# Contents

## Introduction 3

1 Number Theory 4

1.1 Basic Notions and the Fundamental Theorem of Arithmetic 4

1.2 Congruences 9

1.3 The Euler-Fermat Theorem 11

1.3.1 Euler’s Phi Function 11

1.3.2 Residue Systems 13

1.3.3 The Euler-Fermat Theorem 13

1.4 Linear Congruences 14

1.4.1 Existence of solutions 15

1.4.2 Simultaneous Congruences 17

1.5 Number-theoretic Algorithms 18

1.5.1 Effectiveness of Algorithms 18

1.5.2 Basic Arithmetic Operations 20

1.5.3 Modular Exponentiation 21

1.5.4 The Calculation of the Greatest Common Divisor 23

1.5.5 Solution of Linear Congruences 24

1.5.6 Primality tests 26

1.5.7 Public Key Cryptography 29

2 Linear Algebra 31

2.1 Analytic Geometry in the Space 31

2.1.1 The Coordinate System 31

2.1.2 Equations of a Line 33

2.1.3 Equation of a Plane 35

2.2 The Space $\mathbb{R}^n$ 35

2.2.1 The Notion of $\mathbb{R}^n$ 35

2.2.2 Subspaces of $\mathbb{R}^n$ 37

2.2.3 Generated Subspace 38

2.2.4 Linear Independence 41

2.2.5 The I-G Inequality 43

2.2.6 Basis and Dimension 44

References 50

Index 51
Introduction

These notes are based on the lecture notes [9] written by Dávid Szeszlér in Hungarian and cover the material of the course Introduction to the Theory of Computing I. given in every fall semester at the Faculty of Electrical Engineering and Informatics of Budapest University of Technology and Economics. The text follows closely the structure of the Hungarian version, many parts of it are just translations of the original.

The material is divided into two chapters, the first one covers the basics of number theory and also some applications. In the second one we discuss the basics of linear algebra. We will only see a special case of a general theory, though this is without doubt the most important special case. Not only that it provides a very useful tool in almost every branch of mathematics, but it has a fundamental role in many parts of computer science.

I would like to thank Rita Csákány for reading these notes and making comments.
1 Number Theory

Number theory is one of the oldest branches of mathematics. It investigates the properties of the integers, many basic notions of it were defined and named by the ancient Greeks. It provided many of the most famous problems of mathematics, some of them turned out to be very challenging and deep. After hundreds and thousands of years there are still several unsolved questions among them.

Despite all this there was no general interest towards number theory outside mathematics until the first important application appeared in 1977, when Ronald Rivest, Adi Shamir and Len Adleman discovered the so called RSA algorithm (what was named after the initials of its creators). It is used to encrypt and decrypt messages with the help of public keys, i.e. keys that can be given to anyone without endangering the privacy. The connection with cryptography made this branch very important in computer science, especially in the age of the internet. In this chapter we discuss the basics of number theory and describe some of its applications, including the RSA algorithm.

1.1 Basic Notions and the Fundamental Theorem of Arithmetic

In this section we discuss the basic notions of number theory. Most of the definitions and theorems should be familiar to anyone from high school, but here we also give the exact proofs of the claims. Unless it is told otherwise, every variable denotes an integer in this chapter.

Definition 1.1.1. If $a, b \in \mathbb{Z}$ are integers, then we say that $a$ is a divisor of $b$ (or $a$ divides $b$, $b$ is a multiple of $a$) if there is an integer $c \in \mathbb{Z}$ such that $b = ac$. This is denoted by $a \mid b$. If $a$ does not divide $b$, then we write $a \nmid b$. The number $a$ is a proper divisor of $b$ if $a \mid b$ and $1 < |a| < |b|$ hold.

Note that other authors may not exclude the number 1 from the set of proper divisors. One checks easily that $13 \mid 91$, $-7 \mid 63$, $2 \mid 0$ and $-8 \nmid -36$ hold. At first sight it is maybe surprising that $0 \mid 0$ holds too since $0 = 0 \cdot c$ for every $c \in \mathbb{Z}$. But this does not mean that the operation "dividing by zero" is defined. The divisors of 10 are $\pm 1$, $\pm 2$, $\pm 5$ and $\pm 10$ while the proper divisors of 10 are $\pm 2$ and $\pm 5$.

Definition 1.1.2. The integer $p \in \mathbb{Z}$ is called prime if $|p| > 1$ and $p$ does not have a proper divisor. In other words: $p = ab$ holds if and only if $a = \pm 1$ or $b = \pm 1$. If $|p| > 1$ and $p$ is not prime, then it is called a composite number. The numbers 0 and $\pm 1$ are neither prime nor composite.

Examples of prime numbers are 3, 103 and $-7$. The negative primes are just the opposites of the positive primes.

Remark. Many authors call the above defined numbers irreducibles and define the notion of prime numbers by the property that if $p \mid ab$ holds for a product, then $p \mid a$ or $p \mid b$ must also hold. Since these two definitions give the same notion for integers, we do not follow this practice. The reason why others do it is that number theory can be worked out in "larger domains" and in general the two notions may differ. We will see such examples later but aside from these we restrict ourselves to the set of integers and recommend the book [6] to the interested reader.
The following theorem has a crucial role in number theory (which is reflected in its name) and also shows the importance of primes:

**Theorem 1.1.1** (Fundamental Theorem of Arithmetic). Every integer different from 0 and ±1 can be represented as a product of primes. This representation is unique up to the order and the sign of the factors.

For example two different representations of the number 100 are $2 \cdot 2 \cdot 5 \cdot 5$ and $(-5) \cdot 2 \cdot (-2) \cdot 5$, which shows that uniqueness cannot be achieved in the theorem without disregarding the order and the sign of the prime factors. We can also see why it is useful to exclude the numbers ±1 from the set of primes. Otherwise the representation would not be unique since we could write $4 = 2 \cdot 2 = 1 \cdot 2 \cdot 2$. On the other hand, the numbers 0 and ±1 can not be written as product of primes, they must be excluded in the theorem. Note that prime numbers can be considered as products that have only one factor and then the statement of the theorem remains true for them too.

**Proof of existence of the factorization in Theorem 1.1.1.** We give a simple process which provides the factorization for any $n \in \mathbb{Z}$ with $|n| > 1$. We will store a factorization all along, initially this will be the number $n$ itself (a product with one factor). Once we have a product $n = a_1 a_2 \ldots a_k$ where all the $a_i$'s are prime numbers we stop. If at least one of the factors, say $a_i$ is composite, then it has a proper divisor. That is, we can choose some $b, c \in \mathbb{Z}$ with $|b|, |c| > 1$ such that $a_i = bc$. We replace the factor $a_i$ with $bc$ in the product and proceed. In every step we increase the number of factors by 1 and the absolute value of every factor is at least 2. Hence after at most $\log_2 |n|$ steps our procedure ends and gives the required factorization. □

Before we complete the proof of the fundamental theorem, we make some remarks and show some (counter)examples. First note that the (at this point still unproved) uniqueness part is the "powerful" part of the fundamental theorem. Namely, it assures that the obtained factorization gives the arithmetic structure of the numbers and this way it makes possible to calculate all of their divisors, for example.

Although the fundamental theorem may seem evident, it is not too hard to give such "domains" where it does not hold. For instance, let us forget about the odd numbers for a moment. The set of even numbers is similar to the integers. By this we mean that the sum, difference and product of two even numbers is also even. Moreover, the notion of divisibility can be defined the same way as before. But here we do not have a unique factorization: for example $36 = 2 \cdot 18 = 6 \cdot 6$ and none of these representations can be split up further. The reader may notice that our definition for the prime numbers is not applicable here, because the number 1 is not an element of our set (i.e. it is not even). However, it is not hard to modify the definition so that it yields the right notion.

A more sophisticated example is the set of complex numbers of the form $a + ib\sqrt{5}$, where $a$ and $b$ are integers and $i$ is the imaginary unit, i.e. $i^2 = -1$. Again, this is closed under addition and multiplication, but also contains the number 1. It is true that $9 = 3^2 = (2-i\sqrt{5})(2+i\sqrt{5})$ but these factors do not have "proper divisors". Of course we should clarify what a proper divisor means here, but we do not go into the details, we refer to the book [6] instead.

As a final remark, we mention that though these domains may seem artificial for the first sight, still examples similar to the last one occur naturally in number theory. For example, they play a major role in problems like Fermat’s Last Theorem which was formulated in 1637 and was proved by Andrew Wiles in 1994. The theorem states that for any exponent $n \in \mathbb{N}$
greater than 2 the equation $x^n + y^n = z^n$ does not have an integer solution. Many special cases and similar problems can be treated relatively easily, but they are beyond the scope of these notes.

**Proof of uniqueness of the factorization in Theorem 1.1.1.** It is clearly enough to show that every positive integer greater than 1 can be written uniquely (up to order) as a product of primes. So assume that $n \in \mathbb{N}$, $n > 1$. We prove by induction. The assertion is true for every prime, in particular for $n = 2$, so assume that $n > 2$ is composite and the assertion is true for every $1 < n' < n$. If $n = p_1 \ldots p_r = q_1 \ldots q_s$ such that the $p_i$’s and $q_j$’s are primes, then $r, s \geq 2$ (since $n$ is not a prime). If $p_i = q_j$ holds for some $i$ and $j$, then dividing $n$ by this prime we get two non-empty products giving a smaller number $n'$. By induction the remaining primes on the two sides of the equality differ by order only, hence the same holds for the original products.

It remains to handle the case when $p_i \neq q_j$ for every $i$ and $j$. After a possible relabeling we may assume that $p_1 \leq p_i$ and $p_1 \leq q_j$ hold for every $i$ and $j$. Let us define then $n' = (q_1 - p_1)q_2 \ldots q_s$. We have assumed $q_1 \geq p_1$ and $q_1 \neq p_1$, hence $n > n' > q_1 - p_1 \geq 1$ follows (since $s \geq 2$). We now show $n'$ has a factorization which contains $p_1$ and another one without $p_1$. This contradicts our hypothesis and this contradiction shows that this case is impossible and the theorem is proved. If $q_1 - p_1 = 1$, then we can simply omit this factor from the product to obtain an appropriate representation of $n'$. Otherwise $q_1 - p_1$ can be written uniquely (up to order) as a product of primes by induction. Replacing this factor by this product in the definition of $n'$ above we get a factorization of $n'$. Since $p_1 \nmid q_1$ (because $q_1$ is prime) we also have that $p_1 \nmid q_1 - p_1$. So $p_1$ does not occur among the primes in the factorization of $q_1 - p_1$. Recall that $p_1 \neq q_j$ is also true, hence we get a factorization without the prime $p_1$.

Finally,

$$n' = (q_1 - p_1)q_2 \ldots q_s = q_1q_2 \ldots q_s - p_1q_2 \ldots q_s = p_1p_2 \ldots p_r - p_1q_2 \ldots q_s = p_1(p_2 \ldots p_r - q_2 \ldots q_s).$$

Replacing $p_2 \ldots p_r - q_2 \ldots q_s$ by an optional prime factorization of it or simply omit this factor in the case when it equals 1 we get a prime factorization of $n'$ including $p_1$. This is a contradiction, and the proof of the theorem is now complete. \hfill \Box

The fundamental theorem was proved for the set of integers, but then it follows also for the natural numbers: every positive integer greater than 1 has a prime factorization which is unique up to order. This makes it possible to define the *canonical representation* of the positive integers. We obtain this by collecting the identical primes in the factorization into powers and by ordering the powers by the magnitude of the bases. That is, we get the form $n = p_1^{\alpha_1} \ldots p_k^{\alpha_k}$, where $p_1 < p_2 < \cdots < p_k$ are primes and $\alpha_1, \ldots, \alpha_k$ are positive integers. Observe that this canonical representation is unique, though many times we only require that the prime bases in this representation are pairwise different (and not necessarily ordered by magnitude). Hopefully this causes no confusion in the future. As an example, the canonical representation of the number 600 is $2^3 \cdot 3^1 \cdot 5^2$ (of course we often omit the exponent 1).

Many times it is useful to allow the exponent zero in the representation. For example it makes possible to use the same primes in the representations of two different numbers, as in the following
**Proposition 1.1.2.** Let us assume that $p_1, \ldots, p_k$ are pairwise different positive primes and $n = p_1^{\alpha_1} \ldots p_k^{\alpha_k}$, where $\alpha_1, \ldots, \alpha_k$ are non-negative integers. Then the positive integer $m$ divides $n$ if and only if $m = p_1^{\beta_1} \ldots p_k^{\beta_k}$, where $0 \leq \beta_1 \leq \alpha_1, \ldots, 0 \leq \beta_k \leq \alpha_k$ are integers.

**Proof.** If $m$ is of the form given in the proposition, then $n = ml$, where $l = p_1^{\alpha_1-\beta_1} \ldots p_k^{\alpha_k-\beta_k}$, hence $m \mid n$.

Now assume that $m \mid n$ and that the canonical representation of $m$ is $q_1^{\tau_1} \ldots q_s^{\tau_s}$. Then $n = ml$ for some $l \in \mathbb{Z}$. We can get a factorization of $n$ by multiplying the factorization of $m$ and $l$. But then by the uniqueness part of the fundamental theorem every $q_i$ must coincide with some $p_j$. This means that $m$ can be written as $p_1^{\beta_1} \ldots p_k^{\beta_k}$ where some of the exponents may be 0. Assume that an exponent, say $\beta_i$, is strictly bigger than $\alpha_i$, then

$$p_i^{\beta_i-\alpha_i} \mid \frac{n}{p_i^{\alpha_i}} = p_1^{\alpha_1} \ldots p_{i-1}^{\alpha_{i-1}} p_{i+1}^{\alpha_{i+1}} \ldots p_k^{\alpha_k}.$$ 

where $\beta_i - \alpha_i \geq 1$. The same way as before we get that $p_i$ must coincide with some $p_j$, $j \neq i$. But this is impossible, since the primes $p_1, \ldots, p_k$ are pairwise distinct.

This last result makes it possible to give a formula for the number of divisors. For a positive integer $n$ the number of its divisors is denoted by $d(n)$ (note that other notations like $\nu(n)$, $\tau(n)$ and $\sigma_0(n)$ are also common).

**Corollary 1.1.3.** If $n > 1$ is an integer and its canonical representation is $n = p_1^{\alpha_1} \ldots p_k^{\alpha_k}$, then

$$d(n) = (\alpha_1 + 1) \ldots (\alpha_k + 1).$$

**Proof.** The product given in the statement is the number of products of the form $p_1^{\beta_1} \ldots p_k^{\beta_k}$, where $0 \leq \beta_1 \leq \alpha_1, \ldots, 0 \leq \beta_k \leq \alpha_k$. By the previous proposition these products give all the divisors of $n$, and by the uniqueness of the prime factorization they give every divisor only once.

**Proposition 1.1.2** also helps us to determine the greatest common divisor and the least common multiple of two numbers. Although these notions are basically defined by their names, we give the formal definitions:

**Definition 1.1.3.** If $n, m \in \mathbb{Z}$ are integers and at least one of them is non-zero, then their greatest common divisor (often abbreviated by gcd) is the largest positive integer which divides both $n$ and $m$. The greatest common divisor of $n$ and $m$ is denoted by $(n, m)$ or gcd($n, m$). The integers $n$ and $m$ are called co-prime if $(n, m) = 1$ holds.

**Definition 1.1.4.** If $n, m \in \mathbb{Z} \setminus \{0\}$ are non-zero integers, then their least common multiple (abbreviated by lcm) is the smallest positive number that is divisible by both $n$ and $m$. The least common multiple of $n$ and $m$ is denoted by $[n, m]$ or lcm($n, m$).

Note that if $n$ is an integer, then the divisors and multiples of $n$ and $-n$ are the same, hence we have $(n, m) = (|n|, |m|)$ and $[n, m] = [|n|, |m|]$. Also, for any positive integer $n$ we have $(n, 0) = n$. Hence for the rest of this section we restrict ourselves to the case when $n$ and $m$ are positive integers.

Now we are going to use the prime factorization of the numbers to compute their greatest common divisor and least common multiple (we will address the effectiveness of this method later).
Proposition 1.1.4. If \( p_1, \ldots, p_k \) are pairwise different positive primes, \( n = p_1^{\alpha_1} \cdots p_k^{\alpha_k} \) and \( m = p_1^{\beta_1} \cdots p_k^{\beta_k} \), where \( \alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_k \) are non-negative integers, then

\[
(n, m) = p_1^{\min\{\alpha_1, \beta_1\}} \cdots p_k^{\min\{\alpha_k, \beta_k\}},
\]

\[
[n, m] = p_1^{\max\{\alpha_1, \beta_1\}} \cdots p_k^{\max\{\alpha_k, \beta_k\}}.
\]

Before the proof we show an example. If \( n = 600 \) and \( m = 84 \), then their canonical representations are \( 600 = 2^3 \cdot 3^1 \cdot 5^2 \) and \( 84 = 2^2 \cdot 3^1 \cdot 7^1 \). Observe that there are different primes in these factorizations, hence to apply the previous proposition we have to write them differently, using all the primes that occur in the two products. That is, \( 600 = 2^3 \cdot 3^1 \cdot 5^2 \cdot 7^0 \) and \( 84 = 2^2 \cdot 3^1 \cdot 5^0 \cdot 7^1 \). Of course, unlike in the case of the canonical representation here it is necessary to allow the exponent zero. Now the formulae above are applicable: \( 600, 84 \) = \( 2^2 \cdot 3^1 \cdot 5^0 \cdot 7^0 = 12 \) and \( 600, 84 \) = \( 2^3 \cdot 3^1 \cdot 5^2 \cdot 7^1 = 4200 \).

Proof of Proposition 1.1.4. By Proposition 1.1.2, for any positive integer \( d \) the properties \( d \mid n \) and \( d \mid m \) hold simultaneously if and only if \( d = p_1^{\gamma_1} \cdots p_k^{\gamma_k} \), where \( 0 \leq \gamma_i \leq \alpha_i \) and \( 0 \leq \gamma_i \leq \beta_i \), i.e. \( 0 \leq \gamma_i \leq \min\{\alpha_i, \beta_i\} \) for every \( i \). This holds also for \( (n, m) \), and since \((n, m)\) is the greatest among the positive divisors, we must have equality in the previous inequalities, otherwise we could get a greater divisor by increasing an exponent. The proof of the other claim is similar and left to the reader.

Note that this proof gives more. Namely, the greatest common divisor of two numbers has the following special property:

Corollary 1.1.5. Let \( n, m \in \mathbb{N}^+ \) be positive integers. Then the common divisors of \( n \) and \( m \) are the divisors of their greatest common divisor, i.e. \( d \mid n \) and \( d \mid m \) holds simultaneously if and only if \( d \mid (n, m) \).

Proof. The greatest common divisor of \( n \) and \( m \) divides both numbers, i.e. \( n = (n, m) \cdot c_1 \) and \( m = c_2 \cdot (n, m) \) for some \( c_1, c_2 \) integers. If \( d \mid (n, m) \), then \( (n, m) = de \), so \( n = d(c_1) \) and \( m = d(c_2) \), that is, both \( d \mid n \) and \( d \mid m \) hold.

On the other hand, if both \( d \mid n \) and \( d \mid m \) hold, then the formula for \( (n, m) \) in the previous statement and the first sentence of the previous proof together with Proposition 1.1.2 give that \( d \mid (n, m) \).

Exercise 1.1.1. Assume that \( n, m \in \mathbb{N}^+ \) are positive integers and let \( \langle n, m \rangle \) denote the least positive integer for which both \( n \mid m \cdot \langle n, m \rangle \) and \( m \mid n \cdot \langle n, m \rangle \) hold. Give a formula for \( \langle n, m \rangle \) that is similar to the ones in Proposition 1.1.4.

We close this section by a basic theorem about the number of primes:

Theorem 1.1.6. The number of primes is infinite.

Proof. It is enough to prove that there are infinitely many positive primes. So in the proof every prime is assumed to be positive.

Assume on the contrary that the number of primes is finite, say \( k \). Let \( p_1, \ldots, p_k \) be the list of all primes. Then \( N = p_1 \cdots p_k + 1 \) is bigger than 1, hence it has a prime factorization. Since \( N \) is not divisible by any of the primes \( p_1, \ldots, p_k \), every prime in the factorization of \( N \) must be different from them, and this is a contradiction.
1.2 Congruences

The set of integers is closed under addition, subtraction and multiplication, but this is not the case with the fourth basic operation. The result of a division is not always an integer (and we cannot divide by 0 at all). What we can do is division with remainders. Namely, for every \( a, b \in \mathbb{Z}, b \neq 0 \) there exist integers \( q, r \) such that \( a = qb + r \) where \( 0 \leq r \leq |b| - 1 \). This is obvious since if we regard the integers below \( a \) (and also \( a \) itself), then we can find one within the distance \(|b| - 1\) which is divisible by \( b \). Since among \(|b|\) consecutive numbers there is exactly one which is divisible by \( b \), we get that the number \( r \) (and then also \( q \)) is determined uniquely. The number \( r \) is called the remainder (and \( q \) is the quotient). For example, if we divide \(-30\) by \( 9 \), then the remainder is \( 6 \) (since \(-30 = (-4) \cdot 9 + 6\)). This makes it possible to define the congruence relation:

**Definition 1.2.1.** Let \( a, b, m \in \mathbb{Z} \) be integers and \( m \neq 0 \). We say that \( a \) and \( b \) are congruent (or \( a \) is congruent to \( b \)) modulo \( m \) if they give the same remainder when we divide them by \( m \). This is denoted by \( a \equiv b \pmod{m} \) or \( a \equiv b \pmod{m} \). The number \( m \) is called the modulus of the congruence.

For example, \( 17 \equiv 52 \pmod{7} \) (because both of them gives the remainder \( 3 \)) and \( 33 \equiv -30 \pmod{9} \) (here the remainder is \( 6 \)). The notation of the congruence resembles the notation of equality, and this is not a coincidence. It expresses that we consider \( a \) and \( b \) the same when we count with the remainders. The following equivalent definition of the congruence is often useful:

**Proposition 1.2.1.** If \( a, b, m \in \mathbb{Z}, m \neq 0 \), then \( a \equiv b \pmod{m} \) if and only if \( m \mid a - b \).

**Proof.** Let us denote the remainder of \( a \) modulo \( m \) by \( r_a \). Similarly, let \( r_b \) be the remainder of \( b \). Then \( a = qa + r_a \) and \( b = qb + r_b \) for some \( q, q_b \) integers. If \( r_a = r_b \), then \( m \mid a - b = (qa - qb)m \). On the other hand, if \( r_a \neq r_b \), then \( a - b = (qa - qb)m + r_a - r_b \), where \( 0 \neq |r_a - r_b| < m \), and hence \( m \mid a - b \) (because the distance between two multiples of \( m \) is at least \( m \)).

The following proposition shows why using the congruence relation makes the computations often easier:

**Proposition 1.2.2.** Assume that \( a \equiv b \pmod{m} \) and \( c \equiv d \pmod{m} \) hold for some integers \( a, b, c, d, m \in \mathbb{Z}, m \neq 0 \) and let \( k \in \mathbb{Z} \) be an arbitrary integer. Then the following hold:

(i) \( a + c \equiv b + d \pmod{m} \),

(ii) \( a - c \equiv b - d \pmod{m} \),

(iii) \( ac \equiv bd \pmod{m} \),

(iv) \( a^k \equiv b^k \pmod{m} \).

**Proof.** By the previous proposition our assumption is equivalent to the conditions \( m \mid a - b \) and \( m \mid c - d \). From these we get that \( m \mid (a - b) + (c - d) = (a + c) - (b + d) \), which means that \( a + c \equiv b + d \pmod{m} \) (again, by the previous proposition). Similarly, we have \( m \mid (a - b) - (c - d) = (a - c) - (b - d) \), hence \( a - c \equiv b - d \pmod{m} \) hold. This proves (i) and (ii).
To show (iii) we note that if \( m \mid a - b \), then \( m \mid c(a - b) = ac - bc \) follows. The same way, we get from \( m \mid c - d \) that \( m \mid b(c - d) = bc - bd \). But the sum of numbers divisible by \( m \) is again divisible by \( m \), hence we have \( m \mid ac - bc + bc - bd = ac - bd \), and this is equivalent to \( ac \equiv bd \pmod{m} \).

Finally, (iv) follows from (iii): if we set \( c = a \) and \( d = b \), then (iii) gives \( a^2 \equiv b^2 \pmod{m} \). Now we apply (iii) to the latter congruence and to \( a \equiv b \pmod{m} \), and this way we obtain \( a^3 \equiv b^3 \pmod{m} \). Continuing this way we get \( a^k \equiv b^k \pmod{m} \) after \( k - 1 \) steps. \( \square \)

We often use the previous statements in the special case when \( c = d \). As obviously \( c \equiv c \pmod{m} \), we get that if \( a \equiv b \pmod{m} \), then also \( a \pm c \equiv b \pm c \pmod{m} \) and \( ac \equiv bc \pmod{m} \). But the analogous claim does not hold for the division. Of course to be able to divide a congruence by a number \( c \) we must have integers on both sides which are divisible by \( c \). But one has to be careful even in that case, for example \( 40 \equiv 64 \pmod{12} \), but dividing by \( 8 \) we get \( 5 \equiv 8 \pmod{12} \), which is false. The right form of the division rule is the following:

**Theorem 1.2.3.** Let \( a, b, c, m \in \mathbb{Z} \) be integers, \( m \neq 0 \) and \( d = (c, m) \) (the greatest common divisor of \( m \) and \( c \)). Then \( ac \equiv bc \pmod{m} \) if and only if \( a \equiv b \pmod{m/\gcd(m, c)} \).

**Proof.** If \( c' = \frac{c}{\gcd(c, m)} \) and \( m' = \frac{m}{\gcd(m, c)} \), then \( c' \) and \( m' \) are integers since \( d \) is a common divisor of \( c \) and \( m \). Moreover \( (c', m') = 1 \), otherwise the number \( d \cdot (c', m') \) would be a common divisor of \( m \) and \( c \) which is bigger than \( d \), and this contradicts the definition of the greatest common divisor.

Now \( ac \equiv bc \pmod{m} \) if and only if \( m \mid ac - bc = c(a - b) \) by Proposition 1.2.1. That is, we have \( c(a - b) = mk \) for some integer \( k \). Dividing both sides by \( d \) we get the equivalent equation \( c'(a - b) = m'k \). If \( m' \mid a - b \), then at least one prime divisor of \( m' \) must divide \( c' \) by the fundamental theorem, but since \( m' \) and \( c' \) are co-prime (which means that their greatest common divisor is 1), this is impossible. It follows that \( m' \nmid a - b \), i.e. \( a \equiv b \pmod{m'} \).

On the other hand, if \( a \equiv b \pmod{m'} \), then \( m' \mid a - b \) and hence \( m' \mid c'(a - b) \). This means that \( c'(a - b) = m'k \) for some integer \( k \), and we have already seen that this is equivalent to \( ac \equiv bc \pmod{m} \). \( \square \)

**Corollary 1.2.4.** Assume that \( a, b, c, m \in \mathbb{Z} \), \( m \neq 0 \) and \( (m, c) = 1 \) (that is, \( c \) and \( m \) are co-prime). Then \( ac \equiv bc \pmod{m} \) if and only if \( a \equiv b \pmod{m} \).

**Exercise 1.2.1.** What is the remainder when we divide

\[ a) \ 100^{100} \text{ by } 11; \quad b) \ 654^{321} \text{ by } 655; \quad c) \ 111^{41} \text{ by } 35? \]

**Solution.** We use the properties of the congruence relation that are given in Proposition 1.2.2.

\( a) \) Since \( 11 \mid 99 \) we have \( 100 \equiv 1 \pmod{11} \). Raising both sides to the power 100 and using property (iv) we get that \( 100^{100} \equiv 1^{100} = 1 \pmod{11} \) (and hence the remainder is 1).

\( b) \) Observe that \( 654 \equiv -1 \pmod{655} \), hence \( 654^{321} \equiv (-1)^{321} = -1 \pmod{655} \) by property (iv). The remainder of \( 654^{321} \) is then 654.

\( c) \) First note, that \( 111 \equiv 6 \pmod{35} \), so \( 111^{41} \equiv 6^{41} \pmod{35} \). At this point the result is not clear, but notice that \( 6^2 \equiv 1 \pmod{35} \). From this we obtain that \( 6^{40} = (6^2)^{20} \equiv 1^{20} = 1 \pmod{35} \), and then \( 6^{41} = 6^{40} \cdot 6 \equiv 1 \cdot 6 \pmod{35} \), i.e. the remainder is 6. \( \square \)
1.3 The Euler-Fermat Theorem

The aim of this section is to show that for appropriate values of \( a, m \) and \( k \) the congruence \( a^k \equiv 1 \pmod{m} \) holds. We make use of this later in the RSA algorithm. One must be careful though, since if \( (a, m) = d > 1 \), then of course \( d \nmid a^k - 1 \) for any integer \( k > 0 \) (because \( d \mid a^k \)). On the other hand, in the case when \( a \) and \( m \) are co-prime we can find an appropriate \( k \) which depends only on \( m \) and not on \( a \). To be able to formulate the precise statement we will need a tool which we introduce below.

1.3.1 Euler’s Phi Function

Two numbers that are congruent to each other behave similarly from many points of view. The following statement says that even their greatest common divisor with \( m \) agrees:

**Proposition 1.3.1.** Assume that \( a, b, m \in \mathbb{Z} \) and \( m \neq 0 \). If \( a \equiv b \pmod{m} \) holds, then \( (a, m) = (b, m) \).

**Proof.** Assume that \( a \equiv b \pmod{m} \), i.e. \( m \mid a - b \). This means that \( b = a + km \) for some \( k \in \mathbb{Z} \). If \( d = (a, m) \), then since \( d \mid a \) and \( d \mid km \), we get that \( d \mid a + km = b \). In other words, \( d \) is a common divisor of \( b \) and \( m \). It follows that \( d = (a, m) \leq (b, m) \), because the latter number is the greatest among the positive common divisors. Since the role of \( a \) and \( b \) is symmetric, we have \( (b, m) \leq (a, m) \) as well, and the claim follows.

**Corollary 1.3.2.** If \( a \equiv b \pmod{m} \), then \( (a, m) = 1 \) if and only if \( (b, m) = 1 \).

**Definition 1.3.1.** If \( n \geq 1 \), then we denote by \( \varphi(n) \) the number of those integers in the interval \([1, n]\) which are co-prime to \( n \), that is,

\[
\varphi(n) = |\{k \in \mathbb{N} : 1 \leq k \leq n, (k, n) = 1\}|.
\]

The function \( \varphi \) is called Euler’s phi function.

The congruence relation modulo \( n \) divides the set of integers into disjoint classes, these are called residue classes modulo \( n \). Two integers belong to the same class if and only if they are congruent. The system of residue classes modulo \( n \) is complete in the sense that every integer belongs to a class. Since every class contains exactly one element in the interval \([1, n]\), we get by the previous Corollary that \( \varphi(n) \) is the number of the residue classes modulo \( n \) which contain numbers that are co-prime to \( n \).

We determine the value of \( \varphi(10) \). Among the numbers 1, 2, \ldots, 10 the even numbers and the multiples of 5 have a common divisor with 10 greater than 1, but the remaining numbers are co-prime to 10. These are 1, 3, 7 and 9, hence \( \varphi(10) = 4 \). If \( n = p \) is prime, then all the numbers 1, \ldots, \( p - 1 \) are co-prime to \( p \), so \( \varphi(p) = p - 1 \). It is also easy to determine the value of \( \varphi \) for prime powers:

**Lemma 1.3.3.** If \( p \) is a prime and \( \alpha \geq 1 \) is a positive integer, then \( \varphi(p^\alpha) = p^\alpha - p^{\alpha-1} \).

**Proof.** The numbers among 1, \ldots, \( p^\alpha \) that are co-prime to \( p^\alpha \) are the ones which are not divisible by \( p \). So we exclude the numbers \( kp \), where \( k \) is a positive integer and \( kp \leq p^\alpha \), i.e. \( k \leq p^{\alpha-1} \). This proves the claim.

The computation of \( \varphi \) based on the definition becomes tiresome for a general composite number. However, we can use the following lemma and the canonical form of the number to give a formula for \( \varphi(n) \).
Lemma 1.3.4. If \( a \) and \( b \) are co-prime positive integers, then \( \varphi(ab) = \varphi(a)\varphi(b) \).

Remark. A function defined on the set of positive integers is called multiplicative if it has the property described in the lemma. As co-prime numbers have no common primes in their canonical representations, it follows easily from Corollary 1.1.3 that the function \( d(n) \) defined in the first section is multiplicative. To learn more about multiplicative arithmetic functions see e.g. [5].

We postpone the proof of the lemma and first apply it to give a formula for \( \varphi(n) \):

**Theorem 1.3.5.** If \( n > 1 \) is a positive integer with canonical representation \( n = p_1^{\alpha_1} \cdots p_k^{\alpha_k} \), then

\[
\varphi(n) = (p_1^{\alpha_1} - p_1^{\alpha_1-1}) \cdots (p_k^{\alpha_k} - p_k^{\alpha_k-1}).
\]

**Proof.** First note that form the previous lemma if follows by induction that if \( a_1, \ldots, a_k \) are pairwise co-prime numbers (i.e. \( (a_i, a_j) = 1 \) for every \( 1 \leq i, j \leq k, i \neq j \)), then \( \varphi(a_1 \cdots a_k) = \varphi(a_1) \cdots \varphi(a_k) \). Indeed, the lemma gives this for \( k = 2 \). Assume that \( k > 2 \) and the statement is true for \( k - 1 \). If \( a_1, \ldots, a_k \) are pairwise co-prime numbers, then \( (a_1 \cdots a_{k-1}, a_k) = 1 \), because if a prime divides both \( a_k \) and the product, then this prime occurs in the canonical representation of some \( a_i \) where \( 1 \leq i \leq k - 1 \). But this is impossible since \( (a_i, a_k) = 1 \) holds. Then \( \varphi(a_1 \cdots a_{k-1}a_k) = \varphi(a_1 \cdots a_{k-1})\varphi(a_k) \) by the previous lemma, and using the assumption for \( k - 1 \) numbers we get the claim.

Now we apply this for the numbers \( a_i = p_i^{\alpha_i} \) which are pairwise co-prime, and hence \( \varphi(n) = \varphi(p_1^{\alpha_1}) \cdots \varphi(p_k^{\alpha_k}) \) holds. Finally, applying Lemma 1.3.3 we get the statement of the theorem.

**Proof of Lemma 1.3.4.** First note that for the positive integers \( x, a, b \in \mathbb{N}^+ \) \( (x, ab) = 1 \) holds if and only if both \( (x, a) = 1 \) and \( (x, b) = 1 \) hold. Indeed, we get the prime factorization of \( ab \) by multiplying the factorizations of \( a \) and \( b \), so if a prime divides \( x \) and \( ab \), then it divides \( a \) or \( b \). That is, if \( (x, ab) > 1 \), then \( (x, a) > 1 \) or \( (x, b) > 1 \) must hold. On the other hand, if \( x \) and \( a \) or \( x \) and \( b \) have a common prime divisor, then it divides \( ab \) as well. It follows that \( \varphi(ab) \) is the number of those integers between \( 1 \) and \( ab \) that are co-prime to both \( a \) and \( b \).

We write the numbers \( 1, 2, \ldots, ab \) in a table so that the intersection of the \( i \)th row and \( j \)th column contains the number \( m_{ij} = (i-1)b+j \), where \( 1 \leq i \leq a, 1 \leq j \leq b \). In this table we will search for numbers that are co-prime to both \( a \) and \( b \). The following table shows the case \( a = 3 \) and \( b = 8 \).

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \\
\end{array}
\]

First note that \( m_{i,j} = (i-1)b+j \equiv j \pmod{b} \) for every \( i, j \), hence by Proposition 1.3.1 we have that \( (m_{i,j}, b) = (j, b) \). In particular, \( (m_{ij}, b) = 1 \) holds if and only if \( (j, b) = 1 \). This means that the numbers in the table that are co-prime to \( b \) are those which lie in the \( j \)th column for some \( j \) co-prime to \( b \). This narrows down the scope of our search to \( \varphi(b) \) columns.

Now we are going to count the numbers in the \( j \)th column that are co-prime to \( a \). In fact, we show that any two different numbers in the \( j \)th column are not congruent to each other modulo \( a \), and since there are \( a \) rows in our table, it follows that the numbers in the \( j \)th column form a complete residue system modulo \( a \) and hence there are \( \varphi(a) \) numbers among them that are co-prime to \( a \). Putting this and the result of the previous paragraph together we get the claim.
So assume that \( m_{ij} = (i-1)b+j \equiv m_{kj} = (k-1)b+j \pmod{a} \) for some \( 1 \leq i, k \leq a \). By property (ii) in Proposition 1.2.2 we can subtract \( j \) from both sides, and since \( (a,b) = 1 \), it follows from Corollary 1.2.4 that we can divide the congruence by \( b \). We get that \( i-1 \equiv k-1 \pmod{a} \), that is, \( i \equiv k \pmod{a} \). But since \( 1 \leq i, k \leq a \), we have \( i = k \), i.e. \( m_{ij} = m_{kj} \). This completes the proof of the lemma.

1.3.2 Residue Systems

Every residue class modulo \( m \) can be represented by any one of its members. That is, any member of a class identifies it. We often represent a class by the smallest non-negative integer which belongs to the class, i.e. every class modulo \( m \) can be represented by an integer \( 0 \leq c \leq m - 1 \). Moreover, these non-negative integers represent every class exactly once. The subsets of integers with this property are called complete residue systems. We introduce another important residue system. Recall that if a residue class modulo \( m \) contains a number which is co-prime to \( m \), then every member of that class has the same property by Corollary 1.3.2. We have seen that the number of these classes is \( \varphi(m) \). If each of these classes is represented exactly once, then we call the system reduced.

**Definition 1.3.2.** The system \( R = \{c_1, \ldots, c_k\} \) of integers is called a reduced residue system modulo \( m \) if the following hold:

(i) \( (c_i, m) = 1 \) hold for every \( 1 \leq i \leq k \),

(ii) \( c_i \not\equiv c_j \pmod{m} \) for any \( 1 \leq i, j \leq k, i \neq j \),

(iii) \( k = \varphi(m) \).

The systems \( \{1, 3, 7, 9\}, \{21, 43, 67, 89\} \) and \( \{-1, -3, -3\} \) are reduced modulo 10.

**Proposition 1.3.6.** Assume that \( R = \{c_1, \ldots, c_k\} \) is a reduced residue system modulo \( m \) and \( a \in \mathbb{Z} \) is an arbitrary integer with \( (a, m) = 1 \). Then \( R' = \{ac_1, \ldots, ac_k\} \) is also a reduced residue system modulo \( m \).

**Proof.** We are going to show that the properties (i), (ii) and (iii) in the previous definition hold for \( R' \). To see (i) we set \( d_i = (ac_i, m) \). If \( p \mid d_i \) is a prime, then it occurs in the prime factorization of both \( m \) and \( ac_i \). As we get the prime factorization of \( ac_i \) by multiplying the factorization of \( a \) and \( c_i \), \( p \) must divide at least one of them (and also \( m \)). But this contradicts the assumption \( (a, m) = (c_i, m) = 1 \), and it follows that \( (ac_i, m) = 1 \).

Assume now that \( ac_i \equiv ac_j \pmod{m} \) for some \( 1 \leq i, j \leq k \). Then by Corollary 1.2.4 this is equivalent to \( c_i \equiv c_j \pmod{m} \), because \( (a, m) = 1 \) holds. Since \( R \) is a reduced residue system, this can hold if and only if \( i = j \), so property (ii) is proved.

Finally, the number of the elements of the systems \( R' \) and \( R \) is the same, hence (iii) follows for \( R' \). \( \square \)

1.3.3 The Euler-Fermat Theorem

After this preparation we are in the position to state and prove the so called Euler-Fermat theorem:

**Theorem 1.3.7** (Euler-Fermat theorem). If \( a, m \in \mathbb{Z} \) are integers, \( m \neq 0 \) and \( (a, m) = 1 \), then \( a^{\varphi(m)} \equiv 1 \pmod{m} \) holds, where \( \varphi \) is Euler’s phi function.
Proof. Let \( R = \{c_1, \ldots, c_k\} \) be an arbitrary reduced residue system modulo \( m \). Since \((a, m) = 1\), we have by Proposition \[1.3.6\] that \( R' = \{ac_1, \ldots, ac_k\} \) is also a reduced residue system modulo \( m \). For every remainder \( 0 \leq r \leq m - 1 \) with \((r, m) = 1\) there is exactly one number in both \( R \) and \( R' \) which is congruent to \( r \). Hence we can pair the numbers in \( R \) and \( R' \) so that the pairs are congruent to each other. Then by property (iii) in Proposition \[1.2.2\] we can multiply the numbers in \( R \) and \( R' \) and this way we still get numbers that are congruent to each other:

\[
c_1 \ldots c_k \equiv (ac_1) \ldots (ac_k) = a^{\varphi(m)}c_1 \ldots c_k \pmod{m},
\]

where we used that \( k = \varphi(m) \). Since \((c_i, m) = 1\), it follows from Corollary \[1.2.4\] that we can divide the previous congruence by \( c_i \) for every \( 1 \leq i \leq k \). After doing this for every \( i \) we get the statement of the theorem.

\[\square\]

Corollary 1.3.8 (Fermat’s little theorem). If \( p \) is a positive prime and \( a \in \mathbb{Z} \) is an arbitrary integer, then \( a^p \equiv a \pmod{p} \).

Proof. If \( p \divides a \), then \( p \divides a^p \) also holds, hence \( a^p \equiv 0 \equiv a \pmod{p} \). If \( p \nmid a \), then \((a, p) = 1\), because \( p \) is a prime. Then by the previous theorem we have \( a^{\varphi(p)} = a^{p-1} \equiv 1 \pmod{p} \). Multiplying both sides by \( a \) we get the statement.

\[\square\]

Exercise 1.3.1. What is the remainder when we divide a) \( 11^{111} \) by \( 63 \) b) \( 51^{4132} \) by \( 140 \)?

Solution. a) Since \((11,63) = 1\), we can apply the Euler-Fermat theorem, which gives that \( 11^{\varphi(63)} = 11^{36} \equiv 1 \pmod{63} \) (as \( \varphi(63) = (7^1 - 7^0)(3^2 - 3) = 6 \cdot 6 = 36 \)). Now we apply property (iv) of Proposition \[1.2.2\] for \( k = 3 \). That is, we raise both sides to the 3rd power to get that \( (11^{36})^3 = 11^{108} \equiv 1^3 \equiv 1 \pmod{36} \). That is, \( 11^{111} = 11^{108} \cdot 11^3 \equiv 1 \cdot 11^3 \pmod{63} \), so it remains to determine the remainder of \( 11^3 \). As \( 11^2 = 121 \equiv -5 \pmod{63} \), we obtain that \( 11^3 = 11^2 \cdot 11 \equiv (-5) \cdot 11 = -55 \equiv 8 \pmod{63} \), and hence the remainder is 8.

b) We will apply the Euler-Fermat theorem for the numbers \( a = 51 \) and \( m = 140 \). This can be done since \( 51 = 3 \cdot 17 \) and \( 140 = 2^2 \cdot 5 \cdot 7 \), and hence \((51,140) = 1\). We also have that \( \varphi(140) = (2^2 - 2)(5 - 1)(7 - 1) = 2 \cdot 4 \cdot 6 = 48 \), so \( 51^{48} \equiv 1 \pmod{140} \) holds by the Euler-Fermat theorem. Maybe it is not clear at first sight how this can be used in this situation. But as before, we have \( 51^{48k} \equiv 1^k \equiv 1 \pmod{140} \) for every \( k \geq 1 \). Although the exponent is not of the form \( 48k \) we still can divide it by 48 with a remainder. That is, we are looking for the smallest non-negative integer \( r \) such that \( 41^{32} \equiv r \pmod{48} \). Luckily, \((41, 48) = 1\) holds, hence we can apply the Euler-Fermat theorem again. As \( \varphi(48) = (2^4 - 2^2)(3 - 1) = 16 \), we have \( 41^{16} \equiv 1 \pmod{48} \) and hence \( 41^{16 \cdot 2} = 41^{32} \equiv 1 \pmod{48} \). This can be written as \( 41^{32} = 48k + 1 \) for some integer \( k \), and then \( 51^{4132} = 51^{48k+1} = 51^{48k} \cdot 51 \equiv 51 \pmod{140} \), i.e. the remainder is 51.

\[\square\]

1.4 Linear Congruences

In this section we address the following question: if \( a, b, m \in \mathbb{Z} \), \( m \neq 0 \) are given, then what are the numbers for which the congruence \( ax \equiv b \pmod{m} \) holds? This problem is called a linear congruence, because we have information about the first power of the unknown number \( x \).

First we note, that if a linear congruence has a solution \( x_0 \), then \( ax_0 \equiv ax_1 \pmod{m} \) holds for every \( x_1 \) which is congruent to \( x_0 \) modulo \( m \). In other words, if \( x_0 \) is a solution, then every number in its residue class modulo \( m \) is also a solution. Hence the set of the solutions is a union of residue classes, and we will give the solutions by giving only one representative.
from each class which contains solutions, that is, we will write \( x \equiv x_0 \pmod{m} \) (and give this way the whole class of \( x_0 \)).

For example, let us examine the congruence \( 3x \equiv 2 \pmod{5} \). Multiplying by 2 we get \( 6x \equiv 4 \pmod{5} \). But \( 6x \equiv x \pmod{5} \), hence the only option for the solution is the class \( x \equiv 4 \pmod{5} \). This is indeed a solution since \( 3 \cdot 4 \equiv 2 \equiv 2 \pmod{5} \).

Let us try to solve the congruence \( 10x \equiv 5 \pmod{30} \). If we look at this congruence, we may observe that a number of the form \( 10x \) has a zero in the end when we write it in the decimal system. On the other hand, if a number gives the remainder 5 when we divide it by 30, then it must end with the digit 5. This means that this congruence has no solutions.

In this section we determine the conditions that are sufficient and necessary for a linear congruence or a system of linear congruences to have a solution. We will also determine the number of the solutions. We give a method in the next section, which determines the solutions "efficiently". The word "efficiently" will also get a more or less precise meaning in the next section.

### 1.4.1 Existence of solutions

In the last example above we did not have a solution for a linear congruence, and the true reason for this is that the modulus and the coefficient of \( x \) had a common divisor which did not divide the right hand side. We formalize this in the following

**Theorem 1.4.1.** The linear congruence \( ax \equiv b \pmod{m} \) is solvable if and only if \((a,m) \mid b\). If this condition holds, then \((a,m)\) is the number of the different residue classes which contain all the solutions.

We usually say briefly that the number of solutions modulo \( m \) is \((a,m)\).

**Proof.** First we show that if the congruence is solvable, then \( d := (a,m) \mid b \). Let \( x_0 \) be a solution of the congruence. Then \( m \mid ax_0 - b \) holds, and as \( d \mid m \), we have that \( d \mid ax_0 - b \). But \( d \mid a \mid ax_0 \) holds as well, hence \( d \mid ax_0 - (ax_0 - b) = b \) follows.

Next we show that if \((a,m) = 1\), then the congruence is solvable. We set \( x_0 = a^{\varphi(m)-1}b \), then by the Euler-Fermat theorem we get that \( ax_0 = a^{\varphi(m)}b \equiv b \pmod{m} \), i.e. \( x_0 \) is indeed a solution.

Now assume that \( d = (a,m) \mid b \) and set \( a' = a/d, b' = b/d \) and \( m' = m/d \). Then \( a', b' \) and \( m' \) are integers, and \((a',m') = 1\) (otherwise \((a',m') \cdot d\) would be a common divisor of \( a \) and \( m \) which is greater than \( d \)). By Theorem [1.2.3] the congruence \( ax \equiv b \pmod{m} \) is equivalent to \( a'x \equiv b' \pmod{m'} \), and by the previous paragraph this latter congruence has a solution, and hence so does the original congruence.

Now we turn to the number of solutions. Assume that \( x_1 \) is an arbitrary solution of the congruence. Now \( x_2 \) is another one if and only if \( ax_1 \equiv b \equiv ax_2 \pmod{m} \). By Theorem [1.2.3] this is equivalent to \( x_1 \equiv x_2 \pmod{m'} \). So every solution is of the form \( x_1 + km' \) for some \( k \in \mathbb{Z} \), and any of these numbers is a solution. Now \( x_1 + k_1m' \equiv x_1 + k_2m' \pmod{m} \) holds if and only if \( k_1 \equiv k_2 \pmod{m/m'} \), and as \( m/m' = d \), this means that the solutions of the original congruence come from \( d \) distinct residue classes modulo \( m \).

Note that the last paragraph of the proof gives the set of all solutions once we have found one single solution. Namely, if \( x_1 \) is a solution, then \( x_1 + km' \) \((k = 0, 1, \ldots, (a,m) - 1)\) are the representatives of all distinct residue classes modulo \( m \) which contain the solutions, each of them is represented only once.
One may observe that the second and third paragraph of the proof also gives a method to determine a first solution, however this is not useful in practice, because it is often hopeless to make the calculations fast. But the first part of this method is important from the practical point of view. Namely, given a congruence \( ax \equiv b \pmod{m} \) with \( d = (a, m) \mid b \), we only have to solve the equivalent congruence \( a'x \equiv b' \pmod{m'} \), where \( a' = a/d, \ b' = b/d, \ m' = m/d \) and \( (a', m') = 1 \). The solution of this congruence will be a solution of the original one as well.

**Exercise 1.4.1.** Solve the following congruences:

\[
\begin{align*}
\text{a)} & \quad 68x \equiv 12 \pmod{98}, \\
\text{b)} & \quad 59x \equiv 4 \pmod{222}.
\end{align*}
\]

**Solution.** a) Both sides of the congruence are divisible by 4, and \( (4, 98) = 2 \), so this congruence is equivalent to
\[
17x \equiv 3 \pmod{49}
\]
by Theorem 1.2.3. That is, we divided both sides by 4, but we had to divide the modulus by the greatest common divisor of 4 and 98 as well. Now we multiply both sides by 3 to obtain
\[
51x \equiv 9 \pmod{49}.
\]
Observe that \( 51 \equiv 2 \pmod{49} \) and hence \( 51x \equiv 2x \pmod{49} \) holds. Also, \( 9 \equiv 58 \pmod{49} \), so from the previous congruence we infer
\[
2x \equiv 58 \pmod{49},
\]
and dividing both sides by 2 we have
\[
x \equiv 29 \pmod{49}.
\]
There are two residue classes modulo 98 which contain numbers that are congruent to 29 modulo 49, namely the class of 29 and the class of \( 29 + 49 = 78 \). One checks easily that these numbers satisfy the the original congruence (and then so does every number in their classes). So the solutions are \( x \equiv 29 \pmod{98} \) and \( x \equiv 78 \pmod{98} \).

One may observe that all steps that we made gave an equivalent form of the former congruence (and not just a consequence of the former ones). We emphasized this at the first step, but then we multiplied and divided by a number which was co-prime to the modulus, so the result was equivalent to the former congruence. Hence it is fact superfluous to check our solutions, all of them must satisfy the original congruence. Also note that Theorem 1.4.1 gives us the number of solutions modulo 98 at the beginning, there are \( (98, 68) = 2 \) of them. We could also refer to this, and then if we get only two possibilities for the solutions, then both of them must be correct.

b) First we multiply the congruence by 4 to get
\[
236x \equiv 16 \pmod{222},
\]
and since \( 236 \equiv 14 \pmod{222} \), we can write this as
\[
14x \equiv 16 \pmod{222}.
\]
Dividing by 2 (and using Theorem 1.2.3) we get that
\[
7x \equiv 8 \pmod{111}.
\]
Now we multiply this last congruence by 16:

\[ 112x \equiv 128 \pmod{111}, \]

and since 112 \(\equiv 1\) and 128 \(\equiv 17\) (mod 11), we conclude

\[ x \equiv 17 \pmod{111}. \]

We get two classes modulo 222, one of them is represented by 17 while the other one by 128. However, a computation shows that 59 \cdot 128 \equiv 4 \pmod{222} holds but 59 \cdot 17 \equiv 115 \pmod{222}. How is this possible? Did we make a mistake? We can find the answer at the first step. It was right in the sense that 236x \equiv 16 \pmod{236} follows from the original congruence but it is not equivalent to it. But this latter congruence is equivalent to 59x \equiv 4 \pmod{111} by Theorem 1.2.3, and the set of the solutions of this latter one is larger (because here 59x - 4 must be divisible only by 111 and not by 222). Also, Theorem 1.4.1 tells us that the number of solutions modulo 222 is (59, 222) = 1, so if we somehow obtain more possibilities, then only one of them can solve the original congruence. Note that this phenomenon occurs every time when we make a non-equivalent transformation at some of the steps. \(\square\)

### 1.4.2 Simultaneous Congruences

In many applications of number theory we are faced with problems where many congruences must hold simultaneously. In the remaining part of the section we handle this problem. We start by solving two congruences at the same time.

**Theorem 1.4.2.** The system of congruences \(x \equiv a_1 \pmod{m_1}\) and \(x \equiv a_2 \pmod{m_2}\) is solvable if and only if \((m_1, m_2) | a_1 - a_2\). If this condition holds, then solutions form a single residue class modulo \([m_1, m_2]\) (where \([m_1, m_2]\) is the least common multiple of \(m_1\) and \(m_2\)).

**Proof.** The system of congruences is solvable if and only if there is an \(x\) of the form \(m_2y + a_2 \equiv a_1 \pmod{m_1}\) such that \(m_2y \equiv a_1 - a_2 \pmod{m_1}\). This is equivalent to the solvability of the congruence \(m_2y \equiv a_1 - a_2 \pmod{m_1}\). By Theorem 1.4.1 this is solvable if and only if \((m_1, m_2) | a_1 - a_2\).

Now assume that this latter condition holds, then the congruence \(m_2y \equiv a_1 - a_2 \pmod{m_1}\) has \((m_1, m_2)\) different solutions modulo \(m_1\). If \(y_0\) is a solution, then the other solutions modulo \(m_1\) are \(y_0 + km_1/d, \) where \(d = (m_1, m_2)\) and \(0 \leq k \leq m_1 - 1\). This means that the solutions form exactly 1 residue class modulo \(m_1/d\), so they are of the form \(y_0 + km_1/d, \) where \(k \in \mathbb{Z}\). Then the solutions of the original system are of the form \(m_2(y_0 + km_1/d) + a_2 = m_2y_0 + km_1m_2/d + a_2\), that is, they form a residue class modulo \(m_1m_2/d = [m_1, m_2]\). This last equality is an easy consequence of Proposition 1.1.4. \(\square\)

**Corollary 1.4.3** (Chinese remainder theorem). Assume that \(m_1, \ldots, m_k\) are pairwise coprime integers, then the system of congruences \(x \equiv a_1 \pmod{m_1}, \ldots, x \equiv a_k \pmod{m_k}\) is solvable, and the solutions form a single residue class modulo \(m_1 \ldots m_k\).

**Proof.** We prove the statement by induction. For \(k = 2\) this is a special case of the previous theorem (because \((m_1, m_2) = 1\)). Assume that \(k > 2\) and the statement is true for \(k-1\). Then the system that consists of the first \(k-1\) congruences is equivalent to a single congruence \(x \equiv a_0 \pmod{m_1 \ldots m_{k-1}}\). Together with \(x \equiv a_k \pmod{m_k}\) this forms a system which is solvable by the previous theorem, and there is exactly 1 solution modulo \(m_1 \ldots m_k\). Here we used that \((m_1 \ldots m_{k-1}, m_k) = 1\), this follows the same way like the analogous claim in the proof of Theorem 1.3.5. \(\square\)
Exercise 1.4.2. Solve the following system of congruences:

\[ x \equiv 11 \pmod{42} \quad \text{and} \quad x \equiv 10 \pmod{199}. \]

Solution. Since \((42, 199) = 1\), we get from the previous theorem that there is one single solution modulo \(42 \cdot 199 = 8358\). By the first congruence we can write \(x = 42y + 11\) for some integer \(y\). Substituting this in the second congruence we get \(42y + 11 \equiv 10 \pmod{199}\), that is, \(42y \equiv -1 \equiv 198 \pmod{199}\). We can divide by 6 because \((6, 199) = 1\). We obtain \(7y \equiv 33 \equiv 630 \pmod{199}\). Finally, dividing this by 7 we get \(y \equiv 90 \pmod{199}\). Since we made the transformations of the congruences in every step so that the latter congruence was equivalent to the former one, we get that \(y\) must be of the form \(199z + 90\). Then \(x = 42y + 11 = 42(199z + 90) + 11 = 8358z + 3791\), i.e. \(x \equiv 3791 \pmod{8358}\) is the only solution modulo 8358. □

1.5 Number-theoretic Algorithms

1.5.1 Effectiveness of Algorithms

At the design of an algorithm one of the first questions which has to be dealt with is the expected running time of an implementation. This question is not always easy to answer, different running times are acceptable for different tasks. Sometimes every millisecond matters while in other cases the program can run for days. Of course the running time always depends on the hardware, but what is more important that in general a program runs longer for a bigger input. Here we regard the running time as a function of the size of the input.

As a first example we examine the following task which we call prime factorization: the input is an integer \(N\) and we are looking for its prime factorization. There is a simple method which gives the result: starting from 2 we try to divide \(N\) by every integer, and if we find a divisor \(p\), then we continue the procedure for the number \(N/p\) (and it is enough to start searching from the number \(p\)). Note that every divisor that we find this way will be a prime number. When \(N\) is composite, then \(N = ab\) for some \(1 < a \leq b\), and hence \(a^2 \leq ab = N\). This means that \(N\) has a divisor which is at most \(\sqrt{N}\), so if we do not find a divisor until \(\sqrt{N}\), then \(N\) is prime. This procedure clearly gives the expected result, it is easy to perform it for small numbers even without a calculator, but computers can determine the prime factorization this way for numbers with 10-20 digits. This may look satisfactory for the first sight, but in practice we often work with much larger numbers. For example if \(N\) is has 81 digits in its decimal representation, then \(N \geq 10^{80}\), so \(\sqrt{N} \geq 10^{40}\). This means that if \(N\) is a prime, then our program makes at least \(10^{40}\) divisions before it terminates. The fastest supercomputer today makes less than \(10^{18}\) elementary floating point operations in one second, which means than it would take more time for that computer to run this algorithm than the age of the universe.

Of course this does not mean that it is impossible to give an algorithm for this task which has an acceptable running time - but unfortunately no one was able to find one yet. The situation changes a lot when we only want to decide if our number is prime. That is, the output here is "prime" or "composite", and we may have no information about the divisors in the latter case. We will learn about algorithms which solve this problem for numbers with several hundred digits in a reasonable time.

Now we try to describe what an "efficient" algorithm is. There is a definition which is more or less satisfactory both for theory and applications (leaving many questions unanswered though): we consider an algorithm efficient if it has polynomial running time.
Definition 1.5.1. For an algorithm $A$ the size of its input is the number of bits that are used to store the input. The algorithm $A$ is said to be of polynomial (running) time (or shortly: polynomial) if its running time (i.e. the number of steps of $A$) can be bounded from above by a polynomial of the size of its input, that is, if there exist a positive real number $c \in \mathbb{R}^+$ and a positive integer $k \in \mathbb{N}^+$ such that for every input of size $n \geq 1$ the algorithm $A$ terminates after at most $cn^k$ steps.

One may observe that the definition above is not precise from a mathematical point of view. First of all, it is not clear what we mean by a step of an algorithm (even the notion of algorithm is undefined). Also, the memory of a computer is not a mathematical object, so the size of an input is not accurately defined. For now, we work with this somewhat intuitive definition and leave the precise work for a later course. We will give an algorithm by a pseudocode or by a C programming code. We will also assume that executing a line of our code means a series of bit operations made by the processor of the computer and the number of these operations is called the number of steps then.

Let us return to our prime factorization algorithm. What can we say about its running time? Of course there are cases when the algorithm finds the prime divisors fast, for example when $N$ is a power of 2. But a polynomial algorithm must run in polynomial time for every input. The size of the input is the number of digits of $N$ written in the numeral system of base 2. This is exactly $n = \lceil \log_2 N \rceil + 1$, and hence $2^{n-1} \leq N$ holds. If $N$ is a prime number, then our algorithm makes $\lceil \sqrt{N} \rceil$ divisions. Now

$$\lceil \sqrt{N} \rceil \geq \lceil (\sqrt{2})^{n-1} \rceil \geq (\sqrt{2})^{n-1} - 1 \geq 0.7 \cdot 1.4^n$$

if $n$ is big enough (here we used that $\sqrt{2} > 1.4$ and $(\sqrt{2})^{-1} > 0.7$). That is, the number of steps can be bounded from below by an exponential function of the input size when $N$ is a prime. Since there are infinitely many primes by Theorem 1.1.6, there are arbitrary large $N$’s for which this bound holds. As an exponential function grows faster than any polynomial function, this algorithm cannot be polynomial.

This method will be applied many times when we show that an algorithm is not polynomial. Namely, in many cases one can give a lower bound for the number of steps in terms of the input size which grows faster than any polynomial. In these notes we will always use exponential lower bounds for this purpose, but of course in general there are cases when other type of functions are needed.

In this chapter the input of an algorithm is always a set of integers so the size of the input is the sum of the number of digits of these numbers (represented in the binary system). As we have already seen, for a single number $N$ this is $\lceil \log_2 N \rceil + 1$. But since $\log_{10} N = \log_2 10 \cdot \log_2 N$, the notion of polynomial algorithm does not change if we regard the size of the input as the number of digits in the decimal representation. Moreover, this holds for a numeral system of any base, though we usually work with the binary or the decimal system. In short: an algorithm is polynomial in terms of the number of decimal digits if and only if it is polynomial in the sense of Definition 1.5.1.

As a final remark of this introductory section we mention that although from a theoretical point of view an algorithm with input size $n$ and running time $cn^k$ is polynomial and hence said to be "effective" for any $c \in \mathbb{R}^+$ and $k \in \mathbb{N}^+$, in practice the exponent is required to be small (e.g 1 or 2), otherwise the algorithm becomes too slow for the applications even for relatively small inputs.

Exercise 1.5.1. The following pseudocode gives an algorithm which computes the least common multiple of the numbers $a, b > 0$. Decide if it is polynomial or not.
### 1.5.2 Basic Arithmetic Operations

For those who have some experience with programming languages it may seem evident that the basic arithmetic operations are built into the languages or even into the hardwares. Still we elaborate on this topic, since there are algorithms behind these built-in operations as well, and what is more important, the number-theoretic algorithms of this chapter often have inputs with thousands of digits, and usually there are no built-in functions that treat such large numbers - we may have to write them.

By basic arithmetic operations we mean the addition, subtraction and multiplication of two numbers, and the division of them with a remainder. Fortunately we know effective algorithms for these tasks, namely the ones that we learned in elementary school, when we performed these operations by hand.

First we take a closer look at the addition. Assume that the number of digits of $a$ and $b$ are $k$ and $l$, respectively where $k \geq l$ (we can assume this, because in the other case we can interchange $a$ and $b$), so the size of the input is $k + l$. For simplicity we may write $b$ as a $k$-digit number (writing $k - l$ zeros at the beginning of the number). Then we can carry out the addition in one loop. We go along the digits of the numbers from right to left and do the same operations in the body of the loop, namely we add the actual digits and the remainder carried over from the previous run of the body (this remainder is set to be zero at the beginning), fill the actual digit in the result and overwrite the remainder. This sum can be stored in a table (as the summands and the remainder are bounded), so the body of the loop only makes at most $c$ bitwise operations for some constant $c$. We repeat the loop $k$ times, so the procedure stops after at most $ck \leq cn$ steps. This means that the algorithm is polynomial, moreover, it is very efficient even among the polynomial algorithms, because the exponent of the input size in the bound is 1. The algorithms with this property are said to be of linear running time.

The subtraction can be implemented similarly and we also get a linear running time.

After the previous example it is not hard to see that the multiplication can be accomplished via $k$ multiplications of a number by a one-digit number and $k - 1$ additions. This yields at most $c(k + l)^2 = cn^2$ steps, so the multiplication is also polynomial (though this algorithm is not linear but quadratic). Note that there are faster algorithms for this task. These are more complicated and in practice one saves time only for large inputs, but in many applications, especially in cryptography they are useful. Historically the first one of these was Karatsuba’s algorithm which terminates after at most $cn^{\log_2 3} \approx cn^{1.58}$ steps. There are even (asymptotically) faster methods, see e.g. the Toom-Cook algorithm or the Schönhage-Strassen algorithm. We just mention that the division can also be done in at most $cn^2$ steps by the usual algorithm that we use when calculating by hand. We leave the details to the reader.
1.5.3 Modular Exponentiation

It is easy to see that one cannot raise to powers in polynomial running time. Indeed, if the exponent is (not fixed, but) also part of the input, then even the number of the digits of the result is bigger than any polynomial of the input size, and hence the result cannot be written down in polynomial time. For example, if \(a = 2\) and \(b\) has \(k\) decimal digits, then the number of digits of \(a^b = 2^b\) is greater than

\[
\log_{10} 2^b = b \cdot \log_{10} 2 \geq 10^{k-1} \cdot \log_{10} 2 > 0.03 \cdot 10^k.
\]

That is, the number of digits of \(2^b\) is bounded from below by an exponential function of the input size.

Although we cannot determine the power itself, for the RSA algorithm it is essential to calculate the remainder of powers modulo \(m\). This task is called modular exponentiation. Here inputs are positive integers \(a, b, m \in \mathbb{N}^+\), and the output is the remainder of \(a^b\) modulo \(m\). Then at least the output size cannot be an exponential function of the input size, since the output is smaller than \(m\). As we cannot calculate \(a^b\) and then divide it by \(m\), we could try to calculate first the remainder of \(a\) and then multiply it by \(a\) and calculate the remainder of \(a^2\), and continuing this way, we could obtain the remainder of the numbers \(a^3, a^4, \ldots, a^b\). This requires the calculation of \(b\) remainders, which means that the number of steps is still exponential.

Instead of this we apply repeated squaring, that is, we only determine the remainders of the powers \(a^{2^k}\) for some \(k\)'s, and then multiply some of them. Before the precise description we show an example. We calculate the remainder of \(13^{53}\) modulo 97. In the following calculation we always square the congruence in the previous row:

\[
\begin{align*}
13^1 & \equiv 13 \quad \text{(mod 97),} \\
13^2 & \equiv 169 \equiv 72 \quad \text{(mod 97),}
\end{align*}
\]

\(13^4 = (13^2)^2 \equiv 72^2 = 5184 \equiv 43 \quad \text{(mod 97),}
\]

\(13^8 = (13^4)^2 \equiv 43^2 = 1849 \equiv 6 \quad \text{(mod 97),}
\]

\(13^{16} = (13^8)^2 \equiv 6^2 = 36 \quad \text{(mod 97),}
\]

\(13^{32} = (13^{16})^2 \equiv 36^2 = 1296 \equiv 35 \quad \text{(mod 97).}
\]

It is not necessary to continue, since \(13^{64}\) is already greater than \(13^{53}\). Now we use the binary representation of the exponent \(53 = 110101_2 = 32 + 16 + 4 + 1\), and we write the original power as a product: \(13^{53} = 13^1 \cdot 13^4 \cdot 13^{16} \cdot 13^{32}\). Multiplying the remainders of these factors we get the remainder of the original power:

\[
\begin{align*}
13^5 & = 13^1 \cdot 13^4 \equiv 13 \cdot 43 = 559 \equiv 74 \quad \text{(mod 97),}
13^{21} & = 13^5 \cdot 13^{16} \equiv 74 \cdot 36 = 2664 \equiv 45 \quad \text{(mod 97),}
13^{53} & = 13^{21} \cdot 13^{32} \equiv 45 \cdot 35 = 1575 \equiv 23 \quad \text{(mod 97).}
\end{align*}
\]

Note that we could do the last 3 steps in parallel with the determination of the remainders of the powers in (1), and then it is not necessary to store these remainders.

Now we describe the algorithm for the general positive integers \(a, b, m\). As a first step we may calculate the remainder of \(a\) modulo \(m\), so it is enough to give the algorithm for integers \(0 < a < m, 0 < b\) (if the remainder is 0, then so is the remainder of the power). We store the remainder of a product modulo \(m\), which is set to be 1 initially (the value of the
empty product). We calculate the remainders of \(a^{2^k}\) for the exponents \(k = 0, 1, \ldots, \lfloor \log_2 b \rfloor\). In parallel we multiply the stored product by the actual remainder modulo \(m\) if \(b\) has a digit 1 in the \((k + 1)\)th place (from the right) in its binary representation. We do not assume that the binary representation of \(b\) is given, it will be determined along the way. We use that the first digit of the binary form of a number \(b\) is its remainder modulo 2 and the other digits form the binary representation of \(\left\lfloor \frac{b}{2} \right\rfloor\). So the algorithm is the following:

**MODULAR EXPONENTIATION**

**Input:** the positive integers \(a\) and \(b\) with \(0 < a < m\), \(0 < b < m\)

1. \(c \leftarrow 1\)
2. **while** true **do**
   3. **if** \(b\) is odd, **then**
   4. \(c \leftarrow c \cdot a \mod m\)
   5. \(b \leftarrow \left\lfloor \frac{b}{2} \right\rfloor\)
   6. **if** \(b = 0\), **then**
   7. **print** \(\text{"}a^b \mod m = \text{"}c\); **stop**
   8. \(a \leftarrow a^2 \mod m\)
3. **end while**

Now we show that this algorithm gives us the right result. Let \(a_0\) and \(b_0\) denote the numbers \(a\) and \(b\), respectively, and we also set \(c_0 = 1\). For a positive integer \(k > 0\) let \(a_k\), \(b_k\) and \(c_k\) be the value of the variables \(a\), \(b\) and \(c\), respectively after the \(k\)th run of the body of the loop. The remainder of \(a^k\) modulo \(m\) will be denoted by \(r\). We are going to show by induction that \(a_k^{b_k}c_k \equiv r \pmod{m}\) holds for every \(k \in \mathbb{N}\). This is obviously true for \(k = 0\). Now assume that \(k > 0\) and that the congruence holds for \(k - 1\). If \(b_{k-1}\) is even, then

\[
r \equiv a_k^{b_{k-1}c_{k-1}} = \left(a_k^2\right)^{b_{k-1}/2}c_{k-1} \equiv a_k^{b_k}c_k \pmod{m}
\]

as \(b_k = b_{k-1}/2\) and \(c_k = c_{k-1}\) in this case, and \(a_k \equiv a_k^{2} \pmod{m}\) holds independently from the parity of \(b_{k-1}\). On the other hand, if \(b_{k-1}\) is odd, then \(b_k = (b_{k-1} - 1)/2\) and \(c_k = a_{k-1}c_{k-1}\), hence

\[
r \equiv a_k^{b_{k-1}c_{k-1}} = \left(a_k^2\right)^{b_{k-1}-1}/2 \cdot a_{k-1}c_{k-1} \equiv a_k^{b_k}c_k \pmod{m}.
\]

The algorithm stops in the 4th loop when \(b_k = 0\), then \(b_{k-1} = 1\), and the output is \(c_k\). But we have just proved that \(r \equiv a_k^{b_{k-1}c_{k-1}} \pmod{m}\) holds, moreover, \(a_k^{b_{k-1}c_{k-1}} = a_{k-1}c_{k-1} = c_k\), so we are done.

Finally, we prove that the algorithm is polynomial. Let \(j\), \(k\) and \(l\) denote the number of digits of \(a\), \(b\) and \(m\), respectively. Then the size of the input is \(n = j + k + l\). In the body of the loop we do at most 2 multiplications with inputs less than \(m\), 2 divisions with remainder with inputs less than \(m^2\), and we calculate the half of a number which is at most \(b\). So if we use the algorithms of the previous section (in the following we will always do so), then it follows that in the body of the loop we make at most \(c_1(l^2 + k)\) steps for some constant \(c_1\), and it runs \(\lfloor \log_2 b \rfloor + 1\) times. This latter number is the number of the digits in the binary representation of \(b\), hence it is at most \(c_2k\) for some constant \(c_2\). Then the number of steps if at most \(c_1c_2(l^2 + k)k \leq c_1c_2n^3\). Hence this is indeed a polynomial algorithm, but it is important to note that this is still too slow for the applications in cryptography. There are faster variants of this method, but we do not give any details here.
1.5.4 The Calculation of the Greatest Common Divisor

Proposition 1.1.4 gives us a formula for the greatest common divisor using the canonical representation of the numbers. As we have seen earlier, it is hard to determine the prime factorization in general, so this formula is not applicable in practice. Luckily, there is a much more effective method for this task: the so-called Euclidean algorithm. It is contained in the book "Elements" which was written by the ancient Greek mathematician Euclid ca. 300 BC.

In this task the input consists of the numbers \( a \) and \( m \), and we can assume that \( 0 < a < m \) holds (the cases when \( a = 0 \) or \( a = m \) can be handled easily). To determine \( (a, m) \) we are going to make repeated divisions with remainders: first we divide \( m \) by \( a \), then in the next step we divide \( a \) by the remainder taken from the first step, and in the \( i \)th step we divide the remainder from the \((i - 2)\)th step by the remainder from the \((i - 1)\)th step. We stop when we get 0 as a remainder, and the output will be the last non-zero remainder (or \( a \) if we stop in the first step). First we show an example: we calculate the greatest common divisor of 567 and 1238.

\[
\begin{align*}
(1) & \quad 1238 = 2 \cdot 567 + 104, \\
(2) & \quad 567 = 5 \cdot 104 + 47, \\
(3) & \quad 104 = 2 \cdot 47 + 10, \\
(4) & \quad 47 = 4 \cdot 10 + 7, \\
(5) & \quad 10 = 1 \cdot 7 + 3, \\
(6) & \quad 7 = 2 \cdot 3 + 1, \\
(7) & \quad 3 = 3 \cdot 1 + 0.
\end{align*}
\]

The result is the last non-zero remainder, that is \( (567, 1238) = 1 \). Now we write the previous steps with a general \( m \) and \( a \) (assuming that \( a \nmid m \)):

\[
\begin{align*}
(1) & \quad m = q_1a + r_1 \quad (0 < r_1 < a), \\
(2) & \quad a = q_2r_1 + r_2 \quad (0 < r_2 < r_1), \\
(3) & \quad r_1 = q_3r_2 + r_3 \quad (0 < r_3 < r_2), \\
(4) & \quad r_2 = q_4r_3 + r_4 \quad (0 < r_4 < r_3), \\
& \quad \vdots \\
(k) & \quad r_{k-2} = q_kr_{k-1} + r_k \quad (0 < r_k < r_{k-1}), \\
(k + 1) & \quad r_{k-1} = q_{k+1}r_k + 0,
\end{align*}
\]

Here the output is \( r_k \) (i.e. the last non-zero remainder).

**Proposition 1.5.1.** The output of the previous algorithm is \( (a, m) \).

**Proof.** The statement holds obviously when \( a \mid m \), so we assume that this is not the case. By the first step we have that \( m \equiv r_1 \pmod{a} \), and hence \( (m, a) = (r_1, a) \) by Proposition 1.3.1. The second step gives similarly that \( (a, r_1) = (r_2, r_1) \). Continuing this way we get

\[
(m, a) = (a, r_1) = (r_1, r_2) = \cdots = (r_{k-1}, r_k) = (r_k, 0) = r_k,
\]

and this proves the claim. \( \square \)
For the computation of \( r_i \), we only have to store \( r_{i-1} \) and \( r_{i-2} \), which makes the implementation simpler. Here is a pseudocode for the algorithm:

**EUCLIDEAN ALGORITHM**

Input: the positive integers \( a \) and \( m \) with \( 0 < a < m \)

1. while true do
2. \( r \leftarrow m \mod a \)
3. if \( r = 0 \), then
4. print \( "(a, m) = " ; a ; \) stop
5. \( m \leftarrow a ; \) \( a \leftarrow r \)
6. end while

In every step the remainder is always less than the number we divide by. That is, we have \( a > r_1 > r_2 > \ldots \), hence the algorithm terminates after at most \( a \) loops. However, this does not show that the number of steps is polynomial, since \( a \) can be exponential in terms of the input size (note that this is not always the case, if \( a \) is small and \( m \) is big enough, then the last statement does not hold, but the point is that there is such a case when the size of \( a \) is comparable to the size of the input). But in fact the sequence of the remainders decreases faster, namely the following statement holds:

**Proposition 1.5.2.** The Euclidean algorithm stops after at most \( 2 \lceil \log_2 a \rceil \) loops.

**Proof.** For making the notation simpler, we set \( r_{-1} = m \) and \( r_0 = a \). Then in every loop we make a division with a remainder: \( r_{i-2} = t_i r_{i-1} + r_i \), where \( r_{i-2} > r_{i-1} > r_i \). Since \( r_{i-2} > r_{i-1} \) holds, we must have \( t_i \geq 1 \) (because \( t_i \) is a non-negative integer). It follows that \( r_{i-2} \geq r_{i-1} + r_i > 2r_i \). If the algorithm does not stop after the 2\( k \)th step, then we have

\[
a = r_0 > 2r_2 > 4r_4 > \cdots > 2^kr_{2k} \geq 2^k \cdot 1,
\]

i.e. \( k < \log_2 a \). In other words, it is impossible that we do not stop after \( 2 \lceil \log_2 a \rceil \) steps.

After this preparation we are in the position to show that the Euclidean algorithm is polynomial. If \( m \) has \( k \) digits and \( a \) has \( l \) digits, then the size of the input is \( n = k + l \). Since \( a < m \) and hence every \( r_i < m \), we perform every division on numbers with at most \( k \) digits, so we make at most \( c_1k^2 \) steps in the body of the loop. By the previous proposition we run the loop at most \( 2[\log_2 a] \geq c_2k \) times, so the total number of steps is at most \( ck^3 \leq cn^3 \).

As a final remark we note that the least common multiple can also be determined in polynomial time using the formula \((a, m) \cdot [a, m] = am\).

**1.5.5 Solution of Linear Congruences**

By Theorem 1.4.1, we know that the linear congruence \( ax \equiv b \pmod{m} \) is solvable if and only if \( d = (a, m) \mid b \), and that the number of solutions modulo \( m \) is \( d \). Hence we can use the Euclidean algorithm to decide if a congruence is solvable. We also determined all the solutions using an arbitrary one, so it remains to find the first solution. By Theorem 1.2.3, we only have to solve the equivalent linear congruence \( a'x \equiv b' \pmod{m'} \), where \( a' = a/d, b' = b/d \) and \( m' = m/d \), and here \( a' \) and \( m' \) are co-prime. So in this section we are going to assume that \((a, m) = 1\).
It turns out that a modification of the Euclidean algorithm can be used for solving a linear congruence. We illustrate this on an example, we are going to solve the congruence $567x \equiv 123 \pmod{1238}$. First we write down the congruence $1238x \equiv 0 \pmod{1238}$ which is obviously true for all integers and will be denoted by $(A)$. After that we write down the original congruence $567x \equiv 123 \pmod{1238}$, this will be denoted by $(I)$ (expressing that this is the input). Now we repeat the steps of the Euclidean algorithm, we divide $1238$ by $567$ with a remainder, and subtract from $(A)$ the congruence $(I)$ multiplied by the quotient. This way we get a linear congruence, where the coefficient of $x$ is the remainder. We calculate the smallest remainder modulo $1238$ on the right hand side to keep the numbers bounded during the process. Continuing this way, after the 6th step the coefficient of $x$ will be the greatest common divisor, namely $1$.

\[
\begin{array}{c|c|c|c}
(A) & 1238x \equiv 0 & \pmod{1238} \\
(I) & 567x \equiv 123 & \pmod{1238} \\
(A) - 2 \cdot (I) : (1) & 104x \equiv -246 \equiv 992 & \pmod{1238} \quad [1238 = 2 \cdot 567 + 104] \\
(I) - 5 \cdot (1) : (2) & 47x \equiv -4837 \equiv 115 & \pmod{1238} \quad [567 = 5 \cdot 104 + 47] \\
(1) - 2 \cdot (2) : (3) & 10x \equiv 762 & \pmod{1238} \quad [104 = 2 \cdot 47 + 10] \\
(2) - 4 \cdot (3) : (4) & 7x \equiv -2933 \equiv 781 & \pmod{1238} \quad [47 = 4 \cdot 10 + 7] \\
(3) - 1 \cdot (4) : (5) & 3x \equiv -19 \equiv 1219 & \pmod{1238} \quad [10 = 1 \cdot 7 + 3] \\
(4) - 2 \cdot (5) : (6) & x \equiv -1657 \equiv 819 & \pmod{1238} \quad [7 = 2 \cdot 3 + 1] \\
& & [3 = 3 \cdot 1 + 0] \\
\end{array}
\]

Now we repeat this for the general congruence $ax \equiv b \pmod{m}$, where $(a, m) = 1$, and add the congruence $mx \equiv 0 \pmod{m}$ in the beginning (which is true for every $m \neq 0$). It is clear, that in this case the last non-zero remainder in the Euclidean algorithm is $r_k = 1$ for some $k$. Observe, that because of this we get a congruence of the form $x \equiv c_k \pmod{m}$ in the $k$th step, which gives us the solution of the original linear congruence.

\[
\begin{array}{c|c|c|c}
(A) & mx \equiv 0 & \pmod{m} \\
(I) & ax \equiv b & \pmod{m} \\
(A) - q_1 \cdot (I) : (1) & r_1x \equiv -q_1b \equiv c_1 & \pmod{m} \quad [m = q_1a + r_1] \\
(I) - q_2 \cdot (1) : (2) & r_2x \equiv b - q_2c_1 \equiv c_2 & \pmod{m} \quad [a = q_2r_1 + r_2] \\
(1) - q_3 \cdot (2) : (3) & r_3x \equiv c_1 - q_3c_2 \equiv c_3 & \pmod{m} \quad [r_1 = q_3r_2 + r_3] \\
(2) - q_4 \cdot (3) : (4) & r_4x \equiv c_2 - q_4c_3 \equiv c_4 & \pmod{m} \quad [r_2 = q_4r_3 + r_4] \\
& \vdots & \vdots & \vdots \\
(k) & x \equiv c_{k-2} - q_kc_{k-1} \equiv c_k & \pmod{m} \quad [r_{k-2} = q_kr_{k-1} + r_k = q_kr_{k-1} + 1] \\
& & [r_{k-1} = q_{k+1}r_k + 0] \\
\end{array}
\]

In every step we get a congruence which follows from the previous ones and hence from $ax \equiv b \pmod{m}$. Thus, if $x_0$ is a solution of this congruence, then $x_0 \equiv c_k \pmod{m}$ must hold, i.e. we get the modulo $m$ unique solution. Note though that the congruences that we obtain during the process are not necessarily equivalent to the original one. For example the congruence $104x \equiv 992 \pmod{1238}$ in the example above has 2 different solutions modulo
1238 (while the original congruence has only 1). Also, in the ith step we compute the least positive remainder modulo m (this is denoted by $c_i$) keeping the occurring numbers bounded by m.

By Proposition 1.5.2 we get to the last congruence after at most $2\lceil \log_2 a \rceil$ divisions. Note that after determining the remainders of $a$ and $b$ modulo m we can reach $0 < a < m$ and $0 < b < m$, and then every further number that occurs during the process is bounded by $cm^2$ for some constant c. Using all this it is not hard to see that the algorithm is polynomial, the details are left to the reader.

We give this algorithm also by a pseudocode:

**EUCLIDEAN ALGORITHM FOR THE SOLUTION OF A CONGRUENCE**

**Task:** solution of the congruence $ax \equiv b \pmod{m}$ with $(a,m) = 1$

**Input:** the positive integers $a$, $b$ and $m$ with $0 < a, b < m$

```
1 M ← m; c ← 0; d ← b
2 while true do
3     q ← \left\lfloor \frac{m}{a} \right\rfloor; r ← m \mod a
4     if $r = 0$, then
5         print "The solution is $x \equiv d \pmod{M}$; stop"
6     t ← c − qd \mod M
7     m ← a; a ← r; c ← d; d ← t
8 end while
```

The variables $m$ and $a$ store the previous remainders (and initially $m$ and $a$, respectively), while $c$ and $d$ store the right hand sides of the previous two congruences (which are 0 and $b$ at the beginning, respectively). We also store the value of $m$ in the variable $M$, because we need it in every loop (in line 5 and 6). Finally, we make the code readable by storing the quotient in the variable $q$ since we need it in line 6 (though this is not necessary).

### 1.5.6 Primality tests

We have mentioned in Section 1.5.1 that even though we do not know an efficient algorithm which determines the prime factorization of a number, we can still decide if a number is prime or not. It is maybe surprising that this creates a situation which makes the application of some cryptographic techniques possible, as we will see in the next section.

**The Fermat test**

One of the simplest primality tests is based on the Euler-Fermat theorem (Theorem 1.3.7): if $m$ is prime and $1 \leq a \leq m - 1$ is an integer, then $\varphi(m) = m - 1$ and $a^{m-1} \equiv 1 \pmod{m}$. This means that if we are able to find an $a$ such that $a^{m-1} \not\equiv 1 \pmod{m}$, then $m$ is not prime. The so-called Fermat test works the following way: it generates numbers between 1 and $m - 1$ randomly and computes the remainder of $a^{m-1}$ modulo $m$. If this remainder is not 1, then either $(a,m) > 1$ holds, or $(a,m) = 1$ but $\varphi(m) \neq m - 1$. No matter which case applies, $m$ cannot be prime. Note that we can calculate $(a,m)$ fast, so if we are lucky enough to have the former case, even a divisor of $m$ can be determined.

Of course it can happen that we pick an $a$ such that $a^{m-1} \equiv 1 \pmod{m}$ even for a composite modulus (and then $(a,m) = 1$). If $m$ is composite, then such an $a$ is called a
Fermat liar. On the other hand, if \((a, m) = 1\) and \(a^{m-1} \not\equiv 1 \pmod{m}\), then \(a\) is called a Fermat witness for the compositeness of \(m\). So if \(a\) is a liar, then we may repeat the test several times and hope for finding a witness. It is still not obvious that we find a witness with high probability, but the following theorem assures this if there exists at least one witness:

**Theorem 1.5.3.** If \(m \in \mathbb{N}^+\) is composite and it has a Fermat witness (i.e. a number \(a\) between 1 and \(m\) which is co-prime to \(m\) and for which \(a^{m-1} \not\equiv 1 \pmod{m}\) holds), then at least half of the numbers co-prime to \(m\) between 1 and \(m\) are witnesses.

**Proof.** Let \(a\) be a witness of \(m\) and assume that \(c_1, \ldots, c_k\) are all the liars of \(m\) between 1 and \(m\) (that is, \((c_i, m) = 1\) and \(c_i^{m-1} \equiv 1 \pmod{m}\) for every \(i\)). Let \(a_i\) be the least positive number for which \(a_i \equiv ac_i \pmod{m}\) holds. We show that \(a_1, \ldots, a_k\) are pairwise different witnesses of \(m\), and hence the number of witnesses between 1 and \(m\) is at least the number of liars in this interval. Since all the \(c_i\)'s are co-primes to \(m\), the statement follows.

First observe that since \((a, m) = 1\) and \((c_i, m) = 1\) for every \(1 \leq i \leq k\), it follows from the fundamental theorem of arithmetic that \((ac_i, m) = 1\). Then \(ac_i \equiv a_i \pmod{m}\) and Proposition 1.3.1 implies that \((a_i, m) = (ac_i, m) = 1\), that is, every \(a_i\) is co-prime to \(m\). Moreover, if we raise the congruence \(a_i \equiv ac_i \pmod{m}\) to the \((m-1)\)th power, then we get

\[
a_i^{m-1} \equiv (ac_i)^{m-1} = a^{m-1}c_i^{m-1} \equiv a^{m-1} \not\equiv 1 \pmod{m},
\]

since \(c_i\) is a liar and \(a\) is a witness. That is, we have proved that \(a_i\) is a witness for every \(1 \leq i \leq k\).

It is left to show that the numbers \(a_1, \ldots, a_k\) are pairwise different. So assume that \(a_i = a_j\) for some \(1 \leq i, j \leq k\), and then \(ac_i \equiv ac_j \pmod{m}\). Dividing both sides by \(a\) we get that \(c_i \equiv c_j \pmod{m}\), where the modulus does not change because \((a, m) = 1\). Since \(1 \leq c_i, c_j \leq m\) holds, we must have \(c_i = c_j\). But the \(c_i\)'s are pairwise different, so \(i = j\) follows.

Assume that we give the output "\(m\) is prime" if after 100 tests we do not find a witness.

If \(m\) is composite and it has a witness, then we go wrong with probability at most \(2^{-100}\).

Although this number is positive, it is so small, that it is negligible in practice. But there is a bigger problem: there are numbers which do not have any witnesses.

**Definition 1.5.2.** The positive integer \(m \in \mathbb{N}^+\) is called a Carmichael number if it is composite and for every integer \(a \in \mathbb{N}^+\) with \(1 \leq a \leq m\) and \((a, m) = 1\) the congruence

\[
a^{m-1} \equiv 1 \pmod{m},
\]

holds.

If we run the test for a Carmichael number 100 times, then we get that \(m\) is prime with very high probability. We get the output "\(m\) is composite" only if we pick a divisor of \(m\) at least once out of 100 tries, but this is very unlikely. And even though the Carmichael numbers are rare (the smallest one is 561, the next one is 1105, and there are only 43 below one million), unfortunately there are infinitely many of them (see [2]).

There are modifications of this method that solve this problem, among them the most popular is the so-called Miller-Rabin test (see below). We also note that there exists a primality test with polynomial running time which does not use randomness (it is deterministic, i.e. it always gives the right result). This was shown in [1] by Agrawal, Kayal and Saxena in 2002. However, their method is too slow for applications and hence it is not used in practice.
The Miller-Rabin test

The Miller-Rabin test is similar to the Fermat test in structure, only a few modifications are needed. The criterion \( a^{m-1} \equiv 1 \pmod{m} \) will be substituted by a stricter one. We will use the following simple observation:

**Proposition 1.5.4.** Assume that \( m \) is prime, then \( x^2 \equiv 1 \pmod{m} \) holds if and only if \( x \equiv \pm 1 \pmod{m} \).

**Proof.** If \( x \equiv \pm 1 \pmod{m} \), then squaring this congruence we have \( x^2 \equiv 1 \pmod{m} \), and this is true for every \( m \).

Now assume that \( m \) is prime and \( x^2 \equiv 1 \pmod{m} \) holds, i.e. \( m \mid x^2 - 1 = (x - 1)(x + 1) \). Then by the fundamental theorem of arithmetic we must have \( m \mid x - 1 \) or \( m \mid x + 1 \), which is equivalent to \( x \equiv \pm 1 \pmod{m} \).

In the following we may assume that \( m > 2 \) is odd (and then \( m - 1 \) is even), otherwise \( m \) is composite. In the test we choose an integer \( a \) in the interval \([1, m]\) which is co-prime to \( m \), and check if \( a^{m-1} \equiv 1 \pmod{m} \) holds. If not, then \( m \) cannot be prime. But unlike in the Fermat test, now we do not say that \( a \) is a liar right away in the other case. Instead, we check if the congruence \( a^{m-1} \equiv \pm 1 \pmod{m} \) holds. If this is not true, then by the previous proposition we get that \( m \) is composite. Now if \( \frac{m-1}{2} \) is odd or if \( a^{\frac{m-1}{2}} \equiv -1 \pmod{m} \), then we choose another \( a \) and start the test from the beginning. However, if \( \frac{m-1}{2} \) is even and \( a^{\frac{m-1}{2}} \equiv 1 \pmod{m} \), then the previous proposition gives that \( a^{\frac{m-1}{4}} \equiv \pm 1 \pmod{m} \) must also hold. If not, then we get that \( m \) is composite. Otherwise we say that \( a \) is a liar if the exponent \( \frac{m-1}{2} \) is odd or the remainder is \(-1\). From here we continue the same way with the exponents \( \frac{m-1}{8}, \frac{m-1}{16}, \ldots \) until we get an odd exponent or a remainder different from \(1\). If this remainder is also different from \(-1\), then \( m \) is composite, otherwise we choose another \( a \).

An integer \( a \) co-prime to \( m \) is called a Miller-Rabin witness if choosing \( a \) in the test above we conclude that \( m \) is composite. Observe that if \( a \) is a Fermat witness, then of course it is automatically a Miller-Rabin witness, since the first step of the test is the same. Then it follows immediately from Theorem 1.5.3 that if \( m \) is composite and it is not a Carmichael number, then at least the half of the numbers co-prime to \( m \) in the interval \([1, m]\) are Miller-Rabin witnesses. The advantage of this method is that there are no Carmichael number-type exceptions here, moreover, we can be sure that there are even more witnesses than in the case of the Fermat test. Namely, the following is true:

**Theorem 1.5.5.** If \( m > 4 \) is an integer, then at least three-quarters of the integers in the interval \([1, m - 1]\) are either Miller-Rabin witnesses or not co-prime to \( m \).

The proof of this theorem can be found for example in [8]. We also note that it is conjectured that the least witness is relatively small. More precisely, if the so-called Extended Riemann Hypothesis is true, then we can find an integer \( 1 \leq a \leq 2(\ln m)^2 \) so that either \( (a, m) > 1 \) holds or \( a \) is a Miller-Rabin witness of \( m \). If this was true, then it would also give a deterministic polynomial algorithm for this task, because it would be enough to run the test for the least \( 2(\ln m)^2 \) positive integers. For the details see [3].

**Generation of primes**

Finally, we say a few words about the generation of prime numbers. This is important because big primes play a crucial role in public key cryptography, as we will see in the
next section. A simple method for this task is that we generate numbers randomly and use a
primality test to check if they are primes. It can be shown that there are enough primes among
the integers so that this algorithm finds a prime number within a reasonable time. However,
this requires advanced techniques, but some basic theorems are proved about the number of
primes for example in [5]. This book is recommended for the interested reader because it uses
only limited tools and contains the "elementary" proof of the following theorem (elementary
means that it does not use the theory of analytic functions, but the argument is involved
nonetheless). For a positive number \( x \) let \( \pi(x) \) denote the number of positive primes that are
at most \( x \). For example \( \pi(5) = 3, \pi(10) = 4 \).

**Theorem 1.5.6** (Prime number theorem).

\[
\lim_{x \to \infty} \frac{\pi(x)}{\frac{x}{\ln x}} = 1.
\]

Roughly speaking: \( \pi(n) \approx \frac{n}{\ln n} \), i.e. among the positive integers below \( n \) every \((\ln n)\)th
number is prime. This statement is very far from being precise, but we do not give further
details here.

**1.5.7 Public Key Cryptography**

The method that is described in this section is based on the following: if \( p \) and \( q \) are big
primes (e.g. with 300 digits) and their product \( N = pq \) is public (but \( p \) and \( q \) are not), then
no one is able to calculate the factors \( p \) and \( q \) within a reasonable time.

One of the main tasks of cryptography is to give methods that assure secure communica-
tion. Security means basically the following: a message sent between some participants must
be readable for them but should be left hidden for anyone else who is able to read (some part
of) the information that goes through the channel which connects the participants. In other
words, the sender has to encrypt the message so that only the receiver can decrypt it. En-
crypting and decrypting a message means nothing else than the application of two functions
that are inverses to each other: the message \( x \) is encrypted with the function \( E \) giving the
data \( E(x) \) which is decrypted by the inverse function \( D \) of \( E \), that is, \( D(E(x)) = x \). Our
goal is to find the appropriate functions \( E \) and \( D \).

In a version of this method the functions \( E \) and \( D \) are kept secret, only the participants
know them. A drawback of this that in this case they have to share these functions with
each other, and many times this is inconvenient if not impossible. Public key cryptography
solves this problem the following way: the function \( E \) is made public while the function \( D \) is
kept secret. If for example \( A \) wants to send a message to \( B \), then \( A \) can encrypt it with the
encryption function \( E_B \) that is made public by \( B \), while \( B \) decrypts it with the function \( D_B \)
(which is known only for \( B \)). On the other hand, if \( B \) wants to answer this message, then
the function \( E_A \) is used (the one that is made public by \( A \)), and \( A \) decrypts the answer with
the functions \( D_A \) (that is kept secret by \( A \)).

How is it possible that the function \( E \) is known, but one cannot determine its inverse
\( D = E^{-1} \) ? The situation is similar to the following example: given a text in German and
an English-German dictionary. Theoretically it is possible to translate the text with the help of this
dictionary, but it is quite tedious work the find all the German words, simply
because it is ordered alphabetically by the letters of the English words. Returning to the
functions: the domain of \( E \) (and \( D \)) will be a set of integers \( \{0, 1, \ldots, N - 1\} \) for some \( N \).
The number \( N \) can be chosen so big (e.g. bigger than \( 10^{500} \)), that it takes billions of years
even for supercomputers to go through its elements and calculate all the function values. So the formula for $E$ can be made public as long as the values of $E$ and $D$ can be calculated easily while it is practically impossible to determine a formula for the inverse function $D$ from the public formula of $E$.

It is not obvious to give such functions that work. We describe an example below: the RSA algorithm, which was invented by Rivest, Shamir and Adleman. For this we need the following

**Proposition 1.5.7.** If $p$ and $q$ are distinct positive primes and $N = pq$ then for every $x \in \mathbb{Z}$ integer and $k \in \mathbb{N}^+$ positive integer we have $x^{k\varphi(N) + 1} \equiv x \pmod{N}$.

**Proof.** This follows easily from the Euler-Fermat theorem (Theorem 1.3.7) in the case when $x$ is co-prime to $N$, since then we have $x^{\varphi(N)} \equiv 1 \pmod{N}$, so raising this congruence to the $k$th power and multiplying by $x$ we get the statement.

If $(x, N) \neq 1$, then $p \mid x$ or $q \mid x$. If both divisions hold, then $N \mid x$ and hence $x^{k\varphi(N) + 1} \equiv 0 \equiv x \pmod{N}$. So assume for example that $p \nmid x$ and $q \mid x$ (in the other case the proof is practically the same). Then $(p, x) = 1$, since $p$ is a prime, and by the Euler-Fermat theorem we get that $x^{\varphi(p)} = x^{p-1} \equiv 1 \pmod{p}$. We raise this congruence to the $k(q-1)$th power and multiply both sides by $x$. Using that $\varphi(N) = (p-1)(q-1)$ we get that $x^{k\varphi(N) + 1} = x^{k(p-1)(q-1) + 1} \equiv x \pmod{p}$. But since $q \mid x$, this congruence holds modulo $q$ as well. Finally, as $p \mid x^{k\varphi(N) + 1} - x$ and $q \mid x^{k\varphi(N) + 1} - x$ hold, we obtain that $pq = N \mid x^{k\varphi(N) + 1} - x$ because $p$ and $q$ are distinct primes. \hfill \Box

As the first step of the RSA algorithm we choose two primes $p$ and $q$ with (for example) 300 digits. We set $N = pq$ and choose also a $c$ with $(c, \varphi(N)) = 1$. Then the encryption function will be the following: $E(x) = x^c \pmod{N}$. The values of $E$ can easily be calculated with repeated squaring. Moreover, it turns out that the inverse $D$ of $E$ is of the same form: $D(y) = y^d \pmod{N}$ for some integer $d$. As $D$ is the inverse of $E$, we must have

$$x = D(E(x)) \equiv E(x)^d \equiv x^{cd} \pmod{N}$$

for every $0 \leq x \leq N - 1$. By the previous proposition it is enough to find a $d$ for which $cd = k\varphi(N) + 1$ for some positive integer $k$. In other words, we have to solve the congruence $cx \equiv 1 \pmod{\varphi(N)}$. This congruence is solvable since $(c, \varphi(N)) = 1$ by the choice of $c$. The solution can be calculated efficiently with the methods described in the previous sections.

Observe that we need the value of $\varphi(N) = (p-1)(q-1)$ to determine the value of $d$, and for this we need the prime factorization of $N$. Hence if we keep $p$, $q$, $d$ and $\varphi(N)$ secret, then we can make $c$ and $N$ public, and one cannot determine the function $D$, at least by the method described above.

It could happen that someone determines the function $D$ in some other form (based on $c$ and $N$ only), or that someone finds an efficient algorithm for the factorization of integers. However, this seems very unlikely, and the experience of decades shows that this method is safe. But many problems must be handled by the implementation of the algorithm. For example, the system can be attacked by analyzing the time of the encryption and decryption or by measuring the energy consumption of the computer. Also, the number $N$ and $c$ must be chosen carefully. Just to mention one difficulty: there are such numbers that can be factorized easier than a general $N$. But we do not address these questions, these topics are beyond the scope of these notes.
2 Linear Algebra

Linear algebra is undoubtedly one of the most important branches of mathematics. It is hard to give an exhausting list of its applications inside and outside mathematics. It provides a basic tool also in computer science, and from the countless problems whose solutions require the usage of this theory we address the solution of systems of equations in this chapter to give an introduction to this topic. We also get a glimpse of the connection of linear algebra with geometry through linear transformations. Finally, we are going to introduce the notions of eigenvectors and eigenvalues. All these play a crucial role for example in image processing and visualization, just to mention two evident examples.

We emphasize that in these notes we introduce only one basic example of a more general algebraic structure, although evidently the most important one. But while for many applications this suffices, there are plenty of them which require deeper understanding of the theory. There are countless books and notes in this topic, and many of them build upon some knowledge in abstract algebra. Instead of that we give here a more elementary introduction and recommend some other books for the interested reader. It is important to note that an abstract notion can be understood much easier via examples which makes our approach advantageous. For further reading we recommend the book \[7\] which still concentrates on important special cases of the general theory. For an introduction to abstract algebra we recommend for example the book \[4\].

2.1 Analytic Geometry in the Space

Analytic geometry in the plane can be familiar from high school. It uses algebraic tools to handle geometric objects: points, lines and different curves. This introductory section extends this theory to the space, where we use triples instead of pairs to describe the points. We restrict ourselves to the description of lines and planes, one can read more in \[7\] about methods which allow us to handle other surfaces.

2.1.1 The Coordinate System

On the plane one fixes two orthogonal lines, a positive direction and a unit on each of the lines to obtain a unique representation of every point. One can extend this method to the space where we fix three pairwise orthogonal lines - the axes \(x\), \(y\) and \(z\) - which intersect in one point and determine the point of origin \(O\) this way. We also fix a point different from \(O\) on each axis and these points determine three segments whose other endpoint is the origin and also three directions from the origin towards the selected points. For simplicity we may choose a unit segment so that length and distance can be measured in the space, and in this section we assume that each of the three segments above have length 1. In other words, we fix three unit vectors on the axes which together with \(O\) form a coordinate system. Note that we can still choose their directions on the lines. Then every point \(P\) of the space determines uniquely a (maybe degenerate) rectangular cuboid whose edges are parallel to the three unit vectors and the section \(OP\) is its diagonal (by a degenerate cuboid we mean that its vertices are co-planar). As the directed units are fixed on each of the three axes, we can measure the signed length of the edges of the cuboid and obtain the coordinate triple \((x_0, y_0, z_0)\) for the point \(P\) (so that \(x_0\), \(y_0\) and \(z_0\) are the signed length of the edges parallel to the lines \(x\), \(y\) and \(z\), respectively). Note that this is a one-to-one correspondence between the points of the space and the ordered triples of real numbers.
The coordinate system can be oriented in two different ways, right or left. It is said to be right-oriented if once the right thumb points along the $z$ axis in the positive direction, then the right index finger points along the $x$ axis and the middle finger points along the $y$ axis, both of them in the positive direction. Otherwise the coordinate system is called left-oriented. Note that we usually use right-oriented systems.

We used the word "vector" when we fixed the units on the axes. Now we give its precise meaning: similarly as on the plane, by a space vector we mean a directed line segment in the space so that any two segments with the same length and direction are considered to be the same vector. If the initial point and the endpoint of the segment coincide, then we get the zero vector whose direction is not determined (but its length is 0). If a coordinate system is fixed, then any vector can be given by its coordinates, that is, by the coordinates of the point which is the endpoint of the representative whose initial point is the origin. Such a representative is called a position vector. Vectors are usually denoted by underlined lower case letters or by the triples of their coordinates, e.g. $\mathbf{v} = (7, 2, 3)$ denotes a space vector. We also use another notation: if a vector (or more precisely a representative of it) points from $A$ to $B$ where $A$ and $B$ are some points of the space, then this vector can be denoted by $\overrightarrow{AB}$.

We have seen that once a coordinate system is chosen, the set of space vectors can be identified with the ordered triples of real numbers. By ordered we mean that the order of numbers is fixed (but they are not necessarily ordered by magnitude). The set of triples is denoted by $\mathbb{R}^3$. Here we mention that in analytic geometry we usually write the triples in a row (at least we do so in this section), while in later sections the elements of $\mathbb{R}^3$ will be written in a column. Hopefully this will not be confusing in the following.

**Vector operations**

We can define the addition, subtraction and scalar multiplication of vectors like one does in the case of plane vectors. If $\mathbf{u}$ and $\mathbf{v}$ are space vectors, then $\mathbf{u} + \mathbf{v}$ is defined the following way: we first take an arbitrary representative of $\mathbf{u}$ and then the representative of $\mathbf{v}$ whose initial point and the endpoint of the representative of $\mathbf{u}$ agree (we also say somewhat inaccurately that we translate the vector $\mathbf{v}$ to the endpoint of $\mathbf{u}$). Then the sum is determined by the representative which points from the initial point of (the representative of) $\mathbf{u}$ to the endpoint of (the representative of) $\mathbf{v}$.

Also, if $\lambda \in \mathbb{R}$ is a non-zero real number and $\mathbf{v}$ is a non-zero space vector, then we define $\lambda \mathbf{v}$ the following way: we multiply the length of $\mathbf{v}$ by $|\lambda|$ and the direction of the product will be the same if $\lambda > 0$ and the opposite if $\lambda < 0$. If $\lambda = 0$ or $\mathbf{v} = \mathbf{0}$ is the zero vector, then the result is the zero-vector. In this situation $\lambda$ is called a scalar and this operation is called scalar multiplication. Finally, the difference of $\mathbf{u}$ and $\mathbf{v}$ is defined by $\mathbf{u} - \mathbf{v} := \mathbf{u} + (-1) \cdot \mathbf{v}$.

Like in the case of the plane, the basic properties of these operations remain true. The proofs of the following claims are basically the same as the ones of the analogous statements for plane vectors.

**Theorem 2.1.1.** If $\mathbf{u}$, $\mathbf{v}$ and $\mathbf{w}$ are space vectors, then

(i) $(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$ (the addition is associative),

(ii) $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ (the addition is commutative),

(iii) $\mathbf{u} + \mathbf{0} = \mathbf{u}$,

(iv) there is an additive inverse for any vector, namely $\mathbf{u} + (-1) \cdot \mathbf{v} = \mathbf{0}$ holds.
Moreover, if $\lambda, \mu \in \mathbb{R}$, then

(v) $\lambda(u + v) = \lambda u + \lambda v$.

(vi) $(\lambda + \mu)u = \lambda u + \mu u$.

(vii) $\lambda(\mu u) = (\lambda \mu)u$.

(viii) $1u = u$.

If we fix a coordinate system, then we can easily connect these operations with the coordinates of the vectors. As before, we omit the proof of the following theorem and refer to the case of the plane instead.

**Theorem 2.1.2.** If $u = (u_1, u_2, u_3) \in \mathbb{R}^3$ and $v = (v_1, v_2, v_3) \in \mathbb{R}^3$ are space vectors and $\lambda \in \mathbb{R}$ is a scalar, then

(i) $u + v = (u_1 + v_1, u_2 + v_2, u_3 + v_3)$,

(ii) $u - v = (u_1 - v_1, u_2 - v_2, u_3 - v_3)$,

(iii) $\lambda u = (\lambda u_1, \lambda u_2, \lambda u_3)$.

For the next definition we need to introduce a notation: the length of the vector $u$ is denoted by $|u|$. If $u$ and $v$ are non-zero vectors, then their *scalar product* is defined by $u \cdot v = |u| \cdot |v| \cdot \cos \varphi$, where $\varphi$ is the angle of the vectors (i.e. the angle of the lines determined by some representatives of the vectors). If any one of the vectors are zero, then the scalar product is defined to be zero.

The scalar product can be used to decide if two non-zero vectors are orthogonal. Namely, if $u$ and $v$ are non-zero, then $u \cdot v = |u| \cdot |v| \cdot \cos \varphi = 0$ holds if and only if $\cos \varphi = 0$, i.e. $\varphi = 90^\circ$. What makes this latter observation useful is that the scalar product can be expressed easily with the help of the coordinates:

**Theorem 2.1.3.** If $u = (u_1, u_2, u_3) \in \mathbb{R}^3$ and $v = (v_1, v_2, v_3) \in \mathbb{R}^3$ are space vectors, then $u \cdot v = u_1v_1 + u_2v_2 + u_3v_3$.

Again, the proof of this theorem is basically the same as the analogous one about plane vectors.

### 2.1.2 Equations of a Line

A line on the plane is determined by one of its points and a vector which is parallel or orthogonal (perpendicular) to it. In the space the situation is different: if we fix a point of a line and a vector which is orthogonal to it, then this does not determine the line uniquely. Indeed, if we rotate the line around the axis which is parallel to the vector and contains the point, then we obtain other lines with the same property, and the union of these lines is a plane which is orthogonal to the vector.

But the other option still works, so assume that $l$ is a line of the space, then we fix a point $P_0(x_0, y_0, z_0) \in l$ and a non-zero vector $v = (a, b, c)$ which is parallel to $l$. We are going to construct all the points of the line. Assume that $P(x, y, z)$ is an arbitrary point of the space, then it lies on $l$ if and only if the vector $\overrightarrow{P_0P}$ is parallel to $v$ or it is the zero vector. To avoid many cases we say that the zero vector is parallel to every vector. If $\overrightarrow{P_0}$ is the vector that
points from the origin to $P_0$, and similarly, $\overrightarrow{P}$ is the vector that points from the origin to $P$, then $\overrightarrow{P_0P} = \overrightarrow{P} - \overrightarrow{P_0}$. Now two vectors are parallel to each other if and only if they have the same or the opposite direction, or if one of them is zero. We obtain that $P \in l$ holds if and only if $\overrightarrow{P_0P} = \lambda \overrightarrow{v}$ for some $\lambda \in \mathbb{R}$, or equivalently, $\overrightarrow{P} = \overrightarrow{P_0} + \lambda \overrightarrow{v}$. By Theorem 2.1.2 we have that $\overrightarrow{P_0} + \lambda \overrightarrow{v} = (x_0 + \lambda a, y_0 + \lambda b, z_0 + \lambda c)$, hence our condition is equivalent to

$$
\begin{align*}
x &= x_0 + \lambda a, \\
y &= y_0 + \lambda b, \\
z &= z_0 + \lambda c,
\end{align*}
$$

where $\lambda \in \mathbb{R}$ is an appropriate real number. These equations are called the parametric equations of the line $l$. When the parameter $\lambda$ runs through the set of real numbers, the triples $(x, y, z)$ run through the points of $l$.

While these equations give exactly the points of $l$, they are inconvenient when we want to decide if a given point $P(x, y, z)$ is on $l$, because we first have to compute the parameters for which the first, second and third equation of (2) are true. If the same $\lambda$ suits for all of them, that means that the point $P$ is on $l$. However, this argument gives another description, which is often better for our goals. We summarize this in the following

**Theorem 2.1.4.** Assume that $l$ is a line in the space parallel to the vector $\overrightarrow{v} = (a, b, c)$, and the point $P_0(x_0, y_0, z_0)$ lies on $l$. Then an arbitrary point $P(x, y, z)$ lies on $l$ if and only if

a) $a \neq 0$, $b \neq 0$ and $c \neq 0$, and

$$
\frac{x - x_0}{a} = \frac{y - y_0}{b} = \frac{z - z_0}{c},
$$

or

b) $a \neq 0$, $b \neq 0$ and $c = 0$, and

$$
\frac{x - x_0}{a} = \frac{y - y_0}{b}, \quad \text{and} \quad z = z_0,
$$

or an analogous condition holds if exactly one of $a$, $b$ and $c$ is 0, or

c) $a = b = 0$ and $c \neq 0$, and

$$
x = x_0, \quad y = y_0,
$$

or an analogous condition holds if exactly one of $a$, $b$ and $c$ is non-zero.

**Proof.** We already have that $P \in l$ holds if and only if the equations in (2) hold simultaneously for some $\lambda \in \mathbb{R}$. If $a \neq 0$, $b \neq 0$ and $c \neq 0$, then we can express $\lambda$ from the equations and we get that this system of equations is solvable if and only if the condition in a) holds.

If $a \neq 0$ and $b \neq 0$ but $c = 0$, then the third equation in (2) gives $z = z_0$ while expressing $\lambda$ from the first two equations we get the same value if and only if the system is solvable. This gives the condition in b). The cases where exactly one of $a$, $b$ and $c$ is zero can be handled the same way.

Finally, if $a = b = 0$ and $c \neq 0$, then the first first two equations of (2) give $x = x_0$ and $y = y_0$, while the third equation holds automatically for $\lambda = \frac{x-x_0}{c}$ and hence it does not give further restrictions, so we get the condition in c). The cases where exactly one of $a$, $b$ and $c$ is non-zero can be handled the same way.
2.1.3 Equation of a Plane

We have already seen that a point and a vector determine the plane that contains the point and orthogonal to the vector. Now we describe this plane, i.e. we give a condition which can be used to decide if a point is contained in the plane. Given a plane $S$, a vector $\mathbf{n} \neq 0$ is called a normal vector of $S$ if it is orthogonal to it. Note that a normal vector of a plane is not unique, every non-zero scalar multiple of it is also a normal vector of the same plane, moreover, we obtain all normal vectors of the plane this way.

**Theorem 2.1.5.** Let $S$ be a plane which contains the point $P_0(x_0, y_0, z_0)$ and assume that $\mathbf{n} = (a, b, c)$, $\mathbf{n} \neq 0$ is a normal vector of $S$. Then an arbitrary point $P(x, y, z)$ lies on $S$ if and only if $ax + by + cz = ax_0 + by_0 + cz_0$ holds.

**Proof.** Let $\mathbf{p} = (x, y, z)$ and $\mathbf{p}_0 = (x_0, y_0, z_0)$ be the vectors that point from the origin to $P$ and $P_0$, respectively. Now $P \in S$ is and only if $\overrightarrow{P_0P} = \mathbf{p} - \mathbf{p}_0 = (x - x_0, y - y_0, z - z_0)$ is parallel to $S$ and hence orthogonal to $\mathbf{n}$. That is, we have the equivalent condition

$$0 = (\mathbf{p} - \mathbf{p}_0) \cdot \mathbf{n} = a(x - x_0) + b(y - y_0) + c(z - z_0)$$

by Theorem 2.1.3. Reordering this equation we get the statement of the theorem. \qed

It follows also that every equation of the form $ax + by + cz = d$ determines a plane, where $a, b, c, d \in \mathbb{R}$ are real numbers and at least one of $a, b$ and $c$ is non-zero. Indeed, assume for example that $a \neq 0$, the other cases are similar. Now the plane which contains the point $(d/a, 0, 0)$ and orthogonal to the vector $(a, b, c)$ is given by the equation above.

2.2 The Space $\mathbb{R}^n$

In this section we generalize the notion of the plane and the space. While it is hard two visualize more than three pairwise orthogonal lines, the identification of the points with the set of coordinate tuples provides an appropriate starting point for this work.

2.2.1 The Notion of $\mathbb{R}^n$

In the case of the plane and the space we used pairs and triples of real numbers to describe the points. In the following step we forget about the geometric background (at least for a while) and proceed in the following way: we are going to work with $n$-tuples for a general positive integer $n$. The points represented vectors before, and accordingly we call the $n$-tuples vectors and use the analogues of the formulae in Theorem 2.1.2 to define operations on them:

**Definition 2.2.1.** Let $n \geq 1$ be a positive integer, then the set of $n$-tuples (i.e. sequences of length $n$) of real numbers is denoted by $\mathbb{R}^n$. We write the elements of $\mathbb{R}^n$ in columns, and define the addition operation and the scalar multiplication for a scalar $\lambda \in \mathbb{R}$ by the following formulae:

$$\left(\begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array}\right) + \left(\begin{array}{c} y_1 \\ y_2 \\ \vdots \\ y_n \end{array}\right) = \left(\begin{array}{c} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{array}\right), \quad \text{and} \quad \lambda \cdot \left(\begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_n \end{array}\right) = \left(\begin{array}{c} \lambda x_1 \\ \lambda x_2 \\ \vdots \\ \lambda x_n \end{array}\right).$$

The elements of $\mathbb{R}^n$ are called vectors, and they are often denoted by underlined lower case letters. The numbers that form the $n$-tuples are called the coordinates of the vectors.
From now on by a vector we do not mean a directed segment, they are simply columns of numbers. The terms "plane vector" and "space vector" are used to refer to the geometric objects.

As before, we define the difference of the vectors $\mathbf{u}$ and $\mathbf{v}$ by $\mathbf{u} - \mathbf{v} := \mathbf{u} + (-1)\mathbf{v}$, which means coordinate-wise difference. The vector with only zero coordinates are called the zero vector and it is often denoted by $\mathbf{0}$. The statement of the following theorem is basically the same as the statement of Theorem 2.1.1:

**Theorem 2.2.1.** If $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^n$ and $\lambda, \mu \in \mathbb{R}$, then

(i) $(\mathbf{u} + \mathbf{v}) + \mathbf{w} = \mathbf{u} + (\mathbf{v} + \mathbf{w})$ (the addition is associative),

(ii) $\mathbf{u} + \mathbf{v} = \mathbf{v} + \mathbf{u}$ (the addition is commutative),

(iii) $\mathbf{u} + \mathbf{0} = \mathbf{u}$,

(iv) there is an additive inverse for any vector, namely $\mathbf{u} + (-1)\mathbf{u} = \mathbf{0}$ holds.

(v) $\lambda(\mathbf{u} + \mathbf{v}) = \lambda\mathbf{u} + \lambda\mathbf{v}$.

(vi) $(\lambda + \mu)\mathbf{u} = \lambda\mathbf{u} + \mu\mathbf{u}$.

(vii) $\lambda(\mu\mathbf{u}) = (\lambda\mu)\mathbf{u}$.

(viii) $1\mathbf{u} = \mathbf{u}$.

**Proof.** All these properties follow easily from the properties of the addition and multiplication of real numbers. We prove (v) as an example and leave the proof of the other statements to the reader. So assume that $\lambda \in \mathbb{R}$, $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$,

$$\mathbf{u} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad \text{and} \quad \mathbf{v} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}.$$

Then

$$\lambda(\mathbf{u} + \mathbf{v}) = \lambda \cdot \left( \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \right) = \lambda \cdot \begin{pmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_n + y_n \end{pmatrix}$$

$$= \begin{pmatrix} \lambda(x_1 + y_1) \\ \lambda(x_2 + y_2) \\ \vdots \\ \lambda(x_n + y_n) \end{pmatrix} = \begin{pmatrix} \lambda x_1 + \lambda y_1 \\ \lambda x_2 + \lambda y_2 \\ \vdots \\ \lambda x_n + \lambda y_n \end{pmatrix}$$

$$= \lambda \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} + \lambda \cdot \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \lambda \mathbf{u} + \lambda \mathbf{v}.$$
Remark. Observe that the previous theorem and Theorem 2.1.1 contain the same statement about completely different objects. That is, here we talk about column vectors while in Section 2.1.1 we worked with directed line segments. But these theorems enlighten that these objects are very similar from the algebraic point of view. We say that they form a vector space over $\mathbb{R}$. This means nothing else than they satisfy the statement of these theorems. We just want to point out that although we will work only with $\mathbb{R}^n$ in the following, we have already seen another example of a vector space, and the truth is that this kind of structures occur many times in many different situations. We also mention that in Section 2.1.1 we identified the set of space vectors with $\mathbb{R}^3$ after we fixed a coordinate system. This example foreshadows the important role of $\mathbb{R}^n$, but we still cannot say that the set of space vectors and $\mathbb{R}^3$ are basically the same, because we have to choose a coordinate system for the identification. In other words, the coordinates of a space vector look different for different choices of the coordinate system. Speaking loosely, we can reach that the space vectors look like $\mathbb{R}^3$ but they do not look like $\mathbb{R}^3$ naturally. It is probably very hard to understand this concept at first sight, but in fact it is not necessary, since we do not even use the notion of vector space later, we concentrate only on the special case $\mathbb{R}^n$ instead and recommend the book [7] for the interested reader.

2.2.2 Subspaces of $\mathbb{R}^n$

In geometry it is clear that a plane contains infinitely many copies of a line and the space contains infinitely many copies of a plane. They are in some sense "smaller", but we can still restrict the vector operations to these subsets. In the following we are going to study the subsets of $\mathbb{R}^n$ which have this property.

**Definition 2.2.2.** Assume $\emptyset \neq V \subseteq \mathbb{R}^n$ is a non-empty subset if $\mathbb{R}^n$. We say that $V$ is a **subspace** of $\mathbb{R}^n$ if the following hold:

(i) if $u, v \in V$, then $u + v \in V$,

(ii) if $u \in V$ and $\lambda \in \mathbb{R}$, then $\lambda u \in V$.

If $V$ is a subspace of $\mathbb{R}^n$, then this is denoted by $V \leq \mathbb{R}^n$.

In other words, $V$ is a subspace of $\mathbb{R}^n$ if it is non-empty and closed under addition and scalar multiplication. The subsets $V = \mathbb{R}^n$ and $V = \{0\}$ satisfy these conditions and hence they are subspaces. They are called the **trivial subspaces** of $\mathbb{R}^n$. We also get that $0 \in V$ for every subspace $V$. Indeed, if $\underline{u} \in V$ is an arbitrary vector (note that there is a $\underline{u}$ in $V$ since $V$ is non-empty), then by property (ii) we have $0\underline{u} = \underline{0} \in V$.

**Exercise 2.2.1.** Decide if the the following sets of $\mathbb{R}^2$ are subspaces or not:

a) $V_1 = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 : x \geq 0, y \geq 0 \right\}$,

b) $V_2 = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 : x = y \right\}$,

c) $V_3 = \left\{ \begin{pmatrix} x \\ y \\ z \\ w \end{pmatrix} \in \mathbb{R}^4 : x + y + z + w = 0 \right\}$.
Solution. a) If the coordinates of the vector \( u = \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2 \) are positive, then \( u \in V_1 \), but \(-x < 0\) and \(-y < 0\), so \((-1)u \notin V_1\). Hence \( V_1 \) is not closed under scalar multiplication, so it is not a subspace of \( \mathbb{R}^2 \). Note that \( V_1 \) is still closed under addition, since the sum of non-negative numbers is non-negative.

b) If \( u, v \in V_2 \), then \( u = \begin{pmatrix} x \\ y \end{pmatrix} \) and \( v = \begin{pmatrix} x' \\ y' \end{pmatrix} \) for some \( x, y \in \mathbb{R} \), and hence their sum is \( \begin{pmatrix} x + x' \\ y + y' \end{pmatrix} \), which is also in \( V_2 \). Also, for any \( \lambda \in \mathbb{R} \) the product \( \lambda u = \begin{pmatrix} \lambda x \\ \lambda y \end{pmatrix} \) is in \( V_2 \), hence \( V_2 \) is a subspace of \( \mathbb{R}^2 \).

c) If \( u, v \in V_3 \), where
\[
u = \begin{pmatrix} x_1 \\ y_1 \\ z_1 \\ w_1 \end{pmatrix} \quad \text{and} \quad v = \begin{pmatrix} x_2 \\ y_2 \\ z_2 \\ w_2 \end{pmatrix},
\]
then \( w_i = -x_i - y_i - z_i \) for \( i = 1, 2 \). Hence
\[
u + v = \begin{pmatrix} x_1 + x_2 \\ y_1 + y_2 \\ z_1 + z_2 \\ w_1 + w_2 \end{pmatrix} \in V_3,
\]
since \( w_1 + w_2 = -(x_1 + x_2) - (y_1 + y_2) - (z_1 + z_2) \). Similarly, if \( \lambda \in \mathbb{R} \), then \( \lambda w \in V_3 \), because \( \lambda w_1 = -\lambda x_1 - \lambda y_1 - \lambda z_1 \). Thus, \( V_3 \) is a subspace of \( \mathbb{R}^4 \). \( \square \)

Exercise 2.2.2. Show that the lines in \( \mathbb{R}^2 \) that contain the origin are subspaces of \( \mathbb{R}^2 \). Show that the lines and planes in \( \mathbb{R}^3 \) that contain the origin are subspaces of \( \mathbb{R}^3 \).

It will turn out later that these are the only subspaces of \( \mathbb{R}^2 \) and \( \mathbb{R}^3 \) beside the trivial ones.

2.2.3 Generated Subspace

It is a well-known fact that if two plane vectors are not parallel, then all plane vectors can be expressed from them with the vector operations. The analogue of this fact holds also in the space:

Proposition 2.2.2. If \( a, b \in \mathbb{R}^3 \) are space vectors that are not parallel to each other and lie on the plane \( S \) which contains the origin, then every vector \( v \in \mathbb{R}^3 \) that lies on \( S \) can be expressed in the form \( \alpha a + \beta b \).

If \( a, b, c \in \mathbb{R}^3 \) are space vectors such that they do not lie on a plane that contains the origin, then every vector \( v \in \mathbb{R}^3 \) can be expressed in the form \( v = \alpha a + \beta b + \gamma c \).

Proof. Assume first that \( S \leq \mathbb{R}^3 \) is a plane that contains the origin (note that \( S \) is a subspace by Exercise 2.2.2), and \( a, b \in S \) are vectors in \( S \) that are not parallel to each other (and hence both of them are non-zero). If \( v = \overrightarrow{OP} \in S \), where \( O \) is the origin, then let \( e \) be the line which goes through \( O \) and parallel to the vector \( a \), and let \( f \) be the line which goes through \( P \) and parallel to \( b \). Since the lines \( e \) and \( f \) lie on the same plane, they intersect each other in a point \( Q \). Then \( v = \overrightarrow{OQ} + \overrightarrow{QP} \), and since \( \overrightarrow{OQ} \) and \( \overrightarrow{QP} \) are parallel to \( a \) and \( b \), respectively,
we have that $\overrightarrow{OQ} = \alpha \overrightarrow{a}$ and $\overrightarrow{QP} = \beta \overrightarrow{b}$ for some $\alpha, \beta \in \mathbb{R}$, hence the first claim of the theorem follows.

For the second part let $\overrightarrow{a}, \overrightarrow{b}, \overrightarrow{c} \in \mathbb{R}^3$ be vectors that do not lie on a plane. Then none of them is zero, and the origin together with the endpoints of any two of them determines a plane. So if $\overrightarrow{v} \in \mathbb{R}^3$ is an arbitrary vector, then let $S$ be the plane going through the origin which is spanned by $\overrightarrow{a}$ and $\overrightarrow{b}$. The line which goes through $P$ and is parallel to $\overrightarrow{c}$ intersects $S$ in the point $R$ (because it is not parallel to $S$). Now $\overrightarrow{v} = \overrightarrow{OR} + \overrightarrow{PR}$, where $\overrightarrow{PR} = \gamma \overrightarrow{c}$ for some $\gamma \in \mathbb{R}$ (since it is parallel to $\overrightarrow{c}$) and $\overrightarrow{OR} = \alpha \overrightarrow{a} + \beta \overrightarrow{b}$ for some $\alpha, \beta \in \mathbb{R}$ by the first paragraph.

The following definition generalizes the expression $\alpha \overrightarrow{a} + \beta \overrightarrow{b} + \gamma \overrightarrow{c}$ that occurs in the previous theorem:

**Definition 2.2.3.** If $\overrightarrow{v}_1, \overrightarrow{v}_2, \ldots, \overrightarrow{v}_k \in \mathbb{R}^n$ are vectors and $\lambda_1, \lambda_2, \ldots, \lambda_k \in \mathbb{R}$ are scalars, then the **linear combination** of the vectors $\overrightarrow{v}_1, \overrightarrow{v}_2, \ldots, \overrightarrow{v}_k$ with the scalars $\lambda_1, \lambda_2, \ldots, \lambda_k$ is the vector

$$\lambda_1 \overrightarrow{v}_1 + \lambda_2 \overrightarrow{v}_2 + \cdots + \lambda_k \overrightarrow{v}_k.$$

Note that in the definition above the number of the vectors and scalars can be 1, that is, for a vector $\overrightarrow{v}$ and a scalar $\lambda$ the vector $\lambda \overrightarrow{v}$ is a linear combination of $\overrightarrow{v}$. Moreover, we also define the linear combination of an empty set of vectors to be the zero vector $\overrightarrow{0}$.

Now the statement of Proposition 2.2.2 can be rephrased the following way: if three vectors in $\mathbb{R}^3$ do not lie on a plane, then every vector in $\mathbb{R}^3$ can be written as a linear combination of those vectors.

**Theorem 2.2.3.** Let $\overrightarrow{v}_1, \overrightarrow{v}_2, \ldots, \overrightarrow{v}_k \in \mathbb{R}^n$ be arbitrary vectors for some $k \in \mathbb{N}$. If $W \subset \mathbb{R}^n$ is the set of vectors that can be expressed as a linear combination of the vectors $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$, then $W$ is a subspace in $\mathbb{R}^n$.

**Proof.** If $k = 0$, then the only vector which is a linear combination of the empty set of vectors is $\overrightarrow{0}$, hence $W = \{ \overrightarrow{0} \}$, which is indeed a subspace. So assume that $k \geq 1$. We have to show that $W \neq \emptyset$ and that it is closed under addition and scalar multiplication. First note that taking (for example) the linear combination $0 \overrightarrow{v}_1 + 0 \overrightarrow{v}_2 + \cdots + 0 \overrightarrow{v}_k = \overrightarrow{0}$ we have that $\overrightarrow{0} \in W$ and hence $W \neq \emptyset$. Now assume that $\overrightarrow{w}_1, \overrightarrow{w}_2 \in W$, then by the definition of $W$ we have that

$$\overrightarrow{w}_1 = \alpha_1 \overrightarrow{v}_1 + \cdots + \alpha_k \overrightarrow{v}_k \quad \text{and} \quad \overrightarrow{w}_2 = \beta_1 \overrightarrow{v}_1 + \cdots + \beta_k \overrightarrow{v}_k,$$

for some $\alpha_1, \ldots, \alpha_k, \beta_1, \ldots, \beta_k \in \mathbb{R}$ scalars. Now using the properties of the vector operations in Theorem 2.2.1 we get that

$$\overrightarrow{w}_1 + \overrightarrow{w}_2 = (\alpha_1 + \beta_1) \overrightarrow{v}_1 + \cdots + (\alpha_k + \beta_k) \overrightarrow{v}_k \in W,$$

hence it is a linear combination of the vectors $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$. Similarly, for every $\lambda \in \mathbb{R}$ we have

$$\lambda \overrightarrow{w}_1 = (\lambda \alpha_1) \overrightarrow{v}_1 + \cdots + (\lambda \alpha_k) \overrightarrow{v}_k,$$

which is again a linear combination of the vectors $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$ and hence it is in $W$. This completes the proof of the theorem.

**Definition 2.2.4.** If $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k \in \mathbb{R}^n$ are arbitrary vectors, then the subspace $W$ that consists of the linear combinations of these vectors are called the **span** of $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$ and it is denoted by $W = \text{span} \{ \overrightarrow{v}_1, \ldots, \overrightarrow{v}_k \}$. We also say that $W$ is spanned or **generated** by the vectors $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$, and we call the vectors $\overrightarrow{v}_1, \ldots, \overrightarrow{v}_k$ a **generating system** of $W$. 

39
We can rephrase the statement of Proposition 2.2.2 again: if three vectors of $\mathbb{R}^3$ do not lie on a plane, then they span the whole space $\mathbb{R}^3$, or $\mathbb{R}^3$ is generated by them. Note that the vectors $v_1, \ldots, v_k$ are also in the space spanned by them, since for every $1 \leq i \leq k$ we have
\[ v_i = 0v_1 + \cdots + 0v_{i-1} + 1v_i + 0v_{i+1} + \cdots + v_k. \]

Remark. The notation $\text{span}\{v_1, \ldots, v_k\}$ expresses that the subspace is spanned by the elements of the set $S = \{v_1, \ldots, v_k\}$. For an arbitrary set $S \subset \mathbb{R}^n$ one can define the subspace $\text{span} S$ to be the set of all linear combinations of finitely many vectors from $S$. One can show similarly as in the proof of the previous theorem that $\text{span} S$ is indeed a subspace for every (not necessarily finite) subset $S$ of $\mathbb{R}^n$. Also, it is easy to see that this new definition gives the same subspace for a finite set $S = \{v_1, \ldots, v_k\}$ as Definition 2.2.4, since every linear combination of some elements of $S$ can be completed to a linear combination of all of its elements by adding the missing vectors multiplied by 0. We also note that $\text{span} \emptyset = \{0\}$.

Exercise 2.2.3. Describe the subspace $\text{span}\{u, v\} \leq \mathbb{R}^3$, where

a) $u = \begin{pmatrix} 1 \\ 0 \\ 5 \end{pmatrix}$, $v = \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix}$, b) $u = \begin{pmatrix} 1 \\ 6 \\ 1 \end{pmatrix}$, $v = \begin{pmatrix} 3 \\ 4 \\ -1 \end{pmatrix}$.

Solution. a) A linear combination of $u$ and $v$ with the scalars $\alpha, \beta \in \mathbb{R}$ is
\[ \alpha \begin{pmatrix} 1 \\ 0 \\ 5 \end{pmatrix} + \beta \begin{pmatrix} 0 \\ 1 \\ -2 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ 5\alpha - 2\beta \end{pmatrix}. \]

If a vector $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$ can be expressed in this form, then we have to choose the scalars $\alpha = x$ and $\beta = y$, and then $5x - 2y = z$ must hold. Also, if this relation holds between the coordinates, then the vector can be expressed as a linear combination of $u$ and $v$ choosing $\alpha = x$ and $\beta = y$. Reordering this equation we get that the subspace spanned by $u$ and $v$ is nothing else than the plane $5x - 2y - z = 0$.

b) This problem can be handled like part a), but now we show another method. Since we are in $\mathbb{R}^3$, 3-dimensional geometry can be applied (and in this solution we write the vectors in a row again). Since $u$ and $v$ are not parallel, they span a plane $S$ by Proposition 2.2.2. We are going to determine a normal vector $\mathbf{n} = (a, b, c)$ of $S$ using the scalar product. A normal vector is orthogonal to every vector on $S$, in particular to $u$ and $v$. We have seen in the previous section that this is equivalent to $\mathbf{n} \cdot u = 0$ and $\mathbf{n} \cdot v = 0$. By Theorem 2.1.3 this gives that
\[ a + 6b + c = 0, \\
3a + 4b - c = 0. \]

If we express $c$ from the second equation and substitute in the first one, then we obtain $4a + 10b = 0$. Then $a = 5$ and $b = -2$ is a solution of this, and the vector $(5, -2, 7)$ satisfies both equations, hence it is a normal vector of $S$. Using that $0 \in S$ we get by Theorem 2.1.5 that the equation of the plane is $5x - 2y + 7z = 0$. □
2.2.4 Linear Independence

It can happen that among the vectors that span a subspace \( W \) there are "superfluous" elements, which means that some of the vectors may be omitted while the spanned subspace remains the same. Assume for example that \( a, b \in \mathbb{R}^n \) and \( c \) is the linear combination of \( a \) and \( b \), i.e. \( c = \lambda a + \mu b \) for some \( \lambda, \mu \in \mathbb{R} \). Then every vector in \( \text{span} \{ a, b, c \} \) is of the form

\[
\alpha a + \beta b + \gamma c = \alpha a + \beta b + \gamma(\lambda a + \mu b) = (\alpha + \gamma \lambda)a + (\beta + \gamma \mu)b,
\]

which is a linear combination of \( a \) and \( b \). It follows that \( \text{span} \{ a, b, c \} \subset \text{span} \{ a, b \} \). On the other hand, every linear combination of \( a \) and \( b \) can be written as a linear combination of \( a, b \) and \( c \), since \( \alpha a + \beta b = \alpha a + \beta b + 0c \). Then \( \text{span} \{ a, b \} \subset \text{span} \{ a, b, c \} \), and hence

\[
\text{span} \{ a, b \} = \text{span} \{ a, b, c \}.
\]

If there is no superfluous vector in the above sense, then we say that the vectors are independent:

**Definition 2.2.5.** The collection of vectors \( v_1, v_2, \ldots, v_k \in \mathbb{R}^n \) is called linearly independent, if no one of them can be written as a linear combination of the others. If there is a vector among them, which is a linear combination of the others, then the collection of vectors \( v_1, \ldots, v_k \) is said to be linearly dependent.

Note that the empty set is defined to be linearly independent. If \( k = 1 \), then the definition above gives that a single vector is linearly independent if and only if it is not the linear combination of the empty set, i.e. if and only if it is non-zero. For \( k = 2 \) we get that two vectors are linearly dependent if and only if one of them is the scalar multiple of the other. Note that if the zero vector is among the vectors, then they are linearly dependent, since multiplying any other vector by 0 we get the zero vector as a linear combination of the others.

If \( a, b, c \in \mathbb{R}^3 \) are space vectors such that they do not lie on a plane that contains the origin, then none of them can be written as a linear combination of the other two. Indeed, any two of them generates a plane which goes through the origin by Proposition 2.2.2 and the third one cannot lie on it. For the same reason we get that if \( a, b \) and \( c \) lie on a plane which contains the origin, then they are linearly dependent.

It is important to note that the linear independence or dependence is the property of the whole collection and not of the single vectors. We use the word "collection" instead of "set" in the definition to handle the situation when a vector appears more than once among \( v_1, \ldots, v_k \), because a vector can be an element of a set only once. Note that in the above case this collection will be dependent automatically, because if \( v_i = v_j \) for some \( i \neq j \), then both of them are linear combinations of the other vectors. For example, to express \( v_i \) we choose the scalar 1 as the coefficient of \( v_i \) and multiply the other vectors by 0. On the other hand, if the vectors are pairwise distinct (e.g. when they are linearly independent), then they form a set, so we may say that a set of vectors is independent or dependent. Also, we may omit the word collection or set, and simply say (somewhat loosely) that the vectors are independent or dependent. It follows immediately from the definition that if a set of vectors is independent, then any subset of them is also independent.

The following theorem gives an equivalent condition to the linear independence. It is particularly useful when one wants to decide if a collection of vectors is independent. Note that many authors use it to define linear independence.
Theorem 2.2.4. The collection of the vectors \( \mathbf{v}_1, \ldots, \mathbf{v}_k \in \mathbb{R}^n \) is linearly independent if and only if the equation \( \lambda_1 \mathbf{v}_1 + \cdots + \lambda_k \mathbf{v}_k = \mathbf{0} \) holds only for the scalars \( \lambda_1 = \lambda_2 = \cdots = \lambda_k = 0 \).

Proof. Assume first that \( \lambda_1 \mathbf{v}_1 + \cdots + \lambda_k \mathbf{v}_k = \mathbf{0} \) holds only in the case \( \lambda_1 = \cdots = \lambda_k = 0 \). If one of the vectors, say \( \mathbf{v}_i \), can be expressed as the linear combination of the others, then

\[
\mathbf{v}_i = \alpha_1 \mathbf{v}_1 + \cdots + \alpha_{i-1} \mathbf{v}_{i-1} + \alpha_{i+1} \mathbf{v}_{i+1} + \cdots + \alpha_k \mathbf{v}_k
\]

for some \( \alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_k \in \mathbb{R} \), and by reordering the equation we get that

\[
\alpha_1 \mathbf{v}_1 + \cdots + \alpha_{i-1} \mathbf{v}_{i-1} - 1 \mathbf{v}_i + \alpha_{i+1} \mathbf{v}_{i+1} + \cdots + \alpha_k \mathbf{v}_k = \mathbf{0},
\]

which contradicts our assumption (since the coefficient of \( \mathbf{v}_i \) is nonzero), so the collection of the vectors must be independent.

Now assume that the collection of the vectors \( \mathbf{v}_1, \ldots, \mathbf{v}_k \) is linearly independent. Now if \( \lambda_1 \mathbf{v}_1 + \cdots + \lambda_k \mathbf{v}_k = \mathbf{0} \) holds for some \( \lambda_1, \ldots, \lambda_k \in \mathbb{R} \) such that not all of them are zero, then we choose the index \( i \) such that \( \lambda_i \neq 0 \). But then

\[
\mathbf{v}_i = \frac{-\lambda_1}{\lambda_i} \mathbf{v}_1 - \cdots - \frac{\lambda_{i-1}}{\lambda_i} \mathbf{v}_{i-1} - \frac{\lambda_{i+1}}{\lambda_i} \mathbf{v}_{i+1} - \cdots - \frac{\lambda_k}{\lambda_i} \mathbf{v}_k
\]

contradicting the linear independence (since \( \mathbf{v}_i \) can be expressed as a linear combination of the other vectors). This means that \( \lambda_1 \mathbf{v}_1 + \cdots + \lambda_k \mathbf{v}_k = \mathbf{0} \) can hold only if all of the coefficients are 0. \( \square \)

The linear combination \( 0 \mathbf{v}_1 + 0 \mathbf{v}_2 + \cdots + 0 \mathbf{v}_k = \mathbf{0} \) is called the trivial linear combination of the vectors \( \mathbf{v}_1, \ldots, \mathbf{v}_k \). The previous statement gives that a collection of vectors is independent if and only if there is no linear combination of the vectors which gives the zero vector other than the trivial one.

Exercise 2.2.4. Decide if the following sets of vectors are linearly independent or not.

\[
\begin{align*}
a) & \begin{pmatrix} 1 \\ 2 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 2 \\ 3 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \\
b) & \begin{pmatrix} 1 \\ 2 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 2 \\ 2 \\ 3 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 5 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 4 \\ 3 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix}. \end{align*}
\]

Solution. a) Let us denote the vectors in order by \( a, b, c \) and \( d \). We are going to use the previous theorem, that is, we have to decide if the equation \( \alpha a + \beta b + \gamma c + \delta d = \mathbf{0} \) has a non-trivial solution (i.e. different from \( \alpha = \beta = \gamma = \delta = 0 \)). Substituting the vectors and using the definitions of the vector operations we get that

\[
\alpha a + \beta b + \gamma c + \delta d = \begin{pmatrix} \alpha + \beta \\ 2\alpha + 2\beta + 4\delta \\ 2\alpha + 2\beta + 3\gamma \\ 5\beta + \gamma + \delta \end{pmatrix}.
\]

This linear combination gives the zero vector if and only if the following equations hold:

\[
\begin{align*}
\alpha + \beta &= 0, \\
2\alpha + 2\beta + 4\delta &= 0, \\
2\alpha + 2\beta + 3\gamma &= 0, \\
5\beta + \gamma + \delta &= 0.
\end{align*}
\]
If we multiply the first equation by 2 and subtract it from the second and third equation, we obtain $4\delta = 0$ and $3\gamma = 0$, which means that $\delta$ and $\gamma$ must be 0. Substituting this in the fourth equation we have $5\beta = 0$, i.e. $\beta = 0$, and then $\alpha = 0$ follows from the first equation. It follows that the vectors are linearly independent.

b) We can start the same way as in part a) and infer the system of equations

\[
\begin{align*}
\alpha + \beta + 2\delta &= 0, \\
2\alpha + 2\beta + 4\delta &= 0, \\
2\alpha + 2\beta + 3\gamma &= 0, \\
5\beta + \gamma + \delta &= 0.
\end{align*}
\]

Comparing this to the system in part a) one can notice that only the first equation changed. Also, in this case we get the second equation if we multiply the first one by 2. Hence we get an equivalent system if we omit (for example) the first one. Now we can take the difference of the first two equations in the new system to obtain $3\delta = 4\delta$, i.e. $\delta = 3\gamma$. Substituting this in the last equation we get $\beta = -\frac{7}{20}\gamma$, and then $\alpha = -\beta - \frac{3}{2}\gamma = -\frac{23}{20}\gamma$. We have no other information about the variables. Indeed, if we express every other variable in terms of $\gamma$ and substitute them in the equations, then the coefficient of $\gamma$ becomes 0. This means simply that the value of $\gamma$ can be chosen freely, and then the other values are determined. That is, we do have a non-trivial solution of the system (e.g. $\alpha = -23, \beta = -7, \gamma = 20, \delta = 15$), and hence the vectors are dependent. □

2.2.5 The I-G Inequality

In this section we prove a result which will have a crucial role in the following. We have already seen that 3 vectors in $\mathbb{R}^3$ can form a generating system of $\mathbb{R}^3$ (see Proposition 2.2.2) but 2 vectors cannot. Also, 3 vectors of $\mathbb{R}^3$ can be independent if they do not lie on a plane containing the origin, and it is easy to see that 4 space vectors cannot be independent. That is, a set of independent vectors contains at most as many elements as a generating system. This latter statement holds in general (not only in $\mathbb{R}^3$):

**Theorem 2.2.5 (I-G inequality).** Let $V \leq \mathbb{R}^n$ be a subspace. If $f_1, \ldots, f_k \in V$ is a set of independent vectors and $g_1, \ldots, g_m \in V$ is a generating system in $V$, then $k \leq m$.

For the proof we will use the following lemmas:

**Lemma 2.2.6.** Assume that $f_1, f_2, \ldots, f_k, f_{k+1} \in \mathbb{R}^n$ such that the collection $f_1, f_2, \ldots, f_k$ is linearly independent, while the collection $f_1, f_2, \ldots, f_k, f_{k+1}$ is linearly dependent. Then $f_{k+1} \in \text{span}\{f_1, \ldots, f_k\}$ (i.e. $f_{k+1}$ can be expressed as the linear combination of the other vectors).

**Proof.** As the vectors $f_1, \ldots, f_k, f_{k+1}$ are linearly dependent, by Theorem 2.2.4 we have scalars $\lambda_1, \ldots, \lambda_k, \lambda_{k+1} \in \mathbb{R}$ such that at least one of them is non-zero and

\[
\lambda_1 f_1 + \cdots + \lambda_k f_k + \lambda_{k+1} f_{k+1} = 0.
\]

Here $\lambda_{k+1} \neq 0$ must hold, otherwise 0 would be a non-trivial linear combination of the vectors $f_1, \ldots, f_k$ contradicting the linear independence of this them. Reordering this equation we obtain

\[
f_{k+1} = -\frac{\lambda_1}{\lambda_{k+1}} f_1 - \cdots - \frac{\lambda_k}{\lambda_{k+1}} f_k,
\]

and the statement is proved. □
Lemma 2.2.7 (The exchange lemma). Assume that $V \leq \mathbb{R}^n$ is a subspace. If the collection $f_1, \ldots, f_k \in V$ is linearly independent and $g_1, \ldots, g_m$ is a generating system of $V$, then for every $1 \leq i \leq k$ we can find a $1 \leq j \leq m$ such that the vectors $f_1, \ldots, f_{i-1}, g_j, f_{i+1}, \ldots, f_k$ are linearly independent.

Proof. After a possible renumbering we may assume $i = k$. Let us replace $f_k$ with $g_j$ for some $1 \leq j \leq m$. If the collection $f_1, \ldots, f_{k-1}, g_j$ is independent, then we are done. On the other hand, if it is linearly dependent, then we get by the previous lemma that $g_j \in \text{span} \{ f_1, \ldots, f_{k-1} \}$ (since the vectors $f_1, \ldots, f_{k-1}$ are independent because they form a subset of a set of independent vectors).

Assume that we get a dependent collection for every $j$ in the previous paragraph. This means that $g_1, \ldots, g_m \in \text{span} \{ f_1, \ldots, f_k \}$, and hence every linear combination of the $g_j$'s is in this span, since it is closed under addition and scalar multiplication. But every element of $V$ is a linear combination of the $g_j$'s, because they span $V$. As $f_k \in V$, we obtain that $f_k \in \text{span} \{ f_1, \ldots, f_{k-1} \}$, and this is impossible, because the vectors $f_1, \ldots, f_k$ are independent. This contradiction completes the proof of the lemma.

Proof of Theorem 2.2.5. We apply the previous lemma first to $f_1$ and get the set $g_j, f_2, \ldots, f_k$ of independent vectors for some $1 \leq j \leq m$. In the next step we apply the exchange lemma for this set and the generating system $g_1, \ldots, g_m$, and replace $f_2$ with some $g_j$ still obtaining an independent set $g_j, f_2, f_3, \ldots, f_k$, for some $1 \leq l \leq m$. Continuing this way we can replace all the $f_j$'s such that the result is an independent collection of $k$ vectors consisting of some of the $g_j$'s. Moreover, in this collection the vectors are different, because they are independent. Since the cardinality of the set $\{ g_1, \ldots, g_m \}$ is $m$, we must have $k \leq m$. □

2.2.6 Basis and Dimension

The triples of vectors in $\mathbb{R}^3$ that do not lie on a plane containing the origin have a special property: they are independent and also span the whole space $\mathbb{R}^3$. The sets of vectors with this property have an important role.

Definition 2.2.6. Assume that $V \leq \mathbb{R}^n$ is a subspace. The set of the vectors $b_1, \ldots, b_k \in V$ is called a basis of $V$ if it is linearly independent and spans $V$.

Theorem 2.2.8. Assume that $V \leq \mathbb{R}^n$ is a subspace. If $b_1, \ldots, b_k$ and $c_1, \ldots, c_m$ are bases in $V$, then $k = m$.

Proof. We apply the I-G inequality (Theorem 2.2.5) for the independent set $b_1, \ldots, b_k$ and the generating system $c_1, \ldots, c_m$ in $V$ and obtain that $k \leq m$. Changing the roles of the two bases and applying the I-G inequality again we get $m \leq k$ and the assertion follows. □

Definition 2.2.7. Let $b_1, \ldots, b_k$ be a basis in the subspace $V \leq \mathbb{R}^n$. Then the number $k$ is called the dimension of $V$. The dimension of the subspace $V$ is denoted by dim $V$.

Theorem 2.2.8 assures that the previous definition is correct, since if there is a finite basis in a subspace, then the number of the vectors in it is uniquely determined. But at this point we do not know if there is always a basis in a subspace. Luckily this is the case, i.e. every subspace of $\mathbb{R}^n$ has a dimension (which is finite), but for the proof we need some preparation.
The standard basis

**Notation.** In the following we use the notation $e_i \in \mathbb{R}^n$ for the vector whose coordinates are 0 except for the $i$th one which is 1.

**Proposition 2.2.9.** The set of the vectors $e_1, \ldots, e_n$ is a basis of $\mathbb{R}^n$.

**Proof.** First we write down the linear combination of the vectors $e_1, \ldots, e_n$ with the scalars $\lambda_1, \ldots, \lambda_n$:

\[
\lambda_1 e_1 + \cdots + \lambda_n e_n = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \lambda_1 + \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \lambda_2 + \cdots + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \lambda_n = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{pmatrix}.
\]

It follows immediately that if this linear combination gives the zero vector, then every $\lambda_i$ must be zero, hence the $e_i$’s are independent. Also, if $v \in \mathbb{R}^n$, then we can choose $\lambda_i$ to be the $i$th coordinate of $v$, and this way the linear combination above gives the vector $v$, i.e. the $e_i$’s span $\mathbb{R}^n$. \hfill \Box

**Definition 2.2.8.** The basis $e_1, \ldots, e_n \in \mathbb{R}^n$ defined above is called the **standard basis** of $\mathbb{R}^n$. It is denoted by $E_n$ or $E$ (if $n$ is clear from the context).

It follows immediately that $\dim \mathbb{R}^n = n$. This is in accordance with our intuition, as one calls the space $\mathbb{R}^3$ three-dimensional. The reason for this that in $\mathbb{R}^3$ there are 3 independent directions, often represented by the directions of the axes which correspond to the vectors $e_1, e_2, e_3$ defined above. Though $\mathbb{R}^3$ is three-dimensional also in the sense of Definition 2.2.7, it is still not right to call it the three-dimensional space, since in general there are other subspaces with this property.

**Exercise 2.2.5.** Show that $\mathbb{R}^m$ has an $n$-dimensional subspace for every $0 \leq n \leq m$.

**Exercise 2.2.6.** Let $V \leq \mathbb{R}^4$ be the subspace of $\mathbb{R}^4$ which consists of the vectors in $\mathbb{R}^4$ for which the sum of their coordinates is zero (see part c) of Exercise 2.2.1). Give a basis in $V$.

**Solution.** If

\[
b_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix}, \quad b_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ -1 \end{pmatrix},
\]

then $b_1, b_2, b_3 \in V$. We show that these vectors form a basis in $V$. Now as in the previous proof, we have that

\[
\lambda_1 b_1 + \lambda_2 b_2 + \lambda_3 b_3 = \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ -\lambda_1 - \lambda_2 - \lambda_3 \end{pmatrix}
\]

for every $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}$. So if this linear combination gives the zero vector, then clearly $\lambda_1 = \lambda_2 = \lambda_3 = 0$ must hold, hence the $b_i$’s are independent.
On the other hand, if \(
\begin{pmatrix}
x \\
y \\
z \\
w
\end{pmatrix}
\in V\), then \(w = -x - y - z\), hence we get this vector as a linear combination of the \(b_i\)'s by choosing the coefficients \(\lambda_1 = x\), \(\lambda_2 = y\), \(\lambda_3 = z\). That is, the \(b_i\)'s span \(V\) and hence they form a basis in \(V\). \(\Box\)

Existence of a basis in subspaces

**Theorem 2.2.10.** If \(V \leq \mathbb{R}^n\) is a subspace, then there exists a basis of \(V\).

**Proof.** If \(V = \{0\}\), then the empty set is a basis of \(V\), since it is independent and the zero vector is a linear combination of the empty set by definition (and hence \(\dim V = 0\)). Otherwise there is a non-zero vector \(0 \neq v \in V\), which constitutes a linearly independent set (with 1 element). Hence the statement follows from the next theorem. \(\Box\)

**Theorem 2.2.11.** Assume that \(V \leq \mathbb{R}^n\) is a subspace. If \(f_1, \ldots, f_k\) is an independent set of vectors in \(V\) (where \(k\) is a non-negative integer), then it can be completed to a basis of \(V\) by adding finitely many (possibly zero) elements.

**Proof.** If \(W = \text{span}\{f_1, \ldots, f_k\}\), then \(W \subset V\), because \(V\) is a subspace, so every linear combination of the \(f_i\)'s must be in \(V\) (note that for \(k = 0\) we have \(W = \{0\}\)). If \(W = V\), then we are done. Otherwise there is a \(v \in V \setminus W\). Then by Lemma 2.2.6 the collection \(f_1, \ldots, f_k, v\) must be independent, otherwise \(v\) would be in the span of the \(f_i\)'s. If this larger set already generates \(V\), then we are done. Otherwise we continue the same way.

It remains to show that this procedure stops after finitely many steps. But this is true, since by Proposition 2.2.9 there is a generating system in \(\mathbb{R}^n\) with \(n\) elements, and hence a set of independent vectors in \(\mathbb{R}^n\) can contain at most \(n\) elements by the I-G inequality. \(\Box\)

**Corollary 2.2.12.** Assume that \(V \leq \mathbb{R}^n\) is a subspace with \(\dim V = k\). If the vectors \(f_1, \ldots, f_k \in V\) are linearly independent, then they constitute a basis in \(V\).

**Proof.** By the previous theorem the set of the vectors \(f_1, \ldots, f_k\) can be completed to a basis by adding finitely many (possibly zero) elements. But since \(\dim V = k\), every basis has exactly \(k\) elements, so the vectors above form a basis. \(\Box\)

An analogous statement holds with a generator system instead of independent vectors:

**Theorem 2.2.13.** Assume that \(V \leq \mathbb{R}^n\) is a subspace with \(\dim V = k\). If the vectors \(g_1, \ldots, g_k \in V\) span \(V\), then they constitute a basis in \(V\).

**Proof.** As \(\dim V = k\), there are vectors \(f_1, \ldots, f_k \in V\) which form a basis, and hence they are linearly independent. If we repeat the proof of Theorem 2.2.5 with the \(f_i\)'s as independent vectors and the \(g_j\)'s as the vectors that span the subspace, we get the statement. \(\Box\)

**Exercise 2.2.7.** Let \(V\) be the subspace of those vectors in \(\mathbb{R}^4\) for which the sum of their coordinates is zero (we have seen in Exercise 2.2.6 that this is indeed a subspace). Give a basis of \(V\) which contains the vector \(f = \begin{pmatrix} 1 \\ 2 \\ 3 \\ -6 \end{pmatrix}\).
**Solution.** We have seen in Exercise 2.2.6 that \( \dim V = 3 \), so by Corollary 2.2.12 it is enough to give 3 independent vectors in \( V \) such that one of them is \( f \). We are going to add \( b_2 \) and \( b_3 \) from the solution of Exercise 2.2.6. Note that the two vectors \( \overline{f}, b_1 \) form an independent set since they are not the scalar multiples of each other. Every linear combination of them is of the form

\[
\alpha \overline{f} + \beta b_1 = \alpha \begin{pmatrix} 1 \\ 2 \\ 3 \\ -6 \end{pmatrix} + \beta \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix} = \begin{pmatrix} \alpha + \beta \\ 2\alpha \\ 3\alpha \\ -6\alpha - \beta \end{pmatrix}.
\]

It is easy to see that \( b_2 \notin \text{span}\{\overline{f}, b_1\} \). Indeed, in the other case we would have \( 2\alpha = 1 \) and \( 3\alpha = 0 \), which is impossible. Then by Lemma 2.2.6 we get that the vectors \( \overline{f}, b_1, b_2 \) are independent, because otherwise \( b_2 \) would be the linear combination of the other two vectors. As we mentioned above, it follows from this that they form a basis. \( \square \)

**Coordinate Vectors**

If \( b_1, \ldots, b_k \) is a basis in the subspace \( V \), then it spans \( V \), that is, every vector can be expressed as a linear combination of the basis vectors. What makes bases special among the generating systems of \( V \) is that this representation of the vectors is unique:

**Theorem 2.2.14.** Assume that \( V \subseteq \mathbb{R}^n \) is a subspace. Then the vectors \( b_1, \ldots, b_k \in V \) form a basis of \( V \) if and only if every \( v \in V \) can be expressed uniquely as a linear combination of them (i.e. if \( v = \lambda_1 b_1 + \cdots + \lambda_k b_k = \mu_1 b_1 + \cdots + \mu_k b_k \) holds, then \( \lambda_i = \mu_i \) for every \( 1 \leq i \leq k \)).

**Proof.** Assume first, that every vector in \( V \) can be written uniquely as the linear combination of the \( b_i \)'s. Then of course the \( b_i \)'s generate \( V \). Moreover, since the trivial linear combination of them gives the zero vector (which is in \( V \)), no other linear combination can be the zero vector by our assumption, which means by Theorem 2.2.4 that the vectors \( b_1, \ldots, b_k \) are independent, i.e. they form a basis in \( V \).

Now assume that the \( b_i \)'s form a basis in \( V \). Then they span \( V \) by definition, so every \( v \in V \) is a linear combination of them. Assume that for a \( v \in V \) we have

\[
v = \lambda_1 b_1 + \cdots + \lambda_k b_k = \mu_1 b_1 + \cdots + \mu_k b_k,
\]

then reordering this equality we get that

\[
0 = (\lambda_1 - \mu_1)b_1 + \cdots + (\lambda_k - \mu_k)b_k.
\]

But since the \( b_i \)'s are linearly independent, we obtain by Theorem 2.2.4 that all of the coefficients above are zero, that is, \( \lambda_i = \mu_i \) for every \( 1 \leq i \leq k \). \( \square \)

Now we can fix a basis \( B = \{b_1, \ldots, b_n\} \) in any subspace \( V \subseteq \mathbb{R}^n \). We remark that this notation is somewhat misleading since it suggests that this basis is a set. Which is true of course, but here the order of the basis vectors will be also important for us (and not just the elements of the set \( B \)). From now on, once we say that we fix a basis we mean that we fix an ordered basis, i.e. the set \( B \) and the order of the vectors in \( B \). Still we stick to this (unprecise) notation above since it is common in the literature.

Once a(n ordered) basis is fixed in a subspace \( V \), one can represent every vector of \( V \) uniquely as a linear combination of the basis elements, and the coefficients of the basis vectors can be assigned to the vector that is represented. That is, every basis determines a "coordinate system" in \( \mathbb{R}^n \). The \( n \) coefficients can be written as a column vector, i.e. as an element of \( \mathbb{R}^n \).
Definition 2.2.9. Assume that \( V \subseteq \mathbb{R}^n \) is a subspace, \( \mathbf{v} \in V \) and \( B = \{b_1, \ldots, b_k\} \) is a basis in \( V \). If \( \mathbf{v} = \lambda_1 b_1 + \cdots + \lambda_k b_k \) (and then the coefficients \( \lambda_1, \ldots, \lambda_k \) are determined uniquely), then the vector
\[
[v]_B := \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_k \end{pmatrix} \in \mathbb{R}^k
\]
is called the coordinate vector of \( \mathbf{v} \) relative to \( B \).

If \( V = \mathbb{R}^n \) and \( B = E_n \) is the standard basis, then \( \mathbf{v} = [\mathbf{v}]_B \) for every \( \mathbf{v} \in V \) (this follows easily from the proof of Proposition 2.2.9). On the other hand, for any other basis \( B \) the coordinate vector relative to \( B \) can be different from the vector as we are going to see in the next example.

Exercise 2.2.8. Let
\[
B = \left\{ b_1 = \begin{pmatrix} 1 \\ 6 \\ 1 \end{pmatrix}, b_2 = \begin{pmatrix} 3 \\ 4 \\ -1 \end{pmatrix}, b_3 = \begin{pmatrix} 1 \\ -8 \\ 2 \end{pmatrix} \right\} \subset \mathbb{R}^3, \quad \mathbf{v} = \begin{pmatrix} 3 \\ 4 \\ 9 \end{pmatrix} \in \mathbb{R}^3.
\]
Show that \( B \) is a basis in \( \mathbb{R}^3 \) and determine the coordinate vector \([\mathbf{v}]_B\).

Solution. At this point we have plenty of available tools, so we show three different solutions for the first part of the exercise (and mention a fourth way). In the first two solutions we show that \( B \) spans \( \mathbb{R}^3 \). Since there are 3 elements in \( B \) and \( \dim \mathbb{R}^3 = 3 \), we obtain by Theorem 2.2.13 that \( B \) is a basis.

One can observe that \( \frac{1}{10}(2b_2 + b_3) = \mathbf{e}_1 \), so \( \mathbf{e}_1 \in \text{span} \{b_1, b_2, b_3\} = V \). As the spanned subspace \( V \) is closed under addition and scalar multiplication, we have that \( \frac{1}{10}(b_1 + b_2 - 4\mathbf{e}_1) = \mathbf{e}_2 \in V \), and then \( b_1 - \mathbf{e}_1 - 6\mathbf{e}_2 = \mathbf{e}_3 \in V \). So every linear combination of \( \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \) is in \( V \), but this is the standard basis of \( \mathbb{R}^3 \), hence \( \text{span} \{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\} = \mathbb{R}^3 \subset V \subset \mathbb{R}^3 \), i.e. \( V = \mathbb{R}^3 \).

One can show this using 3-dimensional geometry. In Exercise 2.2.2 we calculated the equation of the plane that is spanned by \( b_1 \) and \( b_2 \). This equation is \( 5x - 2y + 7z = 0 \), and substituting the coordinates of \( b_3 \) we see that it is not on this plane, hence by Proposition 2.2.2 the vectors in \( B \) span \( \mathbb{R}^3 \). Note that we could also use the method that was shown in part a) of Exercise 2.2.3 to show that \( B \) spans the whole space.

Now we apply Corollary 2.2.12 to show that \( B \) is a basis. Again, because of the cardinality of \( B \) it is enough to show that the vectors \( b_1, b_2 \) and \( b_3 \) are linearly independent. As in Exercise 2.2.4 we have to solve the following system of equations:
\[
\alpha + 3\beta + \gamma = 0, \\
6\alpha + 4\beta - 8\gamma = 0, \\
\alpha - \beta + 2\gamma = 0.
\]
Multiplying the last equation by 4 and adding it to the second one we get that \( 10\alpha = 0 \), i.e. \( \alpha = 0 \). Substituting this in the third equation we obtain \( \beta = 2\gamma \), while from the first equation we infer \( \gamma = -3\beta = -6\gamma \), and hence \( \beta = \gamma = 0 \), so the vectors in \( B \) are linearly independent by Theorem 2.2.4 and hence form a basis.
It remains to calculate the coordinate vector of $\underline{v}$ relative to $B$. For this we have to solve the equation $\alpha \underline{b}_1 + \beta \underline{b}_2 + \underline{b}_3 = \underline{v}$, which leads us (by equating the coordinates on the two sides) to the system

\[
\begin{align*}
\alpha + 3\beta + \gamma &= 3, \\
6\alpha + 4\beta - 8\gamma &= 4, \\
\alpha - \beta + 2\gamma &= 9.
\end{align*}
\]

Again, we multiply the last equation by 4 and add it to the second one to get $10\alpha = 40$, i.e. $\alpha = 4$. Substituting this in the first and the third equation we get that

\[
\begin{align*}
3\beta + \gamma &= -1, \\
-\beta + 2\gamma &= 5.
\end{align*}
\]

Now we multiply the second equation by 3 and add it to the first one to obtain $7\gamma = 14$, that is, $\gamma = 2$, and then $\beta = -1$. Hence the coordinate vector of $\underline{v}$ is

\[
[\underline{v}]_B = \begin{pmatrix} 4 \\ -1 \\ 2 \end{pmatrix}.
\]

□
References


[9] D. Szeszlér, *Bevezetés a számításméletbe I* (available online [here])
Index

basis, 44

canonical representation of positive integers, 6
Carmichael number, 27
Chinese remainder theorem, 17
co-prime numbers, 7
complete residue system, 13
composite number, 4
congruence relation, 9
coordinate system, 31
coordinate vector, 48
dimension, 44
division with remainders, 9
divisor, 4
equations of a line, 34
Euclidean algorithm, 23
Euler’s phi function, 11
Euler-Fermat theorem, 13
Fermat liar, 27
Fermat primality test, 26
Fermat witness, 27
Fundamental Theorem of Arithmetic, 5
generating system, 39
greatest common divisor, 7
irreducible numbers, 4
Karatsuba algorithm, 20
least common multiple, 7
left-oriented coordinate system, 32
linear combination, 39
linear congruence, 14
linear running time, 20
linearly dependent vectors, 41
linearly independent vectors, 41
Miller-Rabin test, 27, 28
Miller-Rabin witness, 28
modular exponentiation, 21
modulus of the congruence, 9
multiplicative function, 12

normal vector of a plane, 35
number of divisors, 7
number of primes, 8

parametric equations of a line, 34
polynomial running time, 18
position vector, 32
prime number, 4
proper divisor, 4
reduced residue system, 13
repeated squaring, 21
residue classes, 11
right-oriented coordinate system, 32
RSA algorithm, 30
scalar, 32
scalar multiplication, 32
scalar product, 33
Schönhage-Strassen algorithm, 20
space vector, 32
span, 39
standard basis, 45
subspace, 37

Toom-Cook algorithm, 20
trivial linear combination, 42
trivial subspaces, 37
vector, 35
vector operations, 32
vector space, 37
zero vector, 32, 36