Moonmath manual

April 8, 2021

Lorem **ipsum** dolor sit amet, consectetur adipiscing elit. Pellentesque semper viverra dictum. Fusce interdum venenatis leo varius vehicula. Etiam ac massa dolor. Quisque vel massa faucibus, facilisis nulla nec, egestas lectus. Sed orci dui, egestas non felis vel, fringilla pretium odio. *Aliquam* vel consectetur felis. Suspendisse justo massa, maximus eget nisi a, maximus gravida mi.

Here is a citation for demonstration: Lamport et al. [1982]

Chapter 1

Introduction

Nisi vitae suscipit tellus mauris a diam maecenas sed. Amet nisl purus in mollis nunc sed id semper. Et odio pellentesque diam volutpat commodo sed egestas egestas fringilla. Ultricies mi eget mauris pharetra et. Dictum non consectetur a erat nam at lectus urna. Elementum curabitur vitae nunc sed velit. Tincidunt nunc pulvinar sapien et ligula. Turpis cursus in hac habitasse. Hac habitasse platea dictumst quisque sagittis purus. Nam libero justo laoreet sit. Diam ut venenatis tellus in metus vulputate. Lacinia at quis risus sed vulputate. Porta nibh venenatis cras sed felis eget.

Chapter 2

Ecosystem

Testing

Tortor vitae purus faucibus ornare suspendisse sed nisi. Tellus in hac habitasse platea dictumst vestibulum rhoncus est. Commodo elit at imperdiet dui accumsan. Leo a diam sollicitudin tempor id eu nisl. Fermentum et sollicitudin ac orci. Elementum nibh tellus molestie nunc non blandit massa. Sed lectus vestibulum mattis ullamcorper velit sed. Aliquet enim tortor at auctor. Tincidunt arcu non sodales neque sodales ut. Lectus proin nibh nisl condimentum id venenatis. Vestibulum rhoncus est elit ullamcorper dignissim. Quam adipiscing vitae proin sagittis nisl. Sit amet tellus cras adipiscing. Semper eget duis at tellus at urna condimentum mattis pellentesque. Est velit egestas dui id ornare. Facilisis volutpat est velit egestas dui id ornare arcu. Vitae justo eget magna fermentum iaculis eu non. Gravida neque convallis a cras semper.

Chapter 3

Preliminaries

Introduction and summary of what we do in this chapter

3.1 Cryptological Systems

The science of information security is referred to as *cryptology.* In the broadest sense, it deals with encryption and decryption processes, with digital signatures, identification protocols, cryptographic hash functions, secrets sharing, electronic voting procedures and electronic money. EXPAND

3.2 SNARKS

3.3 complexity theory

Before we deal with the mathematics behind zero knowledge proof systems, we must first clarify what is meant by the runtime of an algorithm or the time complexity of an entire mathematical problem. This is particularly important for us when we analyze the various snark systems...

For the reader who is interested in complexity theory, we recommend, or example or , as well as the references contained therein.

3.3.1 Runtime complexity

The runtime complexity of an algorithm describes, roughly speaking, the amount of elementary computation steps that this algorithm requires in order to solve a problem, depending on the size of the input data.

Of course, the exact amount of arithmetic operations required depends on many factors such as the implementation, the operating system used, the CPU and many more. However, such accuracy is seldom required and is mostly meaningful to consider only the asymptotic computational effort.

In computer science, the runtime of an algorithm is therefore not specified in individual calculation steps, but instead looks for an upper limit which approximates the runtime as soon as the input quantity becomes very large. This can be done using the so-called *Landau notation* (also called big -$\mathcal{O}$-notation) A precise definition would, however, go beyond the scope of this work and we therefore refer the reader to .

For us, only a rough understanding of transit times is important in order to be able to talk about the security of crypographic systems. For example, $\mathcal{O}(n)$ means that the running time of the algorithm to be considered is linearly dependent on the size of the input set n, $\mathcal{O}(n^k)$ means that the running time is polynomial and $\mathcal{O}(2^n)$ stands for an exponential running time (chapter 2.4).

An algorithm which has a running time that is greater than a polynomial is often simply referred to as *slow.*

A generalization of the runtime complexity of an algorithm is the so-called *time complexity of a mathematical problem*, which is defined as the runtime of the fastest possible algorithm that can still solve this problem (chapter 3.1).

Since the time complexity of a mathematical problem is concerned with the runtime analysis of all possible (and thus possibly still undiscovered) algorithms, this is often a very difficult and deep-seated question .

For us, the time complexity of the so-called discrete logarithm problem will be important. This is a problem for which we only know slow algorithms on classical computers at the moment, but for which at the same time we cannot rule out that faster algorithms also exist.

3.4 Number Theory

To understand zk-SNARKS, the so-called modulus or remainder class arithmetic is of decisive importance.

To understand this, we start with a brief introduction to the theory of prime numbers. The decisive factor here is the so-called fundamental theorem of arithmetic, which says that every natural number can be represented as a finite product of prime numbers.

Then we describe what congruences are, define residue class rings and develop their arithmetic. We pay special attention to those residual classes where the modulus is a prime number, because the resulting sets are so called prime fields ....

The classic **?** is recommended to the interested reader who wants to delve into number theory. In addition, **?** is a somewhat more application-oriented book, which certainly has a lot to offer the reader interested in cryptology.

In the following, $\mathbb{Z}$ always denotes the ring of integers, $\mathbb{N}$ the set of positive integers and $\mathbb{N}_0$ the set of non-negative integers.

3.4.1 prime numbers

A prime number $p \in \mathbb{N}$ is a natural number $p \geq 2$, which is an integer and without a remainder, only divisible by itself and by 1. Such a prime number is called *odd* if it is not the number 2. We write $\mathbb{P}$ for the set of all prime numbers and $\mathbb{P}_{\geq 3}$ for the set of all odd prime numbers.

As the Greek mathematician Euclid was able to prove by contradiction in the famous *theorem of Euclid*, there can be no largest prime number. The set of all prime numbers is thus infinite **?**.

Since prime numbers are especially natural numbers, they can be ordered according to size, so that one can get the sequence

$$p_n := 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, \ldots \tag{3.1}$$

of all prime numbers (sequence $A000040$ in OEIS or **?** chapter 1.4). In particular, one can speak of small and large prime numbers.

As the following theorem shows, prime numbers are in a certain sense the basic units from which all other natural numbers are composed:

Theorem 3.4.1 (fundamental theorem of arithmetic)**.** *Let $n \in \mathbb{N}_{\geq 2}$ be a natural number. Then prime numbers exist $p_1, p_2, \ldots, p_k \in \mathbb{P}$, such that:*

$$n = p_1 \cdot p_2 \cdot \ldots \cdot p_k \; . \tag{3.2}$$

Except for the permutation of the factors, this representation is unambiguous and is called the prime factorization of n.

Proof. (**?** sentence 6.) □

Remark 1. *The proof of the previous theorem (3.4.1) is not constructive and the question remains how quickly the prime factorization of a natural number can be calculated in principle. This is the famous factoring problem. So much for today is known, there is no method on a classical Turing machine that calculates this representation in polynomial time. The fastest algorithms known today run sub-exponentially, ie $\mathcal{O}((1+\epsilon)^n)$ applies for a $\epsilon > 0$. Interested readers can find more on this exciting topic in* **?** *Chapter 10.*

Remark 2. *It should be pointed out at this point that the American mathematician Peter Williston Shor developed an algorithm in 1994 which can calculate the prime factor representation of a natural number in polynomial time on a quantum computer* **?***.*

The consequence of this is, of course, that cryosystems, which are based on the time complexity of the prime factor problem, are unsafe as soon as practically usable quantum computers are available.

3.4.2 remainder class arithmetic

Congruence or remainder class arithmetic is of central importance for understanding most modern crypto systems. In this section we will therefore take a closer look at this arithmetic. For the notation of remainder class arithmetic in cryptology see also **?** Chapter 3, or **?** Chapter 3.

Congruences and modular arithmetic. Let us first repeat what is meant by integer division with remainder (**?** sentence 1): Let $a \in \mathbb{Z}$ be an integer and $b \in \mathbb{N}$ a natural Number, then there are always two numbers $m \in \mathbb{Z}$ and $r \in \mathbb{N}$, with $0 \leq r < b$ such that

$$a = m \cdot b + r \tag{3.3}$$

holds. This is called *whole number division with remainder*, where a is called the *divident*, b is called the *divisor* and r is called the *remainder.* (**??**) is calculated, for example, with the written division.

If a, b, m and r satisfy the equation (3.3), in the following we always write a div $b := m$ for the integer multiple of b in a and a mod $b := r$ for the remainder in the division. With this notation, for example, -17 div $4 = -5$ and -17 mod $4 = 3$, because $-17 = -5 \cdot 4 + 3$. (Since the remainder is by definition a non-negative number) We also say, that a number a is divided by a number b (integer) if a mod $b = 0$ applies. In this case we also write $a|b$ (**?** Note 1 below).

Another important tool in the theory of integers is the so-called *extended Euclidean algorithm.* This calculates the greatest common divisor $ggT(a, b)$ of two natural numbers a and $b \in \mathbb{N}$, as well as two further integers s and $t \in \mathbb{Z}$, see above that's the equation

$$gcd(a, b) = s \cdot a + t \cdot b \tag{3.4}$$

The following pseudocode shows in detail how to calculate these numbers with the extended Euclidean algorithm (**?** chapter 2.9):

Definition 3.4.2 (Extended Euclidean Algorithm)**.** *Let the natural numbers $a, b \in \mathbb{N}$ be given. Then the so-called extended Euclidean algorithm is given by the following calculation rule:*

```
$r_0 := a, \quad r_1 := b, \quad s_0 := 1, \quad s_1 := 0, \quad k := 1$
while $r_k \neq 0$ do
    $q_k := r_{k-1}$ div $r_k$
    $r_{k+1} := r_{k-1} - q_k \cdot r_k$
    $s_{k+1} := s_{k-1} - q_k \cdot s_k$
    $k \leftarrow k + 1$
end while
```

The result provides the integers $ggT(a, b) := r_k$, as well as $s := s_k$ and $t := (r_k - s_k \cdot a)$ div b, which make up the equation $ggT(a, b) = s \cdot a + t \cdot b$ meet.

Example 1. *To illustrate the algorithm to the reader, we apply it to the pair of numbers* (12, 5) *in this example. We expect:*

k	r_k	s_k	$t_k = (r_k - s_k \cdot a)$ *div* b
0	*12*	*1*	*0*
1	*5*	*0*	*1*
2	*2*	*1*	*-2*
3	*1*	*-2*	*5*

From this one can see that 12 *and* 5 *are relatively prime (since their greatest common factor is* $ggT(12.5) = 1$*) and that the equation* $1 = (-2) \cdot 12 + 5 \cdot 5$ *applies.*

With the help of the whole number division with remainder, the congruence of two whole numbers with respect to a so-called module can be defined as follows (**?** chapter 3.1):

Definition 3.4.3 (congruence)**.** *Let a, $b \in \mathbb{Z}$ be two whole numbers and $n \in \mathbb{N}$ a natural number. Then a and b congruent with respect to the module n, if the equation*

$$a \ mod \ n = b \ mod \ n \tag{3.5}$$

is fulfilled. In this case we also write $a \equiv b \pmod{n}$. If two numbers are not congruent with respect to a given module, we also call them incongruent.

The following sentence describes a fundamental property of modulo arithmetic, which is not known from traditional arithmetic in $\mathbb{Z}$ (**?** chapter 3.11):

Theorem 3.4.4 (Fermat's Little Theorem). *Let $p \in \mathbb{P}$ be a prime number and $k \in \mathbb{Z}$ is an integer, then:*

$$k^p \equiv k \quad (\ mod \ p \ ) , \tag{3.6}$$

Proof. □

Another theorem that is important for calculating with congruences is the following Chinese remainder sentence. Roughly speaking, he shows us how to use congruence systems (**?** chapter 3.15).

Theorem 3.4.5 (Chinese remainder sentence). *For $k \in \mathbb{N}$ the coprime natural numbers $n_1, \dots n_k \in \mathbb{N}$ and the integers $a_1, \dots a_k \in \mathbb{Z}$ are given. Then there exists for the so-called simultaneous congruence*

$$\begin{array}{rl} x \equiv a_1 & (\ mod \ n_1 \ ) \\ x \equiv a_2 & (\ mod \ n_2 \ ) \\ & \dots \\ x \equiv a_k & (\ mod \ n_k \ ) \end{array} \tag{3.7}$$

a solution and all possible solutions of this congruence system are congruent modulo $n_1 \cdot \ldots \cdot n_k$.

Proof. (**?** chapter 3.15) □

Remark 3. *This is the classic Chinese remainder as it was known in ancient China. Under certain circumstances, this set can be extended to non-coprime modules $n_1, \dots, n_k$.*

Example 2. *To illustrate how to solve simultaneous congruences using the Chinese remainder theorem, let's look at the following simultaneous congruence in this example:*

$$\begin{array}{rl} x \equiv 4 & (\ mod \ 7 \ ) \\ x \equiv 1 & (\ mod \ 3 \ ) \\ x \equiv 3 & (\ mod \ 5 \ ) \\ x \equiv 0 & (\ mod \ 11 \ ) \end{array}$$

So here is $N = 7 \times 3 \times 5 \times 11 = 1155$, as well as $N_1 = 165$, $N_2 = 385$, $N_3 = 231$ and $N_4 = 105$. From this we calculate with the extended Euclidean algorithm

$$\begin{array}{rrcr} 1 = & -47 \times 7 & + & 2 \times 165 \\ 1 = & -128 \times 3 & + & 1 \times 385 \\ 1 = & -46 \times 5 & + & 1 \times 231 \\ 1 = & -19 \times 11 & + & 2 \times 105 \end{array}$$

so we have $e_1 = 2 \cdot 165 = 330$, $e_2 = 1 \cdot 385 = 385$, $e_3 = 1 \cdot 231 = 231$ and $e_4 = 2 \cdot 105 = 210$, from which

$$x = 4 \times 330 + 1 \times 385 + 3 \times 231 + 0 \times 210 = 2398$$

as one solution. Because of $2398 \ mod \ 1155 = 88$ the set of all possible solutions of this simultaneous congruence is through $\{\dots, -2222, -1067, 88, 1243, 2398, \dots\}$ given. In particular, there are infinitely many different solutions.

Residual class rings and prime fields. Congruence Modulo n is an ëquivalence relation on the set of integers, whereby there are exactly n different equivalence classes. In doing so, we identify each equivalence class with the corresponding remainder of the integer division. The addition and multiplication of whole numbers can be continued to an addition and multiplication on the equivalence classes by simply choosing an animated representative from each class. Selects, adds or multiplies them as usual and then selects the equivalence class that contains the result.

This notation is standard, both in cryptology and in elementary number theory, and is necessary in order to be able to use things like inverses, roots, etc. to be able to speak. See Chapter 3 in **?** or Chapter 3 in **?**.

Theorem 3.4.6. *Let $n \in \mathbb{N}_{\geq 2}$ be a fixed, natural number and $\mathbb{Z}_n$ the set of equivalence classes of integers with respect to the congruence modulo n relation. Then $\mathbb{Z}_n$ forms a commutative ring with one element with respect to the addition and multiplication defined above. Besides that, $\mathbb{Z}_n$ is a field if and only if n is a prime number.*

Proof. (**?** sentence 1) Associativity, distributivity and commutativity of both addition and multiplication follow directly from the corresponding properties in $\mathbb{Z}$.

The neutral element of addition is given by the equivalence class of 0, i.e. by the set $\{\ldots, -2\cdot n, -n, 0, n, 2\cdot n, \ldots\}$ and that is additive The inverse of an equivalence class $x \in \mathbb{Z}_n$ is given by the equivalence class $nx \in \mathbb{Z}_n$. $\mathbb{Z}_n$ is therefore a commutative group with regard to addition.

The neutral element of the multiplication is given by the equivalence class of 1, i.e. by the set $\{\ldots, -2\cdot +1n, -n+1.1, n+1.2\cdot n+1, \ldots\}$ given. So $\mathbb{Z}_n$ is a commutative ring with one element.

To find the multiplicative inverse of an equivalent class $x \in \mathbb{Z}_n$ with $x \neq 0$, we first assume that n is a prime number. Then $ggt(x,n) = 1$ and from the extended Euclidean algorithm it follows that there are integers $s, t \in \mathbb{Z}$ with $s\cdot x + t\cdot n = 1$ gives. But this results in $s \cdot x = 1$ in $\mathbb{Z}_n$, which shows that x is invertible, with $x^{-1} = s$. So $\mathbb{Z}_n$ is a field if n is a prime number.

On the other hand, if n is not a prime number, then there are natural numbers a and b with $n = a\cdot b$ and $0 < a, b < n$. From this it follows that $a \cdot b = 0$ in $\mathbb{Z}_n$. In this case, $\mathbb{Z}_n$ is not free of zero divisors and therefore not a field. □

Definition 3.4.7 (residue class rings and primes Korper)**.** *(**?** example 3.4.4 or **?** definition 3.1) Let $n \in \mathbb{N}_{\geq 2}$ be chosen. Then we write $(\mathbb{Z}_n, +, \cdot)$ for the one described in the sentence (**??**) commutative ring.*

If $n \in \mathbb{P}$ is also a prime number, then $\mathbb{Z}_n$ is also called Primary body for the characteristic n.

Remark 4. *As can be seen from the proof of the theorem (**??**), inverse elements in prime fields can be calculated as follows: If n is a prime number and $x \in \mathbb{Z}_n$ is given, so one calculates $s \times x + t \times n = 1$ with the extended Euclidean algorithm and $s = x^{-1}$ applies.*

The following important property immediately follows from Fermat's little theorem, for arithmetic is prime fields.

Lemma 3.4.8. *Let $p \in \mathbb{P}$ be a prime number and $\mathbb{Z}_p$ be the prime body of the characteristic p. Then applies for all elements $x \in \mathbb{Z}_p$.*

$$x^p = x \quad or \quad x^{p-1} = 1 forx \neq 0\ . \tag{3.8}$$

Proof. □

In order to better illustrate the definition of the remainder class rings to the reader, we calculate the prime body in the following example $\mathbb{Z}_5$ in detail:

Example 3 (The primary field $\mathbb{Z}_5$)**.** *For $n = 5$ we have five equivalence classes of integers which are congruent modulo 5. We write*

$$\begin{aligned} &0 := \{\ldots, -5, 0, 5, 10, \ldots\}, \quad 1 := \{\ldots, -4, 1, 6, 11, \ldots\}, \quad 2 := \{\ldots, -3, 2, 7, 12, \ldots\} \\ &3 := \{\ldots, -2, 3, 8, 13, \ldots\}, \quad 4 := \{\ldots, -1, 4, 9, 14, \ldots\} \end{aligned}$$

Addition and multiplication can now be transferred to the equivalence classes. This results in the following addition and multiplication tables in $\mathbb{Z}_5$:

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

These two tables are all you need to be able to calculate in $\mathbb{Z}_5$. For example, to determine the multiplicative inverse of a remainder class, look for the entry that results in 1 in the product table. This is the multiplicative inverse of 4 For example 4 and the multiplicative inverse of 2 is 3. In $\mathbb{Z}_5$, $4^{-1} = 4$ and $2^{-1} = 3$.

square numbers and square roots In the last part of this subsection we have to deal with *square numbers* and *square roots* in remainder class rings. To do this, we first define what square roots actually are. We roughly follow Chapter 6.5 in **?** and limit ourselves to natural numbers with prime factor representations of different prime numbers:

Definition 3.4.9 (quadratic remainder and square roots)**.** *Let $n \in \mathbb{N}_{\geq 2}$ a natural number, $n = p_1 \cdot \ldots \cdot p_k$ the prime factor representation of n, so that $p_1, \ldots, p_k$ different in pairs and $\mathbb{Z}_n$ is the remainder class ring modulo n. Has the quadratic equation*

$$x^2 = y \tag{3.9}$$

for a given $y \in \mathbb{Z}_n$ with $y \neq 0$ a solution $x \in \mathbb{Z}_n$, we denote y as quadratic remainder or square number and x as a square root of y. If the quadratic equation has no solution, however, we denote y as quadratic non-remainder. For any $y \in \mathbb{Z}_n$ we write

$$\sqrt{y}_{|n} := \{x \in \mathbb{Z}_n \mid x^2 = y\} \tag{3.10}$$

for the set of all square roots of y in the remainder class ring $\mathbb{Z}_n$. (If y is a quadratic non-remainder, then $\sqrt{y}_{|n} = \emptyset$ and if $y = 0$, then $\sqrt{y}_{|n} = \{0\}$)

Remark 5. *The notation $\sqrt{y}_{|n}$ for the root of square residues is not found in textbooks, but it is quite practical to clearly distinguish between roots in different residue class rings. The symbol $\sqrt{y}$ is generally ambiguous and it must also be specified in which ring this root is actually meant.*

Example 4 (square numbers in $\mathbb{Z}_5$)**.** *Let us consider the example (3) the square numbers are on the main diagonal of the second table. As you can see, in $\mathbb{Z}_5$ you can only get the square root of 0, 1 and 4. Here applies $\sqrt{0}_{|5} = \{0\}$, $\sqrt{1}_{|5} = \{1, 4\}$, $\sqrt{2}_{|5} = \emptyset$, $\sqrt{3}_{|5} = \emptyset$ and $\sqrt{4}_{|5} = \{2, 3\}$.*

From the previous example we know that elements from $\mathbb{Z}_5$ have a maximum of two different square roots. However, as we shall see in the following, this does not apply in general $\mathbb{Z}_n$. More precisely, the maximum number of possible square roots depends on how many prime numbers the prime factorization of n contains.

In order to describe in a given remainder class ring whether a remainder class is a square number (quadratic remainder) or not, we define (**?** chapter 6.5):

Definition 3.4.10 (Legendre symbol)**.** *Let $p \in \mathbb{P}$ be a prime number and $y \in \mathbb{Z}_p$ is a remainder class of the prime body. Then the so-called Legendre symbol of y is defined as follows:*

$$\left(\frac{y}{p}\right) := \begin{cases} 1 & \text{if } y \text{ is a quadratic remainder} \\ -1 & \text{if } y \text{ is a quadratic non-remainder} \\ 0 & \text{if } y = 0 \end{cases} \tag{3.11}$$

Example 5. *If we look again at the example (3) we have the following Legendre symbols*

$$\left(\tfrac{0}{p}\right) = 0, \quad \left(\tfrac{1}{p}\right) = 1, \quad \left(\tfrac{2}{p}\right) = -1, \quad \left(\tfrac{3}{p}\right) = -1, \quad \left(\tfrac{4}{p}\right) = 1\ .$$

The following sentence gives a simple criterion for calculating the legend symbol of a remainder class in a prime body. It should be noted that the additive inverse of 1 in $\mathbb{Z}_p$ is given by $p - 1$. In $\mathbb{Z}_p$ applies $-1 = p - 1$.

Theorem 3.4.11 (Euler criterion)**.** *Let $p \in \mathbb{P}_{\geq 3}$ be an odd one Prime number and $y \in \mathbb{Z}_p$ a remainder class. Then applies*

$$\left(\frac{y}{p}\right) = y^{\frac{p-1}{2}}\ . \tag{3.12}$$

Proof. (**?** proposition 83) □

Now that we know what quadratic residues and non-residues are, the question arises how to compute square roots in residual class rings. One has to distinguish whether the given modulus is a prime number or not. As we shall see, effective calculation methods only exist in the case of a prime number. In the other case, finding square roots is as difficult as prime factorization of the modulus.

So let's first look at the case of the prime numbers. The case $\mathbb{Z}_2$ is quickly described: Because $0^2 = 0$ and $1^2 = 1$ in $\mathbb{Z}_2$ every number is a square number and at the same time a root of itself. We can therefore restrict ourselves to odd prime numbers in the following. As the following theorem (3.4.12) shows, there are two different sub-fields, depending on whether the odd prime number is congruent to 1 or 3 modulo 4 (**?** Chapter 6.5):

Theorem 3.4.12 (square roots in primaries)**.** *Let $n \in \mathbb{P}_{\geq 3}$ be an odd prime number. Then applies*

$$n \equiv 1 \quad (\ mod\ 4\ ) or n \equiv 3 \quad (\ mod\ 4\ )$$

In the first case the roots $\sqrt{y}_{|n}$ of a square number can be calculated by the so-called Tonelli-Schanks algorithm and in the second case $n = 4 \cdot k + 3$ for f̈or a $k \in \mathbb{N}_0$ and the two roots $\sqrt{y}_{|n}$ of a square number are through given the set $\{y^{k+1}, n - y^{k+1}\}$.

Proof. □

Remark 6. *As can be seen from the previous theorem (3.4.12), the computation of square roots in a prime field is particularly quick and easy if the prime number is congruent 3 modulo 4.*

This is also used in

Definition 3.4.13 (Tonelli-Shanks algorithm)**.** *Let p be an odd prime number $p \in \mathbb{P}_{\geq 3}$ and y a quadratic remainder in $\mathbb{Z}_p$. Then the so-called algorithm by Tonneli* **?** *and Shanks* **?** *is defined as follows:*

1. *Find $Q, S \in \mathbb{Z}$ with $p - 1 = Q \cdot 2^S$ such that Q is odd.*
2. *Find an arbitrary quadratic non-remainder $z \in \mathbb{Z}_p$.*
3.
```
$M := S, \quad c := z^Q, \quad t := y^Q, \quad R := y^{\frac{Q+1}{2}}, \quad M, c, t, R \in \mathbb{Z}_p$
while $t \neq 1$ do
    Find the smallest $i$ with $0 < i < M$ and $t^{2^i} = 1$
    $b := c^{2^{M-i-1}}$
    $M := i, \quad c := b^2, \quad t := tb^2, \quad R := R \cdot b$
end while
```

The result is then the two remainder classes $r_1 := R$ and $r_2 := p - R$.

Theorem 3.4.14. *Let p be an odd prime number $p \in \mathbb{P}_{\geq 3}$ and y a quadratic remainder in $\mathbb{Z}_p$. Then the following applies: The Tonelli Shanks algorithm terminates and the two results r_1 and r_2 are the two square roots of y.*

Proof. □

Remark 7. *The algorithm (3.4.13) works in prime fields for any odd prime numbers. From a practical point of view, however, it only makes sense if the prime number is congruent to 1 modulo 4, since in the other case the formula from the proposition 3.4.12, which can be calculated more quickly, can be used.*

Now that we know how to compute roots in primaries, the question remains how to find roots in general residue class rings. As the following theorem shows, these are computed by first decomposing the modulus into prime factors and then the roots is calculated in the corresponding primaries. Then the Chinese remainder theorem is used to calculate the roots in the given ring.

Theorem 3.4.15. *Its $n \in \mathbb{N}_{\geq 2}$ a natural number with prime factor representation $n = p_1 \cdot \ldots \cdot p_k$, so that the prime numbers $p_1, \ldots, p_k$ are different in pairs and $y \in \mathbb{Z}_n$ is a square number. Then the set of all square roots $\sqrt{y}_{|n}$ of y in $\mathbb{Z}_n$ is given by the solution set of all simultaneous congruences*

$$\begin{aligned} x &\equiv x_1 \quad (\ mod\ p_1\ ) \quad , \quad x_1 \in \sqrt{y}_{|p_1} \\ &\cdots \\ x &\equiv x_k \quad (\ mod\ p_k\ ) \quad , \quad x_k \in \sqrt{y}_{|p_k} \end{aligned} \tag{3.13}$$

given, where one element is to be taken from each set of roots. In total, y has $|\sqrt{y}_{|p_1}| \cdot \ldots \cdot |\sqrt{y}_{|p_k}|$ different roots in $\mathbb{Z}_n$.

Proof. □

Example 6. *To clarify the constructive proof of the previous theorem, we want to calculate all square roots of 4 in $\mathbb{Z}_{15}$. Since 15 has the prime factorization $15 = 3 \cdot 5$, we first calculate the square roots of 4 in $\mathbb{Z}_3$ and $\mathbb{Z}_5$.*

Because of $1 \equiv 4 \quad (\ mod\ 3\ )$, the roots in $\mathbb{Z}_3$ result in $\sqrt{4}_{|3} = \sqrt{1}_{|3} = \{1.2\}$. From example (3) we also know that the square roots of 4 in $\mathbb{Z}_5$ are given by $\sqrt{4}_{|3} = \{2, 3\}$.

Next we apply the Chinese remainder theorem. We define $N = 15$, $N_1 = 5$ and $N_2 = 3$. With the extended Euclidean algorithm we then calculate $1 = 2 \times 3 + (-1) \times 5$, with which one $e_1 = -5$ and $e_2 = 6$ determined.

With this you can now easily write the roots of 4 in $\mathbb{Z}_{15}$ through all combinations of the roots in $\mathbb{Z}_3$ and $\mathbb{Z}_5$:

$$\begin{aligned} x_1 = 1 \times (-5) + 2 \times 6 &= 7 \\ x_2 = 1 \times (-5) + 3 \times 6 &= 13 \\ x_3 = 2 \times (-5) + 2 \times 6 &= 2 \\ x_4 = 2 \times (-5) + 3 \times 6 &= 8 \end{aligned} \tag{3.14}$$

In the remainder class ring $\mathbb{Z}_{15}$, 4 is a quadratic remainder and it holds $\sqrt{4}_{|15} = \{2, 7, 8, 13\}$.

From the previous sentence there is a method to constructively compute square roots in remainder class rings. Since one needs the prime factorization of the module for this, it is as complex as the prime factor problem. It remains to be shown that there cannot be a faster process. The following sentence does this:

Theorem 3.4.16. *The computation of all square roots of a square number in a remainder class ring $\mathbb{Z}_n$ is at least as complex as the prime factorization of n.*

Proof. □

3.4.3 Polynome

Following **?** we want to develop the ring of polynomials or the formal power series. To do this, we first specify the underlying quantities, then describe the corresponding addition and multiplication and finally state some norms on them:

Definition 3.4.17 (Polynomials)**.** *Let R be a commutative integrity domain with unit 1. Then we name the expression*

$$\sum_{n=0}^{m} a_n t^n = a_0 + a_1 t + a_2 t^2 + \cdots + a_m t^m \ , \tag{3.15}$$

with the unknown t and the coefficients a_n from R Polynomial with coefficients from R and write $R[t]$ for the set of all polynomials with coefficients from R.

We often simply write $P(t)$ for a polynomial or a formal power series and denote the constant term accordingly with $P(0)$. Furthermore, we always see it as a given, we also simply write "formal power series" when we actually mean "formal power series with coefficients in R".

Example 7. *The so-called zero polynomial (or the zero series) is the polynomial (or the formal power series) $\sum_{n=0}^{\infty} a_n t^n$ which (s) arises when we use all coefficients as Assume zero, ie set $a_n = 0$ for all $n \in \mathbb{N}$. In analogy to the additively neutral element $0 \in R$, we also simply write 0 for this polynomial (this series).*

The so-called single polynomial (or the one series) is the polynomial (or the formal power series) $\sum_{n=0}^{\infty} a_n t^n$ which (s) arises when we $a_0 = Set1$ and assume all other coefficients as zero, ie $a_n = 0$ for ür all $n \in \mathbb{N}$. In analogy to the multiplicatively neutral element $1 \in R$, we also simply write 1 for this polynomial (this series).

Definition 3.4.18 (degree)**.** *The degree $degree(P(t))$ of a polynomial $P(t) \in R[t]$ is defined as follows: If $P(t)$ is the zero polynomial, we set $Grad(P(t)) := -\infty$. For every other polynomial we set $degree(P(t)) = n$ if a_n is the highest non-vanishing coefficient of $P(t)$.*

In order to be able to make the set of these polynomials or the formal power series into a commutative integral ring, we have to introduce the sum and the product of two formal power series and then show that the ring axioms are met. "ullt are. The following definition gives the corresponding operations for formal power series (see e.g. **?**). A definition for polynomials is analogous.

Definition 3.4.19. *Let $\sum_{n=0}^{\infty} a_n t^n$ and $\sum_{n=0}^{\infty} b_n t$ select two formal power series $R[[t]]$. Then your sum and your product defined as follows:*

$$\sum_{n=0}^{\infty} a_n t^n + \sum_{n=0}^{\infty} b_n t^n = \sum_{n=0}^{\infty} (a_n + b_n) t^n \tag{3.16}$$

$$\left(\sum_{n=0}^{\infty} a_n t^n\right) \cdot \left(\sum_{n=0}^{\infty} b_n t^n\right) = \sum_{n=0}^{\infty} \sum_{i=0}^{n} a_i b_{ni} t^n \tag{3.17}$$

In the case of polynomials, it is only necessary to note that the degree of the sum is exactly the maximum of the degrees of the summands and that the degree of the product is exactly the sum of the degrees of the factors.

To see that the ring axioms are fulfilled, it is first clear that both polynomials and formal power series form a commutative group with respect to addition, the neutral element being the zero polynomial 0 and the additive inverse element is given by $-\sum_{n=0}^{\infty} a_n t^n$ The properties of a commutative group then follow from the corresponding properties in R.

For the multiplication one sees immediately that the single polynomial is the neutral element and that commutativity or associativity follow from the corresponding properties in R.

The distributive laws also result from the corresponding rules in R, so that overall it is shown

Since bodies, complete bodies and their algebraic closings play a central role in the analogy between Bernoulli and Carlitz-Bernoulli numbers, in the following we want all of the definitions and definitions that are important to us Briefly repeat and summarize the necessary properties on this topic. For a detailed consideration of the so-called body theory see for example Chapter 13 in **?** and for the mentioned analogy see **?** and **?**.

Definition 3.4.20 (body)**.** *A body* $(\mathbb{K}, +, \cdot)$ *is a set* $\mathbb{K}$*, provided with two links* $+ : \mathbb{K} \times \mathbb{K} \to \mathbb{K}$ *and* $\cdot : \mathbb{K} \times \mathbb{K} \to \mathbb{K}$*, called addition and multiplication, so that the following conditions are met :*

- $(\mathbb{K}, +)$ *is an Abelian group, where the neutral element is denoted here with* 0.
- $(\mathbb{K} \setminus \{0\}, \cdot)$ *is an Abelian group, where the neutral element is called* 1.
- *For all* $a, b, c \in \mathbb{K}$ *the distributive laws apply:*

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \textit{and} \quad (a + b) \cdot c = a \cdot c + b \cdot c$$

The characteristic of a body $\mathbb{K}$ *is the smallest natural number* $n \geq 1$*, for which the* n *-fold sum of the one* 1 *equals zero, ie f ür which* $\sum_{i=1}^{n} 1 = 0$ *applies. If such a* $n > 0$ *exists, the body is also called finite characteristic. If, on the other hand, every finite sum of ones is not equal to zero, then the body will have the characteristic* 0 *and one also speaks of a vanishing characteristic.*

A body $\mathbb{K}$ *is called ß t completely ändig if it is complete as a ring. A complete body is also called non-Archimedean if it is not Archimedean as a ring.*

A body $\mathbb{K}$ *is called algebraically closed if every non-constant polynomial* $P(t) \in \mathbb{K}[t]$ *has a root in* $\mathbb{K}$.

Definition 3.4.21 (The finite bodies)**.** *Let* $p \in \mathbb{N}$ *be a prime number. Then* $\mathbb{K}_p$ *denotes the remainder class body örper* $\mathbb{Z}/p\mathbb{Z}$ *and* $\mathbb{K}_{p^n}$*, for each* $n \in \mathbb{N}$*, the finite K (unique except for isomorphism) örper with* p^n *elements.*

3.4.4 Exponents and Logarithms

Bibliography

Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, pages 382–401, July 1982. URL `https://www.microsoft.com/en-us/research/publication/byzantine-generals-problem/`.