i is part of a decreasing sequence of length at least s. $\square$

We can also use the pigeonhole principle to show that the lower bound from Theorem 16.3 is precise.

Theorem 19.3. *There is a B-decision tree T such that $h(T) \leq 9$ and $\mathrm{val}(T, S) \in S$ for all $S \in \binom{[1000]}{500}$.*

Proof. Let us fix some set $S \in \binom{[1000]}{500}$. Note that $S \cap [501] \neq \emptyset$. Therefore, the minimal element of S belongs to $[501]$. Hence, using an algorithm similar to the algorithm from Chapter 9, we can find the minimal element of S using at most $\lceil \log 501 \rceil = 9$ questions. $\square$

19.1 The Generalized Pigeonhole Principle

One may generalize the pigeonhole principle in the following way. If N objects are placed into k boxes, then there is at least one box containing at least $\lceil N/k \rceil$ objects.

Theorem 19.4 (the generalized pigeonhole principle)**.** *Let X and Y be some sets. Then for any function $f : |X| \to |Y|$ there are $x_1, \ldots, x_\ell \in X$ such that*

- $f(x_i) = f(x_j)$,
- $x_i \neq x_j$ *for any* $i \neq j \in [\ell]$, *and*
- $\ell \geq \lceil |X|/|Y| \rceil$, *where* $\lceil \alpha \rceil$ *denotes the least integer greater than or equal to* α.

Now we illustrate applications of this principle on some examples and prove the statement in the next section.

Using this theorem we can prove that if we draw 9 cards out of a deck of cards, we are guaranteed that at least three of them are of the same suit. Given that, there are 4 suits in the deck, by pigeonhole principle if we put each card into one of the four boxes according to their suits, one of the boxes should have at least $\lceil 9/4 \rceil = 3$ cards.

Another example shows how the generalized pigeonhole principle can be applied to an important part of combinatorics called Ramsey theory.

Assume that in a group of six people, each pair of individuals consists of two friends or two enemies. One may prove that there are either three mutual friends or three mutual enemies in the group.

Let A be one of the six people; of the five other people in the group, there are either three or more who are friends of A, or three or more who are his enemies A. This statements follows from the generalized pigeonhole principle since when five objects are divided into two sets, one of the sets has at least $\lceil 5/2 \rceil = 3$ elements. Without loss of generality we may suppose that B, C, and D are friends of A. If any two of these three individuals are friends, then these two and A form a group of three mutual friends. Otherwise, B, C, and D form a set of three mutual enemies.

19.2 *The Averaging Principle*

Assume that we have a collection of m objects, the ith of which has "size" l_i. We wish to show that at least one of the objects is large. In this situation we can argue that at least one of the objects has size greater or equal to the average size ($\sum l_i/m$).

Theorem 19.5 (the averaging principle)**.** *Every sequence of numbers has a number at least as large as the average and a number at least as small as the average; i.e. for any sequence $a_1, \ldots, a_m$ there are i and j such that*

$$a_i \geq \frac{1}{m}\sum_{i=1}^{m} a_i$$

and

$$a_j \leq \frac{1}{m}\sum_{i=1}^{m} a_i.$$

Proof. We prove only the existence of i, proof of the existence of j is almost the same.

Assume the opposite, i.e. that $a_i < \sum_{i=1}^{n} a_i/m$ for any $i \in [n]$. Note that this implies that $\sum_{i=1}^{n} a_i \leq m \cdot \sum_{i=1}^{n} a_i/m = \sum_{i=1}^{n} a_i$. Which is a contradiction. □

Exercise 19.1. *Finish the proof of Theorem 19.5*

Like the pigeonhole principle, this principle is very simple but the applications of it are surprisingly interesting.

First, it allows to prove the generalized pigeonhole principle.

Proof of Theorem 19.4. Let $Y = [m]$ (it is easy to see that the proof works for any other finite Y). Define the sequence $a_i = |f^{-1}(i)|$. Note that we need to prove that $a_i \geq \lceil |X|/m \rceil$ for some $i \in [m]$

It is clear that $\bigcup_{i=1}^{m} f^{-1}(i) = X$ and that $f^{-1}(i) \cap f^{-1}(j) = \emptyset$ for any $i \neq j \in [m]$. Thus, by the additive principle, $\sum_{i=1}^{m} a_i = |X|$. Hence, by the averaging principle, $a_i \geq |X|/m$ for some $i \in [m]$. However, a_i is an integer, thus $a_i \geq \lceil |X|/m \rceil$. □

Another nice application of the averaging principle allows us to prove that if in some group (with more than one person) the number of pairs of people who know each other is less than $n-1$, then we can split this group into two subgroups such that people from different subgroups do not know each other.

Let us assume that there are n people in the group. We prove the statement using the induction by n.

(the base case) If $n = 2$, there are less than $n - 1 = 1$ pairs of people who know each other, in other words, these two people in the group do not know each other. Thus we can put each of them into a separate subgroup.

(the induction step) Let p_i ($i \in [n]$) be the number of acquaintances of the ith person. Note that $\sum_{i=1}^{n} p_i \leq 2(n-2)$ since we count each pair twice. By the averaging principle, $p_i \leq 2(n-2)/n = 2 - 2/n$ for some $i \in [n]$. Thus p_i is either 0 or 1.

- If $p_i = 0$, we can put the ith person into the first subgroup and everyone else into another.
- If $p_i = 1$ we consider the group of $n - 1$ people without the ith person, by the induction hypothesis, we can split everyone but ith person into two subgroups and since the ith person has only one acquaintance we can put them in the same subgroup.

End of The Chapter Exercises

19.2 Show that among any group of five (not necessarily consecutive) integers, there are two with the same remainder when divided by 4.

19.3 Show that if there are 30 students in a class, then at least two have last names that begin with the same letter.

19.4 Let n be a positive integer. Show that in any set of n consecutive integers there is exactly one divisible by n.

19.5 *(recommended)* Prove that for every sequence of integers $a_1, \ldots,$ a_n there are $k > 0$ and $\ell \geq 0$ such that $k + \ell \leq n$ and $\sum_{i=k}^{k+\ell} a_i$ is divisible by n.

19.6 *(recommended)* Let $S \subseteq [20]$ be a set. Show that if $|S| \geq 13$, then there are $a, b \in S$ such that $a - b = 6$.

19.7 How many numbers must be selected from the set $[6]$ to guarantee that at least one pair of these numbers add up to 7?

19.8 Sasha is training for a triathlon. Over a 30 day period, he pledges to train at least once per day, and 45 times in all. Then there will be a period of consecutive days where he trains exactly 14 times.

19.9 Show that among any $n + 1$ positive integers not exceeding $2n$ there must be an integer that divides one of the other integers. *Hint: Consider the set of holes equal to the set of odd numbers from* 1 *to* $2n$.

19.10 *(recommended)* Let $a_1, a_2, \ldots, a_t$ be positive integers. Show that if $a_1 + a_2 + \cdots + a_t - t + 1$ objects are placed into t boxes, then for some $i \in [t]$, the ith box contains at least a_i objects. *Hint: It is important in this question that $a_1, \ldots, a_t$ are integers.*

19.11 Let $\{(x_1, y_1), \ldots, (x_5, y_5)\} \subseteq \mathbb{Z}^2$ be a set of five distinct points with integer coordinates in the xy plane. Show that the midpoint of the line joining at least one pair of these points has integer coordinates.

20. *Binomial Coefficients*

Permutations and Binomial Coefficients: Introduction to Combinatorics #4

https://youtu.be/HLClazoqqzg

This chapter studies the following question: "how many ways to take k objects out of a box with n objects". We assume that the objects are taken one by one; note that there are four modes for this question.

1. we return objects to the box after we take them and the order in which we take them matters,
2. we are *do not* return the objects and the order in which we take them matters,
3. we return objects to the box after we take them and the order in which we take them *does not* matter,
4. we are *do not* return the objects and the order in which we take them *does not* matter.

The Table 20.1 summarizes the results we are going to prove.

Object's name	Parameters	Formula
Functions	we return objects and the order *is* important	n^k
Injections	we *do not* return objects and the order *is* important	$(n)_k$
Subsets	we *do not* return objects and the order *is not* important	$\binom{n}{k}$
Multisets	we return objects and the order *is not* important	$\binom{n+k-1}{k}$

Table 20.1: Formulas for the numbers of ways to take k objects out of a box with n objects

20.1 *Counting Functions*

Note that if we number objects using numbers from 1 to n, then in the first mode the answer is the same as the number of functions from $[n]$ to $[k]$ since we need to just choose which objects is selected on the ith step for $i \in [k]$.

Let us solve a more general question; assume we have two finite sets X and Y: how many functions exist from X to Y?

Theorem 20.1. *Let X and Y be some finite sets. Y^X represents the set of all functions from X to Y. Then $|Y^X| = |Y|^{|X|}$.*

Proof. For simplicity we prove the statement in the case when $X = [n]$. Fix some finite set Y. We prove the statement using induction by n. The base case for $n = 1$ is obvious, since there are $|Y|$ different functions from $[1]$ to Y. Let us prove the induction step, by the induction hypothesis, $|Y^{[n-1]}| = |Y|^{n-1}$. Note that

$$|Y^{[n]}| = \left|\left\{(f, y) \ : \ f \in Y^{[n-1]}, y \in Y\right\}\right| = \\ |Y^{[n-1]} \times Y| = |Y^{[n-1]}| \cdot |Y| = |Y|^n.$$

□

Corollary 20.1. *There are n^k ways to select k objects out of n if the order matters and we return objects to the box after we pick them.*

Exercise 20.1. *Finish the proof of Theorem 20.1 by proving that the statement holds for any set X.*

However, what if we need to find size of a subset of Y^X satisfying some constraint? For example, we may try to find the size of the set

$$(Y)_X = \left\{f \in Y^X \ : \ f \text{ is an injection}\right\}.$$

First, let us try to do this informally. Assume that $X = [n]$ and $|Y| = m$, to define $f \in (Y)_X$ we need to choose images of $1, 2, \ldots, n$. There are m possible ways to select an image of 1, $m - 1$ ways to define $f(2)$ since we cannot use the value selected for 1 etc. Hence, $|(Y)_X| = m(m-1)\ldots(m-n+1)$ (we denote this number as $(m)_n$).

Theorem 20.2. *Let X and Y be some sets. Then $|(Y)_X| = (|Y|)_{|X|}$.*

Proof. Let us prove this statement for $X = [n]$. We prove this using induction by n. The base case, for $n = 1$, is clear. Now we need to prove the induction step from n to $n + 1$. By the induction hypothesis, for any m, the number of injections from $[n]$ to Y is equal to $(|Y|)_n$.

Fix some m and some set Y of cardinality m. Note that

$$|(Y)_X| = |\left\{(f, v) \in (Y)_{[n-1]} \times [m] \ : \ v \notin \operatorname{Im} f\right\}|.$$

It is easy to see that $|\{(f, v) \ : \ v \notin \operatorname{Im} f\}| = m - n + 1$ for any $f \in (Y)_{[n-1]}$ and

$$\left\{(f, v) \in (Y)_{[n-1]} \times [m] \ : \ v \notin \operatorname{Im} f\right\} = \bigcup_{f \in (Y)_{[n-1]}} \{(f, v) \ : \ v \notin \operatorname{Im} f\}.$$

As a result, $|(Y)_X| = (m)_{n-1} \cdot (m - n + 1) = (m)_n$. □

The special case of this result is that there are $n \cdot (n-1) \cdot \ldots \cdot 1$ different permutations of $[n]$ (recall that the number is denoted by $n!$).

Exercise 20.2. *Finish the proof of Theorem 20.2 by proving that the statement holds for any set X.*

Corollary 20.2. *There are $(n)_k$ ways to select k objects out of n if the order matters and we do not return objects to the box after we pick them.*

20.2 Counting Subsets

In this section we study the version of the question when we do not return the objects back to the box; i.e., we cannot select an object twice.

Recall that we denoted the set of all subsets of X by 2^X. The reason for this notation is that $|2^X| = 2^{|X|}$. A quite famous example of a subset of this set is the set

$$\binom{X}{n} = \{A \subseteq X \,:\, |A| = n\}\,.$$

In other words, it is the set of all possible ways to select n elements from X. Size of the set $\binom{[m]}{n}$ we denote by $\binom{m}{n}$ and call it a binomial coefficient.

Exercise 20.3. *Show that for any two finite sets X and Y, if $|X| = |Y|$, then $\left|\binom{X}{k}\right| = \left|\binom{Y}{k}\right|$.*

Note that by any ordered selection of n object out of m, one may construct an unordered selection of n objects out of m, and each unordered selection is counted $n!$.

Theorem 20.3. *For any $n > k \geq 0$, $\binom{n}{k} = \frac{(n)_k}{k!} = \frac{n!}{k!(n-k)!}$.*

Exercise 20.4. *Show that $\binom{n}{k} = \binom{n}{n-k}$ for any $n > k$.*

The formula in the Theorem 20.3 allows to find the values of binomial coefficients, however, it is not very convenient since $n!$ is growing very fast. Thus the following theorem provides a much more efficient way to compute the values of binomial coefficients.

Theorem 20.4 (Pascal's rule)**.** *For $n > k \geq 1$, $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$.*

Proof. The first, algebraic, proof of this theorem is quite simple, we just notice that

$$\binom{n-1}{k-1} + \binom{n-1}{k} = \frac{(n-1)!}{(k-1)!(n-k)!} + \frac{(n-1)!}{k!(n-k-1)!} =$$
$$\frac{(n-1)!}{(k-1)!(n-k-1)!}\left(\frac{1}{n-k} + \frac{1}{k}\right) = \frac{n!}{k!(n-k)!} = \binom{n}{k}.$$

However, this proof does not explain *why* the statement is true. So we consider an alternative proof, which informally can be explained as follows. Assume we need to choose k objects out of n. There are two possible ways:

- we may select n and choose $k-1$ objects from the rest,
- or we may decide to not select n choose k objects from the rest.

In the first case we have $\binom{n-1}{k-1}$ ways to select objects and in the second case we have $\binom{n-1}{k}$ ways to select objects.

Let us prove the statement a bit more formally. Note that

$$\binom{[n]}{k} = \{A \subseteq [n] \,:\, |A| = k \text{ and } n \in A\} \cup \{A \subseteq [n] \,:\, |A| = k \text{ and } n \notin A\}.$$

Since these sets are disjoint and $\{A \subseteq [n] \,:\, |A| = k \text{ and } n \notin A\} = \binom{[n-1]}{k}$, we get the following equality

$$\binom{n}{k} = |\{A \subseteq [n] \,:\, |A| = k \text{ and } n \in A\}| + \binom{n-1}{k}.$$

Hence, to finish the proof we need to explain that

$$|\{A \subseteq [n] \,:\, |A| = k \text{ and } n \in A\}| = \binom{n-1}{k-1}.$$

To prove this statement we construct a bijection

$$f : \{A \subseteq [n] \,:\, |A| = k \text{ and } n \in A\} \to \binom{[n-1]}{k}$$

such that $f(A) = A \setminus \{n\}$. It is clear that this is a bijection. Thus, we prove the statement. □

A mnemonic rule for the Pascal's rule is to use Pascal's triangle. [1]

$$\begin{array}{ccccccccc}
 & & & & 1 & & & & \\
 & & & 1 & & 1 & & & \\
 & & 1 & & 2 & & 1 & & \\
 & 1 & & 3 & & 3 & & 1 & \\
1 & & 4 & & 6 & & 4 & & 1
\end{array}$$

In this diagram the kth entry of the nth row (entries and rows have numbers starting from 0) is equal to $\binom{n}{k}$. Thus the rule for the triangle is very simple, the value of an entry is equal to 1 if it is the first or the last in the row or it is equal to the sum of the two entries to the left and right on the row above.

Exercise 20.5. *Show that* $\binom{n}{k} = \binom{n}{n-k}$ *for any integers* $n > k \geq 0$

[1] The pattern of numbers that forms Pascal's triangle was known well before Pascal's time. Halayudha, around 975 explained obscure references to Meru-prastaara, the Staircase of Mount Meru, giving the first surviving description of the arrangement of these numbers into a triangle.

The Persian mathematician Al-Karaji (953–1029) wrote a now lost book which contained the first description of Pascal's triangle. It was later repeated by the Persian poet-astronomer-mathematician Omar Khayyám (1048–1131); thus the triangle is also referred to as the Khayyam triangle in Iran.

Pascal's triangle was known in China in the early 11th century through the work of the Chinese mathematician Jia Xian (1010–1070). In the 13th century, Yang Hui (1238–1298) presented the triangle and hence it is still called Yang Hui's triangle in China.

Pascal's Traité du triangle arithmétique (Treatise on Arithmetical Triangle) was published in 1655. In this, Pascal collected several results then known about the triangle, and employed them to solve problems in probability theory. The triangle was later named after Pascal by Pierre Raymond de Montmort (1708) who called it "Table de M. Pascal pour les combinaisons" (French: Table of Mr. Pascal for combinations) and Abraham de Moivre (1730) who called it "Triangulum Arithmeticum PASCALIANUM" (Latin: Pascal's Arithmetic Triangle), which became the modern Western name.

Binomial Theorem

Now we are ready to prove the theorem which gave the name to binomial coefficients.

Theorem 20.5 (Binomial theorem)**.** *For any real numbers x and y,*

$$\sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} = (x+y)^n.$$

Proof. Informally, the explanation of the equality is as follows. If we consider the product

$$\underbrace{(x+y)\cdot(x+y)\cdot\ldots\cdot(x+y)}_{n \text{ times}},$$

then for every k there are exactly $\binom{n}{k}$ possibilities to obtain $x^k y^{n-k}$. Indeed, to obtain $x^k y^{n-k}$ we need to choose x from n possibilities (corresponding to the multiplier $x+y$) exactly k times.

A formal proof uses the induction by n. The base case is true, since $\sum_{k=0}^{1}\binom{1}{k}x^k y^{1-k} = x + y = (x+y)^1$. Assume that

$$\sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} = (x+y)^n,$$

we wish to prove that

$$\sum_{k=0}^{n+1} \binom{n+1}{k} x^k y^{n+1-k} = (x+y)^{n+1}.$$

Note that

$$\begin{aligned}(x+y)^{n+1} = (x+y)\left(\sum_{k=0}^{n}\binom{n}{k}x^k y^{n-k}\right) = \\ \sum_{k=0}^{n}\binom{n}{k}x^{k+1}y^{n-k} + \sum_{k=0}^{n}\binom{n}{k}x^k y^{n+1-k} = \\ \sum_{k=1}^{n+1}\binom{n}{k-1}x^k y^{n+1-k} + \sum_{k=0}^{n}\binom{n}{k}x^k y^{n+1-k} = \\ \sum_{k=0}^{n+1}\left(\binom{n}{k-1}+\binom{n}{k}\right)x^k y^{n+1-k} = \sum_{k=0}^{n+1}\binom{n+1}{k}x^k y^{n+1-k}.\end{aligned}$$

□

Finally, we need to answer the question in the mode, when the order does not matter and we do not return the objects to the box. The answer to this question is clearly equal to the number of multisets of $[n]$ containing k objects.

Theorem 20.6. *The number of k-element multisets whose elements all belong to $[n]$ is $\binom{n+k-1}{k}$.*

Exercise 20.6. *Prove Theorem 20.6*

Using Theorem 20.5 one may prove several important equalities of sums of binomial coefficients.

Corollary 20.3. *Let $n \in \mathbb{N}$. Then*

1. $\sum_{k=0}^{n}(-1)^k\binom{n}{k} = 0$ *and*
2. $\sum_{k=0}^{n}(k+1)\binom{n}{k} = n2^{n-1}$.

Proof. 1. Let $x = -1$. We may notice that, by Theorem 20.5,

$$\sum_{k=0}^{n}(-1)^k\binom{n}{k} = \sum_{k=0}^{n}1^{n-k}x^k\binom{n}{k} = (1-1)^n = 0.$$

2. This equality is a bit more tricky. Let $x = 1$. Note that

$$\sum_{k=0}^{n}(k+1)\binom{n}{k} = \sum_{k=0}^{n}(k+1)x^k1^{n-k}\binom{n}{k} = \frac{d\sum_{k=0}^{n}x^k\binom{n}{k}}{dx} = n(1+x)^{n-1}.$$

As a result, $\sum_{k=0}^{n}(k+1)\binom{n}{k} = n2^n$.

□

Using the idea of the second equality, we can give an alternative — more explicit — proof of Claim 16.3.1.

Proof of Claim 16.3.1. Let $x = 1/2$. Then the value we would like to compute is $\sum_{k=1}^{n}kx^k$. Note that

$$\sum_{k=1}^{n}kx^k = x\sum_{k=1}^{n}kx^{k-1} = x\frac{d\sum_{k=0}^{n}x^k}{dx} = x\frac{d}{dx}\frac{1-x^{n+1}}{1-x} = x\frac{-(n+1)x^n(1-x)+(1-x^{n+1})}{(1-x)^2} = 2\left(1-\frac{1}{2^{n+1}}-(n+1)\frac{1}{2^{n+1}}\right) = 2-\frac{n+2}{2^n}.$$

□

Counting Groups of Subsets

In this section we study a generalization of the question we study in the previous sections: "How many ways to select ℓ groups made of k_1, $k_2, \ldots, k_\ell$ objects, respectively, out of n". We denote this number by $\binom{n}{k_1\ k_2\ \ldots\ k_\ell\ (n-m)}$, where $m = k_1 + \cdots + k_\ell$.

Clearly selecting these objects is the same as selecting k_1 objects out of n, after that selecting k_2 objects out of $n - k_1$ etc. As a result,

$$\binom{n}{k_1\ k_2\ \dots\ k_\ell\ (n-m)} = \frac{n!}{k_1!(n-k_1)!} \cdot \frac{(n-k_1)!}{k_2!(n-k_1-k_2)!} \cdot \dots \cdot \frac{(n-k_1-k_2-\dots-k_{\ell-1})!}{k_\ell!(n-k_1-k_2-\dots-k_\ell)!} = \frac{n!}{k_1!k_2!\dots k_\ell!(n-k_1-k_2-\dots-k_\ell)!}.$$

Similarly to the Binomial theorem, we can prove the following.

Theorem 20.7 (Multinomial theorem)**.** *For any real numbers x_1, x_2, ..., x_ℓ and integer n,*

$$(x_1 + x_2 + \dots + x_\ell)^n = \sum_{k_1,k_2,\dots,k_\ell\ :\ k_1+k_2+\dots+k_\ell=n} \binom{n}{k_1\ k_2\ \dots\ k_\ell} \prod_{i=1}^{n} x_i^{k_i}.$$

Exercise 20.7. *Prove Theorem 20.7.*

Double Counting:
Introduction to Combinatorics #4

https://youtu.be/OrzMP8nuuho

20.3 Double Counting

The method that was used to prove Theorem 20.4 can be generalized to a method that is called *double counting principle*. The double counting principle states the following "obvious" fact: if the size of a set is counted in two different ways, the answers are the same.

Using this principle we may prove the following theorem.

Theorem 20.8 (Vandermonde's identity)**.** *For any integers $n, m > k$, $\sum_{i=0}^{k} \binom{n}{i}\binom{m}{k-i} = \binom{n+m}{k}$.*

Proof. The idea is as follows, let us imagine that we have n parrots and m crows, and we need to find how many ways to select k birds. It is easy to see that it is equal to $\binom{n+m}{k}$. At the same if we need to select i parrots there are $\binom{n}{i}\binom{m}{k-i}$ ways to do this. Thus the number is also equal to $\sum_{i=0}^{k} \binom{n}{i}\binom{m}{k-i}$. □

However, the method can be used in a more sophisticated way.

Lemma 20.1 (Handshaking Lemma)**.** *Suppose some number of people meet at a party and some shake hands. Assume that no person shakes his or her own hand and furthermore no two people shake hands more than once.*

The number of guests who shake hands an odd number of times is even.

Proof. Let $1, \dots, n$ be the people at the party. We apply double counting to the set of ordered pairs (i, j) for which i and j shake hands with each other at the party. Let d_i be the number of times that i shakes hands, and e be the total number of handshakes that occur. On one hand, the number of pairs is $\sum_{i=1}^{n} d_i$, since for each i the number of choices of j is equal to d_i. On the other hand, each handshake gives rise to two pairs (i, j) and (j, i); so the total is $2e$. Thus $\sum_{i=1}^{n} d_i = 2e$. But, if the sum of n numbers is even, then evenly many of the numbers are odd. (Because if we add an odd number of odd numbers and any number of even numbers, the sum will be always odd). $\square$

End of The Chapter Exercises

20.8 Show that $(x + y)_n = \sum_{k=0}^{n} \binom{n}{k}(x)_k(y)_{n-k}$.

20.9 Show that $\sum_{k=0}^{n} \binom{n}{k}^2 = \binom{2n}{n}$.

20.10 Show that $\sum_{m=k}^{n} \binom{m}{k} = \binom{n+1}{k+1}$. *Hint: Note that the formula on the right corresponds to the number of ways to select $k + 1$ elements out of $n + 1$; m in the summation on the left denotes the maximum of this selected set minus one.*

20.11 Using the previous formula, find the formulas for the following expressions: 1. $\sum_{k=0}^{n} k$, 2. $\sum_{k=0}^{n} k^2$, and 3. $\sum_{k=0}^{n} k^3$.

20.12 Using the binomial theorem, explain the following equalities: 1. $\sum_{k=0}^{n} \binom{2n}{2k} = \sum_{k=0}^{n-1} \binom{2n}{2k+1}$, and 2. $\sum_{k=0}^{n} \binom{2n+1}{2k} = \sum_{k=0}^{n} \binom{2n+1}{2k+1}$.

20.13 *(recommended)* Show that $\sum_{k=0}^{n} \binom{m+k}{k} = \binom{m+n+1}{n}$.

20.14 Show that $\sum_{k=0}^{n} \binom{n-k}{k} = f_{n+1}$, where $f_1 = 1$, $f_2 = 1$, and $f_{n+2} = f_{n+1} + f_n$ for $n > 0$.

20.15 Show that $\binom{n}{m}\binom{m}{k} = \binom{n}{k}\binom{n-k}{m-k}$.

20.16 *(recommended)* Show that $(a + 1)^p \equiv a^p + 1 \pmod{p}$. *Hint: Use the binomial theorem.*

20.17 *(recommended)* We say that a function $f : \{0, 1\}^n \to \{0, 1\}$ depends on the ith argument iff for some $a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n \in \{0, 1\}$

$$f(a_1, \dots, a_{i-1}, 0, a_{i+1}, \dots, a_n) \neq f(a_1, \dots, a_{i-1}, 1, a_{i+1}, \dots, a_n).$$

We also say that the function f depends on all the arguments iff for all $i \in [n]$ it depends on ith argument.

Find the number of functions $f : \{0, 1\}^n \to \{0, 1\}$ depending on all arguments.

20.18 Find the largest coefficient of $(x_1 + x_2 + \cdots + x_k)^k$.

20.19 Prove that, without using Theorem 20.7,

$$\sum_{k_1,k_2,k_3\,:\,k_1+k_2+k_3=n} \binom{n}{k_1\ k_2\ k_3} = 3^n.$$

21. Partitions

The main question we study in this chapter is as follows: "how many ways to put n objects into k boxes". Note that there are four modes for this question:

1. the objects and boxes are identical,
2. the objects are identical but boxes are different,
3. the objects are different but boxes are identical,
4. the objects and boxes are different.

We are going to study the question in all these modes. The Table 21.1 summarizes the results we are going to prove for the cases when all the boxes are not empty.

21.1 Set Partitions

This section considers the case when objects are not identical.

First, we define a notion that allows us to compute the answer in case when all the boxes are the same.

Definition 21.1. *A partition of the set $[n]$ is a collection of non-empty blocks so that each element of $[n]$ belongs to exactly one of these blocks. The number of partitions of $[n]$ into k nonempty blocks is denoted by $S(n,k)$. The numbers $S(n,k)$ are called the* Stirling numbers of the second kind.

It is easy to see that $S(n,1) = 1$ and $S(n,n) = 1$. Moreover, $S(n,k) = 0$ if $k > n$ or $k \leq 0$.

Let us find the value in a more complicated setting, we claim that $S(n, n-1) = \binom{n}{2}$. Indeed, any partition of $[n]$ into $n-1$ blocks consists of $n-1$ singletons and one set with two elements, thus we just need to select these two elements.

Using double counting, one may prove a recursive formula for Stirling numbers of the second kind.

Theorem 21.1. *For any $n > k > 0$,*

$$S(n,k) = S(n-1,k-1) + k \cdot S(n-1,k).$$

Object's name	Parameters	Formula
Surjections	n distinct objects k distinct boxes	$S(n,k)k!$
	n distinct objects any number of boxes	$\sum_{k=i}^{n} S(n,k)k!$
Compositions	n identical objects k distinct boxes	$\binom{n-1}{k-1}$
	n identical objects any number of boxes	2^{n-1}
Set partitions	n distinct objects k identical boxes	$S(n,k)$
	n distinct objects any number of boxes	$B(n)$
Integer partitions	n identical objects k identical boxes	$p_k(n)$
	n identical objects any number of boxes	$p(n)$

Table 21.1: Formulas for the numbers of ways to put n objects into k boxes so that the boxes are not empty

Proof. Let us consider n, note that there are two cases either n forms a singleton in a partition or it is not the only element in the part.

It is easy to see that there are $S(n-1, k-1)$ partitions where n is a singleton and $k \cdot S(n-1, k)$ partitions where n is not a singleton (we multiply by k since there are k possible ways to add n to a partition of $[n-1]$). $\square$

Using this notation, we can express the number of surjections.

Lemma 21.1. *There are exactly $k!S(n,k)$ surjective functions from $[n]$ to $[k]$.*

Proof. Let $\mathcal{S}(n,k)$ be the set of surjections from $[n]$ to $[k]$, $\mathcal{P}(n,k)$ be the set of partitions with non-empty blocks, and $F : \mathcal{S}(n,k) \to \mathcal{P}(n,k)$ such that $F(f) = \left\{f^{-1}(1), \ldots, f^{-1}(k)\right\}$.

It is easy to see that $F(f) = F(g)$ iff there is $h : [k] \to [k]$ such that $f \circ h = g$. Hence, $F^{-1}(f) = k!$ for any $f \in \mathcal{S}(n,k)$. Thus $|\mathcal{S}(n,k)| = k!|\mathcal{P}(n,k)|$. $\square$

Note that the number of surjections from $[n]$ to $[k]$ is equal to the number of ways to put n different objects into k different boxes.

Using this equality, we can prove a surprising result.

Theorem 21.2. *For any real x and positive integer n,*

$$x^n = \sum_{k=0}^{n} S(n,k)(x)_k,$$

where $(x)_k = \prod_{i=0}^{k-1}(x-i)$.

To prove the statement we need the following statement.

Theorem 21.3. *Let p and q be real polynomials. If $p(\ell) = q(\ell)$ for all natural numbers ℓ, then $p(x) = q(x)$ for all real numbers x.*

Proof of Theorem 21.2. Using the previous result, it is enough to prove that for any integer $\ell > 0$,

$$\ell^n = \sum_{k=0}^{n} S(\ell,k)(\ell)_k.$$

Clearly ℓ^n denotes the number of ways to put n different objects into ℓ different boxes. Note that if we have k nonempty boxes, then there are $\binom{n}{k}$ ways to select these boxes and $k!S(\ell,k)$ ways to put objects in these k boxes. Thus formula in the left is equal to the formula on the right. $\square$

Definition 21.2. *The number of all set partitions of $[n]$ into nonempty parts is denoted by $B(n)$, and is called the n*th Bell number*. (We define $B(0) = 0$).*

It is easy to see that the following theorem holds.

Theorem 21.4. *For any $n \geq 0$,*

$$B(n) = \sum_{k=0}^{n} S(n,k).$$

However, it is also possible to express the Bells numbers in terms of themselves.

Theorem 21.5. *For any $n \geq 0$,*

$$B(n+1) = \sum_{i=0}^{n} \binom{n}{i} B(i).$$

Proof. Note that there are $B(n+1)$ ways to split $[n+1]$ into non-empty blocks. At the same time there are $\binom{n}{n-i}$ ways to select elements to put with $n+1$ in the same block (if we know that there are $n-i$ elements with $n+1$ in the block) and $B(i)$ ways to split the rest into blocks. As a result, there are $\sum_{i=0}^{n}\binom{n}{i}B(i)$ to split $[n+1]$ into nonempty blocks. $\square$

21.2 Composition

This section answers the question in the case when the objects are the same but boxes are different. Since all the objects are identical, only the number of objects in each box matters.

Definition 21.3. *A sequence $(a_1, \ldots, a_k)$ of nonnegative integers such that $a_1 + \cdots + a_k = n$ is called a* weak composition *of n into k. If, in addition, all the numbers are positive, the sequence is called a* composition*.*

Using the binomial coefficients we can find the number of weak compositions.

Theorem 21.6. *For all positive integers n and k, the number of weak compositions of n into k is equal to $\binom{n+k-1}{n}$.*

Proof. Let us consider k boxes in line one after each other. Note that if we put balls inside of the boxes we see a line consisting of n balls and $k-1$ walls separating the k boxes from each other. Note that simply knowing in which order the n identical balls and $k-1$ separating walls follow each other is the same as knowing the number of balls in each box. So our problem is equivalent to counting the number of ways to put $k-1$ walls on one of $n+k-1$ positions. □

As a result, we can count the number of compositions.

Corollary 21.1. *For all positive integers n and k, the number of compositions of n into k is equal to $\binom{n-1}{k-1}$.*

Exercise 21.1. *Let $\ell_1, \ldots, \ell_k$ be some nonnegative numbers such that $\ell_1 + \cdots + \ell_k = \ell$. Find the number of weak compositions (in terms of ℓ, k, and n) $(a_1, \ldots, a_k)$ of n into k such that $a_i \geq \ell_i$.*

Corollary 21.2. *The number of all compositions of n is equal to 2^{n-1}.*

21.3 Integer Partitions

Now consider the case when both objects and boxes are identical. In this case, as in the previous we are only interested in numbers of objects in boxes, but in addition, we are not interested in an order of these numbers.

Definition 21.4. *Let n and $a_1 \geq a_2 \geq \cdots \geq a_k \geq 1$ be integers so that $a_1 + \ldots a_k = n$. Then the sequence $(a_1, \ldots, a_k)$ is called a* partition[1] *of the integer n into k parts.*

The number of all the partitions is denoted by $p(n)$ and the number of partitions of n into k parts is denoted by $p_k(n)$.

[1] Note that we used the word partition in two different meanings: one to denote a partition of a set $[n]$ and another to denote the partition of an integer n. In most of the cases the meaning is clear from the context; however, if it is necessary to emphasize that we mean partition of a set, we say set-partition. Note that in some languages there are two different words for these two notions; e.g in French "partition" is used for set-partitions, and "partage" for partitions of the integer n).

There is no good formula allowing to find the value of $p(n)$. Nevertheless, we will prove some properties of $p(n)$. The main tool to explain proofs we are going to discuss are Young diagrams[2]. A Young diagram for a partition $(a_1, \dots, a_k)$ consists of k columns of squares called "boxes" such that in the ith column there are a_i boxes (an example of such a diagram is depicted on 21.1). We can reflect a Young diagram of a partition of n with respect to its main diagonal, we get another shape, representing the *conjugate* partition of n (an example of such transformation is also depicted on 21.1).

[2] A small variation of these diagrams is called Ferrers shapes after an American mathematician Norman Macleod Ferrers.

(a) The Young diagram for the partition $(4, 3, 1, 1)$.

(b) The conjugate of the Young diagram for the partition $(4, 3, 1, 1)$.

Figure 21.1: Young diagrams.

Using these diagrams, it is easy to show the following theorem.

Theorem 21.7. *The number of partitions of n into at most k parts is equal to that of partitions of n into parts not larger than k.*

Proof. Note that if a partition has at most k parts, then the conjugate of this partition has all the parts of size at most k. As, a result, the number of partitions of n into at most k parts is equal to that of partitions of n into parts not larger than k. $\square$

End of The Chapter Exercises

21.2 Let $q(n)$ be the number of partitions of n in which each part is at least two. Then $q(n) = p(n) - p(n-1)$, for all positive integers $n \geq 2$.

21.3 *(recommended)* Find a formula for $S(n, 2)$.

21.4 Find a formula for $S(n, 3)$.

21.5 Find a formula for $S(n, n-2)$.

21.6 *(recommended)* Show that $B(n) \leq n!$.

21.7 Let $m \geq n$ be positive integers. Show that

$$S(m, n) = \sum_{i=1}^{m} S(m-i, n-1) n^{i-1}.$$

21.8 Prove that the number of partitions of n into exactly k parts is equal to the number of partitions of n in which the largest part is exactly k.

21.9 *(recommended)* Prove that the number of partitions of n into at most k parts is equal to that of partitions of $n + k$ into exactly k parts.

22. *Permutations*

Recall that a permutation is a bijection from $[n]$ to $[n]$. We already discussed several properties of them. In this chapter we will discuss some combinatorial properties of them. We denote by S_n the set of all permutations of $[n]$.[1]

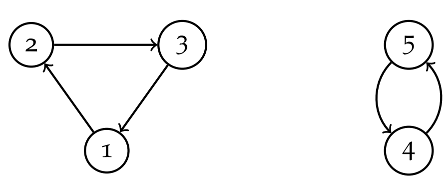
The main operation over permutations is composition, for two permutations p and q we denote their composition $p \circ q$ by pq.[2] Note that this operation is not commutative; i.e. $p \circ q$ is not necessarily equal to $q \circ p$.

Every permutation p can be uniquely determined by the values $p(1), \dots, p(n)$, thus sometimes we denote the permutation f by a sequence $p(1)p(2)\dots p(n)$ (we call it *one-line notation*). For example, the permutation 312 is equal to the function $p : [3] \to [3]$ such that

$$p(x) = \begin{cases} 3 & \text{if } x = 1 \\ 1 & \text{if } x = 2 \\ 2 & \text{if } x = 3 \end{cases}.$$

22.1 *Cycles*

Consider the permutation p equal to 23154 and draw a diagram with 5 points where we draw an arrow from i to j iff $p(i) = j$.



It is easy to see that there are two "cycles" in the diagram. In this section we prove that this is not a coincidence and we also study some properties of permutations with respect to the structure of these cycles.

Definition 22.1. *Let p be a permutation of $[n]$, $x \in [n]$, and i be the smallest integer such that $p^i(x) = \underbrace{p(p(\dots p(}_{i\text{ times}}x)\dots)) = x$. The we say that the entries $x, p(x), \dots, p^{i-1}(x)$ form an i-cycle in p.*

[1] Letter S is used since in the group theory this set is called the symmetric group.

[2] Some authors denote $q \circ p$ by pq.

We denote a permutation $q : [n] \to [n]$ consisting of one cycle $a_1, \dots, a_k$ by $(a_1, \dots, a_k)$; i.e.

$$q(x) = \begin{cases} a_2 & \text{if } x = a_1 \\ a_3 & \text{if } x = a_2 \\ \dots & \\ a_1 & \text{if } x = a_k \\ x & \text{otherwise} \end{cases}.$$

Theorem 22.1. *All permutations can be decomposed into the disjoint unions of their cycles.*

Exercise 22.1. *Prove Theorem 22.1.*

For example, the discussed permutation 23154 can be decomposed into $(1, 2, 3)(4, 5)$.

If an permutation $p : [n] \to [n]$ has c_i cycles of length $i \in [n]$, then we say that $(c_1, c_2, \dots, c_n)$ is the *cycle type* of p. The simplest question we may ask is "how many permutations of a certain cyclic type exist?", the following theorem gives an answer for this question.

Theorem 22.2. *Let $c_1, \dots, c_n$ be some positive integers such that $\sum_{i=1}^{n} ic_i = n$. Then there are $\frac{n!}{c_1!c_2!\dots c_n!1^{c_1}2^{c_2}\dots n^{c_n}}$ permutation of the cyclic type $(c_1, \dots, c_n)$.*

Note that this result allows us to answer the following problem. King Arthur has n Knights of the Round Table; Arthur wonders: how many ways to seat in the round table? In other words he is asking how many permutations of the cyclic type $(0, 0, \dots, 0, 1)$. Hence, the answer for Arthur's question is $n!$ (note that we also need to give a seat to the king).

22.2 Stirling Numbers of The First Kind

In the previous chapter we defined Stirling numbers of the second kind; in this section we define their first kind counterpart.

Definition 22.2. *Let $n > k$ be some integers. We denote the number of permutations of $[n]$ with k cycles by $c(n, k)$. The number $s(n, k) = (-1)^{n-k}c(n, k)$ is called a* Stirling number of the first kind.

The multiplier $(-1)^{n-k}$ seems a bit strange, but we will explain it in Theorem 22.4.

Like the numbers $S(n, k)$, the numbers $c(n, k)$ satisfy a simple recurrent formula.

Theorem 22.3. *Let $n \geq k$ be positive integers. Then*

$$c(n, k) = c(n-1, k-1) + (n-1)c(n-1, k).$$

Exercise 22.2. *Prove Theorem 22.3.*

Theorem 22.4. *For any real x and positive integer n,*

$$(x)_n = \sum_{k=0}^{n} s(n,k)x^k.$$

Now one may see why the multiplier $(-1)^{n-k}$ was necessary by comparing this equality with the equality from Theorem 21.2 stating that

$$x^n = \sum_{k=0}^{n} S(n,k)(x)_k.$$

In other words, Stirling numbers of the second kind are "inverse" to the Stirling numbers of the first kind.

We can interpret this result in terms of linear algebra. Consider the vector space $\mathbb{P}_n$ of real polynomials of degree at most n. It is well known that $1, x, \ldots, x^n$ is the basis of this space; additionally, it is easy to see that $1, (x)_1, \ldots, (x)_n$ is also a basis. Then the matrices $\mathcal{S}$ and $\int$ such that $\mathcal{S}_{i,j} = S(i,j)$ and $\int_{i,j} = s(i,j)$ are change of basis matrices between these two bases.

22.3 Permutations with Restricted Cycle Structure

One of the problem of the representation of a permutation as a collection of cycles is that it is not unique; e.g. $(1,2,3)(4,5)$ and $(5,4)(1,2,3)$ represent the same permutation. To avoid this we introduce a *canonical cycle form*, That is, each cycle will be written with its largest element first, and the cycles will be written in increasing order of their first elements. Thus the permutation's 23154 canonical cycle form is $(3,1,2)(5,4)$.

Using this notation and the next lemma we can discover several nice properties of permutations.

Lemma 22.1. *Let $p : [n] \to [n]$ be a permutation written in canonical cycle notation. Let $\mathcal{G}(p)$ be the permutation obtained from p by omitting the parentheses and reading the entries as a permutation in the one-line notation. Then $\mathcal{G}$ is a bijection from S_n to S_n.*

For example, $\mathcal{G}(23154) = 31254$ and $\mathcal{G}^{-1}(23154) = (2)(3,1)(5,4) = 32154$.

Using this transformation we may prove the following result, which is very technical without this transformation.

Theorem 22.5. *Let n be a positive integer and $x_1, \ldots, x_k \in [n]$ be k different numbers. There are $n!/k$ permutations of $[n]$ such that $x_1, \ldots, x_k$ are in the same cycle.*

Proof. Without loss of generality, $x_1 = n$.

Let $q = q_1 q_2 \dots q_n$ be a permutation of n, and let $\mathcal{G}(p) = q$, where $\mathcal{G}$ is the bijection from Lemma 22.1. Note that the last cycle of p starts with $x_1 = n$, and the entries in that cycle of q are precisely the entries on the right of n in q. Therefore, p contains $x_1, \dots, x_k$ in the same cycle if and only if $x_2, \dots, x_k$ are on the right of n in q. It is easy to see that there are $\binom{n}{k}(k-1)!(n-k)! = \frac{n!}{k}$ such permutations q. $\square$

Another nice result states that for any $i \in [n]$, the probability that i is in a cycle of length k does not depend on k and is equal to $1/n$.

Theorem 22.6. *Let $i \in [n]$. Then for all $k \in [n]$, there are exactly $(n-1)!$ permutations of $[n]$ in which the cycle containing i is of length k.*

Proof. Again, it is sufficient to prove the statement for $i = n$. Let $q = q_1 q_2 \dots q_n$ be a permutation of n, let $\mathcal{G}(p) = q$, where $\mathcal{G}$ is the bijection from Lemma 22.1, and let $q_j = n$. Then the cycle C containing n in p is of length $n - j + 1$ as n itself starts the last cycle. So if we want C to have length k, we must have $j = n + 1 - k$. However, there are clearly $(n-1)!$ permutations of length n that contain n in a given position, and the proof follows. $\square$

22.4 *Superpermutations*

In this section we consider the following problem. In the TV series "The Melancholy of Haruhi Suzumiya" there are 14 episodes. The episodes feature time travel and are chronologically challenging for the viewer. Moreover, they were originally aired in a nonlinear order. When the series went to DVD, the episodes were rearranged. Thus, it is something of an obsession for fans to rewatch the series over and over again, going through in many different chronologies. So the question is as follows: if you want to watch all the episodes of the anime in every possible order, what is the shortest sequence of episodes you need to watch?

Let us first formulate a more formal question.

Definition 22.3. *A sequence $w_1, \dots, w_\ell \in [n]$ is called an n-superpermutation iff for any $p \in S_n$ there is $0 \le i \le \ell - n$ such that $w_{i+1} = p(1)$, $w_{i+2} = p(2), \dots,$ and $w_{i+n} = p(n)$.*

In other words, the question we wish to study can be formulated in the following way: what is the minimal length of a 14-superpermutation?

As usual, we would like to study a more complicated question, what is the minimal length of an n-superpermutation. The answer for this question is unknown; however, there are relatively tight known upper and lower bounds. The known upper bound was proven by Greg Egan in 2008.

Theorem 22.7. *For all $n \geq 4$, there is an n-superpermutation of length at most*

$$n! + (n-1)! + (n-2)! + (n-3)! + n - 3.$$

The Verge:
An anonymous 4chan post could help solve a 25-year-old math mystery

https://bit.ly/2Gj8kpT

However, the problem became especially famous because the best known lower bound was proven by an anonymous author on 4chan. The anonymous proved the following theorem.

Theorem 22.8. *Every n-superpermutation has length at least*

$$n! + (n-1)! + (n-2)! + n - 3.$$

Proof. First we need to define the notion of length between two permutations $p, q \in S_n$. We say that the distance between p and q is equal to $\mathcal{D} = k$ iff there is a word u of length k such that the last n letters of the concatenation of $w = p(1)p(2)\dots p(n)$ and u encodes the permutation q but any the last n symbols of the concatenation of w and any proper prefix of u is not a permutation; otherwise, we say that the distance is equal to $+\infty$.

Note that $n + \mathcal{D}(p_1, \dots, p_\ell) = \sum_{i=1}^{\ell-1} \mathcal{D}(p_i, p_{i+1}) \leq m$, where

$$w_1, w_2, \dots, w_m \in [n]$$

and

$$\{i_1 < i_2 < \dots < i_\ell\} = \{i \in [m-n] \ : \ w_{i+1} = p(1), \dots, w_{i+n} = p(n)\}.$$

In other words, to find the minimal n-superpermutation, we need to find a sequence of permutations $p_1, \dots, p_\ell$ containing all the permutations and with the minimal $\mathcal{D}$.

Instead of proving the statement right away, we prove four lower bounds, each stronger but more complicated than the previous one.

- $(n! + n - 1)$ We prove that

  $$\mathcal{D}(p_1, \dots, p_k) \geq C_0(p_1, \dots, p_k) - 1, \tag{22.1}$$

  where $C_0(p_1, \dots, p_k)$ is equal to the number of permutations occurring in $p_1, \dots, p_k$.

  It is easy to see that $C_0(p_1) = 1$ and $\mathcal{D}(p_1) = 0$ so $\mathcal{D}(p_1) = 0 \geq 1 - 1 = C_0(p_1) - 1$. We may also note that for any $p_{k+1} \in S_n$, $C_0(p_1, \dots, p_{k+1}) \leq C_0(p_1, \dots, p_k) + 1$ and $\mathcal{D}(p_k, p_{k+1}) \geq 1$. Therefore

  $$\begin{aligned} \mathcal{D}(p_1, \dots, p_{k+1}) \geq \mathcal{D}(p_1, \dots, p_k) + 1 \geq \\ C_0(p_1, \dots, p_k) + 1 - 1 \geq C_0(p_1, \dots, p_{k+1}) - 1. \end{aligned}$$

  Combining (22.1) with the fact that if all the permutations occur in the sequence $p_1, \dots, p_\ell$, then $C_0(p_1, \dots, p_\ell) = n!$, we prove that any n-superpermutation has length at least $n! - 1 + n$.

- ($n! + (n-1)! + (n-2)$) To prove this lower bound we need to introduce the notion of a 1-cycle class. A 1-cycle class of permutations of $[n]$ is a subset $\{p_1, \ldots, p_n\} \subseteq S_n$ such that $p_{k+1}(n) = p_k(1)$, and $p_{k+1}(i) = p_k(i+1)$ for $i \in [n-1]$. For example,

  $$\{12345, 23451, 34512, 45123, 51234\}$$

  is a 1-cycle class.

  Let us now prove that

  $$\mathcal{D}(p_1, \ldots, p_k) \geq C_0(p_1, \ldots, p_k) + C_1(p_1, \ldots, p_k) - 1, \tag{22.2}$$

  where $C_1(p_1, \ldots, p_k)$ is equal to the number of complete 1-cycle classes in $p_1, \ldots, p_{k-1}$ (a 1-cycle class $\{q_1, \ldots, q_n\}$ is complete in $p_1, \ldots, p_t$ iff $\{q_1, \ldots, q_n\} \subseteq \{p_1, \ldots, p_t\}$).

  It is easy to see that $C_0(p_1) = 1$, $C_1(p_1) = 0$ and $\mathcal{D}(p_1) = 0$ so $\mathcal{D}(p_1) = 0 \geq 1 + 0 - 1 = C_0(p_1) + C_1(p_1) - 1$.

  It is easy to see that for any $p_{k+1} \in S_n$,

  $$C_0(p_1, \ldots, p_{k+1}) \leq C_0(p_1, \ldots, p_k) + 1$$
  $$C_1(p_1, \ldots, p_{k+1}) \leq C_1(p_1, \ldots, p_k) + 1.$$

  Hence, if $\mathcal{D}(p_k, p_{k+1}) \geq 2$, then (22.2) is true.

  If $\mathcal{D}(p_k, p_{k+1}) = 1$, we claim that only one of C_0 and C_1 increased. Note that p_k and p_{k+1} are in the same 1- cycle class. Therefore

  1. either this cycle is not complete yet and

     $$C_1(p_1, \ldots, p_{k+1}) = C_1(p_1, \ldots, p_k),$$

  2. or we finished the cycle and

     $$C_0(p_1, \ldots, p_{k+1}) = C_0(p_1, \ldots, p_k).$$

  As a result, (22.2) is true.

  Combining (22.2) with the fact that if all the permutations occur in the sequence $p_1, \ldots, p_\ell$, then $C_0(p_1, \ldots, p_\ell) = n!$ and $C_1(p_1, \ldots, p_\ell) \geq (n-1)! - 1$, we prove that any n-superpermutation has length at least $n! + (n-1)! - 1 - 1 + n$.

- ($n! + (n-1)! + (n-2)! + (n-3)$) To prove the final lower bound we need to define 2-cycles. The 2-cycle generated by p is the sequence $p_1, \ldots, p_{n(n-1)}$ such that $p_1 = p$, $\mathcal{D}(p_{in+j}, p_{in+j+1}) = 1$ for $i \geq 0$ and $n \geq j \geq 1$, and $\mathcal{D}(p_{in}, p_{in+1}) = 2$ for $i \geq 1$ (note that the cycle is unique). For example, 12345, 23451, 34512, 45123, 51234, 23415, 34152, 41523, 15234, 52341, 34125, 41253, 12534, 25341, 53412, 41235, 12354, 23541, 35412, 54123 is a 2-cycle generated by 12345, it is also generated by 23415, 34125, and 41235. More generally, we have the following result. If a 2-cycle is generated by p, then it is generated by all $n-1$ permutations obtained by fixing the last entry of p and cyclically permuting the other entries; i.e., by p and the permutations

$$\begin{aligned}
&p(2)\dots p(n-1)p(1)p(n),\\
&p(3)\dots p(n-1)p(1)p(2)p(n),\\
&\dots,\\
&p(n-1)p(1)\dots p(n-2)p(n).
\end{aligned}$$

We say that a sequence $p_1, \dots, p_k$ enters the 2-cycle generated by p if $p_{i+1} = p$ and $\mathcal{D}(p_i, p_{i+1}) \geq 2$. Because each 2-cycle contains only $n(n-1)$ permutations, any sequence containing all the permutations must enter at least $(n-2)!$ different 2-cycles.

Let us now prove that

$$\begin{aligned}
&\mathcal{D}(p_1, \dots, p_k) \geq \\
&\qquad C_0(p_1, \dots, p_k) + C_1(p_1, \dots, p_k) + C_2(p_1, \dots, p_k) - 2, \quad (22.3)
\end{aligned}$$

where $C_2(p_1, \dots, p_k)$ is equal to the number of entered 2-cycles.

It is easy to see that $C_0(p_1) = 1$, $C_1(p_1) = 0$, $C_2(p_1) = 1$, and $\mathcal{D}(p_1) = 0$ so $\mathcal{D}(p_1) = 0 \geq 1 + 0 + 1 - 2 = C_0(p_1) + C_1(p_1) + C_2(p_1) - 2$.

It is easy to see that for any $p_{k+1} \in S_n$,

$$\begin{aligned}
C_0(p_1, \dots, p_{k+1}) &\leq C_0(p_1, \dots, p_k) + 1\\
C_1(p_1, \dots, p_{k+1}) &\leq C_1(p_1, \dots, p_k) + 1\\
C_2(p_1, \dots, p_{k+1}) &\leq C_2(p_1, \dots, p_k) + 1.
\end{aligned}$$

Hence, if $\mathcal{D}(p_k, p_{k+1}) \geq 3$, then (22.3) is true.

If $k = 1$, then we are still inside the last 2-cycle and inside the last 1-cycle class, therefore like in the previous case (22.3) is true.

If $k = 2$, then we claim that if the value of C_1 increases, then the value of C_2 cannot change. Suppose that the value of C_1 increases. This means that the permutation p_k complete the 1-cycle class and we have not visited it before. Since we completed the 1-cycle class, we visited the permutation $q = p_k(2)p_k(3)\dots p_k(n)p_k(1)$ by 2-step. It is also possible to note that q and p_{k+1} generate the same cyclic class and it implies that $C_2(p_1, \dots, p_{k+1}) = C_2(p_1, \dots, p_k)$. As a result, (22.3) is true.

Combining (22.2) with the fact that if all the permutations occur in the sequence $p_1, \dots, p_\ell$, then $C_0(p_1, \dots, p_\ell) = n!$, $C_1(p_1, \dots, p_\ell) \geq$

$(n-1)!-1$, and $C_2(p_1,\dots,p_\ell) \geq (n-2)!$, we prove that any n-superpermutation has length at least $n! + (n-1)! - 1 + (n-2)! - 2 + n$.

□

Using this inequality we may conclude that real fans of "The Melancholy of Haruhi Suzumiya" need to watch at least 93884313611 episodes which takes around 3572462 years.

End of The Chapter Exercises

22.3 *(recommended)* Find an explicit formula for $c(n, n-2)$.

22.4 Prove that for any fixed k, the function $c(n, n-k)$ is a polynomial function of n. Find the degree of that polynomial.

22.5 Let p be a permutation of $[n]$. We associate a permutation matrix $M^{(p)}$ to p as follows. Let $M^{(p)}_{i,j} = 1$ if $p(i) = j$, and let $M^{(p)}_{i,j} = 0$ otherwise. Prove that $|\det M^{(p)}| = 1$.

22.6 Prove that if p and q are two permutations, then $M^{(p)}M^{(q)} = M^{(pq)}$.

22.7 *(recommended)* Prove that permutations p and p^{-1} are of the same cycle type for any permutation p.

22.8 A permutation p is called a nontrivial involution if $p^2 = 12\dots n$, but $p \neq 12\dots n$. Prove that if $n > 1$, the number of nontrivial involutions in S_n is odd.

22.9 Show that any permutation can be obtained as a product of some transpositions; i.e., cycles of length 2.

23. Generating Function

In this chapter we discuss the basics of one of the most general methods we have in combinatorics, the method is called "generating functions". The core idea of this method is to use knowledge we have about mathematical analysis in combinatorics.

23.1 Easy Two Term Recurrences

Let us start from the following problem. Sasha took an insane credit in a bank: he took 100\$ at the beginning and his debt is growing twofold every year. At the beginning of each year John is paying 100\$ to the bank. How big will be his debt in 5 years?

It is easy to see that the answer for this and similar questions can be answered using a recurrence relation. Indeed, if a_i denotes his debt on ith year, then $a_0 = 100$, and $a_{n+1} = 2a_n - 100$. Using this, one may compute all the values of a_i. However, the question became tricky if we want to find an explicit formula for a_i.

To solve this kind of questions we can use beforementioned generating functions.

Definition 23.1. *Let $\{c_n\}_{n\geq 0}$ be a sequence of real numbers. Then the generating function for this sequence is the power series $F(x) = \sum_{n\geq 0} c_n x^n$.*

Note that these power series may not converge for $x \neq 0$. In this chapter, we will not discuss this problem and always pretend that they are converging, for a formal explanation of how to deal with this issue see Appendix A.

Let us use the definition of a_i to find the generating function $G(x)$ for this sequence. Note that $a_{n+1}x^{n+1} = 2a_n x^{n+1} - 100x^{n+1}$. Thus

$$\sum_{n\geq 0} a_{n+1}x^{n+1} = \sum_{n\geq 0} 2a_n x^{n+1} - 100 \sum_{n\geq 0} x^{n+1}.$$

The left-hand side is equal to $G(x) - a_0$ and the right-hand side is equal to $2xG(x) - \frac{100x}{1-x}$. So we can derive the equality

$$G(x) - 100 = 2xG(x) - \frac{100x}{1-x}.$$

Using this equality we can find explicitly a formula for $G(x)$,

$$G(x) = \frac{100}{1-2x} - \frac{100x}{(1-x)(1-2x)}.$$

Let us simplify the formula a bit.

$$G(x) = \frac{100}{1-2x} + \frac{100}{1-x} - \frac{100}{1-2x} = \frac{100}{1-x}.$$

Thus $G(x) = \sum_{n\geq 0} 100x^n$. As a result, $a_n = 100$.

Exercise 23.1. *Find a formula for a_n in the case when $a_0 = 200$.*

Let us consider another, more complicated, example. Consider a sequence $\{a_n\}_{n\geq 0}$ such that $a_{n+1} = 2a_n + n$ for $n \geq 0$ and $a_0 = 1$. As in the previous case let us write an equation for the generating function $G(x)$.

$$G(x) - a_0 = 2xG(x) + \sum_{n\geq 0} nx^{n+1}.$$

First, we find a formula for $\sum_{n\geq 0} nx^n$,

$$\sum_{n\geq 0} nx^{n+1} = \sum_{n\geq 0} x^2 \cdot \frac{dx^n}{dx} = x^2 \cdot \frac{d\sum_{n\geq 0} x^n}{dx} = x^2\left(\frac{1}{1-x}\right)' = \frac{x^2}{(1-x)^2}.$$

Therefore,

$$G(x) = \frac{1-2x+2x^2}{(1-x)^2(1-2x)}.$$

So we need to find a more appropriate formula for $G(x)$. Let us try to find a formula in the form

$$\frac{1-2x+2x^2}{(1-x)^2(1-2x)} = \frac{A}{(1-x)^2} + \frac{B}{1-x} + \frac{C}{1-2x}.$$

To find A, B, and C we multiply both sides by $(1-x)^2$ and set $x = 1$. We get that $A = -1$. We can also multiply by $1-2x$ and substitute $x = 1/2$ and derive that $C = 2$. Now we need to find B, we substitute $x = 0$ to the equation and get $B = 0$. As a result, $G(x) = \frac{-1}{(1-x)^2} + \frac{2}{1-2x}$. Using simple equalities from calculus we can derive $G(x) = \sum_{n\geq 0} -(n+1) + 2^{n+1}x^n$. So $a_n = -(n+1)2^{n+1}$.

23.2 *Recurrences With Two Variables*

To illustrate how to deal with recurrence relations in cases when we have more than one variable, we prove a version of the binomial theorem and derive a formula for binomial coefficients. In order to do it, we consider the recurrence relation

$$\binom{n+1}{k} = \binom{n}{k} + \binom{n}{k-1}.$$

Let us denote $\sum_{k\geq 0}\binom{n}{k}x^k$ by $B_n(x)$. It is clear that

$$B_{n+1}(x) - 1 = (B_n(x) - 1) + xB_n(x).$$

Therefore, $B_{n+1}(x) = (1+x)B_n(x)$. As a result, $B_n(x) = (1+x)^n$; i.e. $\sum_{k\geq 0}\binom{n}{k}x^k = (1+x)^n$. To find a formula for binomial coefficients, we just need to use Taylor's formula, $\binom{n}{k} = \frac{d^k}{dx^k}B_n(x)\big|_{x=0}/k!$. So $\binom{n}{k} = n(n-1)\dots(n-k+1)/k!$.

23.3 *Products of Generating Functions*

Let us consider a new problem, how many ways to design a class consisting of n lectures with theoretical part and laboratory part (the first k days of the quarter form the theoretical part, note that k is not fixed) such that there are two midterms during the theoretical part and one exam during the laboratory part.

Let a_n be the answer. It is easy to see that

$$a_n = \sum_{k=1}^{n-2} k\binom{n-k}{2}.$$

However, this formula does not suggest an explicit formula. Let us write an equation for the generating function for a_n,

$$G(x) = \sum_{n\geq 0}\sum_{k=1}^{n-2} k\binom{n-k}{2}x^n.$$

It is easy to see that this formula implies that

$$G(x) = \left(\sum_{k\geq 0} kx^k\right)\left(\sum_{k'\geq 0}\binom{k'}{2}x^{k'}\right).$$

Thus

$$G(x) = \frac{x}{(1-x)^2}\cdot\frac{x^2}{(1-x)^3} = \frac{x^4}{(1-x)^5} = x^3\sum_{n\geq 0}\binom{n+4}{4}x^n.$$

As a result, $a_n = \binom{n+1}{4}$.

Using this example, we can formulate a general rule.

Theorem 23.1. *Let a_n be the number of ways to build a certain structure on an n-element set, and let b_n be the number of way to build another structure on an n-element set. Let c_n be the number of ways to separate $[n]$ into two parts consisting of numbers $\{1,\dots,k\}$ and $\{k+1,\dots,n\}$ ($k\geq 0$), and then to build a structure of the first type on the first set, and a structure of the second type on the second set.*

Then $H(x) = F(x)G(x)$, where $F(x)$, $G(x)$, and $H(x)$ are generating functions for $\{a_n\}_{n\geq 0}$, $\{b_n\}_{n\geq 0}$, and $\{c_n\}_{n\geq 0}$, respectively.

To illustrate this theorem, let us solve another problem. A company "bolshoy brat" needs to finish two projects. To do this, a manager of the company splits all the employees into two projects and in each project she selects product team and marketing team. How many ways to do this. Let c_n be the number of ways the manager can complete this task. Again, let us split the problem into two parts. Let $A(x)$ be the generating function for the number of ways to split people in the first project into marketing and product teams. It is clear that $A(x) = \sum_{k\geq 0} 2^k x^k = 1/(1-2x)$ since any k element set has 2^k subsets. It is easy to see that the second project has the same generating function. Thus the generating function for $\{c_n\}_{n\geq 0}$, $C(x) = A(x)A(x) = 1/(1-2x)^2$. As a result,

$$C(x) = \frac{1}{2}\sum_{n\geq 1} n2^n x^{n-1} = \frac{1}{2}\sum_{n\geq 0}(n+1)2^{n+1}x^n$$

and $c_n = (n+1)2^{n+1}$.

Exercise 23.2. *Find the number of ways to split an n-day semester into three parts, choose any number of holidays in the first part, an odd number of holidays in the second part, and an even number of holidays in the third part.*

23.4 Compositions of Generating Functions

As usual, we start the section from a problem. All n soldiers of a military squadron stand in a line. The officer in charge splits the line at several places, forming (non-empty) squads. Then she names one person in each unit to be the commander of that unit. Let c_n be the number of ways she can do this. Find an explicit formula for c_n.

If the officer splits the soldiers into k squads, then there are

$$\sum_{n_1,\ldots,n_k: n=n_1+\cdots+n_k} n_1 \cdot n_2 \ldots n_2$$

ways to do this. Hence, the generating function for splitting into squads and selecting commanders in all k squads is equal to $A^k(x)$, where $A(x) = \sum_{n\geq 0} nx^n = \frac{x}{(1-x)^2}$. Therefore, the generating function $C(x)$ for $\{c_n\}_{n\geq 0}$ is equal to $\sum_{k\geq 1} A^k(x)$. As a result,

$$C(x) = \frac{1}{1-A(x)} = 1 + \frac{x}{1-3x+x^2}.$$

It is possible to note that the roots α and β of x^2-3x+1 are equal to $(3\pm\sqrt{5})/2$, respectively. We want to find A and B such that

$$\frac{1}{1-3x+x^2} = \frac{A}{x-\alpha} - \frac{B}{x-\beta}.$$

Thus $1 = (A - B)x - A\beta + B\alpha$. Therefore, we have $A = B$ and $A(\alpha - \beta) = A\sqrt{5} = 1$; i.e. $A = B = \frac{1}{\sqrt{5}}$. By some simple calculations we may conclude that

$$\frac{1}{1 - 3x + x^2} = \frac{1}{\sqrt{5}}\left(\frac{\alpha}{1 - \alpha x} - \frac{\beta}{1 - \beta x}\right).$$

Therefore $C(x) = 1 + \frac{1}{\sqrt{5}} \sum_{n \geq 0} (\alpha^{n+1} - \beta^{n+1}) x^{n+1}$. Hence, $c_0 = 1$ and $c_n = \frac{1}{\sqrt{5}}(\alpha^n - \beta^n)$ for $n > 0$.

The following theorem generalises this observation.

Theorem 23.2. *Let a_n be the number of ways to build a certain structure on an n-element set, and let us assume that $a_0 = 0$. Let c_n be the number of ways to split the set $[n]$ into an unspecified number of disjoint non-empty intervals, then build a structure of the given type on each of these intervals. Set $h_0 = 1$. Denote $F(x) = \sum_{n \geq 0} a_n x^n$ and $G(x) = \sum_{n \geq 0} c_n x^n$. Then $G(x) = \frac{1}{1 - A(x)}$.*

End of The Chapter Exercises

23.3 *(recommended)* Find the generating functions of each of the following sequences (in the simplest form):

1. $a_n = n$;
2. $a_n = \alpha n + \beta$;
3. $a_n = n^2$;
4. $a_n = \alpha n^2 + \beta n + \gamma$;
5. $a_n = 3^n$.

23.4 *(recommended)* Let $F(x)$ be a generating function for the sequence $\{a_n\}_{n \geq 0}$. Write, in terms of $F(x)$, the generating functions of the following sequences:

1. $\{a_n + \alpha\}_{n \geq 0}$;
2. $\{\alpha a_n + \beta\}_{n \geq 0}$;
3. $\{n a_n\}_{n \geq 0}$;
4. $0, a_1, \ldots, a_n, \ldots$;
5. $a_1, \ldots, a_n, \ldots$;
6. $\{a_{n+m}\}_{n \geq 0}$ (m is a constant).

23.5 Let $f(n)$ be the number of subsets of $[n]$ that contain no two consecutive elements, for integer n. Find the recurrence that is satisfied by these numbers, and then find an explicit formula for these numbers.

23.6 Find an explicit formula for a_n if $a_0 = 0$ and for any $n \geq 0$, $a_{n+1} = a_n + 2^n$.

23.7 *(recommended)* Let a_n be the number of ways to pay n dollars using ten-dollar bills, five-dollar bills, and one-dollar bills only. Find the generating function for a_n.

23.8 *(recommended)* Let x_1 and x_2 be two different solutions of the equation $1 - bx - cx^2 = 0$. Show that a sequence $\{f_n\}_{n\geq}$ satisfies the recurrence relation $f_{n+2} = bf_{n+1} + cf_n$ iff $t_n = \alpha x_1^{-n} + \beta x_2^{-n}$ for some $\alpha, \beta \in \mathbb{R}$.

23.9 *(recommended)* Let $\{a_n\}_{n\geq 0}$, $\{b_n\}_{n\geq 0}$ be two sequences such that $b_n = \sum_{k=0}^{n} a_n$ and $F(x)$ be the generating function for $\{a_n\}_{n\geq 0}$. Find the generating function for $\{b_n\}_{n\geq 0}$ in terms of $F(x)$.

23.10 Let $\{a_n\}_{n\geq 0}$, $\{b_n\}_{n\geq 0}$ be two sequences such that $b_n = a_{2n}$ and $F(x)$ be the generating function for $\{a_n\}_{n\geq 0}$. Find the generating function for $\{b_n\}_{n\geq 0}$ in terms of $F(x)$.

Part V

Introduction to Set Theory

24. Cardinalities of Sets

Part IV discusses sizes of sets; however, this notion was never defined formally. This part is going to close this gap and study properties of sizes of sets.

24.1 Definition

One may notice that if we have a bijection f from $[n]$ to a set S we enumerate all the elements of S: $f(1)$, $\ldots$, $f(n)$. This observation allows us to define the cardinality of a set.

Definition 24.1. *Let S be a set, we say that cardinality of S is equal to n (we write that $|S| = n$) iff there is a bijection from $[n]$ to S.*

We also say that a set T is finite if there is an integer n such that $|T| = n$.

Note that this definition does not guarantee that cardinality is unique so we need the following theorem.

Theorem 24.1. *For any set S, if there are bijections $f : [n] \to S$ and $g : [m] \to S$, then $n = m$.*

Proof of Theorem 24.1. Let us consider the inverse g^{-1} of g (it exists by Theorem 17.2 since g is a bijection). Note that $h = g^{-1} \circ f$ is a bijection from $[n]$ to $[m]$.

We prove using induction by n that for any $n, m \in \mathbb{N}$, if there is a bijection h' from $[n]$ to $[m]$, then $n = m$. The base case is for $n = 1$; if $m \geq 2$, then there are $x, y \in [1]$ such that $h'(x) = 1$ and $h'(y) = 2$, but $x \neq y$ and we have only one element in $[1]$.

The induction step is also simple. Assume that there is a bijection h' from $[n+1]$ to $[m]$. We define a function $h'' : [n] \to [m-1]$ as follows:

$$h''(i) = \begin{cases} h'(i) & \text{if } h'(i) < h'(n+1) \\ h'(i) - 1 & \text{otherwise} \end{cases}.$$

We prove that h'' is a bijection.

- Let $i_1 \neq i_2 \in [n]$. If $h'(i_1), h'(i_2) < h'(n+1)$ or $h'(i_1), h'(i_2) \geq h'(n+1)$, then $h''(i_1) \neq h''(i_2)$ since $h'(i_1) \neq h'(i_2)$. Otherwise, without loss of generality we may assume that $h'(i_1) < h'(n+1) < h'(i_2)$ but it implies that $h''(i_1) = h'(i_1) < h'(n+1) \leq h'(i_2) - 1 = h''(i_2)$.

- Let $j \in [m-1]$. We need to consider two cases.

  1. Let $j < h(n+1)$. There is $i \in [n+1]$ such that $h'(i) = j$ since h' is a bijection (note that $i \neq n+1$). Thus $h''(i) = j$.
  2. Otherwise, there is $i \in [n+1]$ such that $h'(i) = j+1$ since h' is a bijection (note that $i \neq n+1$). Thus $h''(i) = j$.

Since h'' is a bijection, the induction hypothesis implies that $n = m-1$. As a result, $n+1 = m$. □

Also, using this definition we may finally prove Theorem 17.3.

Proof of Theorem 17.3. Let $|X| = n$, and $g : [n] \to X$ be a bijection. Note that $f \circ g : [n] \to Y$ is a bijection, hence $|Y| = n$. □

24.2 Generalized Commutative Operations

Using the notation of cardinality we can formalize the summation operation over sets:

$$\sum_{i \in S \,:\, P(i)} f(i) = \sum_{j=1}^{k} f(i_j),$$

where $\{i \in S \;:\; P(i)\} = \{i_1, \ldots, i_k\}$. More formally,

$$\sum_{i \in S \,:\, P(i)} f(i) = \sum_{j=1}^{k} f(g(j)),$$

where $k = |\{i \in S \;:\; P(i)\}|$ and $g : \{i \in S \;:\; P(i)\} \to [k]$ is a bijection.

Theorem 24.2. *The definition of $\sum_{i \in S : P(i)} f(i)$ does not depend on the choice of g; i.e. $\sum_{i=1}^{k} f(g_1(i)) = \sum_{i=1}^{k} f(g_2(i))$ for any two bijections $g_1, g_2 : \{i \in S \;:\; P(i)\} \to [k]$.*

Before we prove this statement we need to give a couple of definitions. We say that a function $h : [n] \to [n]$ is a *permutation* of $[n]$ iff h is a bijection. We also say that $i, j \in [k]$ form the inversion in h iff $h(i) > h(j)$ and $i < j$. We denote by $I(h)$ the number of inversions in h; i.e. $I(h) = |\{(i,j) \;:\; i, j \text{ form an inversion in } h\}|$.

Important examples of permutations are transposition: for any $i, j \in [n]$, $\tau_{i,j} : [n] \to [n]$ such that

$$\tau_{i,j}(x) = \begin{cases} j & \text{if } x = i \\ i & \text{if } x = j \\ x & \text{otherwise} \end{cases}.$$

is called a transposition of i and j.

It is easy to see that $I(h) = 0$ iff $h(i) = i$ for any $i \in [k]$. It is also clear that if i, j form an inversion in h, then $I(h) > I(h')$, where $h' = h \circ \tau_{i,j}$, i.e.

$$h'(x) = \begin{cases} h(j) & \text{if } x = i \\ h(i) & \text{if } x = j \\ h(x) & \text{otherwise} \end{cases}.$$

Proof of Theorem 24.2. Proof of this theorem consists of two parts. First, we prove that

$$\sum_{i=1}^{k} f(g(i)) = \sum_{i=1}^{k} f(g(h(i))) \tag{24.1}$$

for any bijections $g : \{i \in S \ : \ P(i)\} \to [k]$ and $h : [k] \to [k]$.

We prove Equation 24.1 using the induction by $I(h)$.

(the base case) If $I(h) = 0$, then h is the identity function and $g(i) = g(h(i))$. Hence, Equation 24.1 is true.

(the induction step) By the induction hypothesis, for any permutation $h' : [k] \to [k]$, if $I(h') < \ell$, then

$$\sum_{i=1}^{k} f(g(i)) = \sum_{i=1}^{k} f(g(h'(i))).$$

Let us consider a permutation $h : [k] \to [k]$ such that $I(h) = \ell$. Let i and j form an inversion in h (such i and j exist since $I(h) \neq 0$). Let $h' = h \circ \tau_{i,j}$. Note that by the induction hypothesis,

$$\sum_{i=1}^{k} f(g(i)) = \sum_{i=1}^{k} f(g(h'(i)))$$

since $I(h') < I(h) = \ell$ and it is clear that

$$\sum_{i=1}^{k} f(g(h'(i))) = \sum_{i=1}^{k} f(g(h(i))).$$

As a result, Equation 24.1 is true.

Now we are ready to finish proof of the theorem. Consider $g_1, g_2 : \{i \in S \ : \ P(i)\} \to [k]$ and define $h = g_1^{-1} \circ g_2$. Note that $h : [k] \to [k]$ is a permutation and $g_1(h(i)) = g_2(i)$. Thus we proved that

$$\sum_{i=1}^{k} f(g_1(i)) = \sum_{i=1}^{k} f(g(h(i))) = \sum_{i=1}^{k} f(g_2(i)).$$

$\square$

Similarly one may define a generalized union and intersection of sets. Let Ω and S be some sets, $X : S \to 2^{\Omega}$ and $P(i)$ be a predicate. Then

$$\bigcup_{i \in S\ :\ P(i)} X(i) = \bigcup_{i=1}^{k} X(g(i))$$

and

$$\bigcap_{i \in S\ :\ P(i)} X(i) = \bigcap_{i=1}^{k} X(g(i)),$$

where $k = |\{i \in S\ :\ P(i)\}|$ and $g : \{i \in S\ :\ P(i)\} \to [k]$ is a bijection.

Exercise 24.1. *Show that the definitions of $\bigcup_{i \in S : P(i)} X(i)$ and $\bigcap_{i \in S : P(i)} X(i)$ are correct, i.e. that they do not depend on the choice of g.*

End of The Chapter Exercises

24.2 Prove Theorem 17.4.

Part VI

Introduction to Game Theory

25. *Matrix Games*

This part uses mathematical methods to model interactions and conflicts between different actors. To simplify the analysis we are going to consider interactions between *two* actors that are trying to maximize their payoffs.

Definition 25.1. *The* strategic form*, or* normal form*, of a two-person game is a tuple* (X, Y, A, B) *such that* X *and* Y *are some nonempty sets, and* $A, B : X \times Y \to \mathbb{R}$.

We say that this game is a matrix game if X *and* Y *are finite.*

The interpretation is as follows. Simultaneously, the first player chooses a strategy $x \in X$ and the second player chooses a strategy $y \in Y$, each unaware of the choice of the other. Then their choices are made known and the first player wins $A(x, y)$ and the second player wins $B(x, y)$. (Depending on the monetary unit involved, $A(x, y)$ and $B(x, y)$ will be dollars, rubles, euros, etc.) If $A(x, y)$ or $B(x, y)$ is negative, the corresponding player loses the absolute value of this amount.

It is important to note that the notion of a strategy is very broad; e.g., a strategy for a game of chess, is a complete description of how to play the game, of what move to make in every possible situation that could occur. We are going to ignore the fact that in the game of chess it is physically impossible to describe all possible strategies since there are too many of them (in fact, there are more strategies than there are atoms in the known universe). On the other hand, the number of games of tic tac toe is rather small, so that it is possible to study all strategies and find an optimal strategy for each player.

In cases when X and Y are small sets it is convenient to describe such games using tables. Let us consider the game described by Table 25.1. In this game two players put one coin each on the table: if the

	heads	tales
heads	1, -1	-1, 1
tales	-1, 1	1, -1

Table 25.1: Heads and tales game

the coins have the same side up, then the second player pays 1 dollar to the first player, otherwise the first player pays 1 dollar to the second player. In other words, the firs number denotes the payoff of the first player and the second denotes the payoff of the second player.

Exercise 25.1. *Describe the game in normal form corresponding to rock paper scissors.*

An important class of games is zero-sum games.

Definition 25.2. *A game* (X, Y, A, B) *is* zero-sum *if* $A(x, y) = -B(x, y)$ *for all* $x \in X$ *and* $y \in Y$.

It is clear that the game we described is a zero-sum game.

25.1 Domination and Pareto Optimal Strategies

In Part II we studies optimal strategies; however, in case of games in the normal form it is not clear what does it mean optimal. To illustrate this difficulty, let us discuss the most famous game, the prisoner's dilemma:

Game 25.1. *Two members of a criminal gang are arrested and imprisoned. Each prisoner is in solitary confinement with no means of communicating with the other. The prosecutors lack sufficient evidence to convict the pair on the principal charge, but they have enough to convict both on a lesser charge. Simultaneously, the prosecutors offer each prisoner a bargain. Each prisoner is given the opportunity either to betray the other by testifying that the other committed the crime, or to cooperate with the other by remaining silent. The possible outcomes are:*

- *If A and B each betray the other, each of them serves two years in prison.*
- *If A betrays B but B remains silent, A will be set free and B will serve three years in prison (and vice versa).*
- *If A and B both remain silent, both of them will serve only one year in prison (on the lesser charge).*

It is clear that this game can be described using the following table.

	cooperates	defects
cooperates	-1, -1	-3, 0
defects	0, -3	-2, -2

Let us try to put ourselves into these prisoners shoes. If our partner is silent, then it is better for us to defect (in this case we are free immediately); if our partner defects, then is is also better for us to defect (we'll get two years instead of three). Therefore, no matter what our partner does it is always better to defect. Since the game is symmetric both players come to this conclusion and get two years each; however, if both of them cooperates they would serve only one year[1].

Let us generalise the argument we used to justify why defecting is better than cooperating.

Definition 25.3. *Let (X, Y, A, B) be a game in normal form. We say that* $x_1 \in X$ dominates $x_2 \in X$ *iff* $A(x_1, y) \geq A(x_2, y)$ *for all* $y \in Y$. *We also say that* x_1 strictly dominates x_2 *if* $A(x_1, y) > A(x_2, y)$ *for all* $y \in Y$.

Similarly we may define domination of strategies for the second player.

It seems reasonable to never choose x_2 provided that x_1 dominates x_2; in the prisoners dilemma "defects" dominates "cooperates"; hence, it seems choosing "defects" is the best behaviour for rational agents.

However, it also obvious that if both players cooperate is way better than if both of them defects; this observation leads to the following definition.

Definition 25.4. *Let (X, Y, A, B) be a game in the normal form. We say that a pair of strategies $(x, y) \in X \times Y$ is Pareto optimal if either $A(x', y') < A(x, y)$ or $B(x', y') < B(x, y)$ for any $(x', y') \in X \times Y$.*

In other words, a pair of strategies is Pareto optimal if any other choice would decrease the payoff for at least one of the players.

25.2 *Prisoners Dilemma In Real Life*

The reason that prisoner's dilemma is that famous is because there are many examples human interactions as well as interactions in nature that have the same payoff matrix. Let us consider two of them.

Political science. In political science, the prisoner's dilemma is often used to demonstrate the coherence of strategic realism, which holds that in international relations, all countries (regardless of their internal policies or professed ideology), will act in their rational self-interest given international anarchy. A standard example is an arms race like the Cold War: during the Cold War the opposing alliances of NATO and the Warsaw Pact both had the choice to arm or disarm. From each side's point of view, disarming when their opponent continued to arm would have led to military inferiority and possible annihilation. Conversely, arming whilst their opponent disarmed would have led to superiority. If both sides chose to arm, neither could afford to attack the other, but both incurred the high cost of developing and maintaining a nuclear arsenal. If both sides chose to disarm, war would be avoided and there would be no costs.

[1] In 1993 Frank, Gilovich, and Regan conducted an experimental study of the prisoner's dilemma. The subjects were students in their first and final years of undergraduate economics, and undergraduates in other disciplines. Subjects were paired, placed in a typical game scenario, then asked to choose either to "cooperate" or to "defect".

First year economics students, and students doing disciplines other than economics, overwhelmingly chose to cooperate. But 4th year students in economics tended to not cooperate. Therefore, the authors concluded that that the study of economics reduces cooperation in games. The idea is that much of the time cooperation and consideration of other's perspective are irrational in the narrow sense of the word. Thus, learning that cooperation is irrational in some situations is influencing the behavior of the students towards less cooperation, presumably to the negative.

Although the "best" overall outcome is for both sides to disarm, the rational course for both sides is to arm, and this is indeed what happened. Both sides poured enormous resources into military research and armament in a war of attrition for the next thirty years until the Soviet Union could not withstand the economic cost.

Sports. Another example is doping in sports. Two competing athletes have the option to use an illegal and/or dangerous drug to boost their performance. If neither athlete takes the drug, then neither gains an advantage. If only one does, then that athlete gains a significant advantage over their competitor, reduced by the legal and/or medical dangers of having taken the drug. If both athletes take the drug, however, the benefits cancel out and only the dangers remain, putting them both in a worse position than if neither had used doping.

26. Nash Equilibrium

In the previous chapter we discussed several ways to find optimal behaviour for players. However, there is a big class of games where these method do not work; e.g. the game corresponding to Table 25.1.

26.1 No Regrets Strategies

The core ideas leading to understanding of the optimal behaviour are Nash equilibria and the notion of mixed strategies.

To understand the equilibria, consider the pair of strategies "cooperates", "cooperates" in prisoner's dilemma. This situation is bad because each player can change their behaviour and improve his/her situation; i.e., this situation is not stable.

Definition 26.1. *Let (X, Y, A, B) be a game in normal form. We say that a pair of strategies $x \in X$ and $y \in Y$ are in* Nash equilibrium *if*

$$A(x,y) \geq A(x',y)$$
$$\text{and}$$
$$B(x,y) \geq B(x,y')$$

for all $x' \in X$ and $y' \in Y$.

In other words, if two players decided to stick to x and y that are in Nash equilibrium, they do not regret not violating their agreement.

Exercise 26.1. • *Check that the pair of strategies "defects", "defects" are in Nash equilibrium in the prisoner's dilemma.*

• *Check that there are no Nash equilibrium in the game corresponding to Table 25.1.*

Note that some games may have several Nash equilibria. For example, let us consider the following game.

Game 26.1. *Two people are leaving in a flat. They need to call a plumber and stay in the flat while the plumber works.*

• *If none of them call the plumber, the water leaks and each of them will need to spend* 1 *hours cleaning the apartment.*

- *If one of them calls the plumber, he/she will spend* 1 *hour waiting pipes to be fixed.*
- *Alternatively they can call together and wait for one our together.*

This game is a version of volunteer's dilemma*.*

It is clear that this game can be described using the following table.

	do nothing	call the plumber
do nothing	-1, -1	0, -1
call the plumber	-1, 0	-1, -1

We may notice that in this game all pairs except "call the plumber", "call the plumber" are in Nash equilibrium. However, the pair of strategies "do nothing", "do nothing" is not efficient since one may improve the situation for one of the players without harming the other. Moreover, if the players know that this situation is going to repeat it self they may agree to call the plumber in turns and reduce the total time they wasted.

26.2 *Mixed Strategies*

Recall that we still do not have any optimal behaviour in the game corresponding to Table 25.1; however, it seems clear that in real life the best behaviour would be to flip coins and choose strategies accordingly.

This observation leads to the notion of mixed strategies.

Definition 26.2. *Let (X, Y, A, B) be a game in normal form. We say that a probability distribution $\overline{x}$ over X is a mixed strategy for the first player. Similarly we say that a probability distribution $\overline{y}$ over Y is a mixed strategy for the second player.*

We say that the payoffs of the players if they choose mixed strategies $\overline{x}$ and $\overline{y}$ are $A(\overline{x}, \overline{y})$ and $B(\overline{x}, \overline{y})$ such that

$$A(\overline{x}, \overline{y}) = \mathbb{E}_{x \leftarrow \overline{x}} \left[\mathbb{E}_{y \leftarrow \overline{y}} \left[A(x, y) \right] \right]$$

and

$$B(\overline{x}, \overline{y}) = \mathbb{E}_{x \leftarrow \overline{x}} \left[\mathbb{E}_{y \leftarrow \overline{y}} \left[B(x, y) \right] \right].$$

Exercise 26.2. *Let $f : \Omega_1 \times \Omega_2 \to \mathbb{R}$ and let $\mathcal{D}_1$ and $\mathcal{D}_2$ be probability distributions on Ω_1 and Ω_2, respectively. Show that*

$$\mathbb{E}_{\omega_1 \leftarrow \mathcal{D}_1} \left[\mathbb{E}_{\omega_2 \leftarrow \mathcal{D}_2} \left[f(\omega_1, \omega_2) \right] \right] = \mathbb{E}_{\omega_2 \leftarrow \mathcal{D}_2} \left[\mathbb{E}_{\omega_1 \leftarrow \mathcal{D}_1} \left[f(\omega_1, \omega_2) \right] \right].$$

Let us find the payoffs in the game corresponding to Table 25.1 if the players choose heads with the probability $1/2$; i.e., for the strategies $\overline{x}$

and $\overline{y}$ such that $\overline{x}(x) = \overline{y}(y) = 1/2$ for all $x \in X$ and $y \in Y$. It is clear that

$$A(\overline{x}, \overline{y}) = \frac{1}{4}A(\text{heads}, \text{heads}) + \frac{1}{4}A(\text{heads}, \text{tails}) + \\ \frac{1}{4}A(\text{tails}, \text{heads}) + \frac{1}{4}A(\text{tails}, \text{tails}) = 0.$$

It is clear that a mixed strategy $\overline{x}$ such that $\overline{x}(x) = 1$ is essentially the same as x so we call such a strategy a *pure strategy*.

The most important result about the connection between mixed strategies and Nash equilibrium is von Neumann's minimax theorem.

Theorem 26.1 (von Neumann's minimax theorem)**.** *Let (X, Y, A, B) be a zero-sum matrix game. Then there are mixed strategies $\overline{x}$ and $\overline{y}$ that are in Nash equilibrium; i.e.,*

$$A(\overline{x}, \overline{y}) \geq A(\overline{x}', \overline{y})$$
$$\text{and}$$
$$A(\overline{x}, \overline{y}) \leq A(\overline{x}, \overline{y}')$$

for all mixed strategies $\overline{x}'$ and $\overline{y}'$

Note that this theorem can be interpreted as the following equality

$$\max_{\overline{x}} \min_{\overline{y}} A(\overline{x}, \overline{y}) = \min_{\overline{y}} \max_{\overline{x}} A(\overline{x}, \overline{y}),$$

which give this theorem its name.

26.3 *Solution of All* 2×2 *Matrix Games.*

To better understand the notion of Nash equilibrium, let us devise a method to find it for games defined by 2×2 matrices.

Let $X = Y = 0, 1$ and let A be defined by the matrix

$$\begin{pmatrix} a & b \\ d & c \end{pmatrix}.$$

We need to consider two options: either there is an equilibrium in pure strategies, or there is no equilibrium in pure strategies. In the first case, there are x and y such that

$$A(x, y) \geq A(x', y)$$
$$\text{and}$$
$$A(x, y) \leq A(x, y')$$

for all x' and y' (such a point is also called a saddle point).

Assume that the game does not have a saddle point; without loss of generality we may also assume that $a \geq b$. Since $0, 1$ is not a saddle point $b < c$. The pair $1, 1$ is also not a saddle point so $d < c$. Finally, $d < a$ and $a > b$ since $1, 0$ and $0, 0$ are not saddle points either. Consider $\overline{x}$ such that $\overline{x}(0) = p$. Note that $A(\overline{x}, 0) = pa + (1-p)d = d + p(a-d)$ which is increasing when p is increasing and $A(\overline{x}, 1) = pb + (1-p)c = c + p(b-c)$ which is decreasing when p is increasing. Hence, the maximum is achieved when $p = (c-d)/(a-b+c-d)$. Similarly we may consider $\overline{y}$ such that $\overline{y}(0) = q$, and show that q should be equal to $(c-b)/(a-b+c-d)$.

27. *Yao's Principle*

Using game theory we may develop a method for proving lower bounds on complexity of randomized algorithms. To illustrate this method we will study the problem we duscussed in Chapters 9 and 16: Alice choose a number from $[1000]$, Bob can flip a coin and ask yes/no questions; Bob would like to guess it using, on average, as few questions as possible. We claim that Bob needs at least 9 questions.

Theorem 27.1. *Let Ω be a set of B-decision trees and $\mathcal{D}$ be a probability distribution on Ω. If $\mathrm{val}(T, x) = x$ for all $x \in [1000]$ and $T \in \Omega$, then $\mathbb{E}_{\mathcal{D}}\left[\mathrm{val}(T, x)\right] \geq 9$ for some $x \in [1000]$.*

To prove this statement we would need to give formal notions of randmoized B-decision tree complexity and heuristic B-decision tree cmplexity.

Definition 27.1. *Let $f : [1000] \to \mathbb{Z}$ be a function. We say that $\overline{\overline{T}} = (\Omega, \mathcal{T})$ is a* randomized[1] B-decision tree for f of cost C *if Ω is a set of B-decision trees, $\mathcal{T}$ is a probability distribution on Ω, $\mathrm{val}(T, x) = f(x)$ for all $x \in [1000]$, and $\mathbb{E}_{T \leftarrow \mathcal{T}}\left[\mathrm{h}(T, x)\right] = C$.*

We also say that T is a heuristic B-decision tree for f of expected cost C with respect to a distribution $\mathcal{X}$ *if $\mathrm{val}(T, x) = f(x)$ for all $x \in [1000]$ and $\mathbb{E}_{x \leftarrow \mathcal{X}}\left[\mathrm{h}(T, x)\right] = C$.*

The randmoized B-decision tree complexity $\mathrm{R}^{B\text{-dt}}(f)$ of f *is equal to the minimal C such that there is a randomized B-decision tree for f of cost C.*

The heuristic B-decision tree complexity $\mathrm{H}^{B\text{-dt}}_{\mathcal{X}}(f)$ of f with respect to $\mathcal{X}$ *is equal to the minimal C such that there is a B-decision tree for f of expected cost C with respect to $\mathcal{X}$. Finally,* the heuristic B-decision tree complexity $\mathrm{H}^{B\text{-dt}}(f)$ of f *is equal to the maximum of $\mathrm{H}^{B\text{-dt}}_{\mathcal{X}}(f)$ over all distributions $\mathcal{X}$.*

[1] In fact, there are many notions of randomized computation; the algorithms discussed in this chapter are usually called zero-error randomized algorithms.

Nonetheless that we have not had the notion of randomized complexity before, we proved upper bounds in Chapter 16. Unfortunately, proving lower bounds directly for randomized complexity is not easy, but it is usually way easier to prove lower bounds on heuristic complexity. Yao's principle (also called Yao's minimax principle or Yao's lemma) gives a connection between these measures.

Theorem 27.2 (Yao's principle)**.** *Let* $f : [1000] \to \mathbb{Z}$ *be a function. Then* $\mathrm{R}^{B\text{-dt}}(f) = \mathrm{H}^{B\text{-dt}}(f)$.

Using this principle is easy to prove the lower bound.

Proof of Theorem 27.1. Note that the theorem says that $\mathrm{R}^{B\text{-dt}}(I_{[1000]}) \geq 9$; hence, it is enough to prove that $\mathrm{H}^{B\text{-dt}}(I_{[1000]}) \geq 9$. Let us assume the opposite; i.e., that for any distribution $\mathcal{X}$ there is a B-decision tree T such that $\mathrm{val}(T, x) = x$ for all $x \in [1000]$ and $\mathbb{E}_{x \leftarrow \mathcal{X}}[\mathrm{h}(T, x)] < 9$. This implies that there is a B-decision tree T such that $\frac{1}{1000}\sum_{x \in [1000]} \mathrm{h}(T, x) < 9$; we are going to prove that this is impossible. Let us fix such a T.

We prove, using induction by $|S|$ that $\sum_{x \in S} \mathrm{h}(T, x) \geq |S| \log_2 |S|$. The base case for $|S| = 1$ is clear since $\mathrm{h}(T, x) \geq 1$ for all $x \in [1000]$.

Let us prove the induction step. Let $T = \textbf{if } f \textbf{ then } T_0 \textbf{ else } T_1$, and let $S_0 = \{x \in S : f(x) = 0\}$ and $S_1 = \{x \in S : f(x) = 1\}$. It is clear that $\sum_{x \in S} \mathrm{h}(T, x) = |S| + \sum_{x \in S_0} \mathrm{h}(T, x) + \sum_{x \in S_1} \mathrm{h}(T, x)$. By the induction hypothesis, this implies that $\sum_{x \in S} \mathrm{h}(T, x) = |S| + |S_0| \log_2 |S_0| + |S_1| \log_2 |S_1| \geq |S| + |S| \log_2(|S|/2) = |S| \log_2 |S|$.

Therefore, $\frac{1}{1000}\sum_{x \in [1000]} \mathrm{h}(T, x) \geq \log_2 1000 > 9$, which contradicts the assumption. $\square$

Now we are ready to prove Yao's principle.

Proof of Theorem 27.2. We start from proving that $\mathrm{R}^{B\text{-dt}}(f) \geq \mathrm{H}^{B\text{-dt}}(f)$. Assume the opposite; i.e., that $\mathrm{R}^{B\text{-dt}}(f) < \mathrm{H}^{B\text{-dt}}(f)$.

Let $\overline{T} = (\Omega, \mathcal{T})$ be a randomized B-decision tree for f of cost $C < \mathrm{H}^{B\text{-dt}}(f)$. Note that $\mathbb{E}_{T \leftarrow \mathcal{T}}[\mathrm{h}(T, x)] \leq C$ for all $x \in [1000]$. Hence, $\mathbb{E}_{x \leftarrow \mathcal{X}}[\mathbb{E}_{T \leftarrow \mathcal{T}}[\mathrm{h}(T, x)]] \leq C$. However,

$$C \geq \mathbb{E}_{x \leftarrow \mathcal{X}}[\mathbb{E}_{T \leftarrow \mathcal{T}}[\mathrm{h}(T, x)]] = \mathbb{E}_{T \leftarrow \mathcal{T}}[\mathbb{E}_{x \leftarrow \mathcal{X}}[\mathrm{h}(T, x)]] \geq \mathbb{E}_{T \leftarrow \mathcal{T}}\left[\mathrm{H}^{B\text{-dt}}_{\mathcal{X}}(f)\right] = \mathrm{H}^{B\text{-dt}}_{\mathcal{X}}(f).$$

As a result, $C \geq \mathrm{H}^{B\text{-dt}}(f)$, which is a contradiction to the assumption.

Let us prove that $\mathrm{R}^{B\text{-dt}}(f) \leq \mathrm{H}^{B\text{-dt}}(f)$. To prove this inequality we are going to use von Neumann's minimax theorem (Theorem 26.1). Let $X = [1000]$, Y be a set of all B-decision trees T such that $\mathrm{h}(T) \leq 1000$ and $\mathrm{val}(T, x) = f(x)$ for all $x \in [1000]$, and $A(x, T) = -B(x, T) = \mathrm{h}(T, x)$.

Theorem 26.1 implies that there are distributions $\mathcal{X}$ and $\mathcal{T}$ over X and Y respectively such that

$$\mathbb{E}_{x \leftarrow \mathcal{X}}\left[\mathrm{h}(T', x)\right] \geq \mathbb{E}_{x \leftarrow \mathcal{X}}[\mathbb{E}_{T \leftarrow \mathcal{T}}[\mathrm{h}(T, x)]] \geq \mathbb{E}_{T \leftarrow \mathcal{T}}\left[\mathrm{h}(T, x')\right]$$

for all $x' \in X$ and $T' \in Y$. Which implies that

$$\mathrm{H}^{B\text{-dt}}_{\mathcal{X}}(f) \geq \mathbb{E}_{x \leftarrow \mathcal{X}}[\mathbb{E}_{T \leftarrow \mathcal{T}}[\mathrm{h}(T, x)]] \geq \mathrm{R}^{B\text{-dt}}(f).$$

$\square$

Part VII

Introduction to Mathematical Logic

28. Propositional Logic

This part, as it follows fromt the title, is devoted to mathematical logic, a mathematical approach to a branch of philosophy called logic. Logic studies reasoning and mathematical logic studies mathematical reasoning. As we have mentioned in Chapter 1 proofs in mathematics consists of *sentences* of a certain structure that are connected by implications. In addition, as we discussed in Chapter 5, we can build larger sentences from smaller ones using connectives.

Note that in real life the sentences are written using common English which is ambiguous and therefore hard for analysis. So to create a formal description of mathematics we need to create an artificial formal language for mathematics.

First (Chapter 28) we will define a language for propositional (sentential) logic; i.e. the logic which deals only with propositions. Later (Chapter 29) we extend it to a logic which also takes properties of individuals into account.

The process of formalization of propositional logic consists of two main parts:

- present a formal language,
- specify a procedure for obtaining valid or true propositions.

Propositional Formulas:
Introduction to Mathematical Logic #1

`https://youtu.be/X0797bVFf3Y`

28.1 Propositional Formulas

Statements in propositional logic are either some independant atomic statements, or are formed from the atomic one using connectives.

In other words, statements in propositional logic can be defined using propositional formulas (also known as sentential formulas or Boolean formulas).

Definition 28.1. *We say that a finite sequence ϕ of elements of the set $V \cup \{\neg, \vee, \wedge, \to, \text{“(”}, \text{“)”}\}$ is a propositional formula on the variables from V if*

- *either ϕ is equal to x for some $x \in V$,*
- *or ϕ is equal to $(\psi_1 \wedge \psi_2)$, or $(\psi_1 \vee \psi_2)$, or $(\psi_1 \to \psi_2)$,*[1] *where ψ_1 and ψ_2 are propositional formulas on the variables from V,*

[1] The symbol $\to$ is used to denote the implication. Due to historical reasons the standard symbol $\implies$ is rarerely used as a connective in mathematical logic; hence, we will use $\to$ instead of $\implies$ in this part of the book. It is important to note that, sometimes the symbol $\supset$ is also used instead of $\implies$.

- *or ϕ is equal to $\neg\psi$, where ψ is a propositional formula on the variables from V.*

We denote the set of all propositional formulas by PROP_V.

For example, $((x_1 \vee \neg x_2) \wedge x_3)$ is a propositional formula on the variables from $\{x_1, x_2, x_3\}$ (we also say that it is a formula on x_1, x_2, x_3).

Exercise 28.1. *Write the definition of propositional formulas using the terminology "the set generated by ... from ..." (see Chapter 7).*

Hereafter when naming formulas, we will not mention explicitly all the parenthesis. To establish a more compact notation, we adopt the following conventions.

- The outermost parentheses do not need to be explicitly mentioned; e.g., we write "$A \wedge B$" to refer to $(A \wedge B)$.
- The negation symbol applies to as little as possible. For example, $\neg A \wedge B$ denotes $(\neg A) \wedge B$; i.e., $((\neg A) \wedge B)$. Which is not the same as $(\neg(A \wedge B))$.
- The conjunction and disjunction symbols apply to as little as possible, given that convention 2 is to be observed. For example, $A \wedge B \to \neg C \vee D$ is $((A \wedge B) \to ((\neg C) \vee D))$.
- Where one connective symbol is used repeatedly, grouping is to the right: $A \wedge B \wedge C$ is $A \wedge (B \wedge C)$, $A \to B \to C$ is $A \to (B \to C)$.

Interpreting propositional logic is not difficult since the considered entities have a simple structure. The propositions are built up from rough blocks by adding connectives. The simplest parts (atoms) are of the form "cows are animals", "Earth is flat", "$2 \times 2 = 2$", which are simply true or false. We extend this assignment of truth values to composite propositions, by reflection on the meaning of the logical connectives.

Definition 28.2. *A function $v : \mathrm{PROP}_V \to \{T, F\}$ is a valuation if*

- $v(\neg\psi) = \neg v(\psi)$,
- $v(\psi_1 \wedge \psi_2) = v(\psi_1) \wedge v(\psi_2)$,
- $v(\psi_1 \vee \psi_2) = v(\psi_1) \vee v(\psi_2)$, *and*
- $v(\psi_1 \to \psi_2) = v(\psi_1) \to v(\psi_2)$.

We may note that all the valuations are actualy can be defined by the values of variables.

Theorem 28.1. *Let $\rho : V \to \{T, F\}$ be a function (we say that ρ is a propositional assignement). Then there is a unique valuation $[\![\cdot]\!]_\rho$: $\mathrm{PROP}_V \to \{T, F\}$ such that $[\![x]\!]_\rho = \rho(x)$ for any $x \in V$.*

Since any valuation can be defined by the values assigned to variables, we need to introduce the following notation. If $V = \{x_1, \ldots, x_n\}$ and $v_1, \ldots, v_n \in \{\mathrm{T}, \mathrm{F}\}$, then $[\![\cdot]\!]_{x_1=v_1,\ldots,x_n=v_n}$ denotes the valuation such that $[\![x_i]\!]_{x_1=v_1,\ldots,x_n=v_n} = v_i$ for each $i \in [n]$.

For example, the value of a formula $(x_1 \wedge \neg x_2) \vee x_3$ when T is substituted as the value of x_1, T is substituted as the value of x_2, and F is substituted as the value of x_3 is equal to $(\mathrm{T} \wedge \mathrm{F}) \vee \mathrm{F} = \mathrm{F}$.

Note that if ϕ is a formula on the variables from V it does not mean that all the variables from V have to be used. For example, x_1 is a formula on the variables from $\{x_1, x_2\}$; however, x_2 is not used in the formula.

Exercise 28.2. *Define (using structural induction) the set of all the variables that are used in a propositional formula ϕ on variables from a set V.*

Let ϕ be a formula ϕ on the variables from a set V. The definition of a value of a formula requeres us to specify all the values of all the variables from V. However, the following theorem shows that in fact we need to specify only the variables that are actually used in ϕ.

Theorem 28.2. *Let ϕ be a formula ϕ on the variables from a set V, and U be the set of the variables used in ϕ.*

Consider $\rho_1, \rho_2 : V \to \{T, F\}$ such that $\rho_1(x) = \rho_2(x)$ for any $x \in U$. Then $[\![\phi]\!]_{\rho_1} = [\![\phi]\!]_{\rho_2}$.

Proof. We prove the statement using the structural induction.

(base case) Let $\phi = x$ for some $x \in V$. Note that $x \in U$ and $[\![\phi]\!]_{\rho_1} = \rho_1(x) = \rho_2(x) = [\![\phi]\!]_{\rho_2}$.

(induction step) We need to consider the following three cases.

- Let ϕ be equal to $\psi_1 \wedge \psi_2$ such that $[\![\psi 1]\!]_{\rho_1} = [\![\psi 1]\!]_{\rho_2}$ and $[\![\psi 1]\!]_{\rho_2} = [\![\psi 2]\!]_{\rho_2}$. In this case, $[\![\phi]\!]_{\rho_1} = ([\![\psi 1]\!]_{\rho_1} \wedge [\![\psi 2]\!]_{\rho_1}) = ([\![\psi 1]\!]_{\rho_2} \wedge [\![\psi 2]\!]_{\rho_2}) = [\![\phi]\!]_{\rho_2}$.
- Let ϕ be equal to $\psi_1 \vee \psi_2$ such that $[\![\psi 1]\!]_{\rho_1} = [\![\psi 1]\!]_{\rho_2}$ and $[\![\psi 1]\!]_{\rho_2} = [\![\psi 2]\!]_{\rho_2}$. In this case, $[\![\phi]\!]_{\rho_1} = ([\![\psi 1]\!]_{\rho_1} \vee [\![\psi 2]\!]_{\rho_1}) = ([\![\psi 1]\!]_{\rho_2} \vee [\![\psi 2]\!]_{\rho_2}) = [\![\phi]\!]_{\rho_2}$.
- Let ϕ be equal to $\psi_1 \to \psi_2$ such that $[\![\psi 1]\!]_{\rho_1} = [\![\psi 1]\!]_{\rho_2}$ and $[\![\psi 1]\!]_{\rho_2} = [\![\psi 2]\!]_{\rho_2}$. In this case, $[\![\phi]\!]_{\rho_1} = ([\![\psi 1]\!]_{\rho_1} \to [\![\psi 2]\!]_{\rho_1}) = ([\![\psi 1]\!]_{\rho_2} \to [\![\psi 2]\!]_{\rho_2}) = [\![\phi]\!]_{\rho_2}$.

□

Exercise 28.3. *Let ϕ_1, ϕ_2, and ϕ_3 be propositional formulas on the variables from a set V. Show that for any propositional assignement ρ to V, $[\![\phi_1 \wedge (\phi_2 \wedge \phi_3)]\!]_\rho = [\![(\phi_1 \wedge \phi_2) \wedge \phi_3]\!]_\rho$.*

28.2 Conjunctive and Disjuctive Normal Form

Let $\phi_1, \ldots, \phi_n$ be some propositional formulas. Then

- $\bigwedge_{i=1}^{1} \phi_i = \phi_1$ and $\bigvee_{i=1}^{1} \phi_i = \phi_1$, and
- $\bigwedge_{i=1}^{k+1} \phi_i = (\bigwedge_{i=1}^{k} \phi_i) \wedge \phi_{k+1}$ and $\bigvee_{i=1}^{k+1} \phi_i = (\bigvee_{i=1}^{k} \phi_i) \vee \phi_{k+1}$.

In other words $\bigwedge_{i=1}^{n} \phi_i$ and $\bigvee_{i=1}^{n} \phi_i$ denotes the conjunction of the formulas $\phi_1, \ldots, \phi_n$, and $\bigvee_{i=1}^{n} \phi_i$ denotes the disjunction of them.

Exercise 28.4. *Let $\phi_1, \ldots, \phi_n$, $\psi_1, \ldots, \psi_m$, $\chi_1, \ldots, \chi_{n+m}$ be some propositional formulas on the variables from V such that $\chi_i = \phi_i$ for $i \leq n$ and $\chi_i = \psi_{i-n}$ for $n < i \leq m$. Show that $\left[\!\left[(\bigwedge_{i=1}^{n} \phi_i) \wedge (\bigwedge_{i=1}^{n} \psi_i)\right]\!\right]_\rho = \left[\!\left[(\bigwedge_{i=1}^{n+m} \chi_i)\right]\!\right]_\rho$ for any propositional assignement ρ to V.*

Using this notation we may show that propositional formulas can represent all the Boolean functions (functions from $\{T, F\}^n$ to $\{T, F\}$).

Theorem 28.3. *For any function $f : \{T, F\}^n \to \{T, F\}$ there is a formula ϕ on the variables $x_1, \ldots, x_n$ such that $[\![\phi]\!]_{x_1=v_1,\ldots,x_n=v_n} = f(v_1, \ldots, v_n)$ for all $v_1, \ldots, v_n \in \{T, F\}$.*

Let $u \in \{T, F\}$ and $x \in V$. Then x^u denotes a formula on the variables from V such that $x^u = x$ if $u = T$ and $x^u = \neg x$ if $u = F$. Note that $[\![x^u]\!]_\rho = T$ iff $\rho(x) = u$, for any propositional assignement ρ to V. Indeed, if $u = T$, then $x^u = x$ and $T = [\![x^u]\!]_\rho = [\![x]\!]_\rho = \rho(x)$ so $\rho(x) = T = u$; if $u = F$, then $x^u = \neg x$ and $T = [\![x^u]\!]_\rho = [\![(\neg x)]\!]_\rho = \neg\rho(x)$ so $\rho(x) = F = u$.

Exercise 28.5. *Let $\phi_1, \ldots, \phi_k$ are propositional formulas on the variables from V.*

- *Show that $\left[\!\!\left[\left(\bigvee_{i=1}^{k} \phi_i\right)\right]\!\!\right]_\rho = T$ iff $[\![\phi]\!]_\rho = T$ for some $i \in [k]$.*
- *Show that $\left[\!\!\left[\left(\bigwedge_{i=1}^{k} \phi_i\right)\right]\!\!\right]_\rho = T$ iff $[\![\phi]\!]_\rho = T$ for all $i \in [k]$.*

Using this observation and the exercise we can prove Theorem 28.3.

Proof. Let $S = \left\{(u_1, \ldots, u_n) \in \{T, F\}^n \;:\; f(u_1, \ldots, u_n) = T\right\}$. Assume that $S = \left\{(u_{1,1}, \ldots, u_{1,n}), \ldots, (u_{k,1}, \ldots, u_{k,n})\right\}$. By the previous observations

$$\left[\!\!\left[\left(\bigvee_{i=1}^{k} \bigwedge_{j=1}^{n} x_j^{u_{i,j}}\right)\right]\!\!\right]_{x_1=v_1,\ldots,x_n=v_n} = f(v_1, \ldots, v_n)$$

for all $v_1, \ldots, v_n \in \{\mathrm{T}, \mathrm{F}\}$. (Note that we have not considered the case when $S = \emptyset$, in this case f is a constant F function and it is equal to $x_1 \wedge \neg x_1$.) $\square$

One may notice that the formulas we constructed have very specific form, such a form is called disjunctive normal form (DNF).

Definition 28.3. *We say that a propositional formula λ on the variables from V is a* literal *if it is equal to x or to $\neg x$ for some $x \in V$.*

We say that a propositional formula ψ on the variables from V is a term *if ψ is equal to $\bigwedge_{i=1}^{\ell} \lambda_i$, where $\lambda_1, \ldots, \lambda_\ell$ are literals.*

Finally, we say that a propositional formula ϕ on the variables from V is in disjunctive normal form *(DNF) if ϕ is equal to $\bigvee_{i=1}^{k} \psi_i$, where $\psi_1, \ldots, \psi_k$ are terms.*

However, there is nothing special in this order of operations (disjunction of conjunctions). So we can define conjunctive normal form (CNF) too.

Definition 28.4. *We say that a propositional formula ψ on the variables from V is a* clause *if ψ is equal to $\bigvee_{i=1}^{\ell} \lambda_i$, where $\lambda_1, \ldots, \lambda_\ell$ are literals.*

Finally, we say that a propositional formula ϕ on the variables from V is in conjunctive normal form *(DNF) if ϕ is equal to $\bigwedge_{i=1}^{k} \psi_i$, where $\psi_1, \ldots, \psi_k$ are clauses.*

Using the following simple trick we can prove that any function has a representation in CNF. First, we define a function $g(x_1, \ldots, x_n) = \neg f(x_1, \ldots, x_n)$. Secondly, we may notice that

$$\left[\!\!\left[\left(\neg \left(\bigwedge_{i=1}^{k} \bigvee_{j=1}^{n} \phi_{i,j} \right) \right) \right]\!\!\right]_{x_1 = v_1, \ldots, x_n = v_n} = \left[\!\!\left[\left(\bigvee_{i=1}^{k} \bigwedge_{j=1}^{n} \neg \phi_{i,j} \right) \right]\!\!\right]_{x_1 = v_1, \ldots, x_n = v_n}$$

for all $v_1, \ldots, v_n \in \{\mathrm{T}, \mathrm{F}\}$ (see Exercise 15.7). Therefore the negation of a formula in DNF can be easily tranformed into a formula in CNF. Finally, we know that the function g has a representation in DNF, which implies that f has a representation in CNF.

Proofs Using Truth Tables:
Introduction to Mathematical Logic #2

`https://youtu.be/DOAxnmScpPc`

28.3 Truth Tables

Typical theorem in mathematics have the following template: "if some statements are true, then some statement is also true". In propositional logic statements are described using propositional formulas. So our goal is to present a way to describe proofs of results that looks like: if $\phi_1, \ldots, \phi_k$ are true, then ψ is also true.

This section discusses the method which is based on truth tables (we discussed it before in Chapter 5).

We start from an example similar to the proof gaven in the beginning of the first chapter. Assume that we know that if x is a real number such that $x < -2$ or $x > 2$, then $x^2 > 4$. We can derive that if $\neg(x^2 > 4)$, then $\neg(x < -2)$ and $\neg(x > 2)$.

In order to emphasize the logical structure of the argument let us denote the statement $x > 2$ by p, the statement $x < -2$ by q, and the statement $x^2 > 4$ by r. In this case the argument is as follows: if $(p \vee q) \to r$ is true, then $\neg r \to (\neg p \wedge \neg q)$ is true as well.

The simplest way to explain why this argument is true is to use a truth table.

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

Note that each line where $(p \vee q) \to r$ is true has $\neg r \to (\neg p \wedge \neg q)$ true as well. So we proved that the argument is indeed correct.

We may also note that we showed that

$$\big((p \vee q) \to r\big) \iff \big(\neg r \to (\neg p \wedge \neg q)\big)$$

is always true (we say that this propositional formula is a *tautology*). A generalization of this saying the if $p \to q$ is true, then $\neg q \to \neg p$ is also true is called the *contraposition* argument.

Let us now consider another argument. If we know that Joe was a good boy and we know that if Joe is a good boy, then Santa gives a present to Joe. We may conclude that Santa gives a present to Joe. We can similarly to the previous example write this argument using variables and connectives. If we know that p and $p \to q$, we may conclude that q is true.

Exercise 28.6. *Show that $(p \wedge (p \to q)) \to q$ is a tautology.*

Such an argument is called *modus ponens*.

A notion connected to being a tautology is the notion of being satisfiable. We say that a formula (a set of formulas) is *satisfiable* iff there is a substitution to the variables such that the value of the formula is true (the values of all the formulas are true). Note that a formula is not satisfiable (the formula is *unsatisfiable*) iff its negation is a tautology. Therefore, using truth tables one may check whether a formula is satisfiable or not.[2]

[2] Note that the procedure is awfully not efficient since if the formula uses n variables we need to do 2^n operations. Unfortunately, we do not know anything that always works better since satisfiability problem (the problem of determining whether a given formula is satisfiable or not) is NP complete.

28.4 Semantic Implication

As we mentioned at the beginning of the previous section, most of the statements in mathematics are in the form "if some statements are true, then some statement is also true"; this type of statements can be described using the notion of semantic implication. We say that a set Σ of propositional formulas with variables from a set V *semantically implies* a propositional formula ϕ with variables from the set V (we denote it by $\Sigma \models \phi$) iff whenever all the formulas from Σ are true under some propositional assignement to V, the formula ϕ is also true under this propositional assignement; i.e., $\Sigma \models \phi$ iff for any $\rho : V \to \{\mathrm{T}, \mathrm{F}\}$, $[\![\phi]\!]_\rho = \mathrm{T}$ provided that $[\![\psi]\!]_\rho = \mathrm{T}$ for all $\psi \in \Sigma$. (Note that the set Σ may be infinite.)

In the previous section we explained that if we have a finite set Σ, then it is possible to check whether a formula ϕ is semantically implied by Σ. Let us try to find out whether we can do the same for infinite sets Σ.

Partial answer to this question is given by the following theorem.

Theorem 28.4 (compactness theorem)**.** *A set Σ of propositional formulas is satisfiable iff every finite subset is satisfiable.*

Proof. We say that a set is *finitely satisfiable* if every finite subset is satisfiable.

Let us enumerate all the propositional formulas $\alpha_1, \alpha_2, \ldots$. We define a family of sets $\Delta_1, \ldots, \Delta_n, \ldots$ such that $\Delta_1 = \Sigma$ and

$$\Delta_{n+1} = \begin{cases} \Delta_n \cup \{\alpha_{n+1}\} & \text{if } \Delta_n \cup \{\alpha_{n+1}\} \text{ is finitely satisfiable,} \\ \Delta_n \cup \{\neg\alpha_{n+1}\} & \text{otherwise.} \end{cases}$$

Note that all the Δ_n are finitely satisfiable.

Let $\Delta = \cup_{n \in \mathbb{N}} \Delta_n$. It is clear that Δ is finitely satisfiable and for any propositional formula α, either α or $\neg\alpha$ belongs to Δ.

Let us consider a substitution $v_1, \ldots, v_n, \ldots$ to the variables $x_1, \ldots, x_n, \ldots$ such that $v_i = \mathrm{T}$ iff the formula x_i belongs to Δ. We may note that this substitution satisfies any formula $\phi \in \Delta$. □

Using this theorem, we can show that any implication of an infinite set is actually an implication of a finite subset of it.

Corollary 28.1. *Let Σ be a set of propositional formulas over the variables $x_1, x_2, \ldots, x_n, \ldots$, and ϕ be a propositional formula over the same set. If $\Sigma \models \phi$, then there is a finite $\Sigma' \subseteq \Sigma$ such that $\Sigma' \models \phi$.*

Proof. Note that $\Sigma \not\models \phi$ iff $\Sigma \cup \{\phi\}$ is satisfiable.

Let us now assume that for any finite $\Sigma' \subseteq \Sigma$, $\Sigma' \not\models \phi$. This implies that $\Sigma' \cup \{\phi\}$ is satisfiable for all finite Σ'. Therefore, $\Sigma \cup \{\phi\}$ is satisfiable, which is a contradiction to the assumption that $\Sigma \models \phi$. □

Therefore if we wish to check whether a formula ϕ is semantically implied by Σ, we just need to brute-force all the finite subsets of Σ and check whether they semantically imply ϕ. By the previous argument, if ϕ is implied by Σ, this procedure reports "yes" at some point, and in the opposite case it will work infinitely long.

Natural Deduction:
Introduction to Mathematical Logic #3

https://youtu.be/PfVafyptFtM

28.5 Natural Deduction

The problem of the method discussed in Section 28.3 is that we need to consider **all** possible values of the variables. Let us now consider a more complicated example. Imagine that we know that $\neg q$, $p \to q$. Using the contraposition argument and modus ponens we may derive $\neg p$. Indeed, by contraposition we may conclude that $\neg q \to \neg p$ and modus ponens implies that $\neg p$ is true since $\neg q$ is true.

In other words, we can combine several tautologies to prove another tautology. Apparently it is enough to fix some small number of tautologies to derive all other tautologies, we call these tautologies "rules". There are several ways to write such proofs, we are going to use Fitch notation for natural deduction. In this notation any proof is written in several rows, each row in a Fitch-style proof is either:

- an assumption or subproof assumption.

- a sentence justified by the citation of (i) a rule of inference and (ii) the prior line or lines of the proof that license that rule.

We say that there is a natural deduction derivation of ϕ from $\psi_1, \ldots,$ ψ_k. If there is a Fitch-style proof starting with the assumptions $\psi_1, \ldots,$ ψ_k, and finishes with the formula ϕ. Using this scheme we may write the argument we just mentioned as follows.

1	$\neg q$	
2	$p \to q$	
3	$\neg q \to \neg p$	contraposition, 2
4	$\neg p$	modus ponens, 1, 3

In the rest of the section we are going to list all the rules we use.

Conjunctions. In order to introduce a conjunction we can use the following rule.

$$\begin{array}{r|ll} m & A & \\ n & B & \\ & A \wedge B & \wedge\text{I}, m, n \end{array}$$

This rule corresponds to the tautology $(A \wedge B) \to (A \wedge B)$.

In order to eliminate conjunctions we can use the following two rules.

$$\begin{array}{r|ll} m & A \wedge B & \\ & A & \wedge\text{E}, m \end{array} \qquad \begin{array}{r|ll} m & A \wedge B & \\ & B & \wedge\text{E}, m \end{array}$$

These rules correspond to the tautologies $(A \wedge B) \to A$ and $(A \wedge B) \to B$.

Disjuctions. In order to introduce a disjunction we can use the following two rules.

$$\begin{array}{r|ll} m & A & \\ & A \vee B & \vee\text{I}, m \end{array} \qquad \begin{array}{r|ll} m & A & \\ & B \vee A & \vee\text{I}, m \end{array}$$

These rules correspond to the tautologies $A \to (A \vee B)$ and $A \to (B \vee A)$.

In order to eliminate a disjunction we can use the following rule.

$$\begin{array}{r|ll} m & A \vee B & \\ i & \quad \begin{array}{|l} A \\ \hline \end{array} & \\ j & \quad \begin{array}{|l} C \end{array} & \\ k & \quad \begin{array}{|l} B \\ \hline \end{array} & \\ l & \quad \begin{array}{|l} C \end{array} & \\ & C & \vee\text{E}, m, i\text{–}j, k\text{–}l \end{array}$$

This rule corresponds to the tautology $\big((A \vee B) \wedge (A \to C) \wedge (B \to C)\big) \to C$.

Implications. In order to introduce an implication we can use the following two rules.

$$\begin{array}{r|ll} i & \quad \begin{array}{|l} A \\ \hline \end{array} & \\ j & \quad \begin{array}{|l} B \end{array} & \\ & A \to B & \Rightarrow\text{I}, i\text{–}j \end{array}$$

This rule corresponds to the tautology $(A \to B) \to (A \to B)$.

In order to eliminate an implication we can use the following rule.

$$\begin{array}{r|ll} m & A \to B & \\ n & A & \\ & B & \Rightarrow\mathrm{E}, m, n \end{array}$$

This rule corresponds to the tautology $\big((A \to B) \wedge A\big) \to B$.

Negations. In order to introduce a negation we can use the following two rules ($\perp$ is a special symbol representing a false statement).

$$\begin{array}{r|ll} i & \quad\begin{array}{|l} A \\ \hline \end{array} & \\ j & \quad\begin{array}{|l} \perp \end{array} & \\ & \neg A & \neg\mathrm{I}, i\text{–}j \end{array}$$

This rule corresponds to the tautology $(A \to \perp) \to \neg A$.

In order to eliminate a negation we can use the following rule.

$$\begin{array}{r|ll} m & A & \\ n & \neg A & \\ & \perp & \neg\mathrm{E}, m, n \end{array}$$

This rule corresponds to the tautology $(A \wedge \neg A) \to \perp$.

Truths and falsities. Additionally, we have the following two rules.

$$\begin{array}{r|ll} m & \perp & \\ & A & \perp\mathrm{E}, m \end{array} \qquad \begin{array}{r|ll} i & \quad\begin{array}{|l} \neg A \\ \hline \end{array} & \\ j & \quad\begin{array}{|l} \perp \end{array} & \\ & A & \mathrm{IP}, i, j \end{array}$$

An online tool to check natural deduction proofs

https://proofs.openlogicproject.org/

Exercise 28.7. *Check that all the tautologies we mentioned are indeed tautologies.*

28.6 Examples of Derivations

In this section we give several derivations using the rules we just introduced.

First, we prove that if we know that $A \to \neg A$ we can derive that $\neg A$.

1	$A \to \neg A$	
2	$\quad A$	
3	$\quad \neg A$	$\Rightarrow$E, 1, 2
4	$\quad \bot$	$\neg$E, 2, 3
5	$\neg A$	$\neg$I, 2–4

Another statement we are going to prove is that if $A \to (A \wedge \neg A)$ is true, then $\neg A$ is also true.

1	$A \to (A \wedge \neg A)$	
2	$\quad A$	
3	$\quad A \wedge \neg A$	$\Rightarrow$E, 1, 2
4	$\quad \neg A$	$\wedge$E, 3
5	$\quad \bot$	$\neg$E, 2, 4
6	$\neg A$	$\neg$I, 2–5

A bit more complicated is the proof of the law of excluded middle: $A \vee \neg A$.

1		
2	$\quad \neg(A \vee \neg A)$	
3	$\quad\quad A$	
4	$\quad\quad A \vee \neg A$	$\vee$I, 3
5	$\quad\quad \bot$	$\neg$E, 2, 4
6	$\quad \neg A$	$\neg$I, 3–5
7	$\quad A \vee \neg A$	$\vee$I, 6
8	$\quad \bot$	$\neg$E, 2, 8
9	$A \vee \neg A$	IP, 2–8

28.7 Soundness and Completeness

Soundness and Completeness: Introduction to Mathematical Logic #4

https://youtu.be/9Utsppn-M_I

The most important properties of the natural deduction are the following two theorems.

Theorem 28.5 (completeness of natural deductions)**.** *Let ϕ be a propositional formula. If ϕ is a tautology, then there is a proof of ϕ. Moreover if Σ is a finite set of propositional formulas and $\Sigma \models \phi$, then there is a derivation of ϕ from Σ.*

Theorem 28.6 (soundness of natural deductions)**.** *Let ϕ be a proposi-*

tional formula. If there is a proof of ϕ, then ϕ is a tautology. Moreover if Σ is a finite set of propositional formulas and there is a derivation of ϕ from Σ, then $\Sigma \models \phi$.

Proofs of these two theorems are not that difficult but very technical. So prove these statements on examples to at least illustrate them.

Completeness of natural deductions. Proofs of this statement exploit the following idea: if a propositional formula is a tautology, then we can verify this statement using the truth table. So the proof simply brute-forces all the values of the variables of a formula and checks that the formula is indeed true. Consider a tautology $(\neg A \wedge \neg B) \to \neg(A \vee B)$. The proof of this tautology is as follows.

First we derive $A \vee \neg A$ and $B \vee \neg B$, and we use these two formulas to consider cases using the elimination of disjunction.

1		
2	$A \vee \neg A$	the law of excluded middle
3	$B \vee \neg B$	the law of excluded middle

After that, we consider the case when A and B are both true. Note that the assumption of the implication is false in this case. Thus, we just need to assume $\neg A \wedge \neg B$, derive the contradiction, and derive $\neg(A \vee B)$.

4	A	
5	B	
6	$\neg A \wedge \neg B$	
7	$\neg A$	$\wedge$E, 6
8	$\perp$	$\neg$E, 4, 7
9	$\neg(A \vee B)$	$\perp$E, 8
10	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\Rightarrow$I, 6–9

After that, we consider the case when A is true but and B is false. In this case, the assumption of the implication is also false; thus, the proof is the same as in the previous case.

11	$\neg B$	
12	$\neg A \wedge \neg B$	
13	$\neg A$	$\wedge$E, 12
14	$\perp$	$\neg$E, 4, 13
15	$\neg(A \vee B)$	$\perp$E, 14
16	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\Rightarrow$I, 6–9
17	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\vee$E, 2, 5–10, 11–16

The third case is when A is false and B is true. In this case the assumption of the implication is false again, thus the proof is the same as in the previous two cases.

18	$\neg A$	
19	B	
20	$\neg A \wedge \neg B$	
21	$\neg B$	$\wedge$E, 20
22	$\perp$	$\neg$E, 19, 22
23	$\neg(A \vee B)$	$\perp$E, 22
24	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\Rightarrow$I, 20–23

Finally, we consider the case when A and B are false. In this case the assumption of the implication is true, and since the formula is a tautology and $\neg A \wedge \neg B$ is true, we know that $\neg(A \vee B)$ is also true. Assume that $A \vee B$ is true and note that this is impossible. Thus using introduction of the negation we can prove the statement.

25	$\neg B$	
26	$\neg A \wedge \neg B$	
27	$A \vee B$	
28	A	
29	$\bot$	$\neg$E, 18, 28
30	B	
31	$\bot$	$\neg$E, 25, 30
32	$\bot$	$\vee$E, 27, 28–29, 30–31
33	$\neg(A \vee B)$	$\neg$E, 26–32
34	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\Rightarrow$I, 26–33
35	$(\neg A \wedge \neg B) \to \neg(A \vee B)$	$\vee$E, 1, 3–17, 18–34

Soundness of natural deductions. Idea behind the soundness is also simple. We just explain that every line of the proof represent a tautology, including the last one. We illustrate this on the exaple of the proof of $A \vee \neg A$. Recall that the proof of this tautology is the following.

1	
2	$\neg(A \vee \neg A)$
3	A
4	$A \vee \neg A$
5	$\bot$
6	$\neg A$
7	$A \vee \neg A$
8	$\bot$
9	$A \vee \neg A$

1. The second line is just an assumption, so the corresponding tautology is $\neg(A \vee \neg A) \to \neg(A \vee \neg A)$.
2. Line 3 is also an assumption so the corresponding tautology is $\neg(A \vee \neg A) \to (A \to A)$.
3. Line 4 is a formula $A \vee \neg A$ which we derived under assumptions $\neg(A \vee \neg A)$ and A, so the corresponding tautology is $\neg(A \vee \neg A) \to$

$(A \rightarrow (A \vee \neg A))$ (it is a tautology since we replaced A by $A \vee \neg A$ in the conclusion of the formula corresponding to Line 3).

4. Line 5 is a formula $\perp$ which we derived under assumptions $\neg(A \vee \neg A)$ and A, so the corresponding tautology is $\neg(A \vee \neg A) \rightarrow (A \rightarrow \perp)$ (it is a tautology since on Line 4 we explained that $\neg(A \vee \neg A) \rightarrow (A \rightarrow (A \vee \neg A))$).

5. Line 6 is a formula $\neg A$ which we derived under assumptions $\neg(A \vee \neg A)$, so the corresponding tautology is $\neg(A \vee \neg A) \rightarrow \neg A$ (it is a tautology since on Line 5 we explained that $A \rightarrow \perp$ under the assumption $\neg(A \vee \neg A)$).

6. Line 7 is a formula $A \vee \neg A$ which we derived under assumptions $\neg(A \vee \neg A)$, so the corresponding tautology is $\neg(A \vee \neg A) \rightarrow (A \vee \neg A)$ (it is a tautology since on Line 6 we explained that A under the assumption $\neg(A \vee \neg A)$).

7. Line 8 is a formula $\perp$ which we derived under assumptions $\neg(A \vee \neg A)$, so the corresponding tautology is $\neg(A \vee \neg A) \rightarrow \perp$ (it is a tautology since on Line 6 we explained that $A \vee \neg A$ under the assumption $\neg(A \vee \neg A)$).

8. Finally, Line 9 is a formula $A \vee \neg A$ (it is a tautology since we proved that $\neg(A \vee \neg A) \rightarrow \perp$ is a tautology)

End of The Chapter Exercises

28.8 Let ϕ_1 and ϕ_2 be some propositional formulas on the variables from V. Show that for any propositional assignement ρ to V,

- $[\![\neg\,(\phi_1 \wedge \phi_2)]\!]_\rho = [\![(\neg\phi_1 \vee \neg\phi_2)]\!]_\rho$ and
- $[\![\neg\,(\phi_1 \vee \phi_2)]\!]_\rho = [\![(\neg\phi_1 \wedge \neg\phi_2)]\!]_\rho$.

28.9 Let $\phi_1, \ldots, \phi_n$ be some propositional formulas on the variables from V. Show that for any propositional assignement ρ to V,

- $[\![(\neg\,(\bigwedge_{i=1}^{n} \phi_i))]\!]_\rho = [\![(\bigvee_{i=1}^{n} \phi_i)]\!]_\rho$ and
- $[\![(\neg\,(\bigvee_{i=1}^{n} \phi_i))]\!]_\rho = [\![(\bigwedge_{i=1}^{n} \phi_i)]\!]_\rho$.

28.10 Write a natural deduction derivation of $A \vee C$ from hypothesis $(A \wedge B) \vee C$.

28.11 Write a natural deduction derivation of $B \vee C$ from hypothesis $A \rightarrow B$ and $\neg A \rightarrow C$.

28.12 Write a natural deduction derivation of $(W \vee Y) \rightarrow (X \vee Z)$ from hypotheses $W \rightarrow X$ and $Y \rightarrow Z$.

28.13 Let us formulate the pigeonhole principle using propositional formulas. Let $V = \{x_{1,1}, \ldots, x_{n+1,1}, x_{1,2} \ldots, x_{n+1,n}\}$ (informally $x_{i,j}$ is true iff the ith pigeon is in the jth hole). Consider the following propositional formulas on the variables from V.

- L_i ($i \in [n+1]$) is equal to $\bigvee_{j=1}^{n} x_{i,j}$. (Informally this formula says that the ith pigeon is in a hole.)
- R_j ($j \in [n]$) is equal to $\bigvee_{i_1=1}^{n+1} \bigvee_{i_2=i_1+1}^{n+1} (x_{i_1,j} \wedge x_{i_2,j})$. (Informally this formula says that there are two pigeons in the jth hole.)

Show that there is a natural deduction derivation of $\left(\bigwedge_{i=1}^{n+1} L_i\right) \rightarrow (\bigvee_{i=1}^{n} R_i)$.

28.14 Let $\phi = \bigvee_{i=1}^{m} \lambda_i$ be a clause; we say that the width of the clause is equal to m. Let $\phi = \bigwedge_{i=1}^{\ell} \chi_i$ be a formula in CNF (χ_i's are clauses'); we say that the width of ϕ is equal to the maximal width of χ_i for $i \in [\ell]$.

Let $p_n : \{T, F\}^n \to \{T, F\}$ such that $p_n(x_1, \ldots, x_n) = T$ iff the set $\{i : x_i = T\}$ has an odd number of elements. Show that any CNF representation of p_n has width n.

28.15 In this exercise we think about clauses as sets of literals so the order of disjunctions and repetitions of literals are not important. We say that a clause C can be obtained from clauses A and B using the *resolution* rule if $C = A' \vee B'$, $A = x \vee A'$, and $B = \neg x \vee B'$, for some variable x.

We say that a clause C can be derived from clauses $A_1, \ldots, A_m$ using resolutions if there is a sequence of clauses $D_1, \ldots, D_\ell = C$ such that each D_i

- is either obtained from clauses D_j and D_k for $j, k < i$ using the *resolution* rule, or
- is equal to A_j for some $j \in [m]$, or
- is equal to $D_j \vee E$ for some $j < i$ and a clause E.

Show that if an empty clause $\perp$ can be derived from clauses $A_1, \ldots, A_m$ using the resolution rule, then $A_1, \ldots, A_m$ semantically imply $\perp$.

29. Predicate Logic

In the previous chapter we defined natural deductions for propositional logic. But in real mathematics there are many formulas that are not propositional. For example we may wish to prove that if a relation R on M is transitive, then

$$(R(w,x) \wedge R(x,y) \wedge R(y,z)) \implies R(w,z)$$

is true for any $w, x, y, z \in M$. In this chapter we define a logical system that allows us to formally prove such statements.

Predicate Formulas:
Introduction to Mathematical Logic #5

https://youtu.be/yb9NvmXyFfg

29.1 Predicate Formulas

Let us write the previous statement in a formula-like form:

$$\overbrace{(\forall x,y,z \in M\ (R(x,y) \wedge R(y,z)) \implies R(x,z))}^{R \text{ is transitive}} \implies \underbrace{(\forall w,x,y,z \in M\ (R(w,x) \wedge R(x,y) \wedge R(y,z)) \implies R(x,z))}_{\text{the desired conclusion}}.$$

Note that there are several things we need to explain if we wish to define formally formulas like this:

- we need to explain what kind of sets we can use (in this case we need to define M),
- we need to explain what kind of relations we can use (in this case we need to define R),

Another example of a statement we may wish to prove is saying that if $f : M \to M$ is an inverse of itself (i.e. $f(f(x)) = f(x)$ for any $x \in M$), then $f(f(f(x))) = f(x)$ for any $x \in M$; more formally, we may wish to prove a statement

$$\underbrace{(\forall x \in M\ f(f(x)) = x)}_{f \text{is an inverse of itself}} \implies \underbrace{(\forall x \in M\ f(f(f(x))) = f(x))}_{\text{the desired conclusion}}.$$

In order to explain what we mean by such formulas

- we need to explain what kind functions we can use (in this case we need to define f).

Signature. In predicate logic, formula uses just symbols for all these objects. We specify these symbols only when we wish to compute actual truth value of the formula. We also assume that all the quantifiers are over the same set so we do not need a symbol for the set M.

Signature is the way to define the list of all these symbols, it consists of three objects:

- the set (possibly empty) of symbols for relations,
- the set (possibly empty) of symbols for functions,
- arities of these functions and relations (i.e. how many arguments they may take).

An example of a signature is a triple $(\{\text{“R”}\}, \{\text{“f”}\}, \text{ar})$, where

$$\text{ar}(s) = \begin{cases} 2 & \text{if } s = \text{“R”} \\ 1 & \text{if } s = \text{“f”} \end{cases}.$$

This signature is enough to define the formulas we discussed. Now we are ready to define the predicate formulas.

Definition 29.1. *Let $\mathcal{S} = (S_{\text{rel}}, S_{\text{fun}}, a)$ be a signature.*

We say that t is a term *in the signature $\mathcal{S}$ over the variables $x_1, \ldots, x_n$ if*

- *either t is equal to a variable x_i*
- *or t is equal to $f(t_1, \ldots, t_\ell)$, where $f \in S_{\text{fun}}$, $\ell = a(f)$, and $t_1, \ldots, t_\ell$ are terms in the signature $\mathcal{S}$.*

We say that ϕ is a predicate formula *in the signature $\mathcal{S}$ over the variables $x_1, \ldots, x_n$ if*

- *either ϕ is equal to $R(t_1, \ldots, t_\ell)$, where $R \in S_{\text{rel}}$, $\ell = a(R)$, and $t_1, \ldots, t_\ell$ are terms in the signature $\mathcal{S}$.*
- *or ϕ is equal to $(\psi_1 \wedge \psi_2)$, or $(\psi_1 \vee \psi_2)$, or $(\psi_1 \implies \psi_2)$, where ψ_1 and ψ_2 are predicate formulas in the signature $\mathcal{S}$,*
- *or ϕ is equal to $\neg\psi$, where ψ is a predicate formula in the signature $\mathcal{S}$,*
- *or ϕ is equal to $\exists x_i\ \psi$ or $\forall x_i\ \psi$ where ψ is a predicate formula in the signature $\mathcal{S}$.*

In order to compute the truth value of a predicate formula, we need to specify the values of all the free variables and all the symbols from the signature. The specification of the symbols from the signature is called structure; i.e. a structure for a signature $\mathcal{S} = (S_{\text{rel}}, S_{\text{fun}}, a)$ is a triple $(M, F_{\text{rel}}, F_{\text{fun}})$ such that

- $F_{\text{rel}} : S_{\text{rel}} \to \bigcup_{i=0}^{\infty} 2^{M^i}$ such that $F_{\text{rel}}(R) \in 2^{M^{a(R)}}$ and

- $F_{\text{fun}} : S_{\text{fun}} \to \bigcup_{i=0}^{\infty} M^{M^i}$ such that $F_{\text{fun}}(f) \in M^{M^{a(f)}}$.

The set M in the structure is called the domain of the structure.

Definition 29.2. *Let $\mathcal{S} = (S_{\text{rel}}, S_{\text{fun}}, a)$ be a signature and $\mathcal{M} = (M, F_{\text{rel}}, F_{\text{fun}})$ be a structure for $\mathcal{S}$.*

Let t be a term in the signature $\mathcal{S}$ over the variables x_1, …, x_n and $v_1, \dots, v_n \in M$. The value of t with $x_1 = v_1$, …, $x_n = t_n$ with respect to the structure $\mathcal{M}$ is equal

- *either to v_i when $t = x_i$,*
- *or $F_{\text{fun}}(f)(\mu_1, \dots, \mu_{a(f)})$ when $t = f(t_1, \dots, t_{a(f)})$, where μ_i is equal to the value of t_i with $x_1 = v_1$, …, $x_n = v_n$ with respect to the structure $\mathcal{M}$.*

Let ϕ be a formula in the signature $\mathcal{S}$ over the variables $x_1, \dots, x_n$.

- *Let ϕ be equal to $F_{\text{rel}}(R)(t_1, \dots, t_{a(R)})$, where $t_1, \dots, t_n$ are some terms in $\mathcal{S}$. Then the value of ϕ with $x_1 = v_1$, …, $x_n = v_n$ with respect to $\mathcal{M}$ is equal to $R(\mu_1, \dots, \mu_{a(R)})$, where μ_i is equal to the value of t_i with $x_1 = v_1, \dots, x_n = v_n$ with respect to $\mathcal{M}$.*
- *Let ϕ be equal to $\psi_1 \# \psi_2$, where $\# \in \{\vee, \wedge\}$ and ψ_1, ψ_2 are predicate formulas. Then the value of ϕ with $x_1 = v_1, \dots, x_n = v_n$ with respect to $\mathcal{M}$ is equal to $\beta_1 \# \beta_2$, where β_i is equal to the value of ψ_i with $x_1 = v_1$, $\dots$, $x_n = v_n$ with respect to $\mathcal{M}$.*
- *Let ϕ be equal to $\neg\psi$, where ψ is a predicate formula. Then the value of ϕ with $x_1 = v_1, \dots, x_n = v_n$ with respect to $\mathcal{M}$ is equal to $\neg\beta$, where β is equal to the value of ψ with $x_1 = v_1, \dots, x_n = v_n$ with respect to $\mathcal{M}$.*
- *Let ϕ be equal to $\exists x_i\ \psi$, where ψ is a predicate formula. Then the value of ϕ with $x_1 = v_1, \dots, x_n = v_n$ with respect to $\mathcal{M}$ is equal to true iff there is $\mu \in M$ such that the value of ψ with $x_1 = v_1$, …, $x_{i-1} = v_{i-1}$, $x_i = \mu$, $x_{i+1} = v_{i+1}, \dots x_n = v_n$ with respect to $\mathcal{M}$.*
- *Let ϕ be equal to $\forall x_i\ \psi$, where ψ is a predicate formula. Then the value of ϕ with $x_1 = v_1$, …, $x_n = v_n$ with respect to $\mathcal{M}$ is equal to true iff for all $\mu \in M$, the value of ψ with $x_1 = v_1$, …, $x_{i-1} = v_{i-1}$, $x_i = \mu$, $x_{i+1} = v_{i+1}, \dots x_n = v_n$ with respect to $\mathcal{M}$.*

We say that $\mathcal{M}$ is a model of a formula ϕ (written $\mathcal{M} \models \phi$)[1] over the variables x_1, …, x_n iff the value of ϕ with $x_1 = v_1$, …, $x_n = v_n$ with respect to $\mathcal{M}$ is equal to T for all $v_1, \dots, v_n \in \{\mathrm{T}, \mathrm{F}\}$.

We also say that ϕ is true in $\mathcal{M}$ if $\mathcal{M} \models \phi$, and we say that ϕ is false in $\mathcal{M}$ if $\mathcal{M} \not\models \phi$.

Let us consider an example:

[1] Sometimes "$\mathcal{M}$ is a model of ϕ" is written as $\models_{\mathcal{M}} \phi$.

- First, we define a signature $\mathcal{S} = (\{=,<\}, \{+,\cdot\}, \mathrm{ar})$ (if the arities of the symbols are clear from the context, we can write $S = (=,<;+,\cdot)$), where $\mathrm{ar}(x) = 2$ for any $x \in \{<,=,+,\cdot\}$.

- After this we define a structure $\mathcal{M} = (\mathbb{R}, F_{\mathrm{rel}}, F_{\mathrm{fun}})$, where

$$F_{\mathrm{fun}}(f)(x,y) = \begin{cases} x \cdot y & \text{if } f \text{ is } \cdot \\ x+y & \text{if } f \text{ is } + \end{cases}$$

and

$$F_{\mathrm{rel}}(R)(x,y) \begin{cases} x = y & \text{if } R \text{ is } = \\ x < y & \text{if } R \text{ is } < \end{cases}$$

  Note that such a definition is pretty cumbersome, especially considering the fact that we use standard $+$ instead of the symbol $+$, standard $=$ instead of the symbol $=$ etc. So in similar cases we write $\mathcal{M} = (\mathbb{R}; =,<;+,\cdot)$.

- Finally, we consider the formulas in the signature $\mathcal{S}$

$$\forall x\ \forall y\ x + y = y + x$$

and

$$\forall x\ \forall y\ \forall z\ (x < y \implies x + z < y + z).$$

  (Note that we write $a = b$ instead of $= (a, b)$ and $a + b$ instead of $a + b$, this is a common notation when the standard mathematical operations and relations are used in the signature.)

The first formula says that the operation $+$ is commutative, which is true, so the value of the formula with respect to the structure $\mathcal{M}$ should be true. (Note that we do not mention the values of the variables x and y since both of them are not free.) Indeed, consider $a, b \in \mathbb{R}$ note that the value of $x + y = y + x$ with $x = a$ and $y = b$ and with respect to the structure $\mathcal{M}$ is equal to $F_{\mathrm{rel}}(=)(F_{\mathrm{fun}}(+)(a,b), F_{\mathrm{fun}}(+)(b,a))$ which is the same as $a + b = b + a$; thus, the first formula is true.

The second formula says that the inequalities are additive, so it should be also true with respect to the structure $\mathcal{M}$.

Exercise 29.1. *Show that the second formula is true with respect to the structure $\mathcal{M}$.*

Exercise 29.2. *Let us consider a signature $(=;+,\cdot,0,1)$ and two models with this signature: $\mathfrak{R} = (\mathbb{R};=;+,\cdot,0,1)$, and $\mathfrak{Q} = (\mathbb{Q};=;+,\cdot,0,1)$. Find a predicate formula ϕ in this signature such that $\mathfrak{R} \models \phi$ but $\mathfrak{Q} \not\models \phi$.*

Natural Deduction for Predicate Logic: Introduction to Mathematical Logic #6

https://youtu.be/GVht3ES2qqo

29.2 *Natural Deduction*

By analogy with the tautology, in the predicate logic we wish to prove that a formula is true, whenever the structure and the values of the variables we choose. Such formulas are called *logically valid*.

In addition, we may define semantic implication for predicate formulas. We say that a set of predicate formulas Σ in a signature $\mathcal{S}$ semantically implies a formula ϕ ($\Sigma \models \phi$) in the signature iff any structure with the signature $\mathcal{S}$ modeling Σ models ϕ as well.

Natural deduction for the predicate formulas is defined in the same manner as the natural deduction for the propositional formulas but now the lines are predicate formulas and we can use four additional rules.

Universal quantifier. The first logically-valid formula we use as a rule is $A(x) \implies (\forall y\ A(y))$, this rule allows us to introduce a universal quantifier. In order to use the following rule, x should not be a free variable of an open hypothesis.

$$
\begin{array}{r|ll}
m & A(x) & \\
 & \forall y\ A(y) & \forall\mathrm{I},\ m
\end{array}
$$

The second logically-valid formula we use as a rule says that if a statement is true for all the values of a variable, then it is also true when you substitute some specific term instead of the variable, i.e. $(\forall x\ A(x)) \implies A(t))$, this rule allows us to eliminate an universal quantifier.

$$
\begin{array}{r|ll}
m & \forall x\ A(x) & \\
 & A(t) & \forall\mathrm{E},\ m
\end{array}
$$

Existential quantifier. The first formula for the exisential quantifier says that you can name any term in the formula by a variable and formula is still true for some value of the variable. The corresponding formula is $A(t) \implies (\exists x\ A(x))$.

$$
\begin{array}{r|ll}
m & A(t) & \\
 & \exists x\ A(x) & \exists\mathrm{I},\ m
\end{array}
$$

The last rule says that if $A(x)$ is true for some x and we know that $A(y)$ implies B, then we can derive B (note that this is true only when y is not used in B). Thus we can apply the following rule when y is not be a free variable neither of B nor of any open hypothesis.

m	$\exists x\ A(x)$	
i	$\quad \mid\ A(y)$ (hypothesis)	
j	$\quad \mid\ B$	
	B	$\exists$E, m, i–j

29.3 Examples of Derivations

First example $\forall x\ F(x) \lor \neg(\forall x\ F(x))$ is a special form of the law of excluded middle, which we proved in the previous chapter. However, in order to emphasize that the propositional logic can prove all the statements provable in the predicate case we present the proof of this statement as well.

1		
2	$\quad \mid\ \neg(\forall x\ F(x) \lor \neg(\forall x\ F(x)))$ (hypothesis)	
3	$\quad \mid \quad \mid\ \forall x\ F(x)$ (hypothesis)	
4	$\quad \mid \quad \mid\ \forall x\ F(x) \lor \neg(\forall x\ F(x))$	$\lor$I, 3
5	$\quad \mid \quad \mid\ \perp$	$\neg$E, 2, 4
6	$\quad \mid\ \neg(\forall x\ F(x))$	$\neg$I, 3–5
7	$\quad \mid\ \forall x\ F(x) \lor \neg(\forall x\ F(x))$	$\lor$I, 6
8	$\quad \mid\ \perp$	$\neg$E, 2, 8
9	$\forall x\ F(x) \lor \neg(\forall x\ F(x))$	IP, 2–8

Unfortunately, this example just shows that a statement provable in the propositional logic can be proven in the predicate logic. The next example is an example that cannot be expressed in the propositional logic, we prove that if we know that $\forall x \forall y\ R(x,y) \implies R(y,x)$, the we can derive $\forall x \forall y\ ((R(x,y) \implies R(y,x)) \land (R(y,x) \implies R(x,y)))$.

1	$\forall x \forall y\ R(x,y) \implies R(y,x)$	
2	$\forall y\ R(x',y) \implies R(y,x')$	$\forall$E, 1
3	$R(x',y') \implies R(y',x')$	$\forall$E, 2
4	$\forall y\ R(y',y) \implies R(y,y')$	$\forall$E, 1
5	$R(y',x') \implies R(x',y')$	$\forall$E, 4
6	$(R(x',y') \implies R(y',x')) \wedge R(y',x') \implies R(x',y')$	$\wedge$I, 3, 5
7	$\forall y\ (R(x',y) \implies R(y,x')) \wedge (R(y,x') \implies R(x',y))$	$\forall$I, 7
8	$\forall x \forall y\ (R(x,y) \implies R(y,x)) \wedge (R(y,x) \implies R(x,y))$	$\forall$I, 7

29.4 Soundness and Completeness

Like in the propositional case, the most important properties of the natural deduction are the following two theorems.

Theorem 29.1 (completeness of natural deductions, Gödel)**.** *Let ϕ be a predicate formula. If ϕ is logically valid, then there is a proof of ϕ. Moreover, if $\Sigma \models \phi$, for some finite set of predicate formulas Σ, then there is a derivation of ϕ from Σ.*

Theorem 29.2 (soundness of natural deductions)**.** *Let ϕ be a predicate formula. If there is a proof of ϕ, then ϕ is logically valid. Moreover, if there is a derivation of ϕ from Σ, for some finite set of predicate formulas Σ, then $\Sigma \models \phi$.*

End of The Chapter Exercises

29.3 Give a natural deduction derivation of $\forall x\ A(x) \implies \forall x\ B(x)$ from $\forall x\ (A(x) \implies B(x))$.

29.4 Give a natural deduction derivation of $\exists x\ (A(x) \vee B(x))$ from $\exists x\ A(x) \vee \exists x\ B(x)$.

Part VIII

Introduction to Graph Theory

30. The Definition of a Graph

In this chapter we start a very important topic in discrete mathematics, which became even more important with the rise of computers, we start the discussion of graph theory. Graphs are used in mathematics and computer science to describe networks, maps, and dependencies of objects.

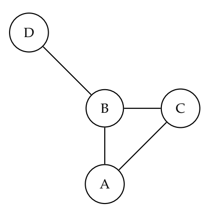
Definition 30.1. *A* graph *G is a pair (V, E) such that $E \subseteq V^2$ is a multiset.*

We say that G is unoriented *iff $(u, v) \in E$ iff $(v, u) \in E$ for any $u, v \in V$. Otherwise the graph is* oriented. *We say that a graph does not have* loops *iff $(u, u) \notin E$ for any $u \in V$. Finally, we say that the graph has* parallel edges *if E is not a set.*

A graph is *simple* iff it has no loops, it has not parallel edges, and it is unoriented.

From now on we will follow a standard convention and think about the set of edges of unoriented graphs as sets of *unordered* pairs.

It is very convenient to draw graphs using pictures like this.



In this picture, each circle corresponds to a vertice and each line corresponds to a an edge; i.e. this diagram describes the graph

$$(\underbrace{\{A, B, C, D\}}_{V}, \underbrace{\{(A, B), (A, C), (B, C), (B, D)\}}_{E}).$$

Note that we already use the convention that in an unoriented graph the pairs are unordered and we have not listed (B, A), (C, A) etc.

To talk about graphs we need to fix the vocubulary. An edge is said to *connect* its endpoints; two vertices that are connected by an edge are called *adjacent*; and a vertex that is an endpoint of a loop is said to be *adjacent to itself*. An edge is said to be *incident* on each of its endpoints, and two edges incident on the same end point are called *adjacent*. A vertex on which no edges are incident is called *isolated*.

One of the most important examples of graphs are complete graphs defined as follows.

Definition 30.2. *Let n be a natural number. A complete graph on n vertices, denoted K_n,*[1] *is a simple graph with in vertices and exactly one edge connecting each pair of distinct vertices.*

[1] Some sources claim that the letter K in this notation stands for the German word komplett, but the German name for a complete graph, vollständiger Graph, does not contain the letter K, and other sources state that the notation honors the contributions of Kazimierz Kuratowski to graph theory.

Exercise 30.1. *Show that for all natural numbers n, the number of edges of K_n is $\frac{n(n-1)}{2}$.*

30.1 Operations on Graphs

Quite often in order to prove a theorem we need to modify a graph. The most often operations are the following four. Let $G = (V, E)$ be a graph, $F \subseteq E$ be a set of edges, $U \subseteq V$ be a set of edges, $e \in E$ be an edge, and $v \in V$ be a vertex.

1. $G[U]$ denotes the graph $(U, \{e \in E \ : \ e \in U^2\})$, $G[U]$ is called the induced subgraph of G on the vertices U;
2. $G[F]$ denotes the graph (V, F), $G[F]$ is called the induced subgraph of G on the edges F;
3. $G - e$ denotes the graph $(V, E \setminus \{e\})$, i.e., the graph G without the edge e.
4. $G - v$ denotes the graph $(V \setminus \{v\}, E \cap (V \setminus \{v\})^2)$, i.e., the graph G without the vertex v.

Note that we used the word "subgraph", in fact we can define formally the meaning of this word.

Definition 30.3. *We say that a graph $H = (U, F)$ is a* subgraph *of $G = (V, E)$ iff $U \subseteq V$ and $F \subseteq E$.*

30.2 Degrees of Vertices

The degree of a vertex is the number of endsegments of edges that "stick out of" the vertex.

Definition 30.4. *Let $G = (V, E)$ be a graph, and v be a vertex. Then* $\deg_G(v) = |\{e \in E \ : \ e \text{is connected to } v\}|$.

Exercise 30.2. *Let $G = (V, E)$ be a graph and $v \in V$ be a vertex. What are the possible values of* $\deg_G(v)$*?*

Note that Lemma 20.1 shows that in any simple graph the number of vertices with an odd degree is even. The essence of the proof of this lemma is the following statement.

Theorem 30.1. *Let $G = (V, E)$ be a simple graph. Then* $\sum_{v \in V} \deg_G(v) = 2|E|$.

End of The Chapter Exercises

30.3 Either draw a graph with the specified properties or explain why no such graph exists:

1. simple graph with five vertices of degrees 1, 2, 3, 3, and 5;
2. simple graph with four vertices of degrees 1, 2, 3, and 3;
3. simple graph with four vertices of degrees 1, 1, 1, and 5;
4. simple graph with four vertices of degrees 1, 2, 3, and 4;
5. simple graph with four vertices of degrees 1, 2, 3, and 5.

30.4 In a group of 25 people, is it possible for each to shake hands with exactly 3 other people?

30.5 Suppose that G is a graph with v vertices and e edges and that the degree of each vertex is at least d_{min} and at most d_{max}. Show that

$$\frac{1}{2}vd_{\text{min}} \leq e \leq \frac{1}{2}vd_{\text{max}}.$$

31. Paths in Graphs

31.1 Connectivity

Imagine you are developing a game, where the map is generated automatically. In this gate there are several areas connected by portals. So you need to check that all the areas in your map are reachable from one another.

First we need to somehow understand what we mean by "reachable", we say that an area A is reachable from an area B if there is a path from A to B. To formalize this notion using graphs we need to introduce a graph corresponding to the map, consider a graph $G = (V, E)$ such that vertices of the graph are areas in your map and $(A, B) \in E$ iff the areas A and B are connected by a portal. So a path from A to B is a sequence of areas $A = C_1, \ldots, C_\ell = B$ such that C_i and C_{i+1} are connected by a portal (i.e. $(C_i, C_{i+1}) \in E$).

Definition 31.1. *Let $G = (V, E)$ be a graph. We say that a path from u to v is a sequence $w_1, \ldots, w_\ell \in V$[1] such that*

- $w_1 = u$, $w_\ell = v$, *and*
- $(w_i, w_{i+1}) \in E$ *for* $i \in [\ell - 1]$.

We say that $u, v \in V$ are connected iff there is a path from u to v. So the graph is connected iff any $u, v \in V$ are connected.

[1] Usually such an object is called a walk, and it is called a path if all the vertices $w_1, \ldots, w_\ell$ are different. However, for our applications it does not matter and we will use the word "path".

Exercise 31.1. *Let $G = ([2n], E)$ be a graph such that $(i, j) \in E$ if $|i - j| =$ 2. Is G connected?*

So, using this notation, we need to check whether the graph corresponding to the map is connected. There are numerous ways to do it, we consider a simple algorithm just to see how it works.

Theorem 31.1. *Algorithm 31.1 checks whether the graph $([n], E)$ is connected.*

Proof. First of all, note that the algorithm has a finite running time since size of S increases by 1 in the cycle starting on line 3. It is also easy to see that if a vertex $v \in Q$ at some point it is in S on line 9. In

Algorithm 31.1: An algorithm checking whether the graph on $[n]$ with the set of edges E is connected.

```
1:  function CONNECTED(n, E)
2:      S ← ∅
3:      Q ← {1}
4:      while Q ≠ ∅ do
5:          Choose an element v from Q
6:          Q ← S \ {v}
7:          S ← S ∪ {v}
8:          Q ← Q ∪ {u ∈ [n] : (v, u) ∈ E and u ∉ S}
9:      end while
10:     return S = [n]
11: end function
```

addition, if $v \in Q$ at some point, then $\{u \in [n] \ : \ (v,u) \in E\} \subseteq S$ on line 9.

Therefore if $u \notin S$ and $(v,u) \in E$, then $v \notin S$. Using this observation we may prove that if $G = ([n], E)$ is connected, then Algorithm 31.1 returns true. Indeed, assume the opposite. Consider $u \in [n] \setminus S$, and $N_i \subseteq [n]$ such that

$$N_0 = \{u\}, N_{i+1} = N_i \cup \{v \in [n] \ : \ w \in N_i, (v,w) \in E\}.$$

Note that by the previous observation if $v \in N_i$, then $v \notin S$. Since G is connected, there is a path $u = v_1, \ldots, v_k = 1$. Note that $u \in N_0$, $v_2 \in N_1, \ldots, v_k \in N_{k-1}$. Therefore $1 = v_k \notin S$ which is a contradiction.

To finish the proof we need to show that if $S = [n]$, then the graph is connected. To prove the statement we prove by induction that there is a path from 1 to any element of S and Q in every iteration of line 3. Indeed, initially S is empty and Q contains only 1. After an iteration of line 3 we choose an element v from Q and by the induction hypothesis there is a path from 1 to v. We add it to S and the statement about S holds, afterwards we add all the neighbours of v to Q. So the statement about Q is also true. $\square$

Not all the graphs are connected, but it is always possible to split the graph into connected parts, such parts are called connected components.

Definition 31.2. *Let $G = (V, E)$ be a graph. We say that $U \subseteq V$ is a connected component if for any $u \in U$ and $v \in V$, $v \in U$ iff there is a path from u to v in G.*

Theorem 31.2. *Let $G = (V, E)$ be a graph. If U_1 and U_2 are connected components of G, then they either equal to each other or disjoint. Moreover there are connected components $V_1, \ldots, V_k$ in G such that $V_1 \cup \cdots \cup V_k = V$ and $V_1, \ldots, V_k$ are disjoint.*

Exercise 31.2. *Let $G = ([2n], E)$ be a graph such that $(i, j) \in E$ if $|i - j| = 2$. Find all the connected components of G.*

Exercise 31.3. *Find a modification of Algorithm 31.1 that can find all the connected components of $([n], E)$.*

31.2 Eulerian Paths

Graph theory originated from a simple question asked by Leonard Euler: "Is it possible to walk through the town of Königsberg, starting and ending at the same place, so that we use each bridge exactly once?" (the map of Königsberg is depicted on Figure 31.1). It is pos-

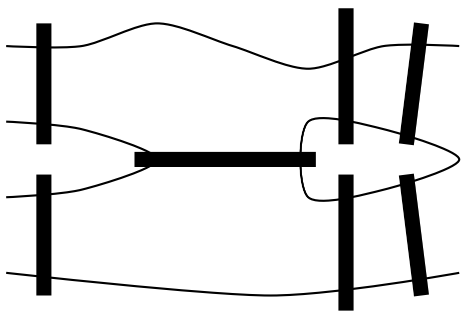


Figure 31.1: Königsberg's map

sible to see that the geometry of the islands is not important for this problem, the only important property is the number of bridges between islands.

In other words, all the necessary information can be described by the graph (the islands are vertices and the bridges are edges) depicted on Figure 31.2. Hence, to formalize the problem we need to give the

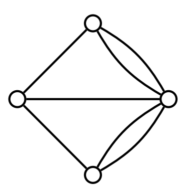


Figure 31.2: The graph of Königsberg's bridges

following definition.

Definition 31.3. *A path $v_1, \ldots, v_k$ in a graph $G = (V, E)$ is called Eulerian if for any edge $(u_1, u_2) \in E$ there is exactly one $i \in [k-1]$ such that $u_1 = v_i$ and $u_2 = v_{i+1}$.*

An Eulerian path is called an Eulerian cycle if $v_1 = v_k$.

Using this definition the question is whether there exists an Eulerian cycle in the graph of Königsberg's bridges.

Exercise 31.4. *Check whether the graph of Königsberg's bridges has an Eulerian cycle or not.*

The following theorem gives a simple criterion that allows us to solve the problem in the general case.

Theorem 31.3. *A connected graph G has an Eulerian cycle if and only if all vertices of G have even degree. (Note that the statement holds even if G has parallel edges).*

Proof. Assume that such a cycle exists If a vertex v appears k times in the cycle, then there are $2k$ edges involving v in the cycle (because, each time v is visited, there is an edge used to step on v and one to leave from v); since the cycle contains all the edges of the graph, v has degree $2k$. Therefore all vertices have even degree. This shows that if a connected graph contains an Eulerian cycle, then every vertex has even degree.

To prove this statement in the other direction, we will prove by induction a stronger statement, we will prove that if G is a graph in which every vertex has even degree, then every connected non-trivial connected component of G (a connected component is trivial if it contains only an isolated vertex of degree zero) has an Eulerian cycle. We will proceed by induction on the number of edges.

If there are zero edges, then every connected component has only one vertex and so it is nothing to prove. This is the base case of the induction.

If we have a graph $G = (V, E)$ with a non-empty set of edges and in which every vertex has even degree, then let $V_1, \dots, V_m$ be the non-trivial connected components of V. If $m \geq 2$, then every connected component has strictly less vertices than G, and so we can apply the inductive hypothesis and find Eulerian cycles in each of $V_1, \dots, V_m$.

It remains to consider the case in which the set V' of vertices of non-zero degree of G are all in the same connected component. Let $G' = G[V']$. Since every vertex of G' has degree at least, there must be a cycle in G'. Let C be a simple cycle (that is, a cycle with no vertices repeated) in G', and let $G'' - C$. Since we have removed two edges from every vertex, we have that G'' is still a graph in which every vertex has even degree. Since G'' has fewer edges than G' we can apply the induction hypothesis, and find an Eulerian cycle in each non-trivial connected component of G''. We can then patch together these Eulerian cycles with C as follows: we traverse C, starting from any vertex; the first time we reach one of the non-trivial connected components of G'', we stop traversing C, and we traverse the Eulerian cycle of the component, then continue on C, until we reach for the first time one of the non-trivial connected components of G'' that we haven't traversed yet, and so on. This describes a Eulerian path into all of G' □

Exercise 31.5. *Finish the proof of Theorem 31.3 by proving that if a graph G has only vertices of an odd degree, then there is a simple cycle in G.*

Corollary 31.1. *A graph G has an Eulerian path starting and ending in two different vertices if and only if in G there are exactly two vertices with odd degrees. (Note that the statement holds even if G has parallel edges).*

Proof. Let $G = (V, E)$ and u and v be the vertices with odd degrees. Let us consider the graph $G + (u, v) = (V, E \cup (u, v))$ (if there are edges between u and v we increase their number by one). Note that all the degrees in $G + (u, v)$ are even. Therefore by Theorem 31.3, there is an Eulerian cycle in $G + (u, v)$. Without loss of generality the cycle is in the form u, v, w_1, ..., w_k, v. Therefore, there is an Eulerian path v, w_1, ..., w_k, v in G. □

31.3 Hamiltonian Paths

Another example of a path that mathematicians are interested in is Hamiltonian path.

Definition 31.4. *Let G be a graph. We say that a path in G is Hamilton if it visits every vertex in G exactly once. We say that such a path is a Hamiltonian cycle if its starting and ending vertices are connected.*[2]

The greatest difference with Eulerian cycles is that it is not known whether there is a fast (polynomial-time) algorithm that allows to find the Hamiltonian cycles in a graph.[3]

It is easy to design an algorithm that checks whether a path exists in $O((n-1)!)$ by just brute forcing all the possible candidates for such a path. However, using the ideas of the inclusion-exclusion principle, we may design a much faster algorithm.

Theorem 31.4. *There is an algorithm with the running time $O(2^n n^3)$ such that it finds the number of Hamiltonian cycles in a graph $G = ([n], E)$.*

Before we prove the theorem, recall that if U and $A_1, \ldots, A_n \subseteq U$ are some finite sets, then

$$\left|\bigcap_{i=1}^{n} A_i\right| = \sum_{X \subseteq [n]} (-1)^{|X|} \left|\bigcap_{i \in X} \overline{A}_i\right|,$$

where $\overline{A}_i = U \setminus A_i$ and $\bigcap_{i \in \emptyset} \overline{A}_i = U$.

Proof of Theorem 31.4. As we mentioned before we the inclusion-exclusion principle to find the number of Hamilton cycles. Let U be the set of all the cycles of length n (length is the number of edges in the path) going

[2] Hamiltonian paths and cycles are named after William Rowan Hamilton who invented the icosian game, now also known as Hamilton's puzzle, which involves finding a Hamiltonian cycle in the edge graph of the dodecahedron.

[3] Proving or disproving that there is a polynomial-time algorithm allowing to check whether a graph G has a Hamiltonian path is one of the Millennium Problems. Clay Mathematics Institute offers a prize of \$1 million to a person who solves the problem.

via the vertex 1 and $A_v \subseteq U$ ($v \in [n]$) be the set of cycles of length n going via the vertices 1 and v.

It is clear that the answer is $|\bigcap_{i=1}^n A_i|$. Therefore it is enough to find all the cardinalities of $\left|\bigcap_{i\in X} \overline{A_i}\right|$. Note that $\bigcap_{i\in X} \overline{A_i}$ is equal to the set of all the cycles of length n going via the vertex 1 in $G - X$. We denote the cardinality of this set by C_X.

To find the value of C_X we use the following notation. Let E_X be the set of edges in $G - X$ and let $T_X(d, x)$ be the number of length d paths from 1 to $x \in [n] \setminus X$ in $G - X$. Clearly $T_X(0, x) = 1$ if $x = 1$ and $T_X(0, x) = 0$ otherwise. In addition, $T_X(d+1, x) = \sum_{y\,:\,(y,x)\in E_X} T_X(d, y)$. Therefore, we may compute $T_X(n, x)$ for all $x \in [n] \setminus X$ in n^3 steps. As a result, we may find the value of $\sum_{X\subseteq[n]}(-1)^{|X|}C_X$ in $2^n n^3$ steps. □

However, one may prove that if all the vertices in a graph have large degree, then the graph has a Hamiltonian cycle.

Theorem 31.5 (Dirac). *Let G be a graph on $n \geq 3$ vertices. If every vertex v in G has degree at least $n/2$, then there is a Hamiltonian cycle in G.*

Proof. For the sake of contradiction, let us assume that G has no Hamiltonian cycle but all the vertices have degree at least degree $n/2$, where n is the number of vertices in G.

Let us start adding edges to G as long as we are not creating a Hamiltonian cycle. When we stop we get a graph $H = (V, E)$ such that all the vertices of H have degree at least $n/2$, H does not have a Hamiltonian cycle, but adding any new edge would create a Hamiltonian cycle.

Consider any two vertices x and y that are not connected by an edge. We know that in the graph $H + (x, y)$ there is a Hamiltonian cycle $x = v_1, \ldots, v_n = y$. Note that $|\{v \in V : (x, v) \in E \text{ or } (y, v) \in E\}| \geq n$ since $\deg_H(x) \geq n/2$ and $\deg_H(y) \geq n/2$. Therefore by the pigeonhole principle, there is $2 \leq i \leq n-1$ such that $(x, v_i) \in E$ and $(v_{i-1}, y) \in E$. As a result, x, v_2, …, v_{i-1}, y, v_{n-1}, …, v_i is a Hamiltonian cycle in H. □

There are plenty of different applications of Hamiltonian paths. Here we describe the one that comes from bioinformatics.

Imagine that we want to read a DNA strand, i.e., determine the order in which nucleotides occur on a strand of DNA. One of the methods, called "Sequencing by Hybridization", is based on Hamiltonian paths.

The method works as follows.

- Attach all possible DNA probes of length k to a flat surface, each probe at a distinct and known location. This set of probes is called the DNA microarray.

- Apply a solution containing fluorescently labeled copies of a DNA fragment to the array.
- The DNA fragment hybridizes with those probes that are complementary to substrings of length k of the fragment.
- Using a spectroscopic detector, determine which probes hybridize to the DNA fragment to obtain the k-mer composition of the DNA fragment.
- Reconstruct the sequence of the DNA fragment from the k-mer composition.

In other words, we need to reconstruct a string s from all $n - k + 1$ substrings of length k; e.g, we need to reconstruct the string TATGGTGC from the strings ATG, GGT, GTG, TAT, TGC, TGG (in this example $k = 3$). (Note that different strings may have the same sets of substrings. Strings GTATCT and GTCTAT correspond to the strings AT, CT, GT, TA, TC when $k = 2$.)

By a given set $p_1, \ldots, p_\ell$ of strings (k-mers) of length k we construct the following graph. There are ℓ vertices corresponding to the strings $p_1, \ldots, p_\ell$; there is an edge between p_i and p_j whenever the same string of length $k - 1$ is a suffix of p_i and a prefix of p_j (for example, TG is a suffix of ATG and a prefix of TGG). It is easy to see that we can find a string corresponding to $p_1, \ldots, p_\ell$ if we have a Hamiltonian path in the graph.

End of The Chapter Exercises

31.6 Is it true that if a graph has a closed Eulerian walk, then it has an even number of edges?

31.7 *(recommended)* Let G be a graph such that there are only 2 vertices with odd degree. Prove that they belong to the same connected component.

31.8 Let $G = (V, E)$ be a connected graph and $c : V \to \{0, 1\}$ be a function.

1. Assume that $\sum_{v \in V} c(v)$ is odd. Show that for any $s : E \to \{0, 1\}$, there is a vertex $v \in V$ such that $\sum_{(u,v) \in E} s(u, v)$ and $c(v)$ have different remainders modulo 2.
2. Assume that $\sum_{v \in V} c(v)$ is even. Show that there is a function $s : E \to \{0, 1\}$ such that $\sum_{(u,v) \in E} s(u, v)$ is odd iff $c(v)$ is odd for all $v \in V$.

31.9 What is the maximal number of edges of a simple graph G on $[n]$ if it is not connected?

32. Trees

Let us consider the following problem. Given a network of several computers, in this network if a computer A receives some message from a computer B, it broadcasts it to all the connected computers except B. However, in such setting there is an issue known as broadcast radiation. Assume we have three computers A, B, and C such that they form a cycle. If A sends something to B and C both of them send received information to C and B, respectively; after that B and C send this information to A and A start sending this information again, which leads to an infinite cycle.[1]

Therefore to avoid such problem we need to disable some connecion so that the graph of this network does not have cycles. In this chapter we are going to study properties of the graphs without cycles.

[1] This problem is a simplified version of a problem that is solved by STP protocols in the modern networks.

Definition 32.1. *We say that a connected graph G is a* tree *iff G does not have cycles.*

32.1 Minimally Connected Graphs

First we may make the following observation.

Theorem 32.1. *Let $G = (V, E)$ be a connected graph. Then the following statements are equivalent.*

- *G is a tree.*
- *G is minimally connected, that is, $G - e$ is not connected for any $e \in E$.*

Proof. Assume that G is minimally connected but G has a cycle $v_1, \ldots,$ v_k. Consider $G' = G - (v_1, v_k)$, we claim that G' is still connected. Indeed, let x and y be some vertices of G'. Since G is connected, there is a path p from x to y. If p does not contain the edge (v_1, v_k), then x and y are connected in G'. If p contains (v_1, v_k), then we replace this edge by the path $v_1, \ldots, v_k$, so x and y are connected in G'. Therefore G is not a minimally connected graph, which is a contradiction.

Let us now assume that G is not minimally connected, we wish to prove that it implies that G is not a tree. Since G is not minimally connected, there is an edge $(x, y) \in E$ such that $G - e$ is connected.

Since $G - (x, y)$ is connected, there is a path $x = v_1, \ldots, v_k = y$ in $G - (x, y)$. Therefore $v_1, \ldots, v_k$ is a cycle in G, which is a contradiction. $\square$

Therefore in order to get a tree from a graph, we just need to delete edges in an arbitrary way until the moment when we cannot delete them anymore.

Corollary 32.1. *For any connected graph $G = (V, E)$, there is a tree $T = (V, E')$ such that T is a subgraph of G. Such a tree is called a spanning tree of G.*

Another question we may ask is how many edges we need to delete in this process. Apparenly, the answer is always $m - n + 1$, where m is the number of edges in the initial graph and n is the number of vertices.

Theorem 32.2. *Let G be a connected graph on n vertices. If G is a tree, then it has $n - 1$ edges. Moreover, if G has $n - 1$ edge, then it is a tree.*

Before we prove the theorem, let us prove the following lemma.

Lemma 32.1. *If a tree T has at least 2 vertices, then it has at least two vertices whose degree is 1.*

Proof. Let us choose a vertex v of T such that its degree is not 1 (if such a vertex does not exist, then we found at least 2 vertices whose degree is 1). Let us start walking from v to its neighbour, then to a new neighbor of this neighbor, and so on, never revisiting a vertex. As T has finite number of vertices, we will eventually have to stop at a vertex u. We claim that the only reason for us to stop at u could be that u is of degree 1. Indeed, the only possible other reason would be that u has neighbors other than the neighbor u' we reached u from, but they have all been visited already. However, that would mean that there are at least two paths from v to u, and that cannot happen in a tree. So u is of degree 1. To get another vertex of degree 1, remember that v is of degree more than 1. So take another neighbor of v, and repeat this argument. This will result in another vertex w of degree 1, and $u \neq w$ as that would again yield two paths from v to u. $\square$

The vertices of a tree that have degree 1 are called *leaves*.

Proof of Theorem 32.2. We prove the statement using induction by n. If $n = 1$, the statement is clearly true. Assume that the statement is true for trees on n vertices. Consider a tree T on $n + 1$ vertices. Consider a leaf ℓ of T. Note that $T - \ell$ is a tree as well, therefore by the induction hypothesis, it has $n - 2$ edges. Hence, T has $n - 1$ edges.

Let us now prove that if a graph G has $n - 1$ edges and is connected, then G is a tree. Assume that it is not a tree, we start deleting edges as long as the graph is connected, we call the resulting graph T. Note that T is minimally connected, so T is a tree. Note that T has n vertices. Therefore, it has $n-1$ eges, which implies that we removed 0 edges and $T = G$. As a result, G is a tree. □

Exercise 32.1. *A graph such that every connected component of this graph is a tree is called a* forest*. Show that a forest with k connected components has $n-k$ edges.*

32.2 Minimum-weight Spanning Trees

In the initial example about the network, we missed an important detail: not all the connections are equally fast. Let us label each connection (edge in our graph) with the weight (the number that represents how slow is this connection). So now we need to choose a spanning tree of the graph of the network so that it has the minimal possible sum of weights.

Definition 32.2. *Let $G = (V, E)$ be a connected graph, and $w : E \to \mathbb{R}$ be weights of edges. Then we say that a spanning tree $T = (V, E')$ of G is a* minimum-weight spanning tree *of G if $\sum_{e \in E'} w(e) \leq \sum_{e \in E''} w(e)$ for any spanning tree $T' = (V, E'')$ of G.*

The number $\sum_{e \in E'} w(e)$ is called the weight *of T.*

It is obvious that such a tree exists. The question is "how to find efficiently the minimum-weight spanning tree".

Exercise 32.2. *Let $G = (V, E)$ be some graph and $w : E \to \mathbb{R}$ be a weight function such that $w(e) = 1$. How to to find efficiently the minimum-weight spanning tree of G?*

Surprisingly, one may find such a minimum-weight spanning tree using a simple greedy algorithm (Algorithm 32.1).

Theorem 32.3. *If the graph $([n], E)$ is connected, then Algorithm 32.1 returns a minimum-weight spanning tree of the graph $([n], E)$.*

To prove this statement we need a technical lemma.

Lemma 32.2. *Let F_1 and F_2 be forests on the same vertex set V. If F_1 has less edges than F_2, then F_2 has an edge e not in F_1 so that the graph $F_1 + e$ is still a forest.*

Proof. Let E_i be the set of edges of F_i. Assume that there such edge does not exist; i.e., $F_1 + e$ has a cycle for any edge $e \in E_2 \setminus E_1$.

Therefore any edge of F_2 is between two vertices in the same component of F_1. Hence, F_2 has at least as many connected components as F_1. Indeed, consider two connected components U_1 and U_2 of F_1 we

Algorithm 32.1: Kruskal's algorithm, the algorithm that returns a minimum-weight spanning tree of the graph on $[n]$ with the set of edges E.

1: **function** MINIMUMSPANNINGTREE(n, E, w)
2: Let $e_1, \dots, e_m$ be the edges from E sorted in the ascending order with respect to w.
3: $i \leftarrow 1$
4: Set T to be an empty graph on $[n]$.
5: **while** $i \leq n$ **do**
6: **if** $T + e_i$ does not have cycles **then**
7: $T \leftarrow T + e_i$
8: **end if**
9: Increase i by 1.
10: **end while**
11: **return** T
12: **end function**

claim that they any $x \in U_1$ and $y \in U_2$ are not connected in F_2 since there are no edges going outside of U_1 and U_2 in F_2.

However, F_i has $n - |E_i|$ connected components, which contradicts to the fact that $|E_1| < |E_2|$. □

Proof of Theorem 32.3. Let $T_1, \dots, T_m$ be the states of T after iterations of line 4 of Algorithm 32.1. Note that T_i does not have cycles for $i \in [m]$. Therefore T_i is a forest for $i \in [m]$.

First we need to prove that Algorithm 32.1 returns a spanning tree; i.e. that T_m is connected. Assume the opposite. Consider two vertices $x, y \in [n]$ such that they are not connected in T. Since $G = ([n], E)$ is connected there is a path $x = v_1, \dots, v_k = y$ in G. Consider the minimal $i \in [k-1]$ such that (v_i, v_{i+1}) is not an edge of T_m. Let $e_j = (v_i, v_{i+1})$. It is easy to see that $T_m + (v_i, v_{i+1})$ does not have cycles so $T_{j-1} + e_j$ does not have cycles as well and $T_j = T_{j-1} + e_j$ which implies that T_m has the edge (v_i, v_{i+1}) which is a contradiction.

Before we start the second part of the proof note that if $w(e) < w(e_i)$, then $T_{i-1} + e_i$ has a cycle.

Now we need to prove that T_m is a minimum-weight spanning tree. Assume that there is a spanning tree H such that the weight of H is less than the weight of T_m. Consider the edges $t_1, \dots, t_{n-1}$ of T_m and the edges $h_1, \dots, h_{n-1}$ of H such that $w(t_1) \leq w(t_2) \leq \cdots \leq w(t_{n-1})$ and $w(h_1) \leq w(h_2) \leq \cdots \leq w(h_{n-1})$. Let us consider the first step when H is better than T_m; i.e., the minimal i so that $\sum_{j=1}^{i} w(h_j) < \sum_{j=1}^{i} w(t_j)$ (obviously $i > 1$).

It is easy to see that $h_i < t_i$. Let $e_j = t_i$ and $H_i = H[h_1, \dots, h_i]$. Since H_i has more edges than T_j there is an edge $h_{i'}$ ($i' < i$) such that $T_{j-1} + h_{i'}$ does not have cycles. Wich is a contradiction since $h_{i'} < h_i < t_i$. □

End of The Chapter Exercises

32.3 *(recommended)* Let G be a graph with k connected components and $n - k$ edges. Show that G is a forest.

32.4 Prove that if G is a simple graph on $[n]$, then at least one of G and its complement is connected. Show an example when they are both connected. The complement $\overline{G}$ of G has the same vertex set as G and (x, y) is an edge in $\overline{G}$ if and only if it is not an edge in G.

32.5 Let H be a simple graph on n vertices that has m edges. Prove that H contains at least $m - n - 1$ cycles.

Part IX

Introduction to Computability Theory

33. Decidable Sets

This part is mainly based on the amazing book "Computable Functions" by Shen and Vereshchagin.

In this part we study what computers can compute and what they cannot compute. Usually study of this subject starts from a formal definition of an algorithm. However, we believe that this is unnecessary nowadays because of rise of computers. One may think about algorithms as programs on some programming language such as C/C++, Java, Python etc.

An algorithms are taking several natural numbers $x_1, \ldots, x_n$ as an input and either print another number y as an output or never terminates. In the first case we say $\mathcal{A}(x_1, \ldots, x_n) = y$ and in the second case we say $\mathcal{A}(x_1, \ldots, x_n)$ never terminates.

33.1 Computable Functions

The first and the most basic definition in the computability theory is the definition of a computable function.

Definition 33.1. *Let $S \subseteq \mathbb{N}^\ell$ and $f : S \to \mathbb{N}$. We say that f is* computable *if there is an algorithm $\mathcal{A}$ such that*

1. *$\mathcal{A}(x) = f(x)$ for any $x \in S$ and*
2. *$\mathcal{A}(x)$ never terminates for any $x \notin S$.*

We say that $\mathcal{A}$ computes *f.*

It is important to note that nonetheless that we say that the algorithms are said to take and print natural numbers, we could allow algorithm to work with strings of bits (elements of the set $\{0,1\}^* = \bigcup_{n \in \mathbb{N}_0} \{0,1\}^n$). Moreover, these two definitions are equivalent since there is a one-to-one correspondence between natural numbers and strings: $x \mapsto 2^n + \sum_{i=1}^{\ell} 2^{i-1} x[i]$, where $x[i]$ denotes the ith symbol of the string i and n is the length of x. It is also clear that using binary strings we may encode all sorts of objects such as pairs of integers, integers, rational numbers etc. (However, in order to encode real number we need more complicated definitions and we are not going to discuss them in here.)

Exercise 33.1. *Let $f : \{0,1\}^* \to \{0,1\}^*$ be the function such that $f(x)$ is reversed x. Show that f is computable.*

One may show that composition of computable functions is computable. Moreover, one may prove the following a bit stronger statement.

Theorem 33.1. *Let $S \subseteq \mathbb{N}$, and let $f : \mathbb{N} \to \mathbb{N}$, and $g : S \to \mathbb{N}$ be computable functions. Then $f \circ g$ is also computable.*

Proof. Since f and g are computable, there are algorightms $\mathcal{F}$ and $\mathcal{G}$ computing f and g respectively.

Let us consider the following algorithm. It is clear that if $x \notin S$,

Algorithm 33.1: The algorithm computing the composition of the functions computed by $\mathcal{F}$ and $\mathcal{G}$

```
function A(x)
    y ← G(x)
    return F(y)
end function
```

then $\mathcal{A}(x)$ never terminates.

However, if $x \in S$, then $y = g(x)$ and therefore the algorithm prints $\mathcal{F}(g(x)) = f(g(x)) = (f \circ g)(x)$. □

33.2 Decidable Sets

Another important notion is the notion of a decidable set.

Definition 33.2. *We say that a set $S \subseteq \mathbb{N}$ is* decidable *iff there is an algorithm $\mathcal{A}$ such that $\mathcal{A}(x) = 1$ if $x \in S$ and $\mathcal{A}(x) = 0$ if $x \notin S$.*

It is easy to note that a set $S \subseteq \mathbb{N}$ is decidable iff the characteristic function χ_S of S is computable. The function χ_S is defined as follows:

$$\chi_S(x) = \begin{cases} 1 & \text{if } x \in S, \\ 0 & \text{otherwise.} \end{cases}$$

Similarly we may define decidable sets of strings, pairs of integers etc.

We illustrate this concept by proving that $U_e = \{q \in \mathbb{Q} \;:\; q > e\}$. It is known that

1. $(1 + \frac{1}{n})^n < e$ and $\lim\limits_{n\to\infty} (1 + \frac{1}{n})^n = e$ and
2. $(1 + \frac{1}{n})^{n+1} > e$ and $\lim\limits_{n\to\infty} (1 + \frac{1}{n})^{n+1} = e$.

Hence, in order to check whether $q \in U_e$ or not, it is enough to either find n such that $(1 + \frac{1}{n})^{n+1} < q$ or $(1 + \frac{1}{n})^n > q$. In order to do this we can check all positive integers one after another and at some point one of the inequalities became true.

Exercise 33.2. *Let $k \in \mathbb{N}$. Show that $[k]$ is decidable.*

However, sometimes it is possible to show that some set is decidable without presenting the algorithm explicitly. For example, consider the $S \subseteq \mathbb{N}$ such that $n \in S$ iff base 10 expansion of π has n consecutive 9s. It is possible to show that S is decidable. Indeed, it is easy to see that either $S = \mathbb{N}$ or $S = [k]$ for some $k \in \mathbb{N}$, however, in both cases the set is decidable.

It is easy to show that decidable sets have several good properties.

Theorem 33.2. *Let $S_1, S_2 \subseteq \mathbb{N}^\ell$ be decidable sets. Then $S_1 \cup S_2$, $S_1 \cap S_2$, and $S_1 \setminus S_2$ are all decidable.*

To prove the theorem, we generalize Theorem 33.1.

Theorem 33.3. *Let $S \subseteq \mathbb{N}^\ell$ and let $f_1, f_2 : S \to \mathbb{N}$ and $g : \mathbb{N}^2 \to \mathbb{N}$ be computable functions. Then the function $h : S \to \mathbb{N}$ such that $h(x) = g(f_1(x), f_2(x))$ is computable.*

Proof of Theorem 33.2. Let us prove that $S_1 \cup S_2$ is decidable. Since S_1 and S_2 are decidable, there are algorithms $\mathcal{A}_1$ and $\mathcal{A}_2$ computing characteristic functions χ_{S_1} and χ_{S_2} for the sets S_1 and S_2, respectively. Consider the function $g : \mathbb{N}^2 \to \mathbb{N}$ such that

$$g(x, y) = \begin{cases} 0 & \text{if } x = 1 \text{ or } y = 1 \\ 1 & \text{otherwise} \end{cases}.$$

It is clear that $h : \mathbb{N} \to \mathbb{N}$ such that $h(x) = g(\chi_{S_1}(x), \chi_{S_2}(x))$ is a characteristic function for $S_1 \cup S_2$. Hence, $S_1 \cup S_2$ is decidable by Theorem 33.3.

The proof of decidability of $S_1 \cap S_2$, and $S_1 \setminus S_2$ is essentially the same. ☐

33.3 Enumeratable Sets

The algorithm constructed in the proof of decidability of U_e consisted of two important parts: first one allowed to show that q is defenetely not in the set U_e and the second part allowed to show that q is defenetely in the set.

This observation leads to the following definition.

Definition 33.3. *We say that a set $S \subseteq \mathbb{N}$ is* (computably) enumerable, *we also say that it is* recursively enumerable *and* semidecidable, *iff there is an algorithm $\mathcal{A}$ such that*

1. *$\mathcal{A}(x) = 1$ for any $x \in S$ and*

2. *either $\mathcal{A}(x) = 0$ or $\mathcal{A}$ never terminates for any $x \notin S$.*

We say that S is semidecided *by $\mathcal{A}$.*

We can easily show that both sets $L_e = \{q \in \mathbb{Q} \,:\, q < e\}$ and $U_e = \{q \in \mathbb{Q} \,:\, q > e\}$ are enumerable. Indeed, we can try all possible $n \in n$ until $q < (1 + \frac{1}{n})^n$ if we find such n, we know that $q \in L_e$. Similarly, we can try all possible n until $q > (1 + \frac{1}{n})^{n+1}$ and if we find such n, then $q \in U_e$. The following allows us to show that the fact that L_e is decidable is not a coincidence.

Theorem 33.4 (Post's Theorem)**.** *Let $S \subseteq \mathbb{N}$. If S is decidable, then S is enumerable. Moreover, if S and $\mathbb{N} \setminus S$ are enumerable, then S is decidable.*

Proof. The first part is obvious. Let us prove the "moreover" part. Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be the algorithms deciding S and $\mathbb{N} \setminus S$ respectively. Then the algorithm $\mathcal{A}$ deciding S is the following: on onput x it runs $\mathcal{A}_1(x)$ and $\mathcal{A}_2(x)$ in parallel and if the first one prints 1, $\mathcal{A}$ prints 1 as well; however, if the second prints 1, $\mathcal{A}$ prints 0.

We need to prove that the algorithm works correctly.

- If $x \in S$, then $\mathcal{A}_1(x) = 1$ and $\mathcal{A}_2(x)$ never terminates. So $\mathcal{A}(x)$ prints 1.
- If $x \notin S$, then $\mathcal{A}_1(x)$ never terminates and $\mathcal{A}_2(x) = 1$. So $\mathcal{A}(x)$ prints 0.

$\square$

The given definition of enumerable set does not explain the name. However, there is an alternative definition that explains it.

Theorem 33.5. *Let $S \subseteq \mathbb{N}$. The set S is decidable iff there is an algorithm $\mathcal{A}$ such that*

1. *$\mathcal{A}(n)$ terminates for any $n \in \mathbb{N}$ and*
2. *$\{\mathcal{A}(n) \,:\, n \in \mathbb{N}\} = S$.*

We say that this $\mathcal{A}$ is enumerating S.

Proof. To prove this theorem, we generalize the idea of **??**. Assume that S is infinite. Let $\mathcal{A}'$ be the algorithm semideciding S and let $\mathcal{A}$ be Algorithm 33.2. It is clear that $\mathcal{A}$ satisfies the constraints of the theorem.

Let us prove the statement in the opposite direction. Let us assume that there is an algorithm $\mathcal{A}'$ enumerating S. Let $\mathcal{A}$ be Algorithm 33.3. We need to prove that $\mathcal{A}$ semidecide the set S.

1. Let us consider some $x \notin S$. In this case, $\mathcal{A}'(n) \neq x$ for any $n \in \mathbb{N}$. Hence, $\mathcal{A}$ never terminates.

Algorithm 33.2: The algorithm enumerating the set that is semidecided by $\mathcal{A}'$.

```
function A(n)
    i ← 1
    Let V be a map from integers to {0,1}
    while V has less than n keys with the value 1 do
        run in parallel
            Let y = A′(i)
            Put (i, y) into V
        end in parallel
        i ← (i + 1)
    end while
    return the ith key in V with the value 1
end function
```

Algorithm 33.3: The algorithm semideciding the set that is enumerated by $\mathcal{A}'$.

```
1: function A(x)
2:     n ← 1
3:     while A′(n) ≠ x do
4:         n ← n + 1
5:     end while
6:     return 1
7: end function
```

2. Let $x \in S$. Then there is $n \in \mathbb{N}$ such that $\mathcal{A}(n) \neq x$. Therefore, the number of iterations of Line 3 in Algorithm 33.3 is finite and the algorithm returns 1.

$\square$

One may also establish a connection between computable functions and enumerable sets.

Theorem 33.6. *Let $S \subseteq \mathbb{N}$.*

1. *The set S is enumerable iff there is a computable function $f : S \to \mathbb{N}$.*

2. *The set S is enumerable iff there is a computable function $f : \mathbb{N} \to \mathbb{N}$ so that $\operatorname{Im} f = S$.*

Proof. 1. To prove this part of the statement from right to left it is enough to note that the function $f : S \to \mathbb{N}$ such that $f(x) = 1$ is computable iff S is enumerable. Indeed, if $\mathcal{A}$ enumerates S, then it computes f; if $\mathcal{A}$ computes f it enumerates S.

To prove it from left to right, we may notice that $g : \mathbb{N} \to \mathbb{N}$ such that $g(x) = 1$ is computable. Therefore $(g \circ f) : S \to \mathbb{N}$ is computable. This implies that S is enumerable since $(g \circ f)(x) = 1$ for all $x \in S$.

2. This part directly follows from Theorem 33.5.

□

In the rest of this part we refer to functions $f : S \to B$, where $S \subseteq A$ as partial functions from $\mathbb{N}$ to $\mathbb{N}$. We say that S is preimage of f. Moreover, when we say that a function from $\mathbb{N}$ to $\mathbb{N}$ is computable we mean that it is a partial computable function. Using this notation Theorem 33.6 can be rephrased as follows.

Corollary 33.1. *1. The set S is enumerable iff there is a computable function $f : \mathbb{N} \to \mathbb{N}$ such that the preimage of f is equal to S.*

2. The set S is enumerable iff there is a computable function $f : \mathbb{N} \to \mathbb{N}$ such that the image of f is equal to S.

Since the notation between functions and partial functions is that similar, in the rest of this part we say that a partial function from A to B is total iff the preimage of the function is equal to A.

End of The Chapter Exercises

33.3 *(recommended)* Let $S \subseteq \mathbb{N}$ be a nonempty set. Show that S is decidable iff there is a function $f : \mathbb{N} \to \mathbb{N}$ such that f is computable, f is nondecreasing, and $\operatorname{Im} f = S$.

33.4 Let $A, B \subseteq \mathbb{N}$ be enumerable sets. Show that $A \times B$ is enumerable.

33.5 Let $F \subseteq \mathbb{N}^2$ be enumerable. Prove that exists a set $S \subseteq \mathbb{N}$ and a computable function $f : S \to \mathbb{N}$ such that $S = \{x \in \mathbb{N} \ : \ (x, y) \in F\}$ and $(x, f(x)) \in F$ for any $x \in S$.

33.6 Let $S = \{n \in \mathbb{N} \ : \ x^n + y^n = z^n \text{ has an integer solution}\}$. Show that S is decidable. (You should not use Fermat's Last Theorem.)

34. Universal Functions

It is known that we may write a program that gets another program as an argument and run it. (Such programs are known as interpreters.) To use this observation we give the following definition and theorem.

Definition 34.1. *We say that a function U is a* universal function *(for the set of univariate computable functions) iff for each $n \in \mathbb{N}$,*

$$U_n : x \mapsto U(n, x)$$

(we say that U_n is a section of U) is computable and any univariate computable function is among U_n's.

Theorem 34.1. *There is a computable universal function U.*

Exercise 34.1. *Assume that every section of a function U is computable. Is it necessarily true that U is computable?*

Similarly to the notion of universal function we may define the notion of universal sets.

Definition 34.2. *Let $F \subseteq 2^{\mathbb{N}}$. We say that $W \subseteq \mathbb{N}^2$ is universal for F if $W_n = \{x \in \mathbb{N} : (n, x) \in F\}$ is an element of F for all $n \in \mathbb{N}$ and any set $S \in F$ is among W_n's.*

Theorem 34.2. *There is an enumerable set W such that it is universal for the set of all enumerable subsets of $\mathbb{N}$.*

34.1 Enumerable but Not Decidable Set

Theorem 34.3. *There is a set $S \subseteq \mathbb{N}$ such that S is enumerable but it is not decidable.*

Proof. Let U be be a universal computable function, it exists by Theorem 34.1. To prove the statement we are going to use the diagonalization method. Let us consider $S = \{n \in \mathbb{N} : U(n, n) = 0\}$.

It is easy to see that S is enumerable. Assume for the sake of contradiction that S is decidable. Let $\mathcal{A}$ be the algorithm deciding S. There is $n \in \mathbb{N}$ such that $\mathcal{A}$ computes U_n since U is universal. Let us now consider two following cases.

1. Assume that $n \in S$. In this case $\mathcal{A}(n) = 1$ since $\mathcal{A}$ decides S. However, $U_n(n) \neq 1$ by the definition of S. These two equalities together leads us to a contradiction since $\mathcal{A}$ computes U_n.
2. Assume that $n \notin S$. In this case $\mathcal{A}(n) = 0$ since $\mathcal{A}$ decides S. However, $U_n(n) = 1$ by the definition of S. These two equalities together leads us to a contradiction since $\mathcal{A}$ computes U_n.

□

Using the proof of this result we can prove the following surprising observation.

Theorem 34.4. *Let U be a universal function. Let* $\mathrm{HALT} : \mathbb{N}^2 \to \{0, 1\}$ *be the function such that* $\mathrm{HALT}(n, x) = 1$ *iff $U_n(x)$ is defined. Then* HALT *is not computable.*

Informally, this theorem says that it is impossible to check whether a given algorithm terminates or not on some input.

34.2 Diagonalization Method

This section will give several other applications of the diagonalization method.

In the previous section we constructed a computable universal function for the set of computable functions of one variable. Now we can prove that it is impossible for total functions.

Theorem 34.5. *There is no computable universal total function for the set of computable total functions of one variable.*

Proof. Assume that such a function U exists. Let us consider the total computable function $d : \mathbb{N} \to \mathbb{N}$ such that $d(n) = U(n, n) + 1$. Since U is a computable universal total function for the set of computable total functions of one variable, there is $m \in \mathbb{N}$ such that $d(n) = U(m, n)$ for any $n \in \mathbb{N}$. Note that this implies that $U(m, m) = d(m) = U(m, m) + 1$, which is a contradiction. □

Note that the crucial point in this argument is that $U(m, m)$ is different from $U(m, m) + 1$; however, if the functions are not total it is possible that $U(m, m)$ is simply not defined. However, a part of the argument can be used, nonetheless.

Theorem 34.6. *There is a computable function $f : \mathbb{N} \to \mathbb{N}$ such that no computable function $g : \mathbb{N} \to \mathbb{N}$ can differ from f everywhere; i.e., for any computable function $g : \mathbb{N} \to \mathbb{N}$ there is $n \in \mathbb{N}$ such that $f(n) = g(n)$. (The last equality says that the values are either equal or both values are not defined for n.)*

Proof. Let U be a universal function for computable functions, and let $d : \mathbb{N} \to \mathbb{N}$ be a partial function such that $d(n) = U(n, n)$. It is clear that d satisfies the statement of the theorem. Indeed, any computable function f is equal to U_n for some n. Hence, $d(n) = U(n, n) = f(n)$. $\square$

Theorem 34.7. *There is a computable function that does not admit a total computable extension.*

Proof. Let d be the function from the previous theorem. Let us consider the partial function $e : \mathbb{N} \to \mathbb{N}$ such that $e(n) = d(n) + 1$. We wish to prove that e does not have a total extension. Let us assume the opposite, let e' be a computable total extension of e. Then e' differs from d everywhere, therefore e' is not computable. $\square$

Note that the last theorem gives another proof of Theorem 34.3. Indeed, let f be the function without total computable extension. Let $S = \operatorname{Im} f$. By Theorem 33.6, S is enumerable. Assume for the sake of contradiction, that S is decidable. Then the total function $g : \mathbb{N} \to \mathbb{N}$ such that

$$g(x) = \begin{cases} f(x) & \text{if } x \in S \\ 0 & \text{if } x \notin S \end{cases}$$

is computable. Moreover, g is an extension of f which leads to a contradiction.

End of The Chapter Exercises

34.2 Let $U \subseteq \mathbb{N}^2$ be any enumerable set of pairs of natural numbers that is universal for the set of all enumerable sets of natural numbers. Prove that its "diagonal section" $K = \{x : (x, x) \in U\}$ is an enumerable undecidable set.

34.3 Let $S \subseteq \mathbb{N}$ be decidable and let

$$D = \{p \in \mathbb{N} : p \text{ is prime and } p \text{ divides some } n \in S\}.$$

Is the set D always decidable?

34.4 Show that there exist countably many disjoint enumerable sets such that any two of them are inseparable (cannot be separated by a decidable set).

35. Gödel Universal Functions

You can read more about such translators on Wikipedia.

`https://en.wikipedia.org/wiki/Source-to-source_compiler`

It is known that there are algorithms that given a program in one programming language can produce a program in another programming language.

However, in this book we are talking about universal functions instead of programming languages. Hence, we may be interested to study the following problem. Let U and V be (computable) universal functions for the set of all univariate computable functions. Given $n \in \mathbb{N}$ find $m \in M$ such that U_m and V_m are equal. Unfortunately, not for every pair of universal sets such m can be found efficiently (see Theorem 35.6). However, there is a special class of universal functions that allow to find such m's efficiently for any computable V.

Definition 35.1. *Let $U : \mathbb{N}^2 \to \mathbb{N}$ be a computable universal function for the class of univariate computable functions. We say that U is* Gödel univeral function *if for any computable function $V : \mathbb{N}^2 \to \mathbb{N}$, there is a computable function $s : \mathbb{N} \to \mathbb{N}$ such that*

$$V(n, x) = U(s(n), x)$$

for all $n, x \in \mathbb{N}$.

Theorem 35.1. *There is a Gödel univeral function.*

Proof. We start the proof of the theorem from proving that there is a computable function $T : \mathbb{N}^3 \to \mathbb{N}$ that is universal for the set of bivariate computable functions. Let us fix a computable bijection $\langle \cdot, \cdot \rangle : \mathbb{N}^2 \to \mathbb{N}$. Let R be a universal function for the set of univariate computable functions, and let $T(n, u, v) = R(n, \langle u, v \rangle)$. it is easy to see that T is indeed a universal function for the set of bivariate computable functions.

Let $U : \mathbb{N}^2 \to \mathbb{N}$ be the function such that $U(\langle n, u \rangle, v) = T(n, u, v)$. We need to show that U is Gödel univeral function. Let us consider some computable function $V : \mathbb{N}^2 \to \mathbb{N}$. There is $n \in \mathbb{N}$ such that $V(u, v) = T(n, u, v)$ for all $u, v \in \mathbb{N}$ since T is universal. Therefore $U(\langle n, u \rangle, v) = V(u, v)$ for all $u, v \in \mathbb{N}$. As a result, we can define $s(u)$ to be equal to $\langle n, u \rangle$. $\square$

Exercise 35.1. *Show that there is a computable bijection $\langle \cdot, \cdot \rangle : \mathbb{N}^2 \to \mathbb{N}$.*

Gödel universal funcitons allow us to efficiently operate with numbers of computable functions. For example, Theorem 33.1 proved that composition of two computable functions is computbale. Moreover, it is easy to see that given the programs computing funcitons f and g we can automatically obtain the function $g \circ f$.

However, we would like to avoid specifics of programin languages in our study of computability theory. Our tool to do so is the notion of a universal function so we need to prove that there is an algorithm that given numbers of any two computable functions a computes a number of their composition.

Theorem 35.2. *Let U be a Gödel universal function for the set of univariate computable functions. Then there is a total computble function $c : \mathbb{N}^2 \to \mathbb{N}$ such that $U(c(p,q),x) = U(p,U(q,x))$ for any $p,q,x \in \mathbb{N}$.*

Proof. Let us consider a computable function $V : \mathbb{N}^2 \to \mathbb{N}$ such that $V(\langle p,q\rangle,x) = U(p,U(q,x))$.There is a total computable function $s : \mathbb{N} \to \mathbb{N}$ such that $U(s(\langle p,q\rangle),x) = V(\langle p,q\rangle,x)$ since U is a Gödel universal function. Hence, if we define $c(p,q)$ to be equal to $s(\langle p,q\rangle)$, we get that $U(c(p,q),x) = U(p,U(q,x))$. ☐

Using the notion of Gödel universal function we can also prove that constructing the shortest program solving a given problem is not feasible. In other words let us consider the problem of producing the shortest algorithm $\mathcal{A}$ by a given $\mathcal{B}$ such that $\mathcal{A}(x) = \mathcal{B}(x)$ for any $x \in \mathbb{N}$. Apparently there is no algorithm that can find such $\mathcal{A}$.

To prove this we need to formalize what we mean by the shortest algorithm and how we encode $\mathcal{A}$.

Theorem 35.3. *Let U be a Gödel universal function, and let $\mathrm{Opt} : \mathbb{N} \to \mathbb{N}$ be the function such that $U_{\mathrm{Opt}(n)}$ is the same as U_n and U_m and U_n are different for any $m < \mathrm{Opt}(n)$. Then O is not computable.*

To prove this statement we need the following auxilary result.

Theorem 35.4. *Let U be a Gödel universal function. Then the set $S = \{n \in \mathbb{N} : U(n,x)$ is not defined for all $x \in \mathbb{N}\}$ is not decidable.*

Proof. Let K be a enumerable but undecidable set. Consider the partial function $V : \mathbb{N}^2 \to \mathbb{N}$ such that

$$V(n,x) = \begin{cases} 0 & \text{if } n \in K \\ \text{undefined} & \text{otherwise} \end{cases}.$$

Note that $V(n,x)$ terminates for some $x \in \mathbb{N}$ iff $n \in K$. Since U is Gödel universal function there is a computable function s such that $V(n,x) = U(s(n),x)$. Hence, $s(n) \in S$ iff $n \in K$. As a result, S is undecidable. ☐

Proof of Theorem 35.3. Let S be the set from the previous theorem. Assume, for the sake of contradiction, that Opt is computable. Let n_0 be the smallest natural number n such that $U(n, x)$ is not defined for all $x \in \mathbb{N}$. It is clear that $n \in S$ iff $\mathrm{Opt}(n) = n_0$. Therefore S is decidable, which is a contradiction. $\square$

Theorems 34.4, 35.3 and 35.4 proved that several properties of algorithms cannot be computed or verified. The following theorem says that this is not a coincidence, and essentially any nontrivial property of algorithms cannot be verified efficiently.

Theorem 35.5 (Rice – Uspensky)**.** *Let* FC *be the set of all computable functions, and let* $\mathcal{P} \subseteq \mathrm{FC}$ *be some nontrivial set of computable funcitons (*$\mathcal{P} \neq \emptyset$ *and* $\mathcal{P} \neq \mathrm{FC}$*). Let* $U : \mathbb{N}^2 \to \mathbb{N}$ *be a universal Gödel function. Then the set* $S = \{n \in \mathbb{N} \ : \ U_n \in \mathcal{P}\}$ *is undecidable.*

Proof. This theorem can be proved using almost the same method as Theorem 35.4.

Let $f : \mathbb{N} \to \mathbb{N}$ be a partial function that is not defined at all $x \in \mathbb{N}$. Without loss of generality we may assume that $f \in \mathcal{P}$. Let $g : \mathbb{N} \to \mathbb{N}$ be a function from $\mathrm{FC} \setminus \mathcal{P}$.

Let K be a enumerable but undecidable set. Consider the partial function $V : \mathbb{N}^2 \to \mathbb{N}$ such that

$$V(n, x) = \begin{cases} g(x) & \text{if } n \in K \\ \text{undefined} & \text{if } n \notin K \end{cases}.$$

Note that $V_n \notin \mathcal{P}$ iff $n \in K$. Since U is Gödel universal function there is a computable function s such that $V(n, x) = U(s(n), x)$. Hence, $s(n) \notin S$ iff $n \in K$. As a result, S is undecidable. $\square$

Let U be a Gödel universal funciton. A simple corollary of the Theorem 35.4 is that the set $\{n \in \mathbb{N} \ : \ U(n, x) \text{ is not defined for all } x \in \mathbb{N}\}$ has infinitely many elements but it is not equal to the set of natural numbers. This simple observation allows us to prove that not all universal functions are Gödel universal functions.

Theorem 35.6. *There is a universal function* $U : \mathbb{N}^2 \to \mathbb{N}$ *such that* U *is not Gödel universal function.*

Proof. Let U be a Gödel universal function. Note that the set $S = \{n \in \mathbb{N} \ : \ U(n, x) \text{ is defined for some } x \in \mathbb{N}\}$ is enumerable. Hence, there is a computable bijection $d : \mathbb{N} \to S$.

Let us consider $V : \mathbb{N}^2 \to \mathbb{N}$ such that $V(i + 1, x) = U(d(i), x)$ for $i, x \in \mathbb{N}$ and $V(1, x)$ undefined for all $x \in \mathbb{N}$. It is clear that V is computable universal function. However, the set

$$\{n \in \mathbb{N} \ : \ V(n, x) \text{ is not defined for all } x \in \mathbb{N}\} = \{1\}$$

cannot be undecidable. As a result, V is not Gödel universal function. □

In fact, Friedberg (Journal of Symbolic Logic 23 (1958), 309-318) constructed a universal function such that any computable function has only one number; i.e., it is possible to create a programming language such that each programming problem has a unique solution in it.

https://doi.org/10.2307/2964290

End of The Chapter Exercises

35.2 Let $U : \mathbb{N}^2 \to \mathbb{N}$ be a computable universal function for the class of univariate computable functions. Assume that for any universal computable function $V : \mathbb{N}^2 \to \mathbb{N}$, there is a computable function $s : \mathbb{N} \to \mathbb{N}$ such that

$$V(n, x) = U(s(n), x)$$

for all $n, x \in \mathbb{N}$.

35.3 Let U be a Gödel universal function. Show that for any computable function $V : \mathbb{N}^3 \to \mathbb{N}$, there is a total computable function $s : \mathbb{N}^2 \to \mathbb{N}$ such that $U(s(m,n), x) = V(m, n, x)$ for all $m, n, x \in \mathbb{N}$.

36. Fixed Point Theorem

Probably the most impresive example of a quine is Quine Relay, a Ruby program that generates Rust program that generates Scala program that generates ...(through 128 languages in total)... REXX program that generates the original Ruby code again.

`https://github.com/mame/quine-relay`

Probbaly one of the most surprinsg phenomenon in the world of esoteric programming is existence of quines, the programs that print themself.

This chapter proves existence of such programs in almost all programming languages.

Theorem 36.1 (Kleene's Fixed Point Theorem)**.** *Let $h : \mathbb{N} \to \mathbb{N}$ be a total computbale function, and let $U : \mathbb{N}^2 \to \mathbb{N}$ be a Gödel universal function. Then there is $n \in \mathbb{N}$ such that U_n is equal to $U_{h(n)}$.*

Proof. Let $f : \mathbb{N} \to \mathbb{N}$ be a computable function such that no computable function $g : \mathbb{N} \to \mathbb{N}$ can differ from f everywhere, such a function exists by Theorem 34.6. Note that there is a total computable function $g : \mathbb{N} \to \mathbb{N}$ such that $U_{f(n)} = U_{g(n)}$ provided that $f(n)$ is defined. Indeed, let us consider $V(x, y) = U(f(x), y)$; since U is a Gödel universal function, there is a total function $g(n)$ such that $V(x, y) = U(g(x), y)$.

Assume for the sake of contradiction that $U_n \neq U_{h(n)}$ for all $n \in \mathbb{N}$. Let $t : \mathbb{N} \to \mathbb{N}$ be a total computable function such that $t(n) = h(g(n))$. It is clear that if f is different from t evrywhere, which contradictis to the definition of f. $\square$

Corollary 36.1. *Let $U : \mathbb{N}^2 \to \mathbb{N}$ be a Gödel universal function. Then there is $n \in \mathbb{N}$ such that $U(n, x) = n$ for all $x \in \mathbb{N}$.*

Proof. Let $q : \mathbb{N} \to \mathbb{N}$ be a computable total function such that $U(q(n), x) = n$ for all $x \in \mathbb{N}$ (such a function exists since U is a Gödel universal function). Note that there is n such that U_n is equal to $U_{q(n)}$ which implies that $U(n, x) = U(q(n), x) = n$ for all $x \in \mathbb{N}$. $\square$

Exercise 36.1. *Prove that there is a program on the programing of your choice that prints its text backwards.*

Potentially, the function h in Kleene's fixed point theorem (Theorem 36.1) may depend on a parameter; however, even in this case there is a fixed point theorem.

Theorem 36.2. *Let $h : \mathbb{N}^2 \to \mathbb{N}$ be a total computbale function, and let $U : \mathbb{N}^2 \to \mathbb{N}$ be a Gödel universal function. Then there is total computable function $m \in \mathbb{N}$ such that $U(h(p, m(p)), x) = U(m(p), x)$ for all $p, x \in \mathbb{N}$*

End of The Chapter Exercises

36.2 Show that there are different $p, q \in \mathbb{N}$ such that $U(p, x) = q$ and $U(q, x) = p$ for all $x \in \mathbb{N}$.

36.3 Let $h : \mathbb{N} \to \mathbb{N}$ be a total computbale function, and let $U : \mathbb{N}^2 \to \mathbb{N}$ be a Gödel universal function. Show that there are infinetely many $n \in \mathbb{N}$ such that $U_n = U_{h(n)}$.

36.4 Prove Theorem 36.2.

37. m-Reductions

In the proofs of Theorems 34.4 and 35.4, to prove that a set S is undecidable we proved that S is decidable iff an undecidable set K should be also decidable, which lead to the conclusion that S is undecidable. This was done using "reduction" argument. We constructed a total computable function $f : \mathbb{N} \to \mathbb{N}$ such that $f(x) \in S$ iff $x \in K$ for all $x \in \mathbb{N}$. In this chapter we will study this method in more details.

Definition 37.1. *Let $A, B \subseteq \mathbb{N}$. We say that A is m-*reducible *to B*[1] *($A \leq_m B$) if there is a total computable function $f : \mathbb{N} \to \mathbb{N}$ such that $x \in A$ iff $f(x) \in B$ for all $x \in \mathbb{N}$. We say that f m-*reduces *A to B.*

[1] The letter "m" here stands for "many-to-one"; however, Sipser's "Introduction to the Theory of Computation" suggests to call such reductions "mapping reductions" giving another life for the letter m in this notation.

This notion allows us to translate properties from one set to another.

Theorem 37.1. *Let $A, B \subseteq \mathbb{N}$ such that $A \leq_m B$.*

- *If B is decidable, then A is decidable.*
- *If B is enumerable, then A is enumerable.*

Exercise 37.1. *Let $A, B, C \subseteq \mathbb{N}$. Show that*

- *$A \leq_m A$, and*
- *if $A \leq_m B$ and $B \leq_m C$, then $A \leq_m C$.*

Notice that the sets $\emptyset$ and $\mathbb{N}$ behave differently than other decidable sets with respect to $\leq_m$.

Remark 37.1. *Let $A, B \subseteq \mathbb{N}$.*

- *$A \leq_m \emptyset$ iff $A = \emptyset$,*
- *$A \leq_m \mathbb{N}$ iff $A = \mathbb{N}$, and*
- *$A \leq_m B$ provided that A and B are dicidable and $B \notin \{\emptyset, \mathbb{N}\}$.*

However, in case of enumerable sets, the situation is not that simple.

Exercise 37.2. *Show that there are enumerable sets $A, B \subseteq \mathbb{N}$ such that $A \not\leq_m B$.*

In other words, enumerable sets form layers of sets increasing with repspect to $\leq_m$. So the questions is whether there is the last layer or not, the following theorem give an affirmative answer to this question.

Theorem 37.2. *In the class of enumerable sets, there are sets maximal with respect to m-reducibility; i.e., there is an enumerable set B sucht that $A \leq_m B$ for any enumerable set A.*

Proof. Let W be a enumerable universal set for the set of all enumerable subsets of $\mathbb{N}$ (it exists by Theorem 34.2). We claim that the set $B = \{\langle n, x\rangle \ : \ (n, x) \in W\}$ satisfies the requirement of the theorem. Indeed, let A be enumerable set. Then there is $n \in \mathbb{N}$ such that $W_n = A$. Hence, it is easy to see that $f(x) \in B$ iff $n \in A$, where $f(x) = \langle n, x\rangle$. $\square$

Definition 37.2. *An enumerable set B maximal with respect to m-reducibility is called m*-complete for the class of enumerable sets*.*

Theorem 37.3. *Let U be a Gödel universal function. Then $\{x \in \mathbb{N} \ : \ U(x, x) \text{ terminates}\}$ is m-complete for the class of enumerable sets.*

Proof. Let $K \subseteq \mathbb{N}$ be a enumerable set. Let us consider a computable function $V : \mathbb{N}^2 \to \mathbb{N}$s such that $V(n, x) = 1$ if $n \in K$ and undefined otherwise. Since U is Gödel universal function, there is a total computable $s : \mathbb{N} \to \mathbb{N}$ such that $V_n = U_{s(n)}$. Hence, $U_{s(n)}$ decides $\mathbb{N}$ if $n \in K$ and $U_{s(n)}$ decides $\emptyset$ if $K \notin K$. Therefore $s(n) \in D$ iff $n \in K$. $\square$

End of The Chapter Exercises

37.3 Prove that the set of all programs that halt on the input 0 is *m*-complete for the class of enumerable sets.

37.4 Prove that the set of all programs that halt on at least one input is *m*-complete for the class of enumerable sets.

38. Arithmetical Hierarchy

Theorem 33.6 proves that any decidable set is the image of some total computable function from $\mathbb{N}$; identifying sets and predicates we give the following reformulation.

Theorem 38.1. *A predicate$A(x)$ of natural numbers is enumerable iff there is a decidable predicate $B(x,y)$ such that the following formula is always true:*

$$A(x) \iff \exists y\ B(x,y).$$

(Here and in the sequel, we are going to write $\exists x\ P(x)$ denotes the statement "there is an x from the domain of P such that $P(x)$ is true.)

A natural questions can be asked: "What can be said about other combinations of quantifiers?".

It is easy to see that if $A(x)$ is a predicate such that

$$A(x) \iff \exists y\ \exists z\ C(x,y,z),$$

where $C(x,y,z)$ is decidable, then $A(x)$ is enumerable. Indeed, let $B(x, \langle y,z\rangle) = C(x,y,z)$; then $B(x,w)$ is decidable and

$$A(x) \iff \exists y\ \exists z\ B(x, \langle y,z\rangle) \iff \exists w\ B(x,w).$$

If $A(x)$ is a predicate such that

$$A(x) \iff \forall y\ B(x,y),$$

where $B(x,y)$ is decidable, then the negation of $A(x)$ is enumerable (we call such sets and predicates *coenumerable*). Indeed,

$$\neg A(x) \iff \exists y\ \neg B(x,y),$$

and $\neg B(x,y)$ is decidable since the compliment of a decidable set is decidable.

This question gives rise to the following definition.

Definition 38.1. *We say that a predicate A on $\mathbb{N}$ belongs to the class Σ_n iff there is a deciable predicate B on $\mathbb{N}^{n+1}$ such that*

$$A(x) \iff \exists y_1\ \forall y_2\ \exists y_3\ \ldots B(x, y_1, \ldots, y_n).$$

Similarly, a predicate A on $\mathbb{N}$ belongs to the class Σ_n iff there is a deciable predicate B on $\mathbb{N}^{n+1}$ such that

$$A(x) \iff \forall y_1\ \exists y_2\ \forall y_3\ \ldots B(x, y_1, \ldots, y_n).$$

Previous observations can be generalized as follows.

Theorem 38.2. *1. The set Σ_1 consists of enumerable sets.*

2. *If a predicate belongs to Σ_n, then its negation belongs to Π_n.*

3. *The set Σ_n does not change if we allow groups of quantifiers of the same type instead of single quantifiers.*

Like enumerable and decidable sets, Σ_n and Π_n sets have several good properties.

Theorem 38.3. *Union and intersection of two Σ_n (Π_n) sets are also Σ_n (Π_n) sets.*

Proof. We prove the statement for the union of two Σ_n sets, all other cases have similar proofs. Let $A_1, A_2 \in \Sigma_n$. By the definition of Σ_n, there are decidable B_1 and B_2 such that

$$A_1(x) \iff \exists y_1\ \forall y_2\ \ldots B_1(x, y_1, \ldots, y_n)$$
and
$$A_2(x) \iff \exists z_1\ \forall z_2\ \ldots B_2(x, z_1, \ldots, z_n).$$

Therefore

$$A_1(x) \wedge A_2(x) \iff \exists y_1, z_1\ \forall y_2, z_2\ \ldots B_1(x, y_1, \ldots, y_n) \wedge B_2(x, z_1, \ldots, z_n);$$

which implies that $A_1 \wedge A_2$ is a Σ_n predicate (Theorem 38.2). □

Exercise 38.1. *Prove that $\Sigma_n \cup \Pi_n \subseteq \Sigma_{n+1} \cup \Pi_{n+1}$ for any $n \in \mathbb{N}_0$.*

We can also prove an extension of Theorem 37.1.

Theorem 38.4. *Let $n \in \mathbb{N}$, and let $A, B \subseteq \mathbb{N}$ such that $A \leq_m B$. If $B \in \Sigma_n$, then $A \in \Sigma_n$.*

Proof. Let f be the reduction from A to B. Let C be a decidable predicate such that

$$B(x) \iff \exists y_1\ \forall y_2\ \ldots C(x, y_1, \ldots, y_n).$$

Note that, by the definition of the reduction,

$$A(x) \iff \exists y_1\ \forall y_2\ \ldots C(f(x), y_1, \ldots, y_n);$$

hence, $A(x) \in \Sigma_n$ since $C(f(x), y_1, \ldots, y_n)$ is decidable. □

Exercise 38.2. *Prove that if a set $A \in \Sigma_n$, then $A \times A \in \Sigma_n$.*

We defined infinitely many classes of sets; however, we have not shown that all these sets are different (except Σ_0 and Π_0 which are equal to the class of all decidable sets). Like in the case of enumebrable sets, to prove this we need to use universal sets.

Theorem 38.5. *For any $n \in \mathbb{N}$, there is a set $U \in \Sigma_n$ universal for Σ_n subsets of $\mathbb{N}$.*

Proof. We prove the statement using induction by n. The base case for $n = 1$ is true by Theorems 34.2 and 38.2.

Let us prove the induction step. Assume $V \in \Sigma_n$ is a universal set for Σ_n subsets of $\mathbb{N}$. First we show that $(\mathbb{N} \setminus V) \in \Pi_n$ is a universal set for Π_n subsets of $\mathbb{N}$. Let $A \in \Pi_n$. Then there is $k \in \mathbb{N}$ such that $\mathbb{N} \setminus A = V_k$ since $(\mathbb{N} \setminus A) \in \Sigma_n$. Therefore, $A = \mathbb{N} \setminus V_k$ which implies that $\mathbb{N} \setminus V$ is universal for Π_n subsets of $\mathbb{N}$.

Note that there is $W' \in \Pi_n$ that is universal for Π_n subsets of $\mathbb{N}^2$. Indeeed, let $W' = \{(n, x, y) \in \mathbb{N}^3 \ : \ (n, \langle x, y \rangle) \in W\} \in \Sigma_n$. Let $A' \in \Pi_n$ and B' be decidable set such that

$$A'(x, y) \iff \exists z_1 \ \forall z_2 \ \ldots B'(x, y, z_1, \ldots, z_n).$$

Consider $A \subseteq \mathbb{N}$ such that

$$A(\langle x, y \rangle) \iff \exists z_1 \ \forall z_2 \ \ldots B'(x, y, z_1, \ldots, z_n);$$

it is clear that $A \in \Pi_n$. Hence, there is $k \in \mathbb{N}$ such that $W_k = A$, which implies that $W'_n = A'$. Therefore W' is universal for Π_n subsets of $\mathbb{N}^2$.

Let us consider the set $U = \{(k, x) \in \mathbb{N}^2 \ : \ \exists y \ (k, x, y) \in W'\}$. We claim that it is universal for Σ_{n+1} subsets of $\mathbb{N}$. Let $A \in \Sigma_{n+1}$. Then there is $B \in \Pi_n$ such that

$$A(x) \iff \exists y \ B(x, y).$$

In addittion, there is $k \in n$ such that $W'_k = B$. Therefore,

$$A(x) \iff \exists y \ W'(k, x, y) \iff U(k, x).$$

Hence, U is universal for Σ_{n+1} subsets of $\mathbb{N}$. □

We proved that the compliment of an enumerable universal set for the class of enumerable sets is not enumerable. Similar result can be shown for Σ_n.

Theorem 38.6. *Let $n \in \mathbb{N}$ and let $W \subseteq \mathbb{N}^2$ be universal for Σ_n subsets of $\mathbb{N}$. Then $W \notin \Pi_n$.*

Proof. Assume that $W \in \Pi_n$, this implies that that the set $D = \{n \in \mathbb{N} \ : \ (n, n) \notin W\}$ belongs to Σ_n. Note that there is $k \in \mathbb{N}$ such that $W_k = D$ since W is universal for Σ_n subsets if $\mathbb{N}$. However, $W_k(k)$ is true only if $W_k(k)$ is false, which is a contradiction. Therefore, $W \notin \Pi_n$. □

Corollary 38.1. *For any $n \in \mathbb{N}$, $\Pi_n \not\subseteq \Sigma_{n+1}$ and $\Sigma_n \not\subseteq \Pi_{n+1}$.*

End of The Chapter Exercises

38.3 Prove that if $A, B \in \Sigma_n$, then $A \setminus B \in \Sigma_{n+1} \cap \Pi_{n+1}$.

Part X

Appendices

A. Formal Power Series

Formal power series is an algebraic analogy of power series from analysis. A formal power series is something like $a_0 + xa_1 + x^2a_2 + \ldots a_n x^n + \ldots$; to describe such an object it is enough to define the sequence $\{a_n\}_{n\geq 0}$ since x is a variable.

Definition A.1. *We say that $F(x)$ is a* formal power series *in the variable x, if $F(x) = \{f_n\}_{n\geq 0}$. To distinguish between formal power series and sequences, we write formal power series as $\sum_{n\geq 0} f_n x^n$. We say that f_n is the coefficient of x^n in $F(x)$.*

We say that two formal power series $F(x)$ and $G(x)$ are equal iff for all $n \geq 0$, the coefficients of x^n in $F(x)$ and $G(x)$ are the same.

The set of all the power series in the variable x is denoted as $\mathbb{R}[[x]]$.

A.1 Arithmetic Operations

We can perform all the standard operations with the formal power series:

$$\sum_{n\geq 0} a_n x^n \pm \sum_{n\geq 0} b_n x^n = \sum_{n\geq 0} (a_n \pm b_n) x^n,$$

$$c \sum_{n\geq 0} a_n x^n = \sum_{n\geq 0} (ca_n) x^n,$$

and

$$\sum_{n\geq 0} a_n x^n \sum_{n\geq 0} b_n x^n = \sum_{n\geq 0} (\sum_{k=0}^{n} a_k b_{n-k}) x^n.$$

These operations satisfy all the properties we may expect from them.

Theorem A.1. *Let $F(x)$, $G(x)$, and $H(x)$ be some formal power series. Then the following equalities hold:*

- $(F(x) + G(x)) + H(x) = F(x) + (G(x) + H(x))$,
- $F(x) + G(x) = G(x) + F(x)$,
- $(F(x)G(x))H(x) = F(x)(G(x)H(x))$,
- $F(x)G(x) = G(x)F(x)$, *and*

- $(F(x) + G(x))H(x) = F(x)H(x) + G(x)H(x)$.

For example, $(1 - x)(1 + x + x^2 + \dots) = 1$. Thus we can say that the series $(1 - x)$ has an inverse, and that inverse is equal to $1 + x + x^2 + \dots$.

Theorem A.2. *A formal power series $\sum_{n \geq 0} f_n x^n$ has an inverse iff $f_0 \neq 0$ and moreover this inverse is unique.*

Proof. Assume that a power series $F(x) = \sum_{n \geq 0} f_n x^n$ has an inverse $G(x) = \sum_{n \geq 0} g_n x^n$. In this case $F \cdot G = 1$ i.e. $f_0 g_0 = 1$ and $f_0 \neq 0$. Moreover, $\sum_{k=0}^{n} f_k g_{n-k} = 0$; from which we can conclude that

$$g_n = -\frac{1}{f_0} \sum_{k>0} f_k g_{n-k}. \tag{A.1}$$

This determine g_n uniquely, as stated.

Conversely, if $f_0 \neq 0$, (A.1) determines the sequence $\{g_n\}_{n \geq 0}$. □

A.2 Composition

Another operation we may need to perform is composition; a composition of the power series $F(x)$ and $G(x)$ is a power series $F(G(x))$; i.e. $F(G(x)) = \sum_{n \geq 0} a_n G^n(x)$, where $F(x) = \sum_{n \geq 0} a_n x^n$ Note that the composition is well-defined iff the coefficient of x^0 in $G(x)$ is 0 or if $F(x)$ is a polynomial.

A.3 Derivative

Let $F(x) = \sum_{n \geq 0} f_n x^n$ be a formal power series. Then the derivative $F'(x)$ (we also denote it as $\frac{d}{dx} F(x)$) of $F(x)$ is equal to $\sum_{n \geq 1} n f_n x^{n-1} = \sum_{n \geq 0} (n+1) f_{n+1} x^n$.

The derivatives of formal power series satisfy the same properties as derivatives of functions.

Theorem A.3. *Let $F(x)$, $G(x)$, and $H(x)$ be some formal power series. Then the following equalities hold:*

- $\frac{d}{dx}(F(x) + G(x)) = F'(x) + G'(x)$, *and*
- $\frac{d}{dx}(F(x)G(x)) = F'(x)G(x) + F(x)G'(x)$.

As a corollary of these statements we can derive a formula for the derivative of $1/F(x)$.

Corollary A.1. *Let $F(x)$ be a formal power series such that $1/F(x)$ exists. In this case $\frac{d}{dx} \frac{1}{F(x)} = -\frac{F'(x)}{F^2(x)}$.*

Proof. Note that $F(x)\frac{1}{F(x)} = 1$. Hence, $\frac{d}{dx}(F(x)\frac{1}{F(x)}) = 0$. Using the formula for the derivative of a product we may conclude that $F'(x)\frac{1}{F(x)} + F(x)\frac{d}{dx}\frac{1}{F(x)} = 0$. As a result, $-\frac{F'(x)}{F^2(x)} = \frac{d}{dx}\frac{1}{F(x)}$. □

Remark A.1. *If $F'(x) = 0$, then $F(x) = a_0$.*

We denote the formal power series $\sum_{n\geq 0}\frac{1}{n!}x^n$ by e^x (since the Taylor series of e^x is equal to $\sum_{n\geq 0}\frac{1}{n!}x^n$).

Remark A.2. *If $F'(x) = F(x)$, then $F(x) = ce^x$ for some $c \in \mathbb{R}$.*