Computer algebra

Friedrich-Schiller-Universität Jena, SS 2005

Jürgen Müller

Contents

1 Computational complexity

We introduce the standard model of algorithmic computing, namely performing operations on finite strings of symbols out of a finite alphabet, which are thought of as being written onto an infinite tape, using a machine running back and forth on the tape, reading and writing symbols according to a specified program.

By **Church's Hypothesis** this idea precisely covers the intuitive notion of algorithmic computability. One of the early occurrences of this type of questions is **Hilbert's 10th problem** on the decidability of the solubility of Diophantine equations, see [6]; it was solved to the negative by Matijasevich (1972).

(1.1) Definition. An **alphabet** is a finite set $\mathcal{X} \neq \emptyset$. The free monoid $\mathcal{X}^* := \bigcup_{n \in \mathbb{N}_0} \mathcal{X}^n$ is called the set of **words** over $\mathcal{X}$. A subset $\mathcal{L} \subseteq \mathcal{X}^*$ is called a **(formal) language**. For $w \in \mathcal{X}^n$ let $l(w) = n \in \mathbb{N}_0$ be the **length** of w. We have $\mathcal{X}^0 = \{\epsilon\}$, where ϵ is the **empty** word.

(1.2) Definition. See [16].
A **(deterministic) Turing machine** over an alphabet $\mathcal{X}$ is a triple $\mathcal{T} := [\mathcal{X} \,\dot\cup\, \mathcal{Y}, \mathcal{S}, \tau]$, where $\mathcal{Y}$ is a **working** alphabet, in particular containing a **blank symbol** $\_ \in \mathcal{Y}$, an **accepting** symbol $\underline{1}$, a **rejecting symbol** $\underline{0}$, and a failure symbol $\underline{\emptyset}$. Moreover, $\mathcal{S}$ is a finite set of **states**, in particular containing an **initial state** s_0 and a **halting state** s_∞, and there is the **transition function**

$$\tau \colon (\mathcal{X} \,\dot\cup\, \mathcal{Y}) \times (\mathcal{S} \setminus \{s_\infty\}) \longrightarrow (\mathcal{X} \,\dot\cup\, \mathcal{Y}) \times \{\leftarrow, \uparrow, \rightarrow\} \times \mathcal{S}.$$

$\mathcal{T}$ acts on the set $(\mathcal{X} \,\dot\cup\, \mathcal{Y})^* \times \mathcal{S} \times (\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$ of **configurations** as follows: The **initial configurations** are given as $[\_, s_0, u]$, where $u \in \mathcal{X}^*$ is called an **input**; an input of several $u_1, \ldots, u_n \in \mathcal{X}^*$ is encoded as $u_1\_u_2\_\ldots\_u_n \in (\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$.

Let $[v, s, w]$ be a configuration, where $s \in \mathcal{S} \setminus \{s_\infty\}$. If $\epsilon \neq v, w \in (\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$, let $v = v'x$ and $w = yw'$, where $x, y \in \mathcal{X} \,\dot\cup\, \mathcal{Y}$; if $v = \epsilon$, let $v' := \epsilon$ and $x := \_$; if $w = \epsilon$, let $w' := \epsilon$ and $y := \_$. Then $\mathcal{T}$ induces the **transition**

$$[v, s, w] \mapsto \begin{cases} [v, \quad s', \quad zw'], & \text{if } \tau(y, s) = [z, \uparrow, s'], \\ [vz, \quad s', \quad w'], & \text{if } \tau(y, s) = [z, \rightarrow, s'], \\ [v', \quad s', \quad xzw'], & \text{if } \tau(y, s) = [z, \leftarrow, s']. \end{cases}$$

For a configuration $[v, s_\infty, w]$ no transition is defined and $\mathcal{T}$ **halts**. We assume that for all inputs leading to such a halting configuration we are in one and the same of the following cases, depending on whether we consider **decision problems** or **function problems**: Either we have $w \in \underline{1}(\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$ or $w \in \underline{0}(\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$, i. e. $\mathcal{T}$ **accepts** or **rejects** the input, respectively; or we have $w \in \underline{\emptyset}(\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$ or $w \in w'\_(\mathcal{X} \,\dot\cup\, \mathcal{Y})^*$, where $w' \in \mathcal{X}^*$, i. e. $\mathcal{T}$ **fails** or **outputs** w'.

(1.3) Example. Let $\mathcal{X} := \{1\}$ and $\mathcal{S} := \{s_0, s_1, s_\infty\}$, and let $\mathcal{T}$ be given by the following transition function τ:

τ	$\_$	1
s_0	$[1, \leftarrow, s_1]$	$[1, \rightarrow, s_0]$
s_1	$[\_, \rightarrow, s_\infty]$	$[1, \leftarrow, s_1]$

Hence e. g. upon input $11 \in \mathcal{X}^2$ we obtain:

$$\_\,\_\boxed{s_0}11\_\,\_ \quad\mapsto\quad \_\,\_1\boxed{s_0}1\_\,\_ \quad\mapsto\quad \_\,\_11\boxed{s_0}\_\,\_ \quad\mapsto$$

$$\_\,\_1\boxed{s_1}11\_ \quad\mapsto\quad \_\,\_\boxed{s_1}111\_ \quad\mapsto\quad \_\boxed{s_1}\_\,111\_ \quad\mapsto\quad \_\,\_\boxed{s_\infty}111\_\,,$$

and for $\epsilon \in \mathcal{X}^0$ we obtain:

$$\_\,\_\boxed{s_0}\_\,\_ \quad\mapsto\quad \_\boxed{s_1}\_\,1\_ \quad\mapsto\quad \_\,\_\boxed{s_\infty}1\_\,.$$

Thus it is easily seen that $\mathcal{T}$ computes the successor function $\mathbb{N}_0 \to \mathbb{N}\colon n \mapsto n+1$, where $\mathbb{N}_0$ is given in unary encoding. For the successor function in binary encoding, see Exercise (8.1).

(1.4) Definition. a) A language $\mathcal{L} \subseteq \mathcal{X}^*$ is called **decidable (recursive)**, if there is a Turing machine $\mathcal{T}$ **deciding** $\mathcal{L}$, i. e. $\mathcal{T}$ halts for all $w \in \mathcal{X}^*$, and accepts w if and only if $w \in \mathcal{L}$, otherwise rejects w.

A language $\mathcal{L} \subseteq \mathcal{X}^*$ is called **recursively enumerable**, if there is a Turing machine $\mathcal{T}$ **accepting** $\mathcal{L}$, i. e. $\mathcal{T}$ halts for $w \in \mathcal{X}^*$ if and only if $w \in \mathcal{L}$.

Note that if $\mathcal{L}$ is decidable, then it is recursively enumerable: Let $\mathcal{T}$ decide $\mathcal{L}$, then $\mathcal{T}'$ accepting $\mathcal{L}$ is a copy of $\mathcal{T}$, except that whenever $\mathcal{T}$ rejects an input, then $\mathcal{T}'$ enters an infinite loop.

b) A Turing machine $\mathcal{T}$ deciding a language $\mathcal{L} \subseteq \mathcal{X}^*$ is called to run in **time** $f\colon \{N, N+1, \ldots\} \to \mathbb{R}_{>0}$, if $\mathcal{T}$ halts after at most $f(l(w))$ transitions, for all $w \in \mathcal{X}^*$ such that $l(w) \geq N$. The **complexity class** $\mathsf{TIME}(f) \subseteq \mathrm{Pot}(\mathcal{X}^*)$ is the set of all languages being decidable in time f.

In particular, we have the complexity class $\mathsf{P} := \bigcup_{k\in\mathbb{N}} \mathsf{TIME}(n \mapsto n^k)$ of languages being decidable in **polynomial time**, and the complexity class $\mathsf{EXP} := \bigcup_{k\in\mathbb{N}} \mathsf{TIME}(n \mapsto c^{n^k})$ of languages being decidable in **exponential time**, where $c > 1$; note that this does not depend on the choice of $c > 1$.

(1.5) Definition. a) A **non-deterministic Turing machine** over an alphabet $\mathcal{X}$ is a triple $\mathcal{T} := [\mathcal{X} \,\dot\cup\, \mathcal{Y}, \mathcal{S}, \tau]$, where $\mathcal{X} \,\dot\cup\, \mathcal{Y}$ and $\mathcal{S}$ are as in (1.2), while the transition function

$$\tau\colon (\mathcal{X} \,\dot\cup\, \mathcal{Y}) \times (\mathcal{S} \setminus \{s_\infty\}) \longrightarrow \mathrm{Pot}\Big((\mathcal{X} \,\dot\cup\, \mathcal{Y}) \times \{\leftarrow, \uparrow, \rightarrow\} \times \mathcal{S}\Big)$$

allows for **choices** and thus **branching**. Let the **non-determinateness** be defined as $d_{\mathcal{T}} := \max\{|\tau(x,s)|; x \in \mathcal{X} \,\dot\cup\, \mathcal{Y}, s \in \mathcal{S} \setminus \{s_\infty\}\} \in \mathbb{N}$. The machine

$\mathcal{T}$ halts if no further transition in either branch is possible. We assume that for all inputs $\mathcal{T}$ on halting either accepts or rejects, or outputs; for acceptance, rejection or the output one of the branches is chosen randomly.

b) A language $\mathcal{L} \subseteq \mathcal{X}^*$ is called **non-deterministically decidable**, if there is a non-deterministic Turing machine $\mathcal{T}$ **deciding** $\mathcal{L}$, i. e. $\mathcal{T}$ halts for all $w \in \mathcal{X}^*$, and we have $w \in \mathcal{L}$ if and only if there is a branch accepting w, otherwise all branches reject w; note the asymmetry in the treatment of acceptance and rejection. The complexity class $\mathsf{NTIME}(f)$ is the set of all languages being non-deterministically decidable in time f.

In particular, we have the complexity class $\mathsf{NP} := \bigcup_{k\in\mathbb{N}} \mathsf{NTIME}(n \mapsto n^k)$ of languages being decidable in **non-deterministic polynomial time**. Let coNP be the complexity class of languages $\mathcal{L} \subseteq \mathcal{X}^*$ such that $(\mathcal{X}^* \setminus \mathcal{L}) \in \mathsf{NP}$. Analogously, let coP be the complexity class of languages $\mathcal{L} \subseteq \mathcal{X}^*$ such that $(\mathcal{X}^* \setminus \mathcal{L}) \in \mathsf{P}$. Obviously we have $\mathsf{coP} = \mathsf{P} \subseteq \mathsf{NP} \cap \mathsf{coNP}$.

The most outstanding open problem of computational complexity theory is the **Conjecture:** We have $\mathsf{P} \neq \mathsf{NP}$ and $\mathsf{NP} \neq \mathsf{coNP}$ as well as $\mathsf{P} \neq \mathsf{NP} \cap \mathsf{coNP}$.

(1.6) Proposition. We have $\mathsf{NTIME}(f) \subseteq \bigcup_{c>1} \mathsf{TIME}(n \mapsto c^{f(n)})$. Thus in particular we have $\mathsf{NP} \subseteq \mathsf{EXP}$.

Proof. See [13, Thm.2.6] or Exercise (8.3). $\sharp$

(1.7) Proposition. Given a language $\mathcal{L} \subseteq \mathcal{X}^*$, where $|\mathcal{X}| \geq 2$, then we have $\mathcal{L} \in \mathsf{NP}$ if and only if there is a relation $\mathcal{R} \subseteq \mathcal{X}^* \times \mathcal{X}^*$ such that:
i) We have $\mathcal{L} = \{w \in \mathcal{X}^*; [w, v] \in \mathcal{R} \text{ for some } v \in \mathcal{X}^*\}$.
ii) There is $k \in \mathbb{N}$ such that $l(v) \leq l(w)^k$, for all $[w, v] \in \mathcal{R}$.
iii) Letting $\mathcal{L}_\mathcal{R} := \{w\_v; [w, v] \in \mathcal{R}\} \subseteq \mathcal{X}^*\_\mathcal{X}^*$, we have $\mathcal{L}_\mathcal{R} \in \mathsf{P}$.

Given $w \in \mathcal{L}$, an element $v \in \mathcal{X}^*$ such that $[w, v] \in \mathcal{R}$ is called a **polynomial certificate** for w.

Proof. Let $\mathcal{R}$ be as in the assertion. Then $\mathcal{L}$ is decided by a non-deterministic Turing machine, which for $w \in \mathcal{X}^*$ first finds a certificate $v \in \mathcal{X}^*$ of polynomial length $l(v) \leq l(w)^k$, hence in polynomial time, and then decides in polynomial time whether $[w, v] \in \mathcal{R}$. Hence we have $\mathcal{L} \in \mathsf{NP}$.

Conversely, let $\mathcal{L} \in \mathsf{NP}$ be decided by the non-deterministic Turing machine $\mathcal{T}$, running in polynomial time and having non-determinateness $d_\mathcal{T}$. Each finite sequence choices of $\mathcal{T}$ can be encoded $d_\mathcal{T}$-adically into an element of $\mathbb{N}_0$, and hence $|\mathcal{X}|$-adically into an element of $\mathcal{X}^*$. Thus we define $\mathcal{R} \subseteq \mathcal{X}^* \times \mathcal{X}^*$ by letting $[w, v] \in \mathcal{R}$ if and only if $v \in \mathcal{X}^*$ is the encoding of a sequence of choices of an accepting computation for $w \in \mathcal{X}^*$. Hence by construction of $\mathcal{R}$ we have i) and ii). Moreover, for w_v it can be checked in linear time whether v indeed encodes an accepting computation for w, hence we also have iii). $\sharp$

(1.8) Definition. a) A **(one-sided) Monte-Carlo machine** for a language $\mathcal{L} \subseteq \mathcal{X}^*$ is a non-deterministic Turing machine $\mathcal{T}$ halting for all $w \in \mathcal{X}^*$, having an **error bound** $0 < \epsilon < 1$ such that $\mathcal{T}$ accepts $w \in \mathcal{L}$ in at least a fraction of ϵ of the branches, while $\mathcal{T}$ rejects $w \notin \mathcal{L}$ in all branches.

Hence acceptance is correct, but rejection might be incorrect with an error probability $1 - \epsilon$. Note that we may fix an error bound $0 < \epsilon_0 < 1$ a priorly: If $\epsilon < \epsilon_0$, then $\mathcal{T}$ is repeated k times, until $(1 - \epsilon)^k \leq (1 - \epsilon_0)$.

b) The complexity class RP of languages being decidable in **randomized polynomial time** is the set of languages possessing a Monte-Carlo machine running in polynomial time. Hence we have $\mathsf{P} \subseteq \mathsf{RP} \subseteq \mathsf{NP}$.

Let coRP be the complexity class of languages $\mathcal{L} \subseteq \mathcal{X}^*$ such that $(\mathcal{X}^* \setminus \mathcal{L}) \in \mathsf{RP}$. Let $\mathsf{ZPP} := \mathsf{RP} \cap \mathsf{coRP}$ be the complexity class of languages being decidable in randomized polynomial time with **zero probability error**:

For $\mathcal{L} \in \mathsf{ZPP}$ let $\mathcal{T}'$ and $\mathcal{T}''$ be Monte-Carlo machines for $\mathcal{L}$ and $\mathcal{X}^* \setminus \mathcal{L}$, respectively, both with error bound $0 < \epsilon < 1$. A **Las-Vegas machine** for $\mathcal{L}$ is a non-deterministic Turing machine $\mathcal{T}$ defined as follows: $\mathcal{T}$ runs both $\mathcal{T}'$ and $\mathcal{T}''$, if $\mathcal{T}'$ accepts then $\mathcal{T}$ accepts, if $\mathcal{T}''$ accepts then $\mathcal{T}$ rejects, and otherwise repeats this. Hence it is not guaranteed that $\mathcal{T}$ halts, but if it halts then the answer is correct. Moreover, $\mathcal{T}$ halts after at most k repetitions with a probability of at least $1 - (1 - \epsilon)^k$.

c) The complexity class BPP of languages being decidable in polynomial time with **bounded probability error** is the set of languages possessing a non-deterministic Turing machine $\mathcal{T}$, called a **two-sided Monte-Carlo machine**, running in polynomial time halting for all $w \in \mathcal{X}^*$, having an error bound $\frac{1}{2} < \epsilon < 1$ such that $\mathcal{T}$ accepts $w \in \mathcal{L}$ in at least a fraction of ϵ of the branches, and $\mathcal{T}$ rejects $w \notin \mathcal{L}$ in at least a fraction of ϵ of the branches.

Note that we may fix an error bound $\frac{1}{2} < \epsilon_0 < 1$ a priorly, by running $\mathcal{T}$ repeatedly, k times say, and accepting an input if and only if it is accepted by a strict majority of the runs. This is seen as follows: The i-th run of $\mathcal{T}$ on $w \in \mathcal{L}$ is considered as a random variable X_i assuming the values 1 and 0 with probability ϵ and $1 - \epsilon$, respectively. Letting $X := \sum_{i=1}^k X_i$ be the sum of the independent random variable X_i, rejection is equivalent to $X \leq \frac{k}{2}$, or equivalently $\sum_{i=1}^k (1 - X_i) = k - X \geq \frac{k}{2}$; note that $E(1 - X_i) = 1 - \epsilon$. Let $\vartheta := \min\{1, \frac{2\epsilon - 1}{2(1-\epsilon)}\}$, hence $0 < \vartheta \leq 1$; note that $\frac{2\epsilon-1}{2(1-\epsilon)} \geq 1$ if and only if $\epsilon \geq \frac{3}{4}$. This yields $(1 + \vartheta)(1 - \epsilon) \leq \frac{1}{2}$, and thus by Chernoff's bound, see Exercise (8.4), we have $P[k - X \geq \frac{k}{2}] \leq P[k - X \geq (1 + \vartheta)(1 - \epsilon)k] \leq e^{-\frac{1}{2}\vartheta^2(1-\epsilon)k}$. Hence we have $P[k - X \geq \frac{k}{2}] \leq 1 - \epsilon_0$ whenever $k \geq \frac{-2\ln(1-\epsilon_0)}{\vartheta^2(1-\epsilon)}$.

Finally, we have $\mathsf{BPP} = \mathsf{coBPP}$ and $\mathsf{RP} \cup \mathsf{coRP} \subseteq \mathsf{BPP}$, while it is conjectured that $\mathsf{BPP} \not\subseteq \mathsf{NP}$ holds.

(1.9) Definition. a) Let $\mathcal{X}$ be an alphabet and let $\mathcal{R} \subseteq \mathcal{X}^* \times \mathcal{X}^*$ be a relation. The **function problem** associated to $\mathcal{R}$ is, given $w \in \mathcal{X}^*$, find a **solution**

$v \in \mathcal{X}^*$ such that $[w, v] \in \mathcal{R}$, if such a v exists at all, otherwise report failure. A Turing machine $\mathcal{T}$ **solves** the function problem $\mathcal{R}$, if $\mathcal{T}$ halts for all $w \in \mathcal{X}^*$, and outputs a solution, if any solution exists at all, and fails otherwise.

There are straightforward notions of running time and complexity classes. Moreover, this immediately generalizes to non-deterministic Turing machines, and there is a straightforward notion of **Monte-Carlo machines** for function problems. Note that the latter are usually called **Las-Vegas machines** for function problems, which does not seem appropriate, since the straightforward notion of Las-Vegas machines does not make sense.

b) The function problems associated to $\mathcal{L} \in \mathsf{NP}$ are the function problems associated to the polynomial certificate relations $\mathcal{R}$ for $\mathcal{L}$. Let FNP be the complexity class of function problems associated to languages in NP. In particular, function problems in FNP are solvable by non-deterministic Turing machines running in polynomial time. Let $\mathsf{FP} \subseteq \mathsf{FNP}$ be the complexity class of function problems being solvable by Turing machines running in polynomial time; it is conjectured that $\mathsf{FP} \neq \mathsf{FNP}$ holds.

c) A language $\mathcal{L} \subseteq \mathcal{X}^*$ **reduces in polynomial time** to a language $\mathcal{L}' \subseteq \mathcal{X}^*$, if there is a function problem in FP, associated to a relation $\mathcal{R} \subseteq \mathcal{X}^* \times \mathcal{X}^*$, such that for all $w \in \mathcal{X}^*$ there is $v \in \mathcal{X}^*$ such that $[w, v] \in \mathcal{R}$, i. e. failure does not occur, and for all $[w, v] \in \mathcal{R}$ we have $w \in \mathcal{L}$ if and only if $v \in \mathcal{L}'$.

A Turing machine deciding $\mathcal{L}'$ is called an **oracle** for $\mathcal{L}$. Languages $\mathcal{L}$ and $\mathcal{L}'$ are called **polynomial time equivalent**, if $\mathcal{L}$ reduces in polynomial time to $\mathcal{L}'$ and vice versa. Given a complexity class C of languages, $\mathcal{L}' \in \mathsf{C}$ is called **C-complete** if each $\mathcal{L} \in \mathsf{C}$ reduces in polynomial time to $\mathcal{L}'$. Given a complexity class C of languages, a function problem is called **C-hard** if each $\mathcal{L} \in \mathsf{C}$ reduces in polynomial time to that function problem.

2 Integer arithmetic

(2.1) Definition. Let $D \subseteq \mathbb{N}_0$ such that $\{N, N+1, \ldots\} \subseteq D$ for some $N \in \mathbb{N}_0$, and let $f\colon D \to \mathbb{R}$ be an **eventually positive function**, i. e. we have $f(n) > 0$ for all $n \geq N$.

Let $O(f)$ be the set of eventually positive functions $g\colon D \to \mathbb{R}$ such that the sequence $[\frac{g(n)}{f(n)}; n \geq N] \subseteq \mathbb{R}_{>0}$ is bounded. Let $o(f)$ be the set of eventually positive functions $g\colon D \to \mathbb{R}$ such that $\lim_{n\to\infty} \frac{g(n)}{f(n)} = 0$; hence $g \in O(f)$. These symbols are also called **Landau symbols**.

Let $g\colon D \to \mathbb{R}$ be an eventually positive function. Then the functions g and f are called **asymptotically equivalent** $g \sim f$, if $\lim_{n\to\infty} \frac{g(n)}{f(n)} = 1$; hence $f \in O(g)$ and $g \in O(f)$.

For $h\colon \mathbb{R} \to \mathbb{R}$ let $h(O(f)) := \{h \circ g\colon D \to \mathbb{R}; g \in O(f)\}$ and $h(o(f)) := \{h \circ g\colon D \to \mathbb{R}; g \in o(f)\}$. Moreover, we use a straightforwardly generalized notation for functions in several variables, or for functions defined on subsets of $\mathbb{R}$ unbounded to the right.

E. g. we have Stirling's formula $\lim_{n\to\infty} \frac{n!\cdot e^n}{n^n\cdot\sqrt{2\pi n}} = 1$, see [5, Formula 96.2], and thus $n! \sim (\frac{n}{e})^n \cdot \sqrt{2\pi n}$. Hence we have $\ln(n!) \sim n(\ln(n)-1)+\frac{1}{2}\cdot\ln(n)+\ln(\sqrt{2\pi})$, and thus $\ln(n!) \sim n\ln(n)$.

E. g. letting $\pi(n) := |\{p \in \mathbb{N}; p \leq n, p \text{ prime}\}| \in \mathbb{N}_0$, for $n \in \mathbb{N}$, by the Prime Number Theorem, see [4, Ch.22], we have $\pi(n) \sim \frac{n}{\ln(n)}$.

(2.2) Definition. The number of digits to the base $1 \neq z \in \mathbb{N}$ necessary to represent $n = \sum_{i=0}^{b} n_i z^i \in \mathbb{N}$, where $n_i \in \{0,\ldots,z-1\}$, is given as the **bit length** $b_z(n) := 1+b = 1+\lfloor \log_z(n)\rfloor = 1+\lfloor\frac{\ln(n)}{\ln(z)}\rfloor$, where $\lfloor\cdot\rfloor$ denote lower Gaussian brackets. Note that for $n \in \mathbb{Z}$ we only need an additional sign. Hence for the input length of $n \in \mathbb{Z}$ into a Turing machine we have $1+b_z(|n|) \in O(\ln(n))$.

The computational complexity of integer arithmetic is counted in **bit operations**, i. e. and, or, exclusive or, not and shift on bit strings, hence for the base $z=2$. More generally, typical generalized bit operations could be **Byte operations**, **word operations** and **long word operations** with respect to the bases $z=2^8$, $z=2^{32}$ and $z=2^{64}$, respectively. Using multiple string Turing machines, see Exercise (8.2), it is easily seen that the time needed for these operations indeed is polynomial in the input length $1+b_z(|n|)$.

Hence we treat bit operations as oracles. An algorithm using integer arithmetic, whose input up to sign is $n \in \mathbb{N}$, is called an **$L_{\alpha,c}$-time algorithm**, for $0 \leq \alpha \leq 1$ and $c > 0$, if it needs $L_{\alpha,c} := O(e^{c(\ln(n))^\alpha(\ln(\ln(n)))^{1-\alpha}})$ bit operations. Hence for $\alpha = 0$ we have $L_{\alpha,c} = O(\ln^c(n))$, thus the algorithm runs in polynomial time with respect to bit operations, and thus is a polynomial time algorithm. Moreover, for $\alpha = 1$ we have $L_{\alpha,c} = O(e^{c\ln(n)}) = O(n^c)$, thus the algorithm runs in exponential time with respect to bit operations, and thus is a exponential time algorithm. Finally, for $0 < \alpha < 1$ we have $cx^\alpha \ln^{1-\alpha}(x) \in o(x)$, see Exercise (8.5), thus the algorithm runs in **subexponential time** with respect to bit operations, i. e. it needs $O(e^{h(\ln(n))})$ bit operations, for some eventually positive function $h(x) \in o(x)$.

(2.3) Algorithm: Ring operations.
These in general are addition, subtraction and multiplication, as well as division by units; the latter of course do not play a role for integers.

a) Addition is described as follows: Let $n \geq m \in \mathbb{N}$ and $b := b_z(n)$, for some $1 \neq z \in \mathbb{N}$. Hence we have $n = \sum_{i=0}^{b-1} n_i z^i$, and we may assume $m = \sum_{j=0}^{b-1} m_j z^j$, by letting $m_j := 0$ for $j \in \{b_z(m),\ldots,b-1\}$:

```
1. $\delta \leftarrow 0$.
2. for $k \in [0,\ldots,b-1]$ do
     $s_k \leftarrow n_k + m_k + \delta$
```

```
    if $s_k \geq z$ then
        $s_k \leftarrow s_k - z$
        $\delta \leftarrow 1$
    else $\delta \leftarrow 0$
3. $s_b \leftarrow \delta$
4. return $[s_0, \ldots, s_b]$
```

Hence we have $n+m = \sum_{k=0}^{b} s_k z^k$. For each k this needs a fixed number of bit operations, and hence needs $O(b_z(n)) = O(\ln(n))$ bit operations. For subtraction see Exercise (8.7); again this needs $O(\max\{b_z(n), b_z(m)\})$ bit operations.

b) Multiplication is described as follows: Let $n, m \in \mathbb{N}$ and $b_n := b_z(n)$ as well as $b_m := b_z(m)$. Hence we have $nm = \sum_{i=0}^{b_n-1} \sum_{j=0}^{b_m-1} n_i m_j z^{i+j} = \sum_{k=0}^{b_n+b_m-1} (\sum_{l=\max\{0,k-b_m+1\}}^{\min\{b_n-1,k\}} n_l m_{k-l}) \cdot z^k$. Using the former formula yields:

```
1. for $k \in [0, \ldots, b_n + b_m - 1]$ do $s_k \leftarrow 0$
2. for $i \in [0, \ldots, b_n - 1]$ do
    $\delta \leftarrow 0$
    for $j \in [0, \ldots, b_m - 1]$ do
        $s \leftarrow s_{i+j} + n_i m_j + \delta$        # $s = (s \mod z) + \lfloor \frac{s}{z} \rfloor \cdot z$
        $s_{i+j} \leftarrow s \mod z$
        $\delta \leftarrow \lfloor \frac{s}{z} \rfloor$
    $s_{i+b_m} \leftarrow \delta$
3. return $[s_0, \ldots, s_{b_n+b_m-1}]$
```

Hence we have $nm = \sum_{k=0}^{b_n+b_m-1} s_k z^k$. For each i and j this needs a fixed number of bit operations, thus needs $O(b_z(n) b_z(m)) = O(\ln(n)\ln(m))$ bit operations.

(2.4) Algorithm: Karatsuba (1962).
Let $k \in \mathbb{N}_0$ and $b = 2^k$, as well as $1 \neq z \in \mathbb{N}$ and $m, n \in \mathbb{N}$ such that $m, n < z^b$, hence we have $b_z(m), b_z(n) \leq b$. Let $m = m' \cdot z^{\frac{b}{2}} + m''$ and $n = n' \cdot z^{\frac{b}{2}} + n''$, where $0 \leq m', m'', n', n'' < z^{\frac{b}{2}}$. Then we have $m \cdot n = m'n'z^b + (m'n'' + m''n') \cdot z^{\frac{b}{2}} + m''n''$, where $m'n'' + m''n' = m'n' + m''n'' + (m' - m'')(n'' - n')$, and in particular $|m' - m''|, |n' - n''| < z^{\frac{b}{2}}$. Let $K(m, n, b)$ be defined as follows:

```
1. if $b = 1$ then return $mn$
2. if $b > 1$ then
    $r \leftarrow K(m', n', \frac{b}{2})$
    $s \leftarrow K(m'', n'', \frac{b}{2})$
    $t \leftarrow K(|m' - m''|, |n' - n''|, \frac{b}{2})$
3. return $rz^b + (r + s \pm t) \cdot z^{\frac{b}{2}} + s$
```

Hence by induction with respect to $b \in \mathbb{N}$ we have $K(m, n, b) = mn$. We show that this **divide and conquer** technique needs $O(b^{\log_2(3)})$ bit operations. Since we may assume that $n \geq m$ and $\frac{b}{2} < b_z(n) \leq b$, this amounts to $O((2\ln(n))^{\log_2(3)}) = O((\ln(n))^{\log_2(3)})$ bit operations; note that $\log_2(3) < \frac{159}{100}$:

Let $\kappa(b) \in \mathbb{N}$ be the number of bit operations needed to compute $K(\cdot, \cdot, b)$.

Then we have $\kappa(1) = 1$, and for $b > 1$ we have 3 calls of $K(\cdot, \cdot, \frac{b}{2})$ as well as additions and shifts, thus $\kappa(b) = 3\kappa(\frac{b}{2}) + \gamma b$, for some $\gamma > 0$. By induction we get $\kappa(b) = 3^k \cdot \kappa(\frac{b}{2^k}) + \gamma b \cdot \sum_{i=0}^{k-1} (\frac{3}{2})^i = 3^k + \gamma 2^k \cdot \frac{(\frac{3}{2})^k - 1}{\frac{3}{2} - 1} = 3^k + 2\gamma \cdot (3^k - 2^k) = (2\gamma + 1) \cdot 3^k - \gamma 2^{k+1}$. Hence we have $\kappa(b) \in O(3^k) = O(3^{\log_2(b)}) = O((2^{\log_2(3)})^{\log_2(b)}) = O(b^{\log_2(3)})$. ♯

(2.5) Algorithm: Quotient and remainder.
Let $m \geq n \in \mathbb{N}$, hence there are unique $q, r \in \mathbb{N}_0$ such that $r < n$ and $m = qn + r$. Note that in particular to compute in the ring $\mathbb{Z}/\langle n \rangle$ we need the computation of remainders.

Let $b' := b_z(m)$ and $b'' := b_z(n)$, for some $1 \neq z \in \mathbb{N}$. Replacing $[m, n]$ by a suitable multiple $[km, kn]$, for some $1 \leq k < z$, we may assume that $n_{b''-1} \geq \lfloor \frac{z}{2} \rfloor$. Moreover, after replacing n by nz^l for some $l \in \mathbb{N}_0$, i. e. after a suitable shift, we may assume that we have $b_z(n) = b$ and $b_z(m) \in \{b, b+1\}$, where $b \in \{b', b'+1\}$. To compute q, we compute $q' := \min\{\lfloor \frac{m_b z + m_{b-1}}{n_{b-1}} \rfloor, z - 1\}$. We show that $q' - 2 \leq q \leq q'$:

We have $n_{b-1} q' \geq m_b z + m_{b-1} - (n_{b-1} - 1)$. Hence $m - q'n \leq m - q' n_{b-1} z^{b-1} \leq m - (m_b z + m_{b-1}) z^{b-1} + (n_{b-1} - 1) z^{b-1} = (n_{b-1} - 1) z^{b-1} + \sum_{j=0}^{b-2} m_j z^j < n_{b-1} z^{b-1} \leq n$. As we have $q \leq z - 1$ anyway, we conclude $q \leq q'$. Moreover, we have $q' \leq \frac{m}{n_{b-1} z^{b-1}} < \frac{m}{n - z^{b-1}}$ and $q = \lfloor \frac{m}{n} \rfloor > \frac{m}{n} - 1$. Assume to the contrary that $3 \leq q' - q < \frac{m}{n - z^{b-1}} - (\frac{m}{n} - 1) = \frac{m \cdot z^{b-1}}{n(n - z^{b-1})} + 1$. Thus we have $\frac{m}{n} > 2 \cdot (n_{b-1} - 1)$, and hence $z - 4 \geq q' - 3 \geq q = \lfloor \frac{m}{n} \rfloor \geq 2 \cdot (n_{b-1} - 1) \geq z - 3$, a contradiction. Thus $q' - 2 \leq q$. ♯

Computing $[km, kn]$ needs $O(b')$ bit operations, the shifts need $O(b''(b' - b''))$ bit operations. To compute the quotient q at most 3 trials are necessary, as $b_z(q') = 1$ the trial multiplication to compute $q'n$ needs $O(b) = O(b')$ bit operations, and the addition $r := m - qn$ as well needs $O(b')$ bit operations. This amounts to $O(\max\{b', b''(b' - b'')\})$ bit operations, where $b''(b' - b'') \geq b'$ whenever $b' > b''$; as $m \geq n$ this hence needs $O(\ln(m)\ln(n)) \subseteq O(\ln^2(m))$ bit operations.

(2.6) Algorithm: Binary modular exponentiation.
Let $e, n \in \mathbb{N}$ and $m \in \{0, \ldots, n-1\}$.

```
1. $r \leftarrow 1$
2. while $e > 0$ do
     if $1 \equiv e \bmod 2$ then $r \leftarrow rm \bmod n$
     $e \leftarrow \lfloor \frac{e}{2} \rfloor$.
     $m \leftarrow m^2 \bmod n$.
3. return $r$.
```

Using the binary representation of $e \in \mathbb{N}$ shows that $r \in \{0, \ldots, n-1\}$ such that $r \equiv m^e \bmod n$. Moreover, as $b_2(e) \in O(\ln(e))$ and both multiplication and computing remainders need $O(\ln^2(n))$ bit operations, we need $O(\ln(e) \cdot$

$\ln^2(n))$ bit operations; note that the classical exponentiation algorithm needs $O(e \cdot \ln^2(n))$ bit operations, and hence needs exponential time.

(2.7) Algorithm: Extended Euclidean algorithm.
Let $m, n \in \mathbb{N}$.

```
1. $r_0 \leftarrow m$, $s_0 \leftarrow 1$, $t_0 \leftarrow 0$
2. $r_1 \leftarrow n$, $s_1 \leftarrow 0$, $t_1 \leftarrow 1$
3. $i \leftarrow 1$
4. while $r_i \neq 0$ do
     $r_{i+1} \leftarrow r_{i-1} \bmod r_i$
     $q_i \leftarrow \lfloor \frac{r_{i-1}}{r_i} \rfloor$      # quotient and remainder
     $s_{i+1} \leftarrow s_{i-1} - q_i s_i$
     $t_{i+1} \leftarrow t_{i-1} - q_i t_i$
     $i \leftarrow i + 1$
5. return $[r_{i-1}, s_{i-1}, t_{i-1}]$
```

We have $r_0 = s_0 m + t_0 n$ and $r_1 = s_1 m + t_1 n$, and by induction on $i \geq 1$ we have $r_{i+1} = r_{i-1} - q_i r_i = (s_{i-1}m + t_{i-1}n) - q_i \cdot (s_i m + t_i n) = s_{i+1}m + t_{i+1}n$. As we have $r_i < r_{i-1}$ for all $i \geq 1$, the algorithm halts, after step $i := l + 1$ say, returning $[r_l, s_l, t_l] =: [d, s, t]$. The number l of steps needed is discussed in Exercise (8.10). Thus we have $d = sm + tn$, and hence $\gcd(m, n) \mid d$. Conversely, running the algorithm reversely shows that $d \mid r_i$ for all $i \geq 0$, hence $d \mid m, n$ and thus $d \mid \gcd(m, n)$. Thus $[d, s, t] \subseteq \mathbb{Z}$ such that $0 < d = \gcd(m, n) = sm + tn$; the elements $s, t \in \mathbb{Z}$ are called **Bezout coefficients**. Note that the computation of the coefficients s_i and t_i can be left out, the remaining algorithm is called the Euclidean algorithm.

Let $1 \neq z \in \mathbb{N}$. For $i \in \{1, \ldots, l\}$ we need $O(b_z(r_i) b_z(q_i))$ bit operations to compute $[q_i, r_i]$. As $b_z(q_i) = 1 + \lfloor \log_z(q_i) \rfloor$, we have $O(\sum_{i=1}^{l} b_z(q_i)) = O(b_z(\prod_{i=1}^{l} q_i)) \subseteq O(b_z(r_0))$. Hence computing the quotients and remainders needs $O(\sum_{i=1}^{l} b_z(r_i) b_z(q_i)) \subseteq O(b_z(r_1) \cdot \sum_{i=1}^{l} b_z(q_i)) \subseteq O(b_z(r_1) b_z(r_0))$ bit operations. To compute the linear combination needs $O(\sum_{i=1}^{l} b_z(q_i) b_z(s_i))$ bit operations, where in turn $b_z(s_i) \in O(b_z(s_{i-1}) + b_z(q_{i-1}))$, hence we have $b_z(s_i) \in O(\sum_{j=1}^{i-1} b_z(q_j))$, yielding $O(\sum_{i=1}^{l} \sum_{j=1}^{i-1} b_z(q_i) b_z(q_j))$ bit operations. As above we from this obtain $O(\sum_{j=1}^{l-1} \sum_{i=j+1}^{l} b_z(q_j) b_z(q_i)) \subseteq O(\sum_{j=1}^{l-1} b_z(q_j) b_z(r_j)) \subseteq O(b_z(r_1) \cdot \sum_{j=1}^{l-1} b_z(q_j)) \subseteq O(b_z(r_1) b_z(r_0))$ bit operations. Thus this needs $O(b_z(r_1) b_z(r_0)) = O(b_z(m) b_z(n))$ bit operations; if $m \geq n$ this hence needs $O(\ln^2(m))$ bit operations.

(2.8) Remark: Polynomial arithmetic.
Let R be a commutative ring and let $R[X]$ be the polynomial ring over R in the indeterminate X. For $0 \neq f = \sum_{i=0}^{d} f_i X^i \in R[X]$, where $f_i \in R$ and $f_d \neq 0$, let $\deg(f) := d$ denote its **degree**. The computational complexity of polynomial arithmetic is usually measured in ring operations in R, related to the degrees of the polynomials in $R[X]$ involved. Hence in general this is not directly related

to the number of bit operations needed, since coefficient growth in R has to be taken into account, e. g. for $R = \mathbb{Z}$; it directly relates to the number of bit operations needed in the case of a finite ring R, e. g. for $\mathbb{Z}/\langle n\rangle$ or for finite fields $\mathbb{F}_q$. The algorithms for integer arithmetic straightforwardly generalize to polynomial arithmetic by letting $z := X$, and even have a tendency to become slightly easier, see Exercise (8.16):

Let $0 \neq f, g \in R[X]$, where $\deg(f) \geq \deg(g)$. Addition $f + g$ and subtraction $f - g$ need $O(\deg(f))$ ring operations, while multiplication $f \cdot g$, using the classical technique, needs $O(\deg(f)^2)$ ring operations. The Karatsuba algorithm generalizes to multiplication $f \cdot g$, where $\deg(f), \deg(g) < 2^k$ for some $k \in \mathbb{N}_0$; hence if $\deg(f) \geq \deg(g)$ it needs $O(\deg(f)^{\log_2(3)})$ ring operations. Given $e \in \mathbb{N}$ and assuming $\deg(f) \leq \deg(g)$, to compute $r \in R[X]$ such that $r = 0$ or $\deg(r) < \deg(g)$, and $f^e \equiv r \bmod g$, needs $O(\ln(e) \cdot \deg(g)^2)$ ring operations.

Let $0 \neq g \in R[X]$ such that its leading coefficient $\mathrm{lc}(g) := g_{\deg(g)} \in R^*$ is a unit in R. Hence for $f \in R[X]$ there exist unique $q, r \in R[X]$ such that $r = 0$ or $\deg(r) < \deg(g)$, fulfilling $f = qg + r$. We may assume $\deg(f) \geq \deg(g)$, hence to compute $[q, r] \subseteq R[X]$ needs $O(\deg(f) \cdot (\deg(f) - \deg(g))) \subseteq O(\deg(f)^2)$ ring operations in R; note that only $g_{\deg(g)} \in R^*$ has to be inverted, and that q can be computed without guessing. Finally, $R[X]$ is Euclidean if and only if R is a field; in this case the extended Euclidean algorithm generalizes to $0 \neq f, g \in R[X]$, and needs $O(\deg(f) \cdot \deg(g))$ ring operations.

3 Fast Fourier transform

(3.1) Definition. Let R be a commutative ring and let $n \in \mathbb{N}$. An element $\omega \in R$ is called a **primitive n-th root of unity**, if $\omega^n = 1$ and $\omega^k - 1$ is neither 0 nor a zero-divisor in R, for all $k \in \{1, \ldots, n-1\}$.

Note that $\omega \in R^*$, and if R is an integral domain, then the condition on $\omega^k - 1$ amounts to $\omega^k \neq 1$, for all $k \in \{1, \ldots, n-1\}$. E. g. $\zeta_n := e^{\frac{2\pi\sqrt{-1}}{n}} \in \mathbb{C}$ is the **standard** primitive n-th root of unity in $\mathbb{C}$. E. g. if $q \in \mathbb{N}$ is a prime power, then by Artin's Theorem we have $\mathbb{F}_q^* \cong \mathbb{Z}/\langle q-1\rangle$, which has an element of order n if and only if $n \mid q-1$, thus the finite field $\mathbb{F}_q$ has a primitive n-th root of unity if and only if $n \mid q-1$.

E. g. the ring $(\mathbb{Z}/\langle 8\rangle)$ does not have primitive square roots of unity: We have $(\mathbb{Z}/\langle 8\rangle)^* = \{\pm 1, \pm 3\} \cong (\mathbb{Z}/\langle 2\rangle)^2$, hence for all $1 \neq u \in (\mathbb{Z}/\langle 8\rangle)^*$ we have $u^2 = 1$, but $u - 1$ is a zero-divisor.

(3.2) Lemma. Let R be a commutative ring and let $n \in \mathbb{N}$.
a) Let $\omega \in R$ such that $\omega^n = 1$ and $\omega^{\frac{n}{p}} - 1$ is neither 0 nor a zero-divisor, for all prime divisors $p \mid n$. Then ω is a primitive n-th root of unity.
b) Let $\omega \in R$ be a primitive n-th root of unity, let $k \in \{1, \ldots, n-1\}$ and $m := \frac{n}{\gcd(k,n)} \in \mathbb{N}$. Then ω^k is a primitive m-th root of unity; in particular ω^{-1} is a primitive n-th root of unity. Finally, we have $\sum_{i=0}^{n-1} \omega^{ik} = 0$.

Proof. Recall that $X^b - 1 = (X^a - 1) \cdot \sum_{i=0}^{\frac{b}{a}-1} X^{ai} \in R[X]$, for all $a \mid b \in \mathbb{N}$.
a) We have to show that $\omega^k - 1$ is neither 0 nor a zero-divisor, for all $k \in \{1, \ldots, n-1\}$: Let $d := \gcd(k, n) = xk + yn \in \mathbb{N}$, for suitable $x, y \in \mathbb{Z}$. Thus we have $d < n$ and $d \mid n$, hence there is a prime divisor $p \mid n$ such that $d \mid \frac{n}{p}$. Hence we have $\omega^k - 1 \mid \omega^{xk} - 1 = \omega^{xk+yn} - 1 = \omega^d - 1 \mid \omega^{\frac{n}{p}} - 1$, and since the latter is neither 0 nor a zero-divisor, this also holds for the former.
b) From $n \mid \frac{kn}{\gcd(k,n)} = km$ we have $(\omega^k)^m = 1$. Assume that $(\omega^k)^j - 1$ is 0 or a zero-divisor for some $j \in \{1, \ldots, m-1\}$, then we have $n \mid kj$, hence $m = \frac{n}{\gcd(k,n)} \mid j$, a contradiction. Finally, we have $(\omega^k - 1) \cdot \sum_{i=0}^{n-1} \omega^{ik} = \omega^{kn} - 1 = 0$, and since $\omega^k - 1$ is neither 0 nor a zero-divisor we conclude $\sum_{i=0}^{n-1} \omega^{ik} = 0$. ♯

(3.3) Definition. Let R be a commutative ring and let $n \in \mathbb{N}$. Note that R^n becomes an R-algebra by componentwise addition and multiplication.

a) Let $R[X]_{<n} := \{f = \sum_{i=0}^{n-1} f_i X^i \in R[X]; \deg(f) < n\} \,\dot\cup\, \{0\}$. Hence we have an isomorphism of free R-modules $\kappa\colon R^n \to R[X]_{<n}\colon [f_0, \ldots, f_{n-1}] \mapsto \sum_{i=0}^{n-1} f_i X^i$. Moreover, we have the natural isomorphism of free R-modules $\nu\colon R[X]_{<n} \to R[X]/\langle X^n - 1\rangle$; note that $X^n - 1$ is monic. The R-algebra structure of $R[X]/\langle X^n - 1\rangle$ is transported back to R^n via $\kappa\nu$ as follows:

For $f = \sum_{i=0}^{n-1} f_i X^i$ and $g = \sum_{j=0}^{n-1} g_j X^j$ let $h = \sum_{k=0}^{n-1} h_k X^k \in R[X]_{<n}$ such that $fg \equiv h \bmod (X^n - 1)$. Then we have $fg = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} f_i g_j X^{i+j}$, and thus computing $\mathrm{mod}(X^n - 1)$ yields $h_k := \sum_{i+j \equiv k \bmod n} f_i g_j$. We write $f * g := h \in R[X]_{<n}$ as well as $[f_0, \ldots, f_{n-1}] * [g_0, \ldots, g_{n-1}] := [h_0, \ldots, h_{n-1}]$, called the associated **(positive) wrapped convolution** on $R[X]_{<n}$ and R^n, respectively. Note that if $\deg(f) + \deg(g) < n$, then we have $f * g = fg$.

b) Let $\omega \in R$ be a primitive n-th root of unity. Then the concatenation $\delta_\omega\colon R^n \to R^n$ of $\kappa\colon R^n \to R[X]_{<n}$ and the n-point evaluation map $R[X]_{<n} \to R^n\colon f \mapsto [f(\omega^0), \ldots, f(\omega^{n-1})]$ is called the **discrete Fourier transform**: We have $\delta_\omega\colon [f_0, \ldots, f_{n-1}] \mapsto [\sum_{i=0}^{n-1} f_i (\omega^j)^i; j \in \{0, \ldots, n-1\}] = [f_0, \ldots, f_{n-1}] \cdot \Delta_\omega$, where $\Delta_\omega := [\omega^{ij}; i, j \in \{0, \ldots, n-1\}]_{ij} \in R^{n \times n}$.

To evaluate $f = \sum_{i=0}^{n-1} f_i X^i \in R[X]_{<n}$ for an arbitrary argument $w \in R$, we use the **Horner scheme**:

```
$s \leftarrow 0$
for $i \in [n-1, n-2, \ldots, 0]$ do $s \leftarrow s \cdot w + f_i$
return s
```

Hence we have $s = f(w)$, needing $O(n)$ ring operations. Thus to compute $[f_0, \ldots, f_{n-1}]\delta_\omega$ classically, we have to evaluate $f = \sum_{i=0}^{n-1} f_i X^i$ for the n arguments $\omega^0, \ldots, \omega^{n-1}$, where each evaluation needs $O(n)$ ring operations, hence amounting to $O(n^2)$ ring operations. We can do better than that:

(3.4) Algorithm: Fast Fourier transform (FFT), Cooley-Tukey (1965). Let $n \in \mathbb{N}$ be even, let $\omega \in R$ be a primitive n-th root of unity, and let $f =$

$\sum_{i=0}^{n-1} f_iX^i \in R[X]_{<n}$. Then there are $q, r, q', r' \in R[X]_{<\frac{n}{2}}$ such that $f = q \cdot (X^{\frac{n}{2}} - 1) + r = q' \cdot (X^{\frac{n}{2}} + 1) + r'$. Letting $f = f' \cdot X^{\frac{n}{2}} + f''$, where $f' = \sum_{i=0}^{\frac{n}{2}-1} f_{\frac{n}{2}+i}X^i \in R[X]_{<\frac{n}{2}}$ and $f'' = \sum_{i=0}^{\frac{n}{2}-1} f_iX^i \in R[X]_{<\frac{n}{2}}$, we have $f - (f'' + f') = f'(X^{\frac{n}{2}} - 1)$, hence $f \equiv f'' + f' \bmod (X^{\frac{n}{2}} - 1)$, thus $r = f'' + f'$, and similarly $f - (f'' - f') = f'(X^{\frac{n}{2}} + 1)$, hence $f \equiv f'' - f' \bmod (X^{\frac{n}{2}} + 1)$, thus $r' = f'' - f'$.

From $0 = \omega^n - 1 = (\omega^{\frac{n}{2}} - 1)(\omega^{\frac{n}{2}} + 1)$, since $\omega^{\frac{n}{2}} - 1$ is neither 0 nor a zero-divisor, we conclude $\omega^{\frac{n}{2}} = -1$. Hence for $k \in \{0, \ldots, \frac{n}{2} - 1\}$ we find $f(\omega^{2k}) = q(\omega^{2k}) \cdot ((\omega^{2k})^{\frac{n}{2}} - 1) + r(\omega^{2k}) = q(\omega^{2k}) \cdot (\omega^{kn} - 1) + r(\omega^{2k}) = r(\omega^{2k})$ and $f(\omega^{2k+1}) = q'(\omega^{2k+1}) \cdot ((\omega^{2k+1})^{\frac{n}{2}} + 1) + r'(\omega^{2k+1}) = q'(\omega^{2k+1}) \cdot ((\omega^{nk}\omega^{\frac{n}{2}} + 1) + r'(\omega^{2k+1}) = r'(\omega^{2k+1}) = r'(\omega \cdot \omega^{2k}) = r''(\omega^{2k})$, where $r''(X) := r'(\omega X) \in R[X]_{<\frac{n}{2}}$.

Let $l \in \mathbb{N}_0$ and $n = 2^l$, let $\omega \in R$ be a primitive 2^l-th root of unity, and let $F(f, \omega, n)$ be defined as follows:

```
if $n = 1$ then return $[f_0]$
$r \leftarrow \sum_{i=0}^{\frac{n}{2}-1} (f_i + f_{\frac{n}{2}+i})X^i$
$r'' \leftarrow \sum_{i=0}^{\frac{n}{2}-1} (f_i - f_{\frac{n}{2}+i})\omega^i X^i$
$[r_0, r_2, \ldots, r_{n-2}] \leftarrow F(r, \omega^2, \frac{n}{2})$
$[r_1, r_3, \ldots, r_{n-1}] \leftarrow F(r'', \omega^2, \frac{n}{2})$
return $[r_0, \ldots, r_{n-1}]$
```

Hence we have $F(f, \omega, n) = [f_0, \ldots, f_{n-1}]\delta_\omega$; note that ω^2 is a primitive $\frac{n}{2}$-th root of unity, for $n > 1$. Let $\kappa(n) \in \mathbb{N}$ be the number of ring operations needed to compute $F(\cdot, \cdot, n)$. Hence we have $\kappa(1) = 1$, and for $n > 1$ we have 2 calls of $F(\cdot, \cdot, \frac{n}{2})$, as well as $O(n)$ ring operations to compute r and r''. Hence we have $\kappa(n) = 2\kappa(\frac{n}{2}) + \gamma n$, for some $\gamma > 0$. Thus by induction we find $\kappa(n) = 2\kappa(\frac{n}{2}) + \gamma n = 2(2\kappa(\frac{n}{4}) + \frac{\gamma n}{2}) + \gamma n = 4\kappa(\frac{n}{4}) + 2\gamma n = \ldots = 2^l + \gamma n l = n + \gamma n \log_2(n)$, hence $\kappa(n) \in O(n \ln(n))$.

(3.5) Proposition. The discrete Fourier transform δ_ω is a homomorphism of R-algebras $(R^n, *) \to (R^n, \cdot)$. Moreover, we have $\delta_{\omega^{-1}}\delta_\omega = n \cdot \mathrm{id}_{R^n}$, hence if $n \in R^*$ then δ_ω is an isomorphism.

Proof. We show that $([f_0, \ldots, f_{n-1}] * [g_0, \ldots, g_{n-1}])\delta_\omega = [f_0, \ldots, f_{n-1}]\delta_\omega \cdot [g_0, \ldots, g_{n-1}]\delta_\omega$: For $f = \sum_{i=0}^{n-1} f_iX^i$ and $g = \sum_{j=0}^{n-1} g_jX^j$ we have $fg \equiv f * g \bmod (X^n - 1)$, hence there is $q \in R[X]$ such that $f * g = fg + q(X^n - 1)$. Thus $(f * g)(\omega^k) = f(\omega^k)g(\omega^k) + q(\omega^k)((\omega^k)^n - 1) = f(\omega^k)g(\omega^k)$, for $k \in \{0, \ldots, n-1\}$. Hence $[(f * g)(\omega^0), \ldots, (f * g)(\omega^{n-1})] = [f(\omega^0), \ldots, f(\omega^{n-1})] \cdot [g(\omega^0), \ldots, g(\omega^{n-1})]$.

Since ω^{-1} also is a primitive n-th root of unity, $\delta_{\omega^{-1}} \colon R^n \to R^n$ is a discrete Fourier transform. We show that $\Delta_{\omega^{-1}}\Delta_\omega = n \cdot \mathrm{id}_{R^n}$: For $i, j \in \{0, \ldots, n-1\}$ we have $[\Delta_{\omega^{-1}}\Delta_\omega]_{ij} = \sum_{k=0}^{n-1} \omega^{-ik}\omega^{kj} = \sum_{k=0}^{n-1} \omega^{(j-i)k}$. Thus for $i \neq j$ we have $[\Delta_{\omega^{-1}}\Delta_\omega]_{ij} = 0$, while for $i = j$ we obtain $[\Delta_{\omega^{-1}}\Delta_\omega]_{ii} = n$. $\sharp$

(3.6) Theorem. Let R **support FFT**, i. e. we have $2 \in R^*$ and R has primitive 2^l-th roots of unity for all $l \in \mathbb{N}_0$; this holds e. g. for $R = \mathbb{Z}[\frac{1}{2}, \zeta_{2^l}; l \in \mathbb{N}] \subseteq \mathbb{Q}[\zeta_{2^l}; l \in \mathbb{N}] \subseteq \mathbb{Q}^{\mathrm{ab}} \subseteq \overline{\mathbb{Q}}$. Then to compute the wrapped convolution $f * g$, for $f, g \in R[X]_{<n}$, needs at most $O(n \ln(n))$ ring operations. In particular, to compute the product fg, for $f, g \in R[X]$ such that $\deg(fg) \leq \deg(f) + \deg(g) < n$, needs at most $O(n \ln(n))$ ring operations.

Proof. Let $l \in \mathbb{N}_0$ such that $2^{l-1} < n \leq 2^l$, and let $\omega \in R$ be a primitive 2^l-th root of unity. Let $f = \sum_{i=0}^{2^l-1} f_i X^i$ and $g = \sum_{j=0}^{2^l-1} g_j X^j$ and $f * g = \sum_{k=0}^{2^l-1} h_k X^k$. Then we have $([f_0, \ldots, f_{2^l-1}]\delta_\omega \cdot [g_0, \ldots, g_{2^l-1}]\delta_\omega)\delta_\omega^{-1} = ([f_0, \ldots, f_{2^l-1}] * [g_0, \ldots, g_{2^l-1}])\delta_\omega\delta_\omega^{-1} = [h_0, \ldots, h_{2^l-1}]$. Since componentwise multiplication needs $O(2^l)$ ring operations, and both discrete Fourier transformations δ_ω and $\delta_\omega^{-1} = 2^{-l} \cdot \delta_{\omega^{-1}}$ need $O(2^l \cdot l)$ ring operations, this amounts to $O(2^l \cdot l) = O(n \ln(n))$ ring operations. ♯

Note that if $2 \notin R^*$, then after replacing δ_ω^{-1} by $\delta_{\omega^{-1}}$ no division in R is needed at all, and we obtain $2^l \cdot (f * g)$ instead, where l is as in the above proof.

(3.7) Algorithm: Schönhage-Strassen (1971), polynomial version. Let R be a commutative ring such that $2 \in R^*$, and let $n = 2^l$ for some $l \in \mathbb{N}$. For $0 \neq f, g \in R[X]$ such that $\deg(fg) \leq \deg(f) + \deg(g) < n$, to compute $fg \in R[X]$, it is sufficient to compute the **negative wrapped convolution** $h \in R[X]_{<n}$ such that $fg \equiv h \bmod (X^n + 1)$. Thus we may assume we have given $f, g \in R[X]_{<n}$, and aim to compute $h \in R[X]_{<n}$; the idea is to impose additional primitive roots of unity:

Let $m := 2^{\lfloor \frac{l}{2} \rfloor}$ and $t := \frac{n}{m} = 2^{\lceil \frac{l}{2} \rceil}$, and let $f = \sum_{i=0}^{t-1} f_i X^{mi}$ and $g = \sum_{j=0}^{t-1} g_j X^{mj}$, for suitable $f_i, g_j \in R[X]_{<m}$. Moreover let $f' := \sum_{i=0}^{t-1} f_i Y^i \in R[X, Y]$ and $g' := \sum_{j=0}^{t-1} g_j Y^j \in R[X, Y]$, thus we have $f = f'(X, X^m)$ and $g = g'(X, X^m)$. It suffices to compute $h' \in (R[X])[Y]_{<t}$ such that $f'g' \equiv h' \bmod (Y^t + 1)$: From $f'g' = h' + q'(Y^t + 1)$, for some $q' \in R[X, Y]$, we get $fg = f'(X, X^m)g'(X, X^m) = h'(X, X^m) + q'(X, X^m)(X^{mt} + 1)$, hence $fg \equiv h'(X, X^m) \bmod (X^n + 1)$.

A comparison of coefficients at Y^i, for $i \geq t$, yields $\deg_X(q') \leq \deg_X(f'g') \leq \deg_X(f') + \deg_X(g') < 2m$, and thus $\deg_X(h') < 2m$ as well. Hence h' can be computed in $R[X, Y]/\langle X^{2m} + 1, Y^t + 1\rangle \cong (R[X]/\langle X^{2m} + 1\rangle)[Y]/\langle Y^t + 1\rangle$. Using the natural map $\overline{}\colon R[X] \rightarrow R[X]/\langle X^{2m} + 1\rangle$, let $\overline{f'} := \sum_{i=0}^{t-1} \overline{f_i} Y^i \in (R[X]/\langle X^{2m} + 1\rangle)[Y]$ and $\overline{g'} := \sum_{j=0}^{t-1} \overline{g_j} Y^j \in (R[X]/\langle X^{2m} + 1\rangle)[Y]$. It suffices to compute $\overline{h'} \in (R[X]/\langle X^{2m} + 1\rangle)[Y]_{<t}$ such that $\overline{f'g'} \equiv \overline{h'} \bmod (Y^t + 1)$.

The element $\overline{X} \in R[X]/\langle X^{2m} + 1\rangle$ is a primitive $4m$-th root of unity: We have $\overline{X}^{2m} = -1$ and $\overline{X}^{4m} = 1$, and since $4m = 2^{2 + \lfloor \frac{l}{2} \rfloor}$, it suffices to consider $\overline{X}^{2m} - 1 = -2 \in R[X]/\langle X^{2m} + 1\rangle$, which being a unit is neither 0 nor a zero-divisor. If l is even, we have $t = m$ and $\omega := \overline{X}^2$ is a primitive $2t$-th root of unity in $R[X]/\langle X^{2m} + 1\rangle$, while if l is odd, we have $t = 2m$ and $\omega := \overline{X}$ is a primitive $2t$-th root of unity. Hence in both cases we have $\omega^t = -1$. Thus the above congruence is equivalent to $\overline{f'}(\omega Y)\overline{g'}(\omega Y) \equiv \overline{h'}(\omega Y) \bmod (Y^t - 1)$ in $(R[X]/\langle X^{2m} + 1\rangle)[Y]$.

Thus computing $\overline{h'}$ amounts to wrapped convolution in $(R[X]/\langle X^{2m}+1\rangle)[Y]_{<t}$, based on FFT for $t = 2^{\lceil \frac{l}{2} \rceil} \in R^*$, where multiplication in $R[X]/\langle X^{2m}+1\rangle$ again is negative wrapped convolution in $R[X]_{<2m}$, which is done by recursion for $l \geq 3$; for $l \leq 2$ the classical or the Karatsuba multiplication algorithm is used. Finally we get $\overline{h'}(Y) = \overline{h'}(\omega^{-1}(\omega Y))$ and $h = h'(X, X^m)$. ♯

(3.8) Example. Let $R := \mathbb{F}_5$ as well as $f := 3 + 2X + X^4 \in \mathbb{F}_5[X]$ and $g := 2 + 4X + X^2 + 2X^3 \in \mathbb{F}_5[X]$. Hence we may take $l = 3$, thus $n = 8$ as well as $m = 2$ and $t = 4$. Moreover, we have $f' = (3 + 2X) + Y^2 \in \mathbb{F}_5[X, Y]$ and $g' = (2 + 4X) + (1 + 2X) \cdot Y$. We have $\omega = \overline{X} \in \mathbb{F}_5[X]/\langle X^4 + 1\rangle$, and thus $\overline{f'} = \overline{3 + 2X} + Y^2$ and $\overline{g'} = \overline{2 + 4X} + \overline{1 + 2X} \cdot Y \in (\mathbb{F}_5[X]/\langle X^4 + 1\rangle)[Y]$, hence $\overline{f'}(\omega Y) = \overline{f'}(\overline{X} \cdot Y) = \overline{3 + 2X} + \overline{X^2} \cdot Y^2$ and $\overline{g'}(\omega Y) = \overline{2 + 4X} + \overline{X + 2X^2} \cdot Y \in (\mathbb{F}_5[X]/\langle X^4 + 1\rangle)[Y]$. Classical multiplication yields $\overline{f'}(\omega Y)\overline{g'}(\omega Y) = \overline{h'}(\omega Y) = (\overline{3 + 2X} + \overline{X^2} \cdot Y^2) \cdot (\overline{2 + 4X} + \overline{X + 2X^2} \cdot Y) = \overline{1 + X + 3X^2} + \overline{3X + 3X^2 + 4X^3} \cdot Y + \overline{2X^2 + 4X^3} \cdot Y^2 + \overline{3 + X^3} \cdot Y^3 \in (\mathbb{F}_5[X]/\langle X^4 + 1\rangle)[Y]$; where we have used that $\overline{X^3 + 2X^4} = \overline{3 + X^3} \in \mathbb{F}_5[X]/\langle X^4 + 1\rangle$. Using $\omega^{-1} = \omega^7 = 4\omega^3 = 4\overline{X}^3 \in \mathbb{F}_5[X]/\langle X^4 + 1\rangle$ we get $\overline{h'}(Y) = \overline{1 + X + 3X^2} + \overline{(3X + 3X^2 + 4X^3) \cdot 4X^3} \cdot Y + \overline{(2X^2 + 4X^3) \cdot X^6} \cdot Y^2 + \overline{(3 + X^3) \cdot 4X^9} \cdot Y^3 = \overline{1 + X + 3X^2} + \overline{3 + 3X + 4X^2} \cdot Y + \overline{2 + 4X} \cdot Y^2 + \overline{1 + 2X} \cdot Y^3 \in (\mathbb{F}_5[X]/\langle X^4 + 1\rangle)[Y]$. Hence $h' = (1 + X + 3X^2) + (3 + 3X + 4X^2) \cdot Y + (2 + 4X) \cdot Y^2 + (1 + 2X) \cdot Y^3 \in \mathbb{F}_5[X, Y]$ and thus for $h'(X, X^2) = (1 + X + 3X^2) + (3 + 3X + 4X^2) \cdot X^2 + (2 + 4X) \cdot X^4 + (1 + 2X) \cdot X^6 = 1 + X + X^2 + 3X^3 + X^4 + 4X^5 + X^6 + 2X^7 \in \mathbb{F}_5[X]$ we have $fg \equiv h' \bmod (X^8 + 1)$, where since $\deg(g) + \deg(g) = 7 < 8$ we even have $fg = h'$.

(3.9) Theorem. Let R be a commutative ring such that $2 \in R^*$. Then to compute the negative wrapped convolution $fg \bmod (X^{2^l} + 1)$ of $f, g \in R[X]_{<2^l}$, for some $l \in \mathbb{N}$, needs at most $O(2^l \cdot l \ln(l))$ ring operations. Thus to compute the product $fg \in R[X]$, for $f, g \in R[X]$ such that $\deg(fg) \leq \deg(f) + \deg(g) < n$, for some $n \in \mathbb{N}$, needs at most $O(n \ln(n) \ln(\ln(n)))$ ring operations.

Proof. The second assertion follows from the first one by letting $l \in \mathbb{N}$ such that $2^{l-1} < n \leq 2^l$, which implies $O(2^l \cdot l \ln(l)) = O(n \ln(n) \ln(\ln(n)))$. To prove the first assertion, we count the ring operations needed to perform the Schönhage-Strassen algorithm for polynomials: Let $\tau(l)$ be the number of ring operations needed for input in $R[X]_{<2^l}$, where we may assume $\tau(l) = 1$ for $l \leq 2$. For $l \geq 3$, to compute $\overline{f'}(\omega Y)$ and $\overline{g'}(\omega Y)$ as well as $\overline{h'}(Y) = \overline{h'}(\omega^{-1}(\omega Y))$, each needs $O(t)$ operations in $R[X]/\langle X^{2m} + 1\rangle$, which are shifts of coefficient lists and sign inversions in R, hence each need $O(m)$ ring operations, while to compute $h = h'(X, X^m)$ needs $O(mt)$ ring operations. Thus this amounts to $O(mt)$ ring operations in R. Moreover, FFT needs $O(t \ln(t))$ operations of replacing $Y \leftarrow \omega Y$, thus needs $O(mt \ln(t))$ ring operations in R, and to compute the wrapped convolution by componentwise multiplication we need t multiplications in $R[X]/\langle X^{2m}+1\rangle$, where a multiplication is done recursively by negative wrapped convolution, which needs $\tau(\log_2(2m))$ ring operations.

Thus there is $\gamma > 0$ such that $\tau(l) = \gamma mt\log_2(t) + t\tau(\log_2(2m)) = \gamma \cdot 2^l \cdot \lceil \frac{l}{2} \rceil + 2^{\lceil \frac{l}{2} \rceil} \cdot \tau(\lfloor \frac{l}{2} \rfloor + 1)$. Hence letting $\sigma(l)$ be defined by $\tau(l) = 2^{l-1}(l-2)\sigma(l-1)$ we for $l \geq 3$ get $2^{l-1}(l-2)\sigma(l-1) = \gamma \cdot 2^l \cdot \lceil \frac{l}{2} \rceil + 2^{\lceil \frac{l}{2} \rceil} \cdot 2^{\lfloor \frac{l}{2} \rfloor}(\lfloor \frac{l}{2} \rfloor - 1)\sigma(\lfloor \frac{l}{2} \rfloor)$, hence $\sigma(l-1) = \frac{2\gamma \cdot \lceil \frac{l}{2} \rceil}{l-2} + \frac{2(\lfloor \frac{l}{2} \rfloor - 1)}{l-2} \cdot \sigma(\lfloor \frac{l}{2} \rfloor)$. Since $\lfloor \frac{l}{2} \rfloor - 1 \leq \frac{l-2}{2}$ and $\frac{2 \cdot \lceil \frac{l}{2} \rceil}{l-2} \leq 4$, for $l \geq 3$, there is $\gamma' > 0$ such that $\sigma(l-1) \leq \gamma' + \sigma(\lfloor \frac{l}{2} \rfloor)$. Moreover, since for $l \geq 3$ we have $\lfloor \frac{l}{2} \rfloor \leq \frac{l}{2} = \frac{(l-1)+1}{2} \leq \frac{3(l-1)}{4}$, by induction we get $\sigma(l-1) \leq \gamma' \cdot \log_{\frac{4}{3}}(\frac{l-1}{2}) + \sigma(2) \leq \gamma'' \ln(l-1)$, for some $\gamma'' > 0$. This yields $\tau(l) \in O(2^{l-1}(l-2)\ln(l-1)) = O(2^l \cdot l\ln(l))$. ♯

(3.10) Corollary. Let R be a commutative ring. Then to compute the product fg, for $f, g \in R[X]$ such that $\deg(fg) \leq \deg(f) + \deg(g) < n$, needs at most $O(n\ln(n)\ln(\ln(n)))$ ring operations.

Proof. Using the division-free version of FFT in the Schönhage-Strassen algorithm, we compute $t \cdot fg \in R[X]$, where $2^{l-1} < n \leq 2^l$ and $t = 2^{\lceil \frac{l}{2} \rceil} \in O(n^{\frac{1}{2}})$. Similarly, using the **Schönhage algorithm (1977)**, see Exercise (8.20), employing the division-free version of 3-adic FFT, we compute $t' \cdot fg \in R[X]$, where $3^{l'-1} < n \leq 3^{l'}$ and $t' = 3^{\lfloor \frac{l'}{2} \rfloor} \in O(n^{\frac{1}{2}})$. Hence we compute $s, s' \in \mathbb{Z}$ such that $st + s't' = 1 \in \mathbb{Z}$, which needs $O(\ln^2(n^{\frac{1}{2}})) = O(\ln^2(n))$ bit operations, and $s(t \cdot fg) + s'(t' \cdot fg) \in R[X]$, which needs $O(n)$ ring operations. ♯

(3.11) Algorithm: Schönhage-Strassen (1971), integer version.
Let $a, b \in \mathbb{N}$ such that $b_2(ab) \leq b_2(a) + b_2(b) \leq n = 2^l$, for some $l \in \mathbb{N}$. Hence to compute $ab \in \mathbb{N}$ it is sufficient to compute negative wrapped convolution $ab \bmod (2^n + 1)$. Thus we may assume we have given $a, b \in \{1, \ldots, 2^n\}$; as we have $2^n \equiv -1 \bmod (2^n+1)$, the cases $a = 2^n$ or $b = 2^n$ are easy special cases, hence we may additionally assume that $b_2(a), b_2(b) \leq n = 2^l$:

Let $m := 2^{\lfloor \frac{l}{2} \rfloor}$ and $t := \frac{n}{m} = 2^{\lceil \frac{l}{2} \rceil}$, and let $a = \sum_{i=0}^{m-1} a_i \cdot 2^{ti}$ and $b = \sum_{j=0}^{m-1} b_j \cdot 2^{tj}$ as well as $ab = \sum_{k=0}^{2m-1} c_k \cdot 2^{tk}$, for suitable $a_i, b_j \in \{0, \ldots, 2^t - 1\}$ and $c_k \in \mathbb{N}_0$ Hence we have $c_k = \sum_{i=0}^{k} a_i b_{k-i}$, for $k \in \{0, \ldots, 2m-1\}$, where we let $b_j := 0$ for $j \geq m$. Since $mt = 2^l = n$ we have $2^{mt} = 2^n \equiv -1 \bmod (2^n+1)$, and we get $ab \equiv \sum_{k=0}^{m-1} d_k \cdot 2^{tk} \bmod (2^n+1)$, for $k \in \{0, \ldots, m-1\}$, where $d_k := c_k - c_{m+k} \in \mathbb{Z}$.

Counting the number of summands yields $|c_k| \leq (k+1) \cdot (2^t-1)^2 < (k+1) \cdot 2^{2t}$ and $|c_{m+k}| \leq (m-k-1) \cdot (2^t-1)^2 < (m-k-1) \cdot 2^{2t}$, for $k \in \{0, \ldots, m-1\}$. Hence we have $-(m-k-1) \cdot 2^{2t} < d_k < (k+1) \cdot 2^{2t}$. Since $((k+1)+(m-k-1)) \cdot 2^{2t} = m \cdot 2^{2t}$, it is sufficient to compute $d_k \bmod (m \cdot (2^{2t}+1))$. Since the moduli $m = 2^{\lfloor \frac{l}{2} \rfloor}$ and $2^{2t}+1 = 2^{2 \cdot \lceil \frac{l}{2} \rceil} + 1$ are coprime, we compute $d'_k \equiv d_k \bmod m$

and $d''_k \equiv d_k \bmod (2^{2t}+1)$, and let $\delta_k := (2^{2t}+1)((d'_k - d''_k) \bmod m) + d''_k$. Applying the Chinese remainder theorem, since $2^{2t}+1 \equiv 1 \bmod m$, we have $\delta_k \equiv d'_k \bmod m$ and $\delta_k \equiv d''_k \bmod (2^{2t}+1)$, hence $d_k \equiv \delta_k \bmod (m \cdot (2^{2t}+1))$.

To compute d'_k we proceed as follows: Let $a'_i \equiv a_i \bmod m$ and $b'_j \equiv b_j \bmod m$, for $i,j \in \{0,\ldots,m-1\}$, as well as $\widehat{a} := \sum_{i=0}^{m-1} a'_i \cdot m^{3i}$ and $\widehat{b} := \sum_{j=0}^{m-1} b'_j \cdot m^{3j}$. Hence $\widehat{a}\widehat{b} = \sum_{k=0}^{2m-1} c'_k \cdot m^{3k}$, where $c'_k = \sum_{i=0}^{k} a'_i b'_{k-i}$, for $k \in \{0,\ldots,2m-1\}$, and where we let $b'_j := 0$ for $j \geq m$. Since $0 \leq c'_k < m \cdot m^2$ the c'_k can be read off from $\widehat{a}\widehat{b}$, and we have $d'_k \equiv c'_k - c'_{m+k} \bmod m$, for $k \in \{0,\ldots,m-1\}$.

To compute d''_k we proceed as follows: Let $\omega := 2^{\frac{2t}{m}} \in \mathbb{Z}/\langle 2^{2t}+1\rangle$, then ω is a primitive $2m$-th root of unity: We have $\omega^m = 2^{2t} = -1 \in \mathbb{Z}/\langle 2^{2t}+1\rangle$ and hence $\omega^{2m} = (-1)^2 = 1 \in \mathbb{Z}/\langle 2^{2t}+1\rangle$, and since $2m = 2^{1+\lfloor \frac{l}{2} \rfloor}$, it suffices to consider $\omega^m - 1 = -2 \in \mathbb{Z}/\langle 2^{2t}+1\rangle$, which being a unit is neither 0 nor a zero-divisor. We have $[a_0\omega^0,\ldots,a_{m-1}\omega^{m-1}] * [b_0\omega^0,\ldots,b_{m-1}\omega^{m-1}] = [\delta''_0,\ldots,\delta''_{m-1}]$, where $\delta''_k = \omega^k \cdot \sum_{i+j \equiv k \bmod m} a_i b_j \in \mathbb{Z}/\langle 2^{2t}+1\rangle$, for $k \in \{0,\ldots,m-1\}$. Hence we have $d''_k = \omega^{-k} \cdot \delta''_k \in \mathbb{Z}/\langle 2^{2t}+1\rangle$. Thus $d''_k \bmod (2^{2t}+1)$ can be computed using FFT for $m = 2^{\lfloor \frac{l}{2} \rfloor} \in (\mathbb{Z}/\langle 2^{2t}+1\rangle)^*$, and multiplication $\bmod (2^{2t}+1)$ which for $l \geq 4$ by recursion is negative wrapped convolution. ♯

(3.12) Theorem. To compute the negative wrapped convolution $ab \bmod (2^{2^l}+1)$, for $a,b \in \mathbb{N}$ such that $b_2(a), b_2(b) \leq 2^l$, for some $l \in \mathbb{N}$, needs at most $O(2^l \cdot l\ln(l))$ bit operations. Thus to compute the product $ab \in \mathbb{N}$, for $a,b \in \mathbb{N}$ such that $b_2(ab) \leq b_2(a) + b_2(b) \leq n$, for some $n \in \mathbb{N}$, needs at most $O(n\ln(n)\ln(\ln(n)))$ bit operations.

Proof. The second assertion follows from the first one by letting $l \in \mathbb{N}$ such that $2^{l-1} < n \leq 2^l$, which implies $O(2^l \cdot l\ln(l)) = O(n\ln(n)\ln(\ln(n)))$. To prove the first assertion, we count the ring operations needed to perform the Schönhage-Strassen algorithm for integers: Let $\tau(l)$ be the number of bit operations needed for input of bit length 2^l, where we may assume $\tau(l) = 1$ for $l \leq 3$. To compute ab from the d_k, we need m additions of numbers of bit length $O(\ln(m \cdot 2^{2t}))$, hence $O(m\ln(m \cdot 2^{2t})) \subseteq O(mt) = O(n)$ bit operations; note that the d_k have to be multiplied by 2-powers, which amounts to shifts and actually need not be performed explicitly. To compute δ_k from d'_k and d''_k we need $O(t\ln(m))$ bit operations to compute each remainder and each product, hence altogether $O(mt\ln(m)) \subseteq O(n\ln(m))$ bit operations.

To compute d'_k we need $O(t\ln(m))$ bit operations to compute each remainder, hence altogether $O(mt\ln(m)) = O(n\ln(m))$ bit operations, moreover we need $O((3ml)^{\log_2(3)}) = O(n^{\frac{\log_2(3)}{2}}(\ln(n))^{\log_2(3)}) \subseteq O(n)$ bit operations to compute the single product $\widehat{a}\widehat{b}$ using the Karatsuba algorithm, and we need $O(\frac{3}{2} \cdot l\ln(m))$ bit operations to compute each final remainder, hence altogether $O(ml\ln(m)) = O(n^{\frac{1}{2}}\ln^2(n)) \subseteq O(n)$ bit operations. To compute d''_k we need $O(m\ln(m))$ multiplications of integers of bit length t with 2-powers, which hence amounts to shifts, to compute the Fourier transform, hence $O(mt\ln(m)) = O(n\ln(m))$ bit operations, m recursive calls needing $\tau(\lceil\frac{l}{2}\rceil+1)$ bit operations each, and finally m multiplications of integers of bit length t with 2-powers, hence $O(mt) = O(n)$ bit operations.

Thus there is $\gamma > 0$ such that $\tau(l) = \gamma n\log_2(m) + m\tau(\lceil\frac{l}{2}\rceil+1) = \gamma\cdot 2^l\cdot\lfloor\frac{l}{2}\rfloor + 2^{\lfloor\frac{l}{2}\rfloor}\cdot\tau(\lceil\frac{l}{2}\rceil+1)$. Hence letting $\sigma(l)$ be defined by $\tau(l) = 2^{l-1}(l-3)\sigma(l-1)$ we for $l\geq 4$ get $2^{l-1}(l-3)\sigma(l-1) = \gamma\cdot 2^l\cdot\lfloor\frac{l}{2}\rfloor + 2^{\lfloor\frac{l}{2}\rfloor}\cdot 2^{\lceil\frac{l}{2}\rceil}(\lceil\frac{l}{2}\rceil-2)\sigma(\lceil\frac{l}{2}\rceil)$, hence $\sigma(l-1) = \frac{2\gamma\cdot\lfloor\frac{l}{2}\rfloor}{l-3} + \frac{2(\lceil\frac{l}{2}\rceil-2)}{l-3}\cdot\sigma(\lceil\frac{l}{2}\rceil)$. Since $\lceil\frac{l}{2}\rceil - 2 \leq \frac{l-3}{2}$ and $\frac{2\cdot\lfloor\frac{l}{2}\rfloor}{l-3}\leq 4$, for $l\geq 4$, there is $\gamma' > 0$ such that $\sigma(l-1)\leq\gamma' + \sigma(\lceil\frac{l}{2}\rceil)$. Moreover, since for $l\geq 4$ we have $\lceil\frac{l}{2}\rceil \leq \frac{l+1}{2} = \frac{(l-1)+2}{2} \leq \frac{5(l-1)}{6}$, by induction we get $\sigma(l-1)\leq\gamma'\cdot\log_{\frac{6}{5}}(\frac{l-1}{3}) + \sigma(3) \leq \gamma''\ln(l-1)$, for some $\gamma'' > 0$. This yields $\tau(l)\in O(2^{l-1}(l-3)\ln(l-1)) = O(2^l\cdot l\ln(l))$. ♯

(3.13) Remark. a) In practice, instead of using the Schönhage-Strassen algorithm for integers, we use **3-primes modular FFT** multiplication: Let $z = 2^{64}$, and let $a = \sum_{i=0}^{n-1} a_i z^i \in \mathbb{N}_0$ and $b = \sum_{j=0}^{n-1} a_j z^j \in \mathbb{N}_0$, where $a_i, b_j \in \{0,\ldots,z-1\}$. Moreover, let $A := \sum_{i=0}^{n-1} a_i X^i \in \mathbb{Z}[X]_{<n}$ and $B := \sum_{j=0}^{n-1} b_j X^j \in \mathbb{Z}[X]_{<n}$, hence we have $a = A(z)$ and $b = B(z)$. Let $AB = C := \sum_{k=0}^{2n-1} c_k X^k \in \mathbb{Z}[X]_{<2n-1}$, where $0\leq c_k = \sum_{i=0}^{n-1} a_i b_{k-i} < \sum_{i=0}^{n-1} z^2 \leq nz^2$; we again let $b_j := 0$ for $j<0$ or $j\geq n$; hence we have $ab = C(z)$.

To compute $C\in\mathbb{Z}[X]_{<2n-1}$, we proceed as follows: We assume that $n < \frac{z}{2^3} = 2^{61}$, and choose pairwise different primes $\frac{z}{2} = 2^{63}\leq p_1, p_2, p_3 < z$. Since $nz^2 < 2^{189}$ and $p_1p_2p_3 \geq 2^{3\cdot 63} = 2^{189}$, the Chinese remainder theorem allows to compute C from $C \bmod p_i$, for $i\in\{1,\ldots,3\}$. To compute the product $AB \bmod p_i$ using FFT, we choose **Fourier primes** p such that $p-1$ is divisible by a high 2-power. Actually, all primes $p = k\cdot 2^{57}+1 < z$ such that $2^{57} \mid p$ are given as follows, where $w\in\mathbb{N}$ is minimal such that $\overline{w}\in\mathbb{Z}/\langle p\rangle$ is a primitive 2^{57}-th root of unity:

k	29	71	75	95	108	123
w	21	287	149	55	64	493

Note $p < \frac{z}{2}$ only for $k = 29$, and that for $k = 108$ we even have $2^{59} \mid (p-1)$, which is the only prime $p < z$ having this property, and no higher 2-powers dividing $p-1$ occur for primes $p < z$.

Anyway, we choose 3 of these pairs once and for all. Hence we are able to apply FFT for polynomials such that $2n-1\leq 2^{57}$, hence $n\leq 2^{56}$, thus for $a,b\in\mathbb{N}_0$ such that $b_z(a), b_z(b)\leq 2^{56}$, hence $b_2(a), b_2(b)\leq 2^{6+56} = 2^{62}$, i. e. for $a,b\in\mathbb{N}_0$ which need up to $\sim 4,6\cdot 10^{18}$ bit $\sim 5,8\cdot 10^{17}$ Byte.

b) Having at hand fast multiplication over $\mathbb{Z}$ and $F[X]$, where F is a field, there are fast algorithms for modular multiplication and to compute quotient and remainder, see [3, Ch.9.1], and fast algorithms for polynomial multipoint evaluation and interpolation, for reduction modulo several moduli and Chinese remaindering, see [3, Ch.10]. These are not presented here.

4 Resultants

(4.1) Algorithm: Euclidean algorithm, polynomial versions.
a) Let R be an integral domain, and let $K := \mathrm{Quot}(R)$ be its fields of fractions. Let $0 \neq f = \sum_{i=0}^{d} \frac{f_i'}{f_i''} \cdot X^i \in K[X]$, where $f_i', f_i'' \in R$, and let $\nu := \prod_{i=0}^{d} f_i'' \in R$. Hence we have $\nu f = \sum_{i=0}^{d} f_i X^i \in R[X]$, where $f_i = \frac{\nu f_i'}{f_i''} \in R$. Letting $b(f) := \max\{b(\nu), b(f_0), \ldots, b(f_d)\}$, where $b(\cdot)$ is the input length function for the ring R, the input length of f is given as $(\deg(f)+2) \cdot b(f)$.

We discuss coefficient growth in the quotient and remainder algorithm: Let $0 \neq f, g \in K[X]$ be monic, hence we have $f = (\sum_{i=0}^{d-1} \frac{f_i}{\nu} X^i) + X^d \in K[X]$ and $g = (\sum_{j=0}^{e-1} \frac{g_j}{\mu} X^i) + X^e \in K[X]$, where $d = \deg(f)$ and $e = \deg(g)$ as well as $\nu, \mu, f_i, g_j \in R$. Let $f = qg + \rho r$, where $q, r \in K[X]$ such that $r = 0$ or $\deg(r) < e = \deg(g)$ and r is monic, and $\rho \in K$. Assuming $\deg(g) = e = d-1 = \deg(f) - 1$ we have $q = \frac{\mu f_{d-1} - \nu g_{e-1}}{\nu\mu} + X$ and $\rho r = f - qg = \frac{1}{\nu\mu^2} \cdot (\mu^2(\nu f) - \nu\mu(\mu g)X - (\mu f_{d-1} - \nu g_{e-1})(\mu g))$. Hence we have $b(q) \leq b(f) + b(g) + 1$ and $b(\rho r) \leq b(f) + 2b(g) + 3$, and since r may be computed using the leading coefficient of $\nu\mu^2\rho \cdot r$, we also have $b(r) \leq b(f) + 2b(g) + 3$. Thus letting $b := \max\{b(f), b(g)\}$, we obtain $b(r) \leq 3b + 3$; note that for single quotient and remainder computations this in general indeed occurs.

E. g. let $R = \mathbb{Z}$ as well as $f = r_0 := -5 + 2X + 8X^2 - 3X^3 - 3X^4 + X^6 + X^8 \in \mathbb{Z}[X]$ and $g = r_1 := 21 - 9X - 4X^2 + 5X^4 + 3X^6 \in \mathbb{Z}[X]$, see [8, Ch.4.6.1, pp.426ff.] and [3, Exc.6.42]. The Euclidean algorithm yields $r_2 = \frac{1}{9} \cdot (-3 + X^2 - 5X^4)$ and $r_3 = \frac{1}{25} \cdot (441 - 225X - 117X^2)$ as well as $r_4 = \frac{1}{19\,773} \cdot (-307\,500 + 233\,150X)$ and $r_5 = -\frac{1\,288\,744\,821}{543\,589\,225}$. Note that gcd computations in $\mathbb{Z}$ are used to write rational numbers as quotients of coprime integers.

b) Modifying the Euclidean algorithm to use monic remainders throughout, called the **monic Euclidean algorithm**, for $0 \neq f, g \in K[X]$ monic:

```
1. $\lambda_0 \leftarrow \mathsf{lc}(f)$, $r_0 \leftarrow \frac{1}{\lambda_0} \cdot f$, $s_0 \leftarrow \frac{1}{\lambda_0}$, $t_0 \leftarrow 0$, $n_0 \leftarrow \deg(f)$
2. $\lambda_1 \leftarrow \mathsf{lc}(g)$, $r_1 \leftarrow \frac{1}{\lambda_1} \cdot g$, $s_1 \leftarrow 0$, $t_1 \leftarrow \frac{1}{\lambda_1}$, $n_1 \leftarrow \deg(g)$
3. $i \leftarrow 1$
4. while $r_i \neq 0$ do
     $\widehat{r}_{i+1} \leftarrow r_{i-1} \bmod r_i$
     $q_i \leftarrow \frac{r_{i-1} - \widehat{r}_{i+1}}{r_i}$
     $\lambda_{i+1} \leftarrow \mathsf{lc}(\widehat{r}_{i+1})$
     $r_{i+1} \leftarrow \frac{1}{\lambda_{i+1}} \cdot \widehat{r}_{i+1}$
     $s_{i+1} \leftarrow \frac{1}{\lambda_{i+1}} \cdot (s_{i-1} - q_i s_i)$
     $t_{i+1} \leftarrow \frac{1}{\lambda_{i+1}} \cdot (t_{i-1} - q_i t_i)$
     $n_{i+1} \leftarrow \deg(r_{i+1})$
```

```
    $i \leftarrow i+1$
5. return $[r_{i-1}, s_{i-1}, t_{i-1}]$    # $i = l+1$
```

Assuming that we have a **regular degree sequence**, i. e. we have $n_{i-1} - n_i = 1$ for all $i \geq 1$, we for the monic remainders r_i obtain $b(r_i) \leq 3^i b + \frac{3(3^i-1)}{2}$, for $i \geq 2$, hence $b(r_i) \in O(3^i(b+3))$. Thus assuming $d = \max\{\deg(f), \deg(g)\}$ this yields $b(r_i) \in O(3^d(b+3))$. This hence is an exponential bound in the input lengths of f and g; for $R = \mathbb{Z}$ we will show a polynomial bound in (4.12).

E. g. Letting $r_0 := -5 + 2X + 8X^2 - 3X^3 - 3X^4 + X^6 + X^8 \in \mathbb{Q}[X]$ and $r_1 := \frac{1}{3} \cdot (21 - 9X - 4X^2 + 5X^4) + X^6 \in \mathbb{Q}[X]$ the monic Euclidean algorithm yields $r_2' = \frac{1}{9} \cdot (-3 + X^2 - 5X^4)$, hence $r_2 = \frac{1}{5} \cdot (3 - X^2) + X^4$, and $r_3' = \frac{1}{25} \cdot (147 - 75X - 39X^2)$, hence $r_3 = \frac{1}{13} \cdot (-49 + 25X) + X^2$, as well as $r_4' = \frac{1}{2\,197} \cdot (61\,500 - 46\,630X)$, hence $r_4 = -\frac{6\,150}{4\,663} + X$, and $r_5' = \frac{11\,014\,913}{21\,743\,569}$, hence $r_5 = 1$. Note that gcd computations in $\mathbb{Z}$ are used to write rational numbers as quotients of coprime integers.

c) To avoid computations in K completely, we use **pseudo-division** yielding **pseudo-remainders**: Let $0 \neq f, g \in R[X]$ such that $\deg(f) =: d \geq e := \deg(g)$. Hence there are $q, r \in R[X]$ such that $\mathrm{lc}(g)^{d-e+1} f = qg + r$, and $r = 0$ or $\deg(r) < e = \deg(g)$; note that if we have $\mathrm{lc}(g) = g_e \in R^*$, then this amounts to compute quotient and remainder. Pseudo-division in general leads to exponential growth of the coefficients of the remainders, but still is useful e. g. for multivariate polynomial rings over integral domains.

E. g. letting $f = r_0 := -5 + 2X + 8X^2 - 3X^3 - 3X^4 + X^6 + X^8 \in \mathbb{Z}[X]$ and $g = r_1 := 21 - 9X - 4X^2 + 5X^4 + 3X^6 \in \mathbb{Z}[X]$ again, the Euclidean algorithm using only pseudo-division yields $r_2 = -9 + 3X^2 - 15X^4$ and $r_3 = -59\,535 + 30\,375X + 15\,795X^2$ as well as $r_4 = -1\,654\,608\,338\,437\,500 + 254\,542\,875\,143\,750X$ and $r_5 = 12\,593\,338\,795\,500\,743\,100\,931\,141\,992\,187\,500 \sim 1{,}2 \cdot 10^{34}$.

d) Let R be factorial. For $0 \neq f = \sum_{i=0}^{d} f_i X^i \in R[X]$ the element $\gamma(f) := \gcd(f_0, \ldots, f_d) \in R$ is called the **content** of f, and if $\gamma(f) \in R^*$ then f is called **primitive**. Thus, if we are given $0 \neq f = \sum_{i=0}^{d} \frac{f_i'}{f_i''} \cdot X^i \in K[X]$ as above, where $f_i', f_i'' \in R$, then we might assume $f_i', f_i'' \in R$ to be coprime, and could use $\nu' := \mathrm{lcm}(f_0'', \ldots, f_d'') \in R$ instead of $\nu \in R$, to obtain a primitive polynomial $\nu' f \in R[X]$; note that this requires gcd computations in R.

Moreover, for using pseudo-division we may compute and divide out the contents of the pseudo-remainders, leading to the **primitive Euclidean algorithm**, for $0 \neq f, g \in R[X]$ primitive:

```
1. $r_0 \leftarrow f$, $n_0 \leftarrow \deg(f)$
2. $r_1 \leftarrow g$, $n_1 \leftarrow \deg(g)$
3. $i \leftarrow 1$
4. while $r_i \neq 0$ do
    $\widehat{r}_{i+1} \leftarrow (\mathrm{lc}(r_i)^{n_{i-1}-n_i+1} \cdot r_{i-1}) \bmod r_i$
    $\gamma_{i+1} \leftarrow \gamma(\widehat{r}_{i+1})$    # content
    $r_{i+1} \leftarrow \frac{1}{\gamma_{i+1}} \cdot \widehat{r}_{i+1}$
    $n_{i+1} \leftarrow \deg(r_{i+1})$
    $i \leftarrow i+1$
5. return $r_{i-1}$      # $i = l+1$
```

Note that this needs gcd computations in R, and that the primitive and the monic Euclidean algorithm are equivalent as far as the growth of the coefficients of the remainders is concerned. In practice, since the primitive Euclidean algorithm tends to need less gcd computations in R, it is superior to the monic Euclidean algorithm.

E. g. still letting $f = r_0 := -5+2X+8X^2-3X^3-3X^4+X^6+X^8 \in \mathbb{Z}[X]$ and $g = r_1 := 21-9X-4X^2+5X^4+3X^6 \in \mathbb{Z}[X]$, the primitive Euclidean algorithm yields $\widehat{r}_2 = -9+3X^2-15X^4$, hence $\gamma(\widehat{r}_2) = 3$ and $r_2 = -3+X^2-5X^4$, and $\widehat{r}_3 = -2\,205+1\,125X+585X^2$, hence $\gamma(\widehat{r}_3) = 45$ and $r_3 = -49+25X+13X^2$, as well as $\widehat{r}_4 = -307\,500+233\,150X$, hence $\gamma(\widehat{r}_4) = 50$ and $r_4 = -6\,150+4\,663X$, and $\widehat{r}_5 = 143\,193\,869$, hence $\gamma(\widehat{r}_5) = 143\,193\,869$ and $r_5 = 1$.

e) Let R be factorial. To avoid gcd computations in R completely, but still to get polynomial growth of the coefficients of the remainders, Collins's algorithm (1967), see Exercise (8.25), can be used. It runs completely in $R[X]$ and uses pseudo-division, but instead of making the remainders primitive by dividing out their contents, which would need gcd computations in R, only certain divisors of the contents are divided out. The proof of the validity of Collins's algorithm is based on subresultants. For $R = \mathbb{Z}$, similarly to (4.12), these are also used to prove a polynomial bound for the bit lengths of the coefficients of the remainders in terms of the input lengths of f and g; hence in practice Collins's algorithm is superior to the primitive Euclidean algorithm. Finally, the proof of validity also shows that resultants, although defined as determinants in a linear algebra context, can be computed using this variant of the Euclidean algorithm.

E. g. still letting $f = r_0 := -5+2X+8X^2-3X^3-3X^4+X^6+X^8 \in \mathbb{Z}[X]$ and $g = r_1 := 21-9X-4X^2+5X^4+3X^6 \in \mathbb{Z}[X]$, the Collins algorithm yields $\lambda_1 = 3$ and $\eta_1 = 9$, as well as $r_2 = \widehat{r}_2 = -9+3X^2-15X^4$ and $\lambda_2 = -15$ and $\eta_2 = 25$, as well as $\widehat{r}_3 = -59\,535+30\,375X+15\,795X^2$ and $r_3 = -245+125X+65X^2$ and $\lambda_3 = 65$ and $\eta_2 = 169$, as well as $\widehat{r}_4 = -115\,312\,500+87\,431\,250X$ and $r_4 = 12\,300-9\,326X$ and $\lambda_4 = \eta_4 = -9\,326$, as well as $\widehat{r}_5 = 2\,863\,877\,380$ and $r_5 = \lambda_5 = \eta_5 = 260\,708$.

For a detailed cost analysis of the various algorithms, which we do not present here, see [3, Ch.6]. We set out to develop the necessary machinery for resultants and subresultants, and derive a few of their properties, leading to the polynomial bound in (4.12).

(4.2) Definition. Let R be an integral domain, let $0 \neq f, g \in R[X]$ such that $f = \sum_{i=0}^{n} f_i X^i$ and $g = \sum_{j=0}^{m} g_j X^j$, where $n = \deg(f)$ and $m = \deg(g)$. Moreover, let $\varphi(f,g)\colon R[X]_{<m} \times R[X]_{<n} \to R[X]_{<n+m}\colon [s,t] \mapsto sf+tg$, where for $n = 0$ we let $R[X]_{<n} := \{0\}$. Note that $\varphi(f,g)$ is R-linear and we have $\mathrm{rk}_R(R[X]_{<m} \times R[X]_{<n}) = n+m = \mathrm{rk}_R(R[X]_{<n+m})$.

By the R-bases $[[X^{m-1},0],[X^{m-2},0],\ldots,[1,0],[0,X^{n-1}],[0,X^{n-2}],\ldots,[0,1]]$ of $R[X]_{<m}\times R[X]_{<n}$, and $[X^{n+m-1},X^{n+m-2},\ldots,1]$ of $R[X]_{<n+m}$, for $n+m\geq 1$, we obtain the matrix of $\varphi(f,g)$ as the **Sylvester matrix**

$$S(f,g) := \left[\begin{array}{ccccccc} f_n & f_{n-1} & \cdots & f_0 & & & \\ & f_n & \cdots & f_1 & f_0 & & \\ & & \ddots & & \ddots & \ddots & \\ & & & f_n & \cdots & \cdots & f_0 \\ \hline g_m & g_{m-1} & \cdots & g_0 & & & \\ & g_m & \cdots & g_1 & g_0 & & \\ & & \ddots & & \ddots & \ddots & \\ & & & g_m & \cdots & \cdots & g_0 \end{array}\right] \in R^{(n+m)\times(n+m)},$$

where the upper half consists of $m=\deg(g)$ rows, and the lower half consists of $n=\deg(f)$ rows. Moreover, let $\operatorname{res}(f,g):=\det(S(f,g))\in R$ be the **resultant** of $f,g\in R[X]$. If $n=m=0$ then $S(f,g)\in R^{0\times 0}$ is an empty matrix, and in this case we let $\operatorname{res}(f,g):=\det(S(f,g)):=1\in R$. Treating the zero polynomial as a constant polynomial, this yields $\operatorname{res}(f,0)=0$ if $\deg(f)=n\geq 1$, and $\operatorname{res}(f,0)=1$ if f is constant, and similar statements for $\operatorname{res}(0,g)$.

(4.3) Proposition. Let F be a field, and let $0\neq f,g\in F[X]$ such that $n=\deg(f)$ and $m=\deg(g)$.
a) Then $f,g\in F[X]$ are coprime if and only if $\varphi(f,g)\colon F[X]_{<m}\times F[X]_{<n}\to F[X]_{<n+m}$ is injective, which happens if and only if $\varphi(f,g)$ is bijective, which is equivalent to $\operatorname{res}(f,g)\neq 0\in F$.
b) If $\varphi(f,g)$ is bijective and $n+m\geq 1$, let $[s,t]:=1\varphi(f,g)^{-1}\in F[X]_{<m}\times F[X]_{<n}$, i. e. we have $sf+tg=1$. Then we have $[s_l,t_l]=r_l\cdot[s,t]$, where $s_l,t_l\in F[X]$ are the Bezout coefficients computed by the extended Euclidean algorithm for f and g.

Proof. a) We have to show that $\gcd(f,g)\in F[X]$ is non-constant if and only if there are $s\in F[X]_{<m}$ and $t\in F[X]_{<n}$ such that $[s,t]\neq[0,0]$ and $sf+tg=0$: Let $h:=\gcd(f,g)\in F[X]$ monic, if $\deg(h)\geq 1$ then we let $s:=\frac{-g}{h}\in F[X]_{<m}$ and $t:=\frac{f}{h}\in F[X]_{<n}$. Conversely, let $s\in F[X]_{<m}$ and $t\in F[X]_{<n}$ such that $[s,t]\neq[0,0]$ and $sf+tg=0$, and assume $f,g\in F[X]$ are coprime. Then $s,t\neq 0$, and hence $sf=-tg$ implies $f\mid t$. Since $\deg(t)<n$ this is a contradiction.

b) We consider the extended Euclidean algorithm: For $i\geq 1$ we by induction show that $\deg(s_{i+1})=n_1-n_i$: We have $s_2=s_0-q_1s_1=1$, hence $\deg(s_2)=0=n_1-n_1$, as well as $s_3=s_1-q_2s_2=-q_2s_2$, hence $\deg(s_3)=\deg(q_2)=n_1-n_2$, and for $i\geq 3$ we from $\deg(q_i)=n_{i-1}-n_i$ get $\deg(q_is_i)=n_{i-1}-n_i+n_1-n_{i-1}=n_1-n_i$ and $\deg(s_{i-1})=n_1-n_{i-2}$, hence $\deg(q_is_i)>\deg(s_{i-1})$, and thus from $s_{i+1}=s_{i-1}-q_is_i$ we conclude $\deg(s_{i+1})=\deg(q_is_i)=n_1-n_i$. Similarly, for $i\geq 0$ we by induction show that $\deg(t_{i+1})=n_0-n_i$: We have $t_1=1$, hence $\deg(t_1)=0=n_0-n_0$, and $t_2=t_0-q_1t_1=q_1$, hence $\deg(t_2)=\deg(q_1)=n_0-n_1$;

note that $t_2 = q_1 = 0$ if and only if $n_0 = n < m = n_1$. In the latter case we have $t_3 = t_1 - q_2 t_2 = t_1 = 1$ and hence $\deg(t_3) = 0 = n_0 - n_0 = n_0 - n_2$, while for $i = 2$ and $n_0 \geq n_1$, as well as for $i \geq 3$, we have $\deg(q_i t_i) = n_{i-1} - n_i + n_0 - n_{i-1} = n_0 - n_i$ and $\deg(t_{i-1}) = n_0 - n_{i-2}$, hence $\deg(q_i t_i) > \deg(t_{i-1})$, and thus from $t_{i+1} = t_{i-1} - q_i t_i$ we conclude $\deg(t_{i+1}) = \deg(q_i t_i) = n_0 - n_i$. Hence for the Bezout coefficients we have $\deg(s_l) = n_1 - n_{l-1} < n_1 = \deg(g)$ and $\deg(t_l) = n_0 - \deg(r_{l-1}) < n_0 = \deg(f)$. Since $s_l f + t_l g = r_l \in F^*$ the assertion follows from the injectivity of $\varphi(f, g)$. ♯

(4.4) Corollary. Let R be factorial, and let $f, g \in R[X]$ such that $[f, g] \neq [0, 0]$. Then $\gcd(f, g) \in R[X]$ is non-constant if and only if $\operatorname{res}(f, g) = 0 \in R$.

Proof. For $f, g \neq 0$ the assertion follows by Gauß's Theorem from the above. If $g = 0$ then we have $\gcd(f, g) = f$, and hence the assertion follows directly from the definition of $\operatorname{res}(f, 0)$; if $f = 0$ we argue similarly. ♯

(4.5) Corollary. Let R be an integral domain, and let $0 \neq f, g \in R[X]$ such that $n + m \geq 1$, where $n := \deg(f)$ and $m := \deg(g)$. Then there are $s \in R[X]_{<m}$ and $t \in R[X]_{<n}$ such that $[s, t] \neq [0, 0]$ and $sf + tg = \operatorname{res}(f, g) \in R \subseteq R[X]$.

Proof. Let $K := \operatorname{Quot}(R)$. If $\operatorname{res}(f, g) = 0 \in R \subseteq K$, then $f, g \in K[X]$ are not coprime, hence there are $s' \in K[X]_{<m}$ and $t' \in K[X]_{<n}$ such that $[s', t'] \neq [0, 0]$ and $s'f + t'g = 0$, and thus we let $s := \lambda s' \in R$ and $t := \lambda t' \in R$, for some suitable $0 \neq \lambda \in R$.

If $\operatorname{res}(f, g) \neq 0 \in R \subseteq K$, then $f, g \in K[X]$ are coprime, hence there are $s' \in K[X]_{<m}$ and $t' \in K[X]_{<n}$ such that $[s', t'] \neq [0, 0]$ and $s'f + t'g = 1$. Moreover, $[s', t']$ arises as the solution of the system of K-linear equations $[S_0, \ldots, S_{m-1}, T_0, \ldots, T_{n-1}] \cdot S(f, g) = [0, \ldots, 0, 1] \in K^{n+m}$, where S_i, T_j are indeterminates over K. Since $\det(S(f, g)) = \operatorname{res}(f, g) \neq 0$, this solution is uniquely determined and can be computed using Cramer's rule. Hence there are $s'_i, t'_j \in R$ such that $\frac{1}{\operatorname{res}(f,g)} \cdot [s'_0, \ldots, s'_{m-1}, t'_0, \ldots, t'_{n-1}] \cdot S(f, g) = [0, \ldots, 0, 1] \in K^{n+m}$. Thus letting $s := \operatorname{res}(f, g) \cdot \sum_{i=1}^{m-1} s'_i X^i \in R[X]$ and $t := \operatorname{res}(f, g) \cdot \sum_{j=1}^{n-1} t'_j X^j \in R[X]$ we obtain $sf + tg = \operatorname{res}(f, g) \in R$. ♯

(4.6) Proposition. Let R be an integral domain, let $K := \operatorname{Quot}(R)$, and let $\overline{K}$ be an algebraic closure of K. Moreover, let $0 \neq f, g \in R[X]$ such that $f = f_n \cdot \prod_{i=1}^{n} (X - \sigma_i) \in \overline{K}[X]$ and $g = g_m \cdot \prod_{j=1}^{m} (X - \tau_j) \in \overline{K}[X]$, for suitable $\sigma_i, \tau_j \in \overline{K}$. Then $R \ni \operatorname{res}(f, g) = f_n^m \cdot \prod_{i=1}^{n} g(\sigma_i) = (-1)^{nm} g_m^n \cdot \prod_{j=1}^{m} f(\tau_j) = f_n^m g_m^n \cdot \prod_{i=1}^{n} \prod_{j=1}^{m} (\sigma_i - \tau_j) \in \overline{K}$.

Proof. We may assume $n, m \geq 1$. Let $P := \overline{K}[S_1, \ldots, S_n, T_1, \ldots, T_m]$, where $S_1, \ldots, S_n, T_1, \ldots, T_m$ are indeterminates over $\overline{K}$. Moreover, let $\widehat{f} := f_n \cdot \prod_{i=1}^{n} (X - S_i) \in P[X]$ as well as $\widehat{g} := g_m \cdot \prod_{j=1}^{m} (X - T_j) \in P[X]$. Consider the

Vandermonde matrix $V_{n,m} := [v_{ij}] \in P^{(n+m)\times(n+m)}$, where for $1 \leq i \leq n+m$ we let $v_{ij} := T_j^{n+m-i}$ for $1 \leq j \leq m$, and $v_{ij} := S_{n-j}^{n+m-i}$ for $m+1 \leq j \leq n+m$:

$$V_{n,m} := \left[\begin{array}{lll|lll} T_1^{n+m-1} & \dots & T_m^{n+m-1} & S_1^{n+m-1} & \dots & S_n^{n+m-1} \\ T_1^{n+m-2} & & & & & S_n^{n+m-2} \\ \vdots & & & & & \vdots \\ T_1^0 & \dots & T_m^0 & S_1^0 & \dots & S_n^0 \end{array}\right],$$

where hence the left half consists of m columns, and the right half consists of n columns. Hence $\det(V_{n,m}) = \prod_{1\leq i<j\leq m}(T_i - T_j) \cdot \prod_{1\leq i<j\leq n}(S_i - S_j) \cdot \prod_{j=1}^m \prod_{i=1}^n (T_j - S_i) \neq 0 \in P$. Moreover, let $S(\widehat{f},\widehat{g}) \in P^{(n+m)\times(n+m)}$ the associated Sylvester matrix. Since $\widehat{f}(S_i) = 0 = \widehat{g}(T_j) \in P$, for all $1 \leq i \leq n$ and $1 \leq j \leq m$, we thus have $S(\widehat{f},\widehat{g}) \cdot V_{n,m} =$

$$\left[\begin{array}{lll|lll} T_1^{m-1}\widehat{f}(T_1) & \dots & T_m^{m-1}\widehat{f}(T_m) & 0 & \dots & 0 \\ T_1^{m-2}\widehat{f}(T_1) & & & & & 0 \\ \vdots & & & & & \vdots \\ T_1^0\widehat{f}(T_1) & \dots & T_m^0\widehat{f}(T_m) & 0 & \dots & 0 \\ \hline 0 & \dots & 0 & S_1^{n-1}\widehat{g}(S_1) & \dots & S_n^{n-1}\widehat{g}(S_n) \\ 0 & & & & & S_n^{n-2}\widehat{g}(S_n) \\ \vdots & & & & & \vdots \\ 0 & \dots & 0 & S_1^0\widehat{g}(S_1) & \dots & S_n^0\widehat{g}(S_n) \end{array}\right].$$

The Vandermonde determinant again yields $\det(S(\widehat{f},\widehat{g}) \cdot V_{n,m}) = \prod_{j=1}^m \widehat{f}(T_j) \cdot \prod_{1\leq i<j\leq m}(T_i - T_j) \cdot \prod_{i=1}^n \widehat{g}(S_i) \cdot \prod_{1\leq i<j\leq n}(S_i - S_j) \in P$, thus $\operatorname{res}_X(\widehat{f},\widehat{g}) \cdot \prod_{j=1}^m \prod_{i=1}^n (T_j - S_i) = \prod_{j=1}^m \widehat{f}(T_j) \cdot \prod_{i=1}^n \widehat{g}(S_i)$. Since $\widehat{f}(T_j) = f_n \cdot \prod_{i=1}^n (T_j - S_i) \neq 0 \in P$ and $\widehat{g}(S_i) = g_m \cdot \prod_{j=1}^m (S_i - T_j) \neq 0 \in P$ this yields $\operatorname{res}_X(\widehat{f},\widehat{g}) = f_n^m \cdot \prod_{i=1}^n \widehat{g}(S_i) = (-1)^{mn} g_m^n \cdot \prod_{j=1}^m \widehat{f}(T_j) = f_n^m g_m^n \cdot \prod_{i=1}^n \prod_{j=1}^m (S_i - T_j) \in P$.

Using the $\overline{K}$-algebra homomorphism $\epsilon \colon P[X] \to \overline{K}[X] \colon S_i \mapsto \sigma_i, t_j \mapsto \tau_j$, since $\deg(\widehat{f}) = \deg(f)$ and $\deg(\widehat{g}) = \deg(g)$ we finally have $\operatorname{res}(f,g) = \operatorname{res}_X(\widehat{f},\widehat{g})^\epsilon$. ♯

(4.7) Corollary. Let R be an integral domain, let $K := \operatorname{Quot}(R)$, let $f \in R[X]$ such that $n := \deg(f) \geq 1$. Then $\operatorname{disc}(f) := (-1)^{\frac{n(n-1)}{2}} \cdot \frac{1}{\operatorname{lc}(f)} \cdot \operatorname{res}(f, \frac{\partial f}{\partial X}) \in K$ is called the **discriminant** of f, where $\frac{\partial f}{\partial X} \in R[X]$ is the **formal derivative**.

Letting $\overline{K}$ be an algebraic closure of K, and $f = \operatorname{lc}(f) \cdot \prod_{i=1}^n (X - \sigma_i) \in \overline{K}[X]$, then we have $\operatorname{disc}(f) = \operatorname{lc}(f)^{n+m-1} \cdot \prod_{1\leq i<j\leq n}(\sigma_i - \sigma_j)^2 \in R$, where $m := \deg(\frac{\partial f}{\partial X})$, treating the zero polynomial as a constant polynomial; note that if $\operatorname{char}(R) \not| \, n$ then we have $m = n-1$.

Proof. We have $\frac{\partial f}{\partial X} = \operatorname{lc}(f) \cdot \sum_{k=1}^n \prod_{j\neq k}(X - \sigma_j) \in \overline{K}[X]$. Hence if $\sigma_i = \sigma_j$ for some $i \neq j$, then $(X - \sigma_i) \mid \frac{\partial f}{\partial X} \in \overline{K}[X]$, hence $\frac{\partial f}{\partial X}(\sigma_i) = 0$, and thus

$\mathrm{res}(f, \frac{\partial f}{\partial X}) = 0$; note that this also holds for $\frac{\partial f}{\partial X} = 0$. If the σ_i are pairwise different, then we have $\frac{\partial f}{\partial X} \neq 0$ and $\frac{\partial f}{\partial X}(\sigma_i) = \mathrm{lc}(f) \cdot \prod_{j \neq i}(\sigma_i - \sigma_j) \in \overline{K}$, and thus we obtain $\mathrm{res}(f, \frac{\partial f}{\partial X}) = \mathrm{lc}(f)^m \cdot \prod_{i=1}^n \left(\mathrm{lc}(f) \cdot \prod_{j \neq i}(\sigma_i - \sigma_j)\right) = (-1)^{\frac{n(n-1)}{2}} \cdot \mathrm{lc}(f)^{n+m} \cdot \prod_{1 \leq i < j \leq n}(\sigma_i - \sigma_j)^2$.

It remains to show $\mathrm{disc}(f) \in R$: Since for $n = 1$ we have $\mathrm{disc}(f) = 1$ anyway, we may assume $n \geq 2$ and that the σ_i are pairwise different. Since the product $\prod_{1 \leq i < j \leq n}(\sigma_i - \sigma_j)^2 \in \overline{K}$ is invariant under any permutation of the σ_i, it can be written as a $\mathbb{Z}$-polynomial in $\{e_{n,1}(\sigma_1, \ldots, \sigma_n), \ldots, e_{n,n}(\sigma_1, \ldots, \sigma_n)\}$, where $e_{n,i} := \sum_{1 \leq k_1 < k_2 < \cdots < k_i \leq m}(\prod_{l=1}^i X_{k_l}) \in \mathbb{Z}[X_1, \ldots, X_n]$ is the elementary symmetric polynomial of degree i in the indeterminates $\{X_1, \ldots, X_n\}$. Since $e_{n,i}(\sigma_1, \ldots, \sigma_n) \in \frac{1}{\mathrm{lc}(f)} \cdot R$, for all $i \in \{1, \ldots, n\}$, we also have $\mathrm{lc}(f) \cdot \prod_{1 \leq i < j \leq n}(\sigma_i - \sigma_j)^2 \in R$. Note that if $m = n - 1$, then we may also argue as follows: We have $\mathrm{lc}(f) \mid \mathrm{lc}(\frac{\partial f}{\partial X})$, which using the Sylvester matrix $S(f, \frac{\partial f}{\partial X})$ implies $\mathrm{lc}(f) \mid \mathrm{res}(f, \frac{\partial f}{\partial X}) \in R$. $\sharp$

(4.8) Definition. Let R be an integral domain, let $0 \neq f, g \in R[X]$ such that $f = \sum_{i=0}^n f_i X^i$ and $g = \sum_{j=0}^m g_j X^j$, where $n = \deg(f)$ as well as $m = \deg(g)$. For $k \in \{0, \ldots, \min\{n, m\}\}$ let $\varphi_k(f, g)\colon R[X]_{<m-k} \times R[X]_{<n-k} \to R[X]_{<n+m-2k}\colon [s, t] \mapsto \lfloor \frac{sf+tg}{X^k} \rfloor$, where for $h = \sum_{i \geq N_h} h_i X^i \in R((X)) = \mathrm{Quot}(R[[X]])$, for some $N_h \in \mathbb{Z}$, we let $\lfloor h \rfloor := \sum_{i \geq 0} h_i X^i \in R[[X]]$. Note that $\varphi_k(f, g)$ is R-linear and we have $\mathrm{rk}_R(R[X]_{<m-k} \times R[X]_{<n-k}) = n + m - 2k = \mathrm{rk}_R(R[X]_{<n+m-2k})$; moreover, we have $\varphi_0(f, g) = \varphi(f, g)$.

Similar to the above, using the R-bases $[X^{n+m-2k-1}, X^{n+m-2k-2}, \ldots, 1]$ as well as $[[X^{m-k-1}, 0], [X^{m-k-2}, 0], \ldots, [1, 0], [0, X^{n-k-1}], [0, X^{n-k-2}], \ldots, [0, 1]]$ of $R[X]_{<n+m-2k}$ and $R[X]_{<m-k} \times R[X]_{<n-k}$, respectively, for $n + m > 2k$, we obtain the matrix of $\varphi_k(f, g)$ as the k-th **generalized Sylvester matrix** $S_k(f, g) \in R^{(n+m-2k) \times (n+m-2k)}$, letting $f_i := 0$ and $g_j := 0$ for $i, j < 0$, as

$$S_k(f,g) := \left[\begin{array}{cccccccc}
f_n & f_{n-1} & \cdots & f_{n-m+k+1} & \cdots & f_{k+1} & \cdots & f_{2k-m+1} \\
 & f_n & \cdots & & \cdots & & \cdots & \\
 & & \ddots & & \ddots & & \ddots & \\
 & & & f_n & \cdots & f_m & \cdots & f_k \\
\hline
g_m & g_{m-1} & \cdots & g_{k+1} & \cdots & g_{m-n+k+1} & \cdots & g_{2k-n+1} \\
 & g_m & \cdots & & \cdots & & \cdots & \\
 & & \ddots & & \ddots & & \ddots & \\
 & & & g_m & \cdots & & \cdots & \\
 & & & & \ddots & & \ddots & \\
 & & & & & g_m & \cdots & g_k
\end{array}\right],$$

where the upper and lower halves consist of $m - k$ and $n - k$ rows, respectively. Note that for $k' \geq k$ the matrix $S_{k'}(f, g)$ is a submatrix of $S_k(f, g)$. Moreover, let $\mathrm{res}_k(f, g) := \det(S_k(f, g)) \in R$ be the k-th **subresultant** of $f, g \in R[X]$.

If $n = m = k$ then $S_k(f, g) \in R^{0\times 0}$ is an empty matrix, and in this case we let $\mathrm{res}_k(f, g) := \det(S_k(f, g)) := 1 \in R$. Note that in any case we have $\mathrm{res}_k(g, f) = (-1)^{(n-k)(m-k)} \cdot \mathrm{res}_k(f, g)$.

(4.9) Proposition. Let F be a field, and let $0 \neq f, g \in F[X]$ such that $n = \deg(f)$ and $m = \deg(g)$. In the extended Euclidean algorithm for f and g let $n_i := \deg(r_i)$ be the associated degree, for $i \in \{0, \ldots, l\}$, and let $n_{l+1} := \deg(r_{l+1}) = \deg(0) < 0$. Let $k \in \{0, \ldots, \min\{n, m\}\}$.
a) Then $k \in \{n_1, \ldots, n_l\}$ if and only if $\varphi_k(f, g)\colon F[X]_{<m-k} \times F[X]_{<n-k} \to F[X]_{<n+m-2k}$ is injective, which happens if and only if $\varphi_k(f, g)$ is bijective, which is equivalent to $\mathrm{res}_k(f, g) \neq 0 \in F$.
b) If $\varphi_k(f, g)$ is bijective and $n + m > 2k$, let $i \in \{1, \ldots, l\}$ such that $k = n_i$, and let $[s, t] := 1\varphi_k(f, g)^{-1} \in F[X]_{<m-k} \times F[X]_{<n-k}$. Then we have $[s_i, t_i] = \mathrm{lc}(r_i) \cdot [s, t]$, where $\mathrm{lc}(r_i) \in F^*$ is the leading coefficient of $r_i \in F[X]$.

Proof. a) We show that $k \notin \{n_1, \ldots, n_l\}$ if and only if there are $s \in F[X]_{<m-k}$ and $t \in F[X]_{<n-k}$ such that $[s, t] \neq [0, 0]$ and $sf + tg \in F[X]_{<k}$, i. e. $[s, t] \in \ker(\varphi_k(f, g))$: Let $k \notin \{n_1, \ldots, n_l\}$, and let $i \in \{2, \ldots, l + 1\}$, such that $n_i < k < n_{i-1}$, where we let Let $s := s_i \in F[X]$ and $t := t_i \in F[X]$ in the extended Euclidean algorithm. From the proof of (4.3) we know that $\deg(s_i) = m - n_{i-1}$ for $i \in \{2, \ldots, l + 1\}$, and $\deg(t_i) = n - n_{i-1}$ for $i \in \{1, \ldots, l + 1\}$. This yields $\deg(sf + tg) = \deg(r_i) = n_i < k$ as well as $\deg(s) < m - k$ and $\deg(t) < n - k$, where for $i \geq 2$ we have $\deg(s_i) \geq 0$ and thus $s = s_i \neq 0$.

Let conversely $s \in F[X]_{<m-k}$ and $t \in F[X]_{<n-k}$ such that $[s, t] \neq [0, 0]$ and $r := sf + tg \in F[X]_{<k}$, and let $i \in \{2, \ldots, l + 1\}$, such that $n_i < k \leq n_{i-1}$. We consider the equation $[f, g] \cdot \begin{bmatrix} s_i & s \\ t_i & t \end{bmatrix} = [r_i, r] \in F[X]^2$. Assume that $\det\left(\begin{bmatrix} s_i & s \\ t_i & t \end{bmatrix}\right) = s_it - st_i \neq 0$, then by Cramer's rule we have $f = \frac{r_it-rt_i}{s_it-st_i} \in F(X) = \mathrm{Quot}(F[X])$, which since $\deg(r_it - rt_i) \leq \max\{n_i + \deg(t), \deg(r) + \deg(t_i)\} < \max\{n_i + n - k, k + n - n_{i-1}\} \leq n = \deg(f)$ is a contradiction. Thus we have $s_it = st_i \in F[X]$.

We show that $s_j, t_j \in F[X]$ are coprime for $j \in \{0, \ldots, l+1\}$: For $j \in \{0, \ldots, l\}$ let $R_j := \begin{bmatrix} s_j & s_{j+1} \\ t_j & t_{j+1} \end{bmatrix} \in F[X]^{2\times 2}$. Hence $[f, g] \cdot R_j = [r_j, r_{j+1}]$, where $R_0 = E_2$, and where since $s_{j+2} = s_j - q_{j+1}s_{j+1}$ and $t_{j+2} = t_j - q_{j+1}t_{j+1}$ we have $R_{j+1} = R_j \cdot \begin{bmatrix} 0 & 1 \\ 1 & -q_{j+1} \end{bmatrix}$, for $j \in \{0, \ldots, l - 1\}$. Hence by induction on $j \geq 0$ we have $s_jt_{j+1} - s_{j+1}t_j = \det(R_j) = (-1)^j$, for $j \in \{0, \ldots, l\}$.

Hence from $s_it = st_i$ we conclude $s_i \mid s$. Since $s_i \neq 0$ and $[s, t] \neq [0, 0]$ there is $0 \neq h \in F[X]$ such that $s = s_ih$. This finally yields $m - n_{i-1} \leq \deg(h) + m - n_{i-1} = \deg(s_ih) = \deg(s) < m - k$, hence $k < n_{i-1}$.

b) We have $s_1 = 0$ and $\deg(s_i) = m - n_{i-1} < m - k$ for $i \geq 2$. Moreover, we have $\deg(t_i) = n - n_{i-1} < n - k$ for $i \geq 2$, and for $i = 1$ we have $k = m \leq n$, hence from $n + m > 2k$ we get $k < n$, thus $\deg(t_1) = n - n_0 = 0 < n - k$. We have $\deg(s_i f + t_i g) = \deg(r_i) = n_i = k$, hence $[s_i, t_i]\varphi_k(f, g) = \mathrm{lc}(r_i) \in F^*$. $\sharp$

(4.10) Proposition: Fundamental theorem on subresultants.
Let F be a field and $0 \neq f, g \in F[X]$, and let $\lambda_i \in F^*$, for $i \in \{0, \ldots, l\}$, be as in the monic extended Euclidean algorithm for f and g. Then for $i \in \{1, \ldots, l\}$ we have $\mathrm{res}_{n_i}(f, g) = (-1)^{\sum_{j=1}^{i-1}(n_{j-1}-n_i)(n_j-n_i)} \cdot \lambda_0^{n_1-n_i} \cdot \prod_{j=1}^{i} \lambda_j^{n_{j-1}-n_i}$.

In particular, if $\mathrm{res}(f, g) \neq 0$, i. e. $n_l = 0$ occurs in the degree sequence, then we have $\mathrm{res}(f, g) = (-1)^{\sum_{j=1}^{l-1} n_{j-1}n_j} \cdot \lambda_0^{n_1} \cdot \prod_{j=1}^{l} \lambda_j^{n_{j-1}}$; hence $\mathrm{res}(f, g)$ can be computed using the monic extended Euclidean algorithm.

Proof. See Exercise (8.24).
Let $i \in \{1, \ldots, l-1\}$ and $k \in \{0, \ldots, n_{i+1}\}$. Then the polynomial division $\lambda_{i+1} r_{i+1} = \widehat{r}_{i+1} = r_{i-1} - q_i r_i$ can be interpreted in terms of row operations on $S_k(r_{i-1}, r_i) \in F^{(n_{i-1}+n_i-2k)\times(n_{i-1}+n_i-2k)}$, yielding

$$\left[\begin{array}{ccccccc}
 & & \widehat{r}_{i+1,n_{i+1}} & \cdots & & \cdots & \widehat{r}_{i+1,2k-n_i+1} \\
 & & & \ddots & & & \vdots \\
 & & & & \widehat{r}_{i+1,n_{i+1}} & \cdots & \widehat{r}_{i+1,k} \\
\hline
r_{i,n_i} & \cdots & & \cdots & & \cdots & r_{i,2k-n_{i-1}+1} \\
 & \ddots & & & & & \vdots \\
 & & \ddots & & & & \vdots \\
 & & & r_{i,n_i} & \cdots & \cdots & r_{i,k}
\end{array}\right],$$

where the upper and lower halves consist of $n_i - k$ and $n_{i-1} - k$ rows, respectively. Multiplying the rows in the upper half by $\frac{1}{\lambda_{i+1}}$ and interchanging the upper and lower halves yields a matrix of the form $\left[\begin{array}{c|c} * & ** \\ \hline 0 & S_k(r_i, r_{i+1}) \end{array}\right]$, where now the upper and lower halves consist of $n_{i-1} - n_{i+1}$ and $(n_{i+1} - k) + (n_i - k)$ rows, respectively, and where the upper left and lower right submatrices are square. Since $r_{i,n_i} = 1$ the upper left submatrix is unitriangular, and hence we have $\mathrm{res}_k(r_{i-1}, r_i) = (-1)^{(n_{i-1}-k)(n_i-k)} \cdot \lambda_{i+1}^{n_i-k} \cdot \mathrm{res}_k(r_i, r_{i+1})$. Thus we from $\mathrm{res}_{n_i}(r_{i-1}, r_i) = 1$ obtain $\mathrm{res}_{n_i}(r_0, r_1) = \prod_{j=1}^{i-1} \left((-1)^{(n_{j-1}-n_i)(n_j-n_i)} \cdot \lambda_{j+1}^{n_j-n_i}\right)$, and from $\mathrm{res}_{n_i}(f, g) = \lambda_0^{n_1-n_i} \lambda_1^{n_0-n_i} \cdot \mathrm{res}_{n_i}(r_0, r_1)$ the assertion follows. $\sharp$

(4.11) Definition. Let $0 \neq f := \sum_{i=0}^{n} f_i X^i \in \mathbb{C}[X]$. Then let $\|f\|_1 := \sum_{i=0}^{n} |f_i| \in \mathbb{R}_{>0}$ be the **1-norm**, let $\|f\|_2 := \sqrt{\sum_{i=0}^{n} |f_i|^2} \in \mathbb{R}_{>0}$ be the **2-norm** and let $\|f\|_\infty := \max\{|f_i|; i \in \{0, \ldots, n\}\} \in \mathbb{R}_{>0}$ be the **maximum norm** of f; we let $\|0\|_1 = \|0\|_2 = \|0\|_\infty = 0$. Since for $f \neq 0$ we have $\sum_{i=0}^{n} |f_i|^2 \leq (\sum_{i=0}^{n} |f_i|)^2$ we conclude $\|f\|_\infty \leq \|f\|_2 \leq \|f\|_1 \leq \sqrt{\deg(f) + 1} \cdot \|f\|_\infty$.

(4.12) Theorem. Let $0 \neq f, g \in \mathbb{Z}[X]$ such that $n = \deg(f)$ and $m = \deg(g)$ and $\|f\|_\infty, \|g\|_\infty \leq B$ for some $B > 0$. Then the numerators and denominators of the coefficients of the elements $r_i, s_i, t_i \in \mathbb{Q}[X]$ in the monic extended Euclidean algorithm for f and g are absolutely bounded by $2(n+1)^{\frac{m}{2}} \cdot (m+1)^{\frac{n}{2}} \cdot B^{n+m}$.

Note that if $n \geq m$, then the input length of f is $(n+2) \cdot b(f) = (n+2) \cdot \max\{b(f_0), \ldots, b(f_n)\} = (n+2) \cdot b(\|f\|_\infty) \sim n \cdot \ln(B)$, which also bounds the input length of g, and we have $b(r_i), b(s_i), b(t_i) \in O(\ln((n+1)^n \cdot B^{2n})) \subseteq O(n \cdot \ln(nB))$.

Proof. Let $i \in \{2, \ldots, l\}$ and $k = n_i = \deg(r_i)$; note that hence $n + m > 2k$. Since $\operatorname{lc}(r_i) = 1$ the elements $s_i \in \mathbb{Q}[X]_{<m-k}$ and $t_i \in \mathbb{Q}[X]_{<n-k}$ are given by $[s_i, t_i] = 1\varphi_k(f,g)^{-1}$. Letting $0 \neq \rho_k := \operatorname{res}_k(f,g) \in \mathbb{Z}$, then we have $\rho_k s_i f + \rho_k t_i g = \rho_k r_i$, where by Cramer's rule $\rho_k s_i, \rho_k t_i, \rho_k r_i \in \mathbb{Z}[X]$. By Hadamard's inequality (5.3) we get $|\rho_k| = |\det(S_k(f,g))| \leq \|f\|_2^{m-k} \cdot \|g\|_2^{n-k} \leq (n+1)^{\frac{m-k}{2}} \cdot (m+1)^{\frac{n-k}{2}} \cdot \|f\|_\infty^{m-k} \cdot \|g\|_\infty^{n-k}$. Moreover, applying Cramer's rule to find s_i and t_i, Hadamard's inequality yields $\|\rho_k s_i\|_\infty \leq \|f\|_2^{m-k-1} \cdot \|g\|_2^{n-k} \leq (n+1)^{\frac{m-k-1}{2}} \cdot (m+1)^{\frac{n-k}{2}} \cdot \|f\|_\infty^{m-k-1} \cdot \|g\|_\infty^{n-k}$ and $\|\rho_k t_i\|_\infty \leq \|f\|_2^{m-k} \cdot \|g\|_2^{n-k-1} \leq (n+1)^{\frac{m-k}{2}} \cdot (m+1)^{\frac{n-k-1}{2}} \cdot \|f\|_\infty^{m-k} \cdot \|g\|_\infty^{n-k-1}$. This finally yields $\|\rho_k r_i\|_\infty \leq (m+1) \cdot (\|\rho_k s_i\|_\infty \cdot \|f\|_\infty + \|\rho_k t_i\|_\infty \cdot \|g\|_\infty) \leq 2(n+1)^{\frac{m-k+1}{2}} \cdot (m+1)^{\frac{n-k+1}{2}} \cdot \|f\|_\infty^{m-k} \cdot \|g\|_\infty^{n-k}$; note that for $k = 0$ we have $r_l = 1$ anyway. ♯

We proceed towards modular techniques for gcd computations in $\mathbb{Z}[X]$, which are asymptotically faster than techniques based on resultants, but still require resultants as a theoretical tool. We only present the basic ideas, for variants of modular gcd computations in $\mathbb{Z}[X]$ and for modular gcd computations in $F[X,Y]$, where F is a field, see [3, Ch.6.5, 6.7, 6.11]. Moreover, for an asymptotically fast extended Euclidean algorithm in $\mathbb{Z}$ and $R[X]$, where R is factorial, based on a divide and conquer technique, see [3, Ch.11.1]. Mignotte's inequality actually is proved without using resultants; for a sharper version of Mignotte's inequality, using the same line of proof, see [2, Thm.3.5.1]; for a related even better bound, the **Bombieri norm**, see [8, Exc.4.6.2.21].

(4.13) Proposition: Landau inequality (1905).
Let $0 \neq f = f_n \cdot \prod_{i=1}^n (X - z_i) \in \mathbb{C}[X]$, and let $M(f) := |f_n| \cdot \prod_{i=1}^n \max\{1, |z_i|\} \in \mathbb{R}_{>0}$ be its **Mahler measure**. Then we have $M(f) \leq \|f\|_2$.

Proof. Let first $0 \neq g = \sum_{i=0}^m g_i X^i \in \mathbb{C}[X]$, where we let $g_i := 0$ for $i < 0$ and $i > m$. Then for $z \in \mathbb{C}$ we have $\|(X - z)g\|_2^2 = \sum_{i=0}^{m+1} |g_{i-1} - z g_i|^2 = \sum_{i=0}^{m+1} (g_{i-1} - z g_i)(\overline{g_{i-1}} - \overline{z g_i}) = \|g\|_2^2 \cdot (1 + |z|^2) - \sum_{i=0}^{m+1} (z g_i \overline{g_{i-1}} + \overline{z} g_{i-1} \overline{g_i}) = \sum_{i=0}^{m+1} (\overline{z} g_{i-1} - g_i)(z \overline{g_{i-1}} - \overline{g_i}) = \sum_{i=0}^{m+1} |\overline{z} g_{i-1} - g_i|^2 = \|(\overline{z} X - 1)g\|_2^2$.

As for $M(f)$, we may assume that $|z_1|, \ldots, |z_k| > 1$ and $|z_{k+1}|, \ldots, |z_n| \leq 1$, for some $k \in \{0, \ldots, n\}$. Hence we have $M(f) = |f_n \cdot \prod_{i=1}^k \overline{z_i}|$. Letting $g := f_n \cdot \prod_{i=1}^k (\overline{z_i} X - 1) \cdot \prod_{i=k+1}^n (X - z_i) = \sum_{i=0}^n g_i X^i \in \mathbb{C}[X]$, we have $g_n = f_n \cdot \prod_{i=1}^k \overline{z_i} \in \mathbb{C}$ and thus $M(f)^2 = |g_n|^2 \leq \|g\|_2^2 = \|\frac{g}{\prod_{i=1}^k \overline{z_i} X - 1} \cdot \prod_{i=1}^k (X - z_i)\|_2^2 = \|f\|_2^2$. ♯

(4.14) Proposition. Let $0 \neq f = \sum_{i=0}^{n} f_i X^i \in \mathbb{C}[X]$ such that $n = \deg(f)$ and $0 \neq h = \sum_{j=0}^{m} h_j X^j \in \mathbb{C}[X]$ such that $m = \deg(h)$ and $h \mid f \in \mathbb{C}[X]$. Then we have $\|h\|_1 \leq 2^m \cdot M(h) \leq \frac{|h_m|}{|f_n|} \cdot 2^m \cdot M(f)$.

Proof. Let $f = f_n \cdot \prod_{i=1}^{n}(X - z_i) \in \mathbb{C}[X]$ and $h = h_m \cdot \prod_{j=1}^{m}(X - u_j) \in \mathbb{C}[X]$; note that the $u_j \in \mathbb{C}$ are a subsequence of the $z_i \in \mathbb{C}$. Let $e_{m,i} := \sum_{1 \leq k_1 < k_2 < \cdots < k_i \leq m}(\prod_{l=1}^{i} X_{k_l}) \in \mathbb{C}[X_1, \ldots, X_m]$ be the elementary symmetric polynomial of degree $i \in \{1, \ldots, m\}$ in the indeterminates $\{X_1, \ldots, X_m\}$. Then for $j \in \{0, \ldots, m-1\}$ we have $h_j = (-1)^{m-j} h_m e_{m,m-j}(u_1, \ldots, u_m) \in \mathbb{C}$, and thus we have $|h_j| \leq |h_m| \cdot \sum_{1 \leq k_1 < k_2 < \cdots < k_{m-j} \leq m}(\prod_{l=1}^{m-j} |u_{k_l}|) \leq \binom{m}{j} \cdot M(h)$; note that $|h_m| \leq \binom{m}{m} \cdot M(h)$ as well. Hence we have $\|h\|_1 = \sum_{j=0}^{m} |h_j| \leq M(h) \cdot \sum_{j=1}^{m} \binom{m}{j} = 2^m \cdot M(h) \leq \frac{|h_m|}{|f_n|} \cdot 2^m \cdot M(f)$. ♯

(4.15) Theorem: Mignotte inequality.
Let $0 \neq f, g, h \in \mathbb{Z}[X]$, where $n = \deg(f)$ and $k = \deg(g)$ as well as $m = \deg(h)$, such that $gh \mid f \in \mathbb{Z}[X]$. Then we have $\|g\|_1 \cdot \|h\|_1 \leq 2^{m+k} \cdot \|f\|_2$. In particular, this yields $\|h\|_\infty \leq 2^m \cdot \sqrt{n+1} \cdot \|f\|_\infty$.

Proof. We have $\mathrm{lc}(g)\mathrm{lc}(h) \mid \mathrm{lc}(f)$, and thus Landau's inequality implies $\|g\|_1 \cdot \|h\|_1 \leq 2^{m+k} \cdot M(f) \leq 2^{m+k} \cdot \|f\|_2$, thus proving the first inequality. The second inequality follows from taking $g := 1 \in \mathbb{Z}[X]$. ♯

(4.16) Lemma. Let R be an integral domain, let $I \lhd R$ be a prime ideal, and let $\overline{}\colon R \rightarrow R/I$ denote the natural map; note that R/I again is an integral domain. Let $0 \neq f, g \in R[X]$ such that $\mathrm{lc}(f) \notin I$, i. e. we have $\deg(f) = \deg(\overline{f})$. Then for $k \in \{0, \ldots, \min\{\deg(\overline{f}), \deg(\overline{g})\}\}$ we have $\mathrm{res}_k(\overline{f}, \overline{g}) = 0$ if and only if $\overline{\mathrm{res}_k(f, g)} = 0$.

Note that if $\mathrm{lc}(g) \notin I$ as well, i. e. we have $\deg(g) = \deg(\overline{g})$ as well, we have $\overline{\mathrm{res}_k(f, g)} = \mathrm{res}_k(\overline{f}, \overline{g})$ anyway. Moreover note that without any assumption on $\mathrm{lc}(f)$ and $\mathrm{lc}(g)$ the assertion does not hold in general: E. g. let $R := \mathbb{Z}$ and $p := 2$, as well as $f := -X + 4X^3$ and $g := 1 + 2X$, then we have $\mathrm{res}(f, g) = 0$ and $res(\overline{f}, \overline{g}) = \mathrm{res}(X, 1) = 1$.

Proof. Let $f = \sum_{i=0}^{n} f_i X^i$ and $g = \sum_{j=0}^{m} g_j X^j$, where $n = \deg(f)$ and $m = \deg(g)$. If $n = m = 0$, then $\mathrm{res}_0(f, g) = 1 \notin I$ and $\mathrm{res}_0(\overline{f}, \overline{g}) = 1 \neq 0$, while if $n = 0$ and $m > 0$, we have $\mathrm{res}_0(f, g) = f_{\underline{n}}^m \notin I$ and $\mathrm{res}_0(\overline{f}, \overline{g}) = \overline{f_n}^{\deg(\overline{g})} \neq 0$. Hence let $n \geq 1$. If $\overline{g} = 0$, then $\mathrm{res}_0(\overline{f}, \overline{g}) = 0$, and since $g_j \in I$ for all $j \in \{0, \ldots, m\}$ we have $\mathrm{res}_0(f, g) \in I$ as well. If $\overline{g} \neq 0$ let $j := \deg(\overline{g})$, hence

$j \in \{0, \ldots, m\}$ is maximal such that $\overline{g_j} \neq 0$. For $k \in \{0, \ldots, \min\{n, j\}\}$ we have

$$S_k(f,g) = \left[\begin{array}{cc|cccccc}
f_n & \cdots & f_{n-m+j} & \cdots & & & \cdots & f_{2k-m+1} \\
 & \ddots & & & & & & \vdots \\
\hline
 & & f_n & \cdots & & & \cdots & f_{2k-j+1} \\
 & & & \ddots & & & & \vdots \\
 & & & & f_n & \cdots & \cdots & f_k \\
g_m & \cdots & g_j & \cdots & & & \cdots & g_{2k-n+1} \\
 & \ddots & & & & & & \vdots \\
 & & \ddots & & & & & \vdots \\
 & & & g_m & \cdots & & \cdots & g_k
\end{array}\right],$$

where the lower right submatrix taken modulo I yields $S_k(\overline{f}, \overline{g})$, where all entries in the lower left submatrix are in I, and the upper left submatrix is an upper triangular matrix with f_n's on the diagonal. Thus we obtain $\overline{\mathrm{res}_k(f,g)} = \overline{f_n}^{m-j} \cdot \mathrm{res}_k(\overline{f}, \overline{g})$, and since $f_n^{m-j} \notin I$ the assertion follows also in this case. $\sharp$

(4.17) Proposition. Let R be a principal ideal domain, let $p \in R$ be a prime, and let $\overline{}\colon R \to R/\langle p \rangle =: F$ denote the natural map; note that F is a field. Moreover, let $0 \neq f, g \in R[X]$ such that $p \not| \gcd(\mathrm{lc}(f), \mathrm{lc}(g)) \in R$, and let $h := \gcd(f,g) \in R[X]$. Then we have $\overline{\mathrm{lc}(h)} \in F^*$ and $\deg(h) = \deg(\overline{h}) \leq \deg(\gcd(\overline{f}, \overline{g}))$. Moreover, we have $\deg(\overline{h}) = \deg(\gcd(\overline{f}, \overline{g}))$ if and only if $\overline{h} \sim \gcd(\overline{f}, \overline{g}) \in F[X]$, which in turn holds if and only if $p \not| \mathrm{res}(\frac{f}{h}, \frac{g}{h}) \in R$.

Proof. Since $h \mid f, g$, we have $\mathrm{lc}(h) \mid \mathrm{lc}(f), \mathrm{lc}(g)$, which implies $\mathrm{lc}(h) \mid \gcd(\mathrm{lc}(f), \mathrm{lc}(g))$, thus $p \not| \mathrm{lc}(h)$. Letting $u := \frac{f}{h} \in R[X]$ and $v := \frac{g}{h} \in R[X]$, we obtain $\overline{f} = \overline{u}\overline{h} \in F[X]$ and $\overline{g} = \overline{v}\overline{h} \in F[X]$, hence $\overline{h} \mid \gcd(\overline{f}, \overline{g})$, implying $\deg(\overline{h}) \leq \deg(\gcd(\overline{f}, \overline{g}))$, and showing that equality is equivalent to $\overline{h} \sim \gcd(\overline{f}, \overline{g}) \in F[X]$. Moreover, we have $\overline{h} \sim \gcd(\overline{f}, \overline{g}) \in F[X]$ if and only if $\overline{u}, \overline{v} \in F[X]$ are coprime, which holds if and only if $\mathrm{res}(\overline{u}, \overline{v}) \neq 0 \in F$. Since we may assume that $p \not| \mathrm{lc}(f)$, implying $p \not| \mathrm{lc}(u)$, the assertion $\mathrm{res}(\overline{u}, \overline{v}) \neq 0 \in F$ is equivalent to $\mathrm{res}(\frac{f}{h}, \frac{g}{h}) = \mathrm{res}(u, v) \neq 0 \in R$. $\sharp$

(4.18) Theorem. Let R be a principal ideal domain, let $K := \mathrm{Quot}(R)$, let $p \in R$ be a prime, let $\overline{}\colon R \to R/\langle p \rangle =: F$ denote the natural map, and let $0 \neq f, g \in R[X]$ such that $p \not| \gcd(\mathrm{lc}(f), \mathrm{lc}(g)) \in R$.

Then the degree sequence for $\overline{f}$ and $\overline{g}$ is a subsequence of the degree sequence for f and g, where for $i \in \{1, \ldots, l\}$ the degree n_i occurs in the degree sequence for $\overline{f}$ and $\overline{g}$, if and only if $p \not| \mathrm{res}_{n_i}(f,g) \in R$. In this case, for the elements r_i, s_i, t_i occurring in the monic extended Euclidean algorithm for f and g we have $r_i, s_i, t_i \in R_{(p)}[X] \subseteq K[X]$, and $\overline{r_i}, \overline{s_i}, \overline{t_i} \in F[X]$ occur in the monic extended Euclidean algorithm for $\overline{f}$ and $\overline{g}$, where $\deg(\overline{r_i}) = n_i$.

Proof. For $k \in \{0, \ldots, \min\{\deg(\overline{f}), \deg(\overline{g})\}\}$ we have $\mathrm{res}_k(\overline{f}, \overline{g}) \sim \overline{\mathrm{res}_k(f, g)} \in F$. Hence the degree sequence for $\overline{f}$ and $\overline{g}$ is a subsequence of the degree sequence for f and g, and n_i occurs in the degree sequence for $\overline{f}$ and $\overline{g}$ if and only if $p \not| \mathrm{res}_{n_i}(f, g) \in R$. In this case, letting $k = n_i$, if $n + m > 2k$ then we have $[s_i, t_i] = 1\varphi_k(f, g)^{-1}$, thus $s_i, t_i, r_i \in \frac{1}{\mathrm{res}_k(f,g)} \cdot R[X] \subseteq R_{(p)}[X]$. Hence $[\overline{s_i}, \overline{t_i}]\varphi_k(\overline{f}, \overline{g}) = 1$, thus $\overline{s_i}\overline{f} + \overline{t_i}\overline{g} = \overline{r_i} \in F[X]$; if $k = n = m$ and hence $k = \deg(\overline{f}) = \deg(\overline{g})$, then $i = 1$ and $p \not| \mathrm{lc}(f), \mathrm{lc}(g)$, hence from $s_1 = 0$ and $t_1 = \frac{1}{\mathrm{lc}(g)}$ using $s_1 f + t_1 g = r_1$ we get $\overline{s_1}\overline{f} + \overline{t_1}\overline{g} = \overline{r_1} \in F[X]$. ♯

(4.19) Algorithm: Modular Euclidean Algorithm.
Let $0 \neq f, g \in \mathbb{Z}[X]$ primitive such that $\deg(f) = n \geq m = \deg(g)$, and let $h = \gcd(f, g) \in \mathbb{Z}[X]$ primitive. Then by (4.12) we in particular have $\|h\|_\infty \leq 2(n+1)^{\frac{m}{2}} \cdot (m+1)^{\frac{n}{2}} \cdot \max\{\|f\|_\infty, \|g\|_\infty\}^{n+m} \leq 2(n+1)^n \cdot \max\{\|f\|_\infty, \|g\|_\infty\}^{2n}$, while Mignotte's inequality yields the better bound $\|h\|_\infty \leq 2^m \cdot \sqrt{n+1} \cdot \min\{\|f\|_\infty, \|g\|_\infty\}$.

To compute $h = \gcd(f, g) \in \mathbb{Z}[X]$ we choose a prime $p \in \mathbb{N}$ such that $p \geq 2 \cdot 2^m \cdot \sqrt{n+1} \cdot \min\{\|f\|_\infty, \|g\|_\infty\}$. Then compute $\gcd(\overline{f}, \overline{g}) \in \mathbb{Z}/\langle p\rangle[X]$, and let $\widetilde{h} \in \mathbb{Z}[X]$ such that $\deg(\widetilde{h}) = \deg(\gcd(\overline{f}, \overline{g}))$ and $\|\widetilde{h}\|_\infty \leq \frac{p-1}{2}$ as well as $\overline{\widetilde{h}} = \gcd(\overline{f}, \overline{g})$. Hence if $\widetilde{h} \mid f, g \in \mathbb{Z}[X]$, then $\widetilde{h} \sim \gcd(f, g) \in \mathbb{Z}[X]$. Note that this holds if and only if $p \not| \mathrm{res}(\frac{f}{h}, \frac{g}{h}) \in \mathbb{Z}$, hence only a finite number of primes p have to be excluded; note that by the proof of (4.12) and Mignotte's inequality we have $|\mathrm{res}(\frac{f}{h}, \frac{g}{h})| \leq (n+1)^n \cdot \max\{\|\frac{f}{h}\|_\infty, \|\frac{g}{h}\|_\infty\}^{2n} \leq (n+1)^{2n} \cdot 2^{2n^2} \cdot \max\{\|f\|_\infty, \|g\|_\infty\}^{2n}$.

To compute the monic remainders $r_i \in \mathbb{Q}[X]$ and the coefficients $s_i, t_i \in \mathbb{Q}[X]$, such that $s_i f + t_i g = r_i \in \mathbb{Q}[X]$, for $i \in \{1, \ldots, l\}$, by (4.12) we choose a prime $p \in \mathbb{N}$ such that $p \geq 8(n+1)^m \cdot (m+1)^n$. Hence n_i occurs in the degree sequence for $\overline{f}$ and $\overline{g}$, and thus $s_i, t_i, r_i \in \mathbb{Q}[X]$ can be found from $\overline{r_i}, \overline{s_i}, \overline{t_i} \in \mathbb{Z}/\langle p\rangle[X]$ by rational number reconstruction, see Exercise (8.29).

E. g. for $f = r_0 := -5+2X+8X^2-3X^3-3X^4+X^6+X^8 \in \mathbb{Z}[X]$ and $g = r_1 := 21-9X-4X^2+5X^4+3X^6 \in \mathbb{Z}[X]$, the degree sequence is $n_0 = 8$, $n_1 = 6$, $n_2 = 4$, $n_3 = 2$, $n_4 = 1$, $n_5 = 0$, $n_6 < 0$, and we find $\mathrm{res}_0(f, g) = 260\,708 = 2^2 \cdot 7 \cdot 9\,311$ and $\mathrm{res}_1(f, g) = 9\,326 = 2 \cdot 4\,663$, as well as $\mathrm{res}_2(f, g) = 169 = 13^2$ and $\mathrm{res}_3(f, g) = 0$, as well as $\mathrm{res}_4(f, g) = 25 = 5^2$ and $\mathrm{res}_5(f, g) = 0$, and finally $\mathrm{res}_6(f, g) = 9 = 3^2$.

Moreover, we obtain the degree sequences in the monic extended Euclidean algorithm for $\overline{f}$ and $\overline{g}$: For $p \notin \{2, 3, 5, 7, 13, 4\,663, 9\,311\}$ the degree sequence $[6, 4, 2, 1, 0]$ is unchanged, while for $p = 2$ we get $[6, 4, 2]$, for $p = 3$ we get $[4, 2, 1, 0]$, for $p = 5$ we get $[6, 2, 1, 0]$, for $p \in \{7, 9\,311\}$ we get $[6, 4, 2, 1]$, for $p = 13$ we get $[6, 4, 1, 0]$, and for $p = 4\,663$ we get $[6, 4, 2, 0]$.

5 Lattice base reduction

(5.1) Definition. Let $K \in \{\mathbb{R}, \mathbb{C}\}$ and let $V \neq \{0\}$ be a finite dimensional K-vector space. Let $\langle \cdot, \cdot \rangle \colon V \times V \to K \colon [v, w] \mapsto \langle v, w \rangle$ be a **hermitian sesquilinear form**, i. e. we have $\langle v + v', w \rangle = \langle v, w \rangle + \langle v', w \rangle$ and $\langle v\lambda, w \rangle = \lambda \langle v, w \rangle$ as well as $\langle w, v \rangle = \overline{\langle v, w \rangle}$, for all $v, v', w \in V$ and $\lambda \in K$. In particular, we have $\langle v, w\lambda \rangle = \overline{\lambda} \langle v, w \rangle$. For $U \leq V$ let $U^\perp := \{v \in V; \langle v, u \rangle = 0 \text{ for all } u \in U\} \leq V$. In particular, $\mathrm{rad}(\langle \cdot, \cdot \rangle) := V^\perp$ is called the **radical** of $\langle \cdot, \cdot \rangle$. The form $\langle \cdot, \cdot \rangle$ is called **non-degenerate** if $\mathrm{rad}(\langle \cdot, \cdot \rangle) = \{0\}$. A vector $0 \neq v \in V$ is called **isotropic**, if $\langle v, v \rangle = 0$; the form $\langle \cdot, \cdot \rangle$ is called **anisotropic** if there are no isotropic vectors. If $K = \mathbb{R}$ the form $\langle \cdot, \cdot \rangle$ just is a **symmetric bilinear form**.

Let $q \colon V \to K \colon v \mapsto \langle v, v \rangle$ be the **quadratic form** associated to $\langle \cdot, \cdot \rangle$. Hence we have $q(v\lambda) = \langle v\lambda, v\lambda \rangle = \lambda\overline{\lambda} \langle v, v \rangle = |\lambda|^2 \cdot q(v)$, for all $\lambda \in K$, and $q(v) = \langle v, v \rangle = \overline{\langle v, v \rangle} = \overline{q(v)} \in \mathbb{R}$. The quadratic form q is called **positive definite** if $q(v) > 0$ for all $0 \neq v \in V$. Note that in this case $0 = \langle v, v \rangle = q(v)$ implies $v = 0$, hence $\langle \cdot, \cdot \rangle$ is anisotropic, thus non-degenerate, and for $U \leq V$ we have $V = U \oplus U^\perp$. If q is positive definite, then for $K = \mathbb{C}$ the vector space V is called a **unitary**, for $K = \mathbb{R}$ it is called **Euclidean**.

Note that $\langle \cdot, \cdot \rangle$ can be recovered from q: We have $\frac{1}{2} \cdot (q(v + w) - q(v) - q(w)) = \frac{1}{2} \cdot (\langle v + w, v + w \rangle - \langle v, v \rangle - \langle w, w \rangle) = \frac{1}{2} \cdot (\langle v, w \rangle + \overline{\langle w, v \rangle}) = \mathrm{Re}(\langle v, w \rangle)$, and for $K = \mathbb{C}$ we additionally have $\frac{1}{2} \cdot (q(v + iw) - q(v) - q(iw)) = \frac{1}{2} \cdot (\langle v + iw, v + iw \rangle - \langle v, v \rangle - \langle iw, iw \rangle) = \frac{1}{2} \cdot (-i\langle v, w \rangle + i\langle w, v \rangle) = \frac{-i}{2} \cdot (\langle v, w \rangle - \overline{\langle v, w \rangle}) = \mathrm{Im}(\langle v, w \rangle)$.

(5.2) Algorithm: Gram-Schmidt orthogonalization (1883/1907).
Let V be a unitary or Euclidean vector space with quadratic form q and associated sesquilinear form $\langle \cdot, \cdot \rangle$, and let $B = \{b_1, \ldots, b_n\} \subseteq V$ be a K-basis, where $K \in \{\mathbb{R}, \mathbb{C}\}$. For $i \in \{1, \ldots, n\}$ let by induction $\mu_{ij} := \frac{\langle b_i, b'_j \rangle}{\langle b'_j, b'_j \rangle} \in K$, for $j \in \{1, \ldots, i-1\}$, and $b'_i := b_i - \sum_{j=1}^{i-1} b'_j \mu_{ij} \in V$, as well as $B' := \{b'_1, \ldots, b'_n\}$. Then for $i \in \{1, \ldots, n\}$ we have $U_i := \langle b_1, \ldots, b_i \rangle_K = \langle b'_1, \ldots, b'_i \rangle_K \leq V$. Moreover, b'_i is the image of b_i under the projection $V = U_{i-1} \oplus U_{i-1}^\perp \to U_{i-1}^\perp$, where $U_0 := \{0\}$. In particular $B' \subseteq V$ is an orthogonal K-basis, called the associated **Gram-Schmidt K-basis**, such that the base change matrix ${}_{B'}\mathrm{id}_B \in K^{n \times n}$ is lower unitriangular.

Proof. We have to show that $b_i - b'_i \in U_{i-1}$ and $b'_i \in U_{i-1}^\perp$; then we conclude $U_i = \langle b'_1, \ldots, b'_i \rangle_K$, in particular $b'_i \neq 0$, and $\langle b'_i, b'_j \rangle = 0$ for $j \in \{1, \ldots, i-1\}$. We proceed by induction on $i \in \{1, \ldots, n\}$: For $i = 1$ we have $b_1 - b'_1 = 0 \in U_0$, and $U_0^\perp = V$ anyway. For $i \geq 2$ we have $b_i - b'_i = \sum_{j=1}^{i-1} b'_j \mu_{ij} \in U_i$; and for $k \in \{1, \ldots, i-1\}$ we have $\langle b'_i, b'_k \rangle = \langle b_i, b'_k \rangle - \sum_{j=1}^{i-1} \langle b'_j, b'_k \rangle \cdot \frac{\langle b_i, b'_j \rangle}{\langle b'_j, b'_j \rangle} = \langle b_i, b'_k \rangle - \langle b'_k, b'_k \rangle \cdot \frac{\langle b_i, b'_k \rangle}{\langle b'_k, b'_k \rangle} = 0$; note that by induction we have $U_{i-1} = \langle b'_1, \ldots, b'_{i-1} \rangle_K$, where $b'_1, \ldots, b'_{i-1}$ are pairwise orthogonal. $\sharp$

Note that the $\mu_{ij} \in K$, for $1 \leq j < i \leq n$, the $\langle b_i', b_i' \rangle \in \mathbb{R}$, for $i \in \{1, \ldots, n\}$, and the base change matrix ${}_{B'}\mathrm{id}_B \in K^{n \times n}$ can successively be computed from the **Gram matrix** $Q = [\langle b_i, b_j \rangle]_{ij} \in K^{n \times n}$ of $\langle \cdot, \cdot \rangle$ alone, without explicitly computing the b_i'. Moreover, note that $B' \subseteq V$ is not necessarily an orthonormal K-basis; an orthonormal K-basis can subsequently be found by replacing b_i' by $b_i'' = b_i' \cdot \frac{1}{\sqrt{q(b_i')}}$, for $i \in \{1, \ldots, n\}$.

(5.3) Corollary: Hadamard inequality (1893).
Let $n \in \mathbb{N}$ and $\langle \cdot, \cdot \rangle \colon \mathbb{C}^n \times \mathbb{C}^n \to \mathbb{C}$ be given by $\langle e_i, e_i \rangle := 1$ and $\langle e_i, e_j \rangle := 0$, for $i \neq j \in \{1, \ldots, n\}$, where $\{e_1, \ldots, e_n\} \subseteq \mathbb{C}^n$ is the standard $\mathbb{C}$-basis. Then for $v = [v_1, \ldots, v_n] \in \mathbb{C}^n$ we have $q(v) = \langle v, v \rangle = \sum_{i=1}^n |v_i|^2 \in \mathbb{R}_{\geq 0}$, the **standard quadratic form**, hence $\|v\| := \sqrt{q(v)} = \sqrt{\sum_{i=1}^n |v_i|^2} =\in \mathbb{R}_{\geq 0}$ is the **2-norm**.

Let $A = [a_{ij}] \in \mathbb{C}^{n \times n}$, and let $a_i := [a_{i1}, \ldots, a_{in}] \in \mathbb{C}^n$ denote its rows, for $i \in \{1, \ldots, n\}$. Then we have $|\det(A)| \leq \prod_{i=1}^n \|a_i\| = \prod_{i=1}^n \left(\sum_{j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}}$.

Proof. We may assume that $\det(A) \neq 0$, and let $B := \{a_1, \ldots, a_n\} \subseteq \mathbb{C}^n$, which hence is a $\mathbb{C}$-basis. Let $B' = \{a_1', \ldots, a_n'\} \subseteq \mathbb{C}^n$ be the associated Gram-Schmidt $\mathbb{C}$-basis, and let $A' \in \mathbb{C}^{n \times n}$ be the matrix whose rows are $a_1', \ldots, a_n'$. Since ${}_{B'}\mathrm{id}_B \in \mathbb{C}^{n \times n}$ is lower unitriangular we have $\det(A) = \det(A')$. Moreover, by the orthogonality of $a_1', \ldots, a_n'$ with respect to $\langle \cdot, \cdot \rangle$ we have $A' \cdot \overline{A'}^{\mathrm{tr}} = \mathrm{diag}[\langle a_i', a_i' \rangle; i \in \{1, \ldots, n\}]$. Finally, for $i \in \{1, \ldots, n\}$ we have $q(a_i) = q(a_i' + \sum_{j=1}^{i-1} a_j' \mu_{ij}) = \langle a_i', a_i' \rangle + \sum_{j=1}^{i-1} \langle a_j', a_j' \rangle \cdot |\mu_{ij}|^2 = q(a_i') + \sum_{j=1}^{i-1} q(a_j') \cdot |\mu_{ij}|^2$. Hence in conclusion we have $|\det(A)|^2 = |\det(A')|^2 = \det(A' \cdot \overline{A'}^{\mathrm{tr}}) = \det(\mathrm{diag}[\langle a_i', a_i' \rangle]) = \prod_{i=1}^n q(a_i') \leq \prod_{i=1}^n q(a_i) = \prod_{i=1}^n \|a_i\|^2$. ♯

(5.4) Algorithm. This leads to a modular technique to compute determinants for matrices $A = [a_{ij}] \in \mathbb{Z}^{n \times n}$: Choose pairwise non-associate primes $p_1, \ldots, p_s \in \mathbb{N}$ such that $\prod_{k=1}^s p_k \geq 2 \cdot \prod_{i=1}^n \left(\sum_{j=1}^n |a_{ij}|^2 \right)^{\frac{1}{2}} \geq 2 \cdot |\det(A)|$. Then use the Gauß algorithm over the fields $\mathbb{Z}/\langle p_k \rangle$ to compute the determinants $\det(A) \in \mathbb{Z}/\langle p_k \rangle$, for all $k \in \{1, \ldots, s\}$, and by Chinese remaindering find $\det(A) \in \mathbb{Z}/\langle \prod_{k=1}^s p_k \rangle$.

(5.5) Definition. a) A free $\mathbb{Z}$-module $L \neq \{0\}$ of finite $\mathbb{Z}$-rank together with a positive definite quadratic form q on $L_{\mathbb{R}} := L \otimes_{\mathbb{Z}} \mathbb{R}$ is called a **$\mathbb{Z}$-lattice**. Let $B \subseteq L$ be a $\mathbb{Z}$-basis, and let $Q \in \mathbb{R}^{n \times n}$ be the Gram matrix of the symmetric bilinear form $\langle \cdot, \cdot \rangle$ associated to q, with respect to the $\mathbb{R}$-basis $B \subseteq L_{\mathbb{R}}$. Hence we have $q(v) = v_B Q v_B^{\mathrm{tr}}$, where $v_B \in \mathbb{R}^n$ denotes the coordinate tuple associated to $v \in L_{\mathbb{R}}$. Thus Q is a positive definite symmetric matrix. Let $B' \subseteq \mathbb{R}^n$ be the associated Gram-Schmidt $\mathbb{R}$-basis and $P := {}_{B'}\mathrm{id}_B \in \mathrm{GL}_n(\mathbb{R})$. Then we have $PQP^{\mathrm{tr}} = \mathrm{diag}[\|b_i'\|^2; i \in \{1, \ldots, n\}] \in \mathbb{R}^{n \times n}$. In particular, since $\det(P) = 1$ we have $\det(Q) = \prod_{i=1}^n \|b_i'\|^2 > 0$.

$\mathbb{Z}$-Lattices L and L', having quadratic forms q and q', respectively, are called **isomorphic** or **equivalent**, if there is a **$\mathbb{Z}$-lattice isomorphism** $\varphi\colon L \rightarrow L'$, i. e. φ is a $\mathbb{Z}$-isomorphism such that $q'(v^{\varphi}) = q(v)$, for all $v \in L$. Let $B \subseteq L$ and $B' \subseteq L'$ be $\mathbb{Z}$-bases, where $|B| = |B'| = n$, let $Q, Q' \in \mathbb{R}^{n \times n}$ be the associated Gram matrices, respectively, and let $P := {}_{B}\varphi_{B'} \in \mathrm{GL}_n(\mathbb{Z})$. Then we have $v_B Q v_B^{\mathrm{tr}} = q(v) = q'(v^{\varphi}) = v_B \cdot PQ'P^{\mathrm{tr}} \cdot v_B^{\mathrm{tr}}$, for all $v \in L_{\mathbb{R}}$, and since $\langle \cdot, \cdot \rangle$ can be recovered from q we have $Q = PQ'P^{\mathrm{tr}}$. Thus since $\det(P) \in \mathbb{Z}^* = \{\pm 1\}$ we conclude $\det(Q) = \det(Q')$. Hence $\det(L) := \sqrt{\det(Q)} > 0$ is independent of the choice of a $\mathbb{Z}$-basis of L, and called the **determinant** of L.

b) More generally, if $Q \in \mathbb{R}^{n \times n}$ is any positive definite symmetric matrix, there is an orthogonal matrix $P \in O_n(\mathbb{R})$ such that $PQP^{-1} = PQP^{\mathrm{tr}} = \mathrm{diag}[\beta_1, \ldots, \beta_n] \in \mathbb{R}^{n \times n}$, where $\beta_i > 0$. Letting $B := P^{-1} \cdot \mathrm{diag}[\sqrt{\beta_i}] \in \mathbb{R}^{n \times n}$, we obtain $Q = (P^{-1} \cdot \mathrm{diag}[\sqrt{\beta_i}]) \cdot (\mathrm{diag}[\sqrt{\beta_i}] \cdot P^{-\mathrm{tr}}) = BB^{\mathrm{tr}}$. Hence in the Euclidean $\mathbb{R}$-vector space $\mathbb{R}^n$ carrying the standard quadratic form, by restriction of the quadratic form we have a $\mathbb{Z}$-lattice $L := \langle b_1, \ldots, b_n \rangle_{\mathbb{Z}} \subseteq \mathbb{R}^n$ whose Gram matrix equals Q, where $b_1, \ldots, b_n \in \mathbb{R}^n$ are the rows of B, which are an $\mathbb{R}$-basis of $\mathbb{R}^n$. Note that more generally we may let $B \in \mathbb{R}^{n \times m}$, for $m \geq n$, such that $Q = BB^{\mathrm{tr}}$, where still the rows $b_1, \ldots, b_n \in \mathbb{R}^m$ of B are $\mathbb{R}$-linearly independent.

In particular, given any $\mathbb{Z}$-lattice L with Gram matrix $Q \in \mathbb{R}^{n \times n}$, up to equivalence of $\mathbb{Z}$-lattices we may assume that L is embedded into $\mathbb{R}^n$ as described above. Thus $q(v) = \|v\|^2$ for all $v \in \mathbb{R}^n$, and Hadamard's inequality implies $0 < \det(L) = \sqrt{\det(Q)} = |\det(B)| \leq \prod_{i=1}^n \|b_i\| = \prod_{i=1}^n \sqrt{q(b_i)}$.

c) Given a $\mathbb{Z}$-lattice $L \subseteq \mathbb{R}^n$, then $\min(L) := \min\{\|v\|; 0 \neq v \in L\} \geq 0$ is called the **minimum** of L; an element $v \in L$ such that $\|v\| = \min(L)$ is called a **minimal** or **shortest vector**. Given a $\mathbb{Z}$-basis $\{b_1, \ldots, b_n\} \subseteq L$ and the associated Gram-Schmidt $\mathbb{R}$-basis $\{b_1', \ldots, b_n'\} \subseteq \mathbb{R}^n$, then we show that $\min(L) \geq \min\{\|b_i'\|; i \in \{1, \ldots, n\}\}$, hence we have $\min(L) > 0$, and $L \subseteq \mathbb{R}^n$ is a **discrete** subset, thus the minimum $\min(L)$ is attained:

For $0 \neq v \in L$ we have $v = \sum_{i=1}^k b_i \nu_i$, for suitable $\nu_i \in \mathbb{Z}$, where $k \in \{1, \ldots, n\}$ is chosen such that $\nu_k \neq 0$. Thus $v = \sum_{i=1}^k (b_i' + \sum_{j=1}^{i-1} b_j' \mu_{ij}) \nu_i = b_k' \nu_k + \sum_{i=1}^{k-1} b_i' \nu_i'$, for suitable $\nu_i' \in \mathbb{R}$. Hence $\|v\|^2 = \nu_k^2 \cdot \|b_k'\|^2 + \sum_{i=1}^{k-1} (\nu_i')^2 \cdot \|b_i'\|^2 \geq \|b_k'\|^2$. $\sharp$

(5.6) Definition. Let $L \subseteq \mathbb{R}^n$ be a $\mathbb{Z}$-lattice having $\mathbb{Z}$-basis $B = \{b_1, \ldots, b_n\} \subseteq L$, let $\{b_1', \ldots, b_n'\} \subseteq \mathbb{R}^n$ be the associated Gram-Schmidt $\mathbb{R}$-basis, and let $\mu_{ij} := \frac{\langle b_i, b_j' \rangle}{\|b_j'\|^2} \in \mathbb{R}$, for $1 \leq j < i \leq n$. Then B is called **LLL reduced** if the following holds, with respect to some fixed $\frac{1}{4} < \gamma \leq 1$ and where we let $\alpha := \frac{1}{\gamma - \frac{1}{4}}$:

i) $\mu_{ij}^2 \leq \frac{\alpha - 1}{\alpha}$ for all $1 \leq j < i - 1 \leq n$, and

ii) $|\mu_{i,i-1}| \leq \frac{1}{2}$ for all $i \in \{2, \ldots, n\}$, as well as

iii) Lovasz condition: $\|b_i'\|^2 \geq (\gamma - \mu_{i,i-1}^2) \cdot \|b_{i-1}'\|^2$ for all $i \in \{2, \ldots, n\}$.

Note that the Lovasz condition is equivalent to $\|b_i' + b_{i-1}' \mu_{i,i-1}\|^2 = \|b_i'\|^2 + \mu_{i,i-1}^2 \cdot \|b_{i-1}'\|^2 \geq \gamma \cdot \|b_{i-1}'\|^2$, for all $i \in \{2, \ldots, n\}$, where $b_i' + b_{i-1}' \mu_{i,i-1} \in \mathbb{R}^n$

and $b'_{i-1} \in \mathbb{R}^n$ are the images of $b_i \in \mathbb{R}^n$ and $b_{i-1} \in \mathbb{R}^n$, respectively, under the projection $\mathbb{R}^n \to U_{i-2}^\perp$ in (5.2). Moreover, as $\frac{1}{4} < \gamma \leq 1$ varies, we have $\alpha \geq \frac{4}{3}$, and hence $\frac{\alpha-1}{\alpha} \geq \frac{1}{4}$. Thus condition (i) is fulfilled whenever $|\mu_{ij}| \leq \frac{1}{2}$ for all $1 \leq j < i-1 \leq n$, unifying conditions (i) and (ii). Note that the typical choice is $\gamma = \frac{3}{4}$, yielding $\alpha = 2$.

(5.7) Proposition. Using the notation of (5.6), let B be LLL reduced. Then:
a) For $1 \leq j \leq i \leq n$ we have $\|b_j\| \leq \alpha^{\frac{i-1}{2}} \cdot \|b'_i\|$.
b) We have $\|b_1\| \leq \alpha^{\frac{n-1}{4}} \cdot \det(L)^{\frac{1}{n}}$.
c) We have $\|b_1\| \leq \alpha^{\frac{n-1}{2}} \cdot \min(L)$.
d) We have $\det(L) \leq \prod_{i=1}^n \|b_i\| \leq \alpha^{\frac{n(n-1)}{4}} \cdot \det(L)$.

Proof. For $i \in \{2, \ldots, n\}$ we have $\|b'_i\|^2 \geq (\gamma - \frac{1}{4}) \cdot \|b'_{i-1}\|^2 = \frac{1}{\alpha} \cdot \|b'_{i-1}\|^2$. Hence for $1 \leq j \leq i \leq n$ we get $\|b'_j\|^2 \leq \alpha^{i-j} \cdot \|b'_i\|^2$. Thus $\|b_i\|^2 = \|b'_i\|^2 + \sum_{j=1}^{i-1} \mu_{ij}^2 \cdot \|b'_j\|^2 \leq (1 + (\alpha - 1) \cdot \sum_{j=1}^{i-1} \alpha^{i-j-1}) \cdot \|b'_i\|^2 = \alpha^{i-1} \cdot \|b'_i\|^2$. Hence for $1 \leq j \leq i \leq n$ we have $\|b_j\|^2 \leq \alpha^{j-1} \cdot \|b'_j\|^2 \leq \alpha^{j-1} \cdot \alpha^{i-j} \cdot \|b'_i\|^2 = \alpha^{i-1} \cdot \|b'_i\|^2$, proving a). This yields $\|b_1\|^{2n} \leq \prod_{i=1}^n \alpha^{i-1} \cdot \|b'_i\|^2 = \alpha^{\frac{n(n-1)}{2}} \cdot \prod_{i=1}^n \|b'_i\|^2 = \alpha^{\frac{n(n-1)}{2}} \cdot \det(L)^2$, proving b).

Let $0 \neq v = \sum_{i=1}^k b_i \nu_i = \sum_{i=1}^k b'_i \nu'_i \in L$, for suitable $\nu_i \in \mathbb{Z}$ and $\nu'_i \in \mathbb{R}$, and where $k \in \{1, \ldots, n\}$ is chosen such that $\nu_k \neq 0$, hence we have $\nu'_k = \nu_k \in \mathbb{Z}$. Thus $\|v\|^2 = \sum_{i=1}^k \nu_i'^2 \cdot \|b'_i\|^2 \geq \nu_k'^2 \cdot \|b'_k\|^2 \geq \|b'_k\|^2 \geq \alpha^{-(k-1)} \cdot \|b_1\|^2 \geq \alpha^{-(n-1)} \cdot \|b_1\|^2$, proving c). Finally, the first inequality in d) holds anyway, and we have $\prod_{i=1}^n \|b_i\|^2 \leq \prod_{i=1}^n \alpha^{i-1} \cdot \|b'_i\|^2 = \alpha^{\frac{n(n-1)}{2}} \cdot \prod_{i=1}^n \|b'_i\|^2 = \alpha^{\frac{n(n-1)}{2}} \cdot \det(L)^2$, proving the second inequality in d). ♯

(5.8) Algorithm: LLL, Lenstra-Lenstra-Lovasz (1982).
Let $L \subseteq \mathbb{R}^n$ be a $\mathbb{Z}$-lattice having $\mathbb{Z}$-basis $B = \{b_1, \ldots, b_n\} \subseteq \mathbb{R}^n$, let $B' = \{b'_1, \ldots, b'_n\} \subseteq \mathbb{R}^n$ be the associated Gram-Schmidt $\mathbb{R}$-basis, and let $\frac{1}{4} < \gamma \leq 1$.

```
1. $k \leftarrow 2$
2. while $k \leq n$ do
3.     for $l \in [k-1, \ldots, 1]$ do        # size reduction
           $\mu_{kl} \leftarrow \frac{\langle b_k, b'_l \rangle}{\|b'_l\|^2}$
           $b_k \leftarrow b_k - b_l \cdot \lceil \mu_{kl} \rfloor$
4.     $\mu_{k,k-1} \leftarrow \frac{\langle b_k, b'_{k-1} \rangle}{\|b'_{k-1}\|^2}$
       $\beta_k \leftarrow \|b'_k\|^2 + \mu_{k,k-1}^2 \cdot \|b'_{k-1}\|^2$
       if $\beta_k < \gamma \cdot \|b'_{k-1}\|^2$ then        # check Lovasz condition
           $b_k \leftrightarrow b_{k-1}$        # swap
           $b \leftarrow b'_k + b'_{k-1} \cdot \mu_{k,k-1}$
           $b'_k \leftarrow b'_{k-1} \cdot \frac{\|b'_k\|^2}{\beta_k} - b'_k \cdot \frac{\mu_{k,k-1} \cdot \|b'_{k-1}\|^2}{\beta_k}$
           $b'_{k-1} \leftarrow b$
           if $k \geq 3$ then $k \leftarrow k-1$
```

```
    else $k \leftarrow k+1$
5. return $[b_1, \ldots, b_n]$
```

Here, for $x \in \mathbb{R}$ we let $\lceil x \rfloor := \lfloor x + \frac{1}{2} \rfloor \in \mathbb{Z}$ be the **nearest integer** function.

a) The LLL algorithm successively modifies B and B', where $B \subseteq L$ always is a $\mathbb{Z}$-basis, and where we show that $B' \subseteq \mathbb{R}^n$ always is the associate Gram-Schmidt $\mathbb{R}$-basis; note that the related numbers $\mu_{ij} \in \mathbb{R}$ and $\|b'_i\| \in \mathbb{R}$ are always recomputed using the current sets B and B': In step 3, since $b_l \in U_{k-1}$, where $U_i \leq \mathbb{R}^n$ for $i \in \{0, \ldots, n\}$ is as in (5.2), the Gram-Schmidt $\mathbb{R}$-basis $B' \subseteq \mathbb{R}^n$ is unchanged. Hence after step 3 we have $|\mu_{kl}| = \frac{|\langle b_k, b'_l \rangle|}{\|b'_l\|^2} \leq \frac{1}{2}$, for all $1 \leq l < k$.

In step 4, let $b''_{k-1}, b''_k \in \mathbb{R}^n$ be the elements to replace $b'_{k-1}, b'_k \in B'$ after exchanging $b_{k-1}, b_k \in \mathbb{R}^n$; note that the other elements of B' are unchanged. Since b''_{k-1} is the image of b_k under the projection $V \to U_{k-2}^{\perp}$, we have $b''_{k-1} = b'_k + b'_{k-1} \cdot \mu_{k,k-1}$. Moreover, since b''_k is the image of b_{k-1} under the projection $V \to (U_{k-2} + \langle b_k \rangle_{\mathbb{R}})^{\perp}$, we have $b''_k = b'_{k-1} - b''_{k-1} \cdot \frac{\langle b_{k-1}, b''_{k-1} \rangle}{\|b''_{k-1}\|^2}$. We have $\|b''_{k-1}\|^2 = \|b'_k\|^2 + \mu_{k,k-1}^2 \cdot \|b'_{k-1}\|^2 = \beta_k$. Since $b_{k-1} \in U_{k-1}$ and $b'_k \in U_{k-1}^{\perp}$, as well as $b_{k-1} - b'_{k-1} \in U_{k-2}$ and $b'_{k-1} \in U_{k-2}^{\perp}$, we have $\langle b_{k-1}, b''_{k-1} \rangle = \langle b_{k-1}, b'_k \rangle + \mu_{k,k-1} \cdot \langle b_{k-1}, b'_{k-1} \rangle = \mu_{k,k-1} \cdot \|b'_{k-1}\|^2$. Using this we finally obtain $b''_k = b'_{k-1} - (b'_k + b'_{k-1} \cdot \mu_{k,k-1}) \cdot \frac{\mu_{k,k-1} \cdot \|b'_{k-1}\|^2}{\beta_k} = b'_{k-1} \cdot (1 - \frac{\mu_{k,k-1}^2 \cdot \|b'_{k-1}\|^2}{\beta_k}) - b'_k \cdot \frac{\mu_{k,k-1} \cdot \|b'_{k-1}\|^2}{\beta_k} = b'_{k-1} \cdot \frac{\|b'_k\|^2}{\beta_k} - b'_k \cdot \frac{\mu_{k,k-1} \cdot \|b'_{k-1}\|^2}{\beta_k}$. Thus, if the LLL algorithm terminates, then the Lovasz condition is fulfilled as well, and hence $B \subseteq L$ is an LLL reduced $\mathbb{Z}$-basis, in particular thus proving the existence of LLL reduced $\mathbb{Z}$-bases.

b) We show that the LLL algorithm terminates: For $l \in \{1, \ldots, n\}$ let $Q_l := [\langle b_i, b_j \rangle; i, j \in \{1, \ldots, l\}]_{ij} \in \mathbb{R}^{l \times l}$ be the Gram matrix of $\langle \cdot, \cdot \rangle|_{U_l \times U_l}$, and let $d_l := \det(Q_l)$. Hence we have $d_l := \prod_{i=1}^{l} \|b'_i\|^2 > 0$, and in particular $d_n = \det(L)^2$. Thus $d := \prod_{l=1}^{n} d_l > 0$ changes only if swapping, involving b'_k and b'_{k-1} say, occurs in step 4: In this case we have $\|b''_{k-1}\|^2 = \beta_k$ and $\|b''_k\|^2 = \frac{\|b'_k\|^4}{\beta_k^2} \cdot \|b'_{k-1}\|^2 + \frac{\mu_{k,k-1}^2 \cdot \|b'_{k-1}\|^4}{\beta_k^2} \cdot \|b'_k\|^2 = \|b'_{k-1}\|^2 \cdot \|b'_k\|^2 \cdot \frac{\|b'_k\|^2 + \mu_{k,k-1}^2 \cdot \|b'_{k-1}\|^2}{\beta_k^2} = \frac{\|b'_{k-1}\|^2 \cdot \|b'_k\|^2}{\beta_k}$, thus we obtain $\|b''_{k-1}\|^2 \cdot \|b''_k\|^2 = \|b'_{k-1}\|^2 \cdot \|b'_k\|^2$. Hence d_l is unchanged for $l \leq k-2$ or $l \geq k$, while since $\beta_k < \gamma \cdot \|b'_{k-1}\|^2$ the number d_{k-1} is multiplied by $\frac{\|b''_{k-1}\|^2}{\|b'_{k-1}\|^2} = \frac{\beta_k}{\|b'_{k-1}\|^2} < \gamma \leq 1$, hence $d > 0$ also is multiplied by that number.

If $Q \in \mathbb{Z}^{n \times n}$, i. e. we have an **integral** lattice, then we always have $d \in \mathbb{Z}$. Since d becomes strictly smaller in each swapping step, this possibly occurs only finitely many times, hence the LLL algorithm terminates in this case.

In the general case $Q \in \mathbb{R}^{n \times n}$ we use the **Hermite constant (1846)** $\gamma_n > 0$, for which $\min(L) \leq \gamma_n^{\frac{1}{2}} \cdot \det(L)^{\frac{1}{n}} = \gamma_n^{\frac{1}{2}} \cdot \det(Q)^{\frac{1}{2n}}$ for any $\mathbb{Z}$-lattice $L \subseteq \mathbb{R}^n$, and which in this sense is best possible; see [2, Prop.6.4.1] or [8, Exc.3.3.4.9]. Since $\min(\langle b_1, \ldots, b_l \rangle_{\mathbb{Z}}) \geq \min(L)$, this yields $d_l = \det(Q_l) \geq (\frac{\min(L)^2}{\gamma_l})^l$, for all

$l \in \{1, \ldots, n\}$, and thus $d \geq \min(L)^{n(n+1)} \cdot \prod_{l=1}^{n} \frac{1}{\gamma_l^l} > 0$. Hence if additionally $\gamma < 1$, then in the general case swapping possibly occurs only finitely many times, and the LLL algorithm terminates.

c) Similarly to the Gram-Schmidt orthogonalization, it is possible to start with the Gram matrix $Q \in \mathbb{R}^{n \times n}$ alone, where the suitably adjusted LLL algorithm returns the LLL reduced basis in terms of a base change matrix, see Exercise (8.37). The larger the parameter γ is chosen, the better the LLL reduced basis becomes, but the longer the LLL algorithm runs, although both aspects seem to be rather insensitive to the value of γ, see [2, Ch.2.6.1]. Note that if $B \subseteq \mathbb{Z}^n$ then for the associated Gram matrix we have $Q \in \mathbb{Z}^{n \times n}$ as well, and thus the LLL algorithm completely runs over $\mathbb{Q}$; moreover, the analysis in the proof of (5.9) already indicates the denominators actually occurring, opening up a way to a version of the LLL algorithm running completely over $\mathbb{Z}$, see [2, Ch.2.6.3].

(5.9) Theorem. Let $L \subseteq \mathbb{R}^n$ be a $\mathbb{Z}$-lattice having $\mathbb{Z}$-basis $B = \{b_1, \ldots, b_n\} \subseteq \mathbb{Z}^n$ such that $\|b_i\| \leq A$ for all $i \in \{1, \ldots, n\}$, for some $A > 0$; note that hence the input length of L is in $O(n^2 \ln(A))$. Then the LLL algorithm, with parameter $\frac{1}{4} < \gamma < 1$, needs at most $O(n^4 \ln(A))$ ring operations in $\mathbb{Q}$, where the occurring numerators and denominators have bit length in $O(n \ln(A))$.

Proof. Each evaluation of $\langle \cdot, \cdot \rangle$, and hence of $\| \cdot \|^2$, needs $O(n)$ ring operations in $\mathbb{Q}$. Adding a $\mathbb{Q}$-multiple of a vector to another vector also needs $O(n)$ ring operations in $\mathbb{Q}$. Hence each execution of step 3 needs $O(n^2)$ ring operations in $\mathbb{Q}$, and thus each execution of step 2 as well. Let $B' = \{b'_1, \ldots, b'_n\}$ be the Gram-Schmidt $\mathbb{R}$-basis associated to the current $\mathbb{Z}$-basis B during execution of the LLL algorithm. As in (5.8) let $d_l = \prod_{i=1}^{l} \|b'_i\|^2 \in \mathbb{N}$, for $l \in \{1, \ldots, n\}$, and $d_0 := 1$, as well as $d = \prod_{l=1}^{n} d_l \in \mathbb{N}$. Hence from $\|b'_i\| \leq \|b_i\|$, for $i \in \{1, \ldots, n\}$, we for the initial $\mathbb{Z}$-basis B conclude $d_l \leq \prod_{i=1}^{l} \|b_i\|^2 \leq A^{2l} \leq A^{2n}$, and thus $d = \prod_{l=1}^{n} d_l \leq \prod_{l=1}^{n} A^{2l} = A^{n(n+1)}$. Hence swapping occurs at most $\log_{\frac{1}{\gamma}}(A^{n(n+1)}) \in O(n^2 \ln(A))$ times. Thus this needs at most $O(n^4 \ln(A))$ ring operations in $\mathbb{Q}$. Note that at the very beginning, the Gram-Schmidt $\mathbb{R}$-basis B' associated to the initial $\mathbb{Z}$-basis B has to be computed; this needs $O(n \cdot \frac{n(n-1)}{2}) = O(n^3)$ ring operations in $\mathbb{Q}$.

To estimate the occurring numerators and denominators we derive bounds for $\|b_k\|$ and $\|b'_k\|$ as well as $|\mu_{kl}|$, for $k \in \{1, \ldots, n\}$ and $1 \leq l < k \leq n$, and show that certain $\mathbb{Z}$-multiples of b'_k and μ_{kl} are in $\mathbb{Z}^n$ and $\mathbb{Z}$, respectively:

Let ${}_{B'}\mathrm{id}_B = [\lambda_{ij}] \in \mathbb{Q}^{n \times n}$; note that ${}_{B'}\mathrm{id}_B$ is lower unitriangular. Hence for $1 \leq l < k \leq n$ we from $\langle b_l, b'_k \rangle = 0$ obtain $\sum_{j=1}^{k-1} \lambda_{kj} \cdot \langle b_l, b_j \rangle = -\langle b_l, b_k \rangle \in \mathbb{Z}$. Hence $[\lambda_{k1}, \ldots, \lambda_{k,k-1}] \in \mathbb{Q}^{k-1}$ is the solution of a system of $\mathbb{Q}$-linear equations with associated matrix $Q_{k-1} \in \mathbb{Z}^{(k-1) \times (k-1)}$, thus by Cramer's rule we have $d_{k-1}\lambda_{kj} \in \mathbb{Z}$, for $1 \leq j < k \leq n$. Hence we also have $b'_k d_{k-1} \in \mathbb{Z}^n$, for $k \in \{1, \ldots, n\}$, and thus the denominators of the entries of $b'_k \in \mathbb{Q}^n$ are absolutely bounded by $d_{k-1} \leq A^{2n}$, thus these have bit lengths in $O(n \ln(A))$.

Moreover, for $1 \leq l < k \leq n$ we obtain $d_l \mu_{kl} = d_l \cdot \frac{\langle b_k, b'_l \rangle}{\|b'_l\|^2} = d_{l-1} \cdot \langle b_k, b'_l \rangle = \langle b_k, b'_l d_{l-1} \rangle \in \mathbb{Z}$. Thus the denominator of $\mu_{kl} \in \mathbb{Q}$ is absolutely bounded by $d_l \leq A^{2n}$, thus also have bit length in $O(n \ln(A))$; and as $|\mu_{kl}| \leq \frac{1}{2}$ outside step 3 this estimate also holds for the numerators of those $\mu_{kl} \in \mathbb{Q}$.

Initially we have $\|b'_k\| \leq \|b_k\| \leq A$, for $k \in \{1, \ldots, n\}$, Moreover, for swapping for some $k \geq 2$ we have $\|b''_{k-1}\|^2 = \beta_k < \gamma \cdot \|b'_{k-1}\|^2 \leq \|b'_{k-1}\|^2$, and since b''_k is the image of b_{k-1} under the projection $V \to (U_{k-2} + \langle b_k \rangle_{\mathbb{R}})^\perp$, while b'_{k-1} is the image of b_{k-1} under the projection $V \to U_{k-2}^\perp$, we have $\|b''_k\| \leq \|b'_{k-1}\|$ as well. Hence we have $\|b'_k\| \leq A$, for $k \in \{1, \ldots, n\}$. Thus from $b'_k d_{k-1} \in \mathbb{Z}^n$ we conclude that the numerators of the entries of b'_k are absolutely bounded by $\|b'_k d_{k-1}\| \leq A^{2n+1}$, hence have bit lengths in $O(n \ln(A))$.

Moreover, outside step 3 we have $|\mu_{kl}| \leq \frac{1}{2}$ for all $1 \leq l < k \leq n$, and hence letting $\mu_{kk} := 1$ yields $\|b_k\|^2 = \sum_{l=1}^k \mu_{kl}^2 \cdot \|b'_l\|^2 \leq nA^2$, hence $\|b_k\| \leq n^{\frac{1}{2}} \cdot A$; note that if b_k has not been touched at all, we have $\|b_k\| \leq A$ anyway. Thus the entries in those $b_k \in \mathbb{Z}^n$ have bit lengths in $O(\ln(n) + \ln(A)) \subseteq O(n \ln(A))$.

Hence it remains to consider the behavior of size reduction: At the beginning of step 3 let $m_k := \max\{|\mu_{kl}|; l \in \{1, \ldots, k\}\}$, where again $\mu_{kk} := 1$. Since for $l \in \{1, \ldots, n\}$ we have $\|b'_l\| = (\frac{d_l}{d_{l-1}})^{\frac{1}{2}} \geq (\frac{1}{d_{l-1}})^{\frac{1}{2}}$, for $1 \leq l < k \leq n$ by the Cauchy-Schwarz inequality we get $|\mu_{kl}| = \frac{\langle b_k, b'_l \rangle}{\|b'_l\|^2} \leq \frac{\|b_k\| \cdot \|b'_l\|}{\|b'_l\|^2} \leq d_{l-1}^{\frac{1}{2}} \cdot \|b_k\|$, and thus $m_k \leq \max\{d_{l-1}^{\frac{1}{2}}; l \in \{1, \ldots, k-1\}\} \cdot \|b_k\| \leq A^{n-2} \cdot \|b_k\| \leq n^{\frac{1}{2}} \cdot A^{n-1}$.

During size reduction, for intermediate $\widetilde{\mu}_{kl}$, where $1 \leq l \leq k \leq n$, we have $|\widetilde{\mu}_{k,l}| \leq 2^{k-l} m_k$: This holds true for $l = k$ and $l = k-1$, and for $l \leq k-1$ we have $\widetilde{\mu}_{k,l-1} = \mu_{k,l-1} - \lceil \widetilde{\mu}_{k,l} \rfloor \cdot \mu_{l,l-1}$. Hence by the triangle inequality, and using $m_k + \frac{1}{4} \leq 3 \cdot 2^{k-l-1} m_k$, we get $|\widetilde{\mu}_{k,l-1}| \leq m_k + (2^{k-l} m_k + \frac{1}{2}) \cdot \frac{1}{2} = (1 + 2^{k-l-1}) m_k + \frac{1}{4} \leq 4 \cdot 2^{k-l-1} m_k = 2^{k-(l-1)} m_k$. In particular, we have $|\widetilde{\mu}_{k,l}| \leq 2^{n-1} m_k \leq n^{\frac{1}{2}} \cdot (2A)^{n-1}$. Since $d_l \widetilde{\mu}_{k,l} \in \mathbb{Z}$ the numerator of $\widetilde{\mu}_{k,l}$ is absolutely bounded by $|d_l \widetilde{\mu}_{k,l}| \leq n^{\frac{1}{2}} \cdot 2^{n-1} A^{3n-1}$, hence has bit length in $O(n \ln(A))$.

Finally, for intermediate $\widetilde{b}_k$ during size reduction we have $\|\widetilde{b}_k\|^2 = \sum_{l=1}^k \widetilde{\mu}_{kl}^2 \cdot \|b'_l\|^2 \leq n \cdot n(2A)^{2(n-1)} \cdot A^2 \leq 2^{2(n-1)} n^2 A^{2n}$, hence $\|\widetilde{b}_k\| \leq 2^{n-1} n A^n$. Thus the entries in $\widetilde{b}_k \in \mathbb{Z}^n$ have bit lengths in $O(n \ln(A))$. ♯

For variants of the LLL algorithm, such as the modified LLL algorithm which accepts also linear dependent vectors as input, and applications of these algorithms, such as the computation of minimal vectors or the enumeration of short vectors in lattices, or the computation of kernels and images of integer matrices, or the computation of minimal polynomials of algebraic integers, see [2, Ch.2.6, 2.7]. A detailed discussion of successive minima of lattices and related lattice bases is given in [15, Ch.3.3]. Here we only present the following application:

(5.10) Example: Simultaneous Diophantine approximation.
Let $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$. Then by a Theorem of Dirichlet, there are infinitely many

tuples $[q; p_1, \ldots, p_n] \in \mathbb{Z}^{n+1}$ such that $|\alpha_i - \frac{p_i}{q}| \leq q^{-\frac{n+1}{n}}$, for all $i \in \{1, \ldots, n\}$. Finding simultaneous approximations is interpreted as a short vector problem:

Let $\beta_1, \ldots, \beta_n \in \mathbb{Q}$ be approximations of the $\alpha, \ldots, \alpha_n$, respectively, where the β_i need not have the same denominator. Moreover, let $0 < \epsilon < 1$ and $c := 2^{-\frac{n(n+1)}{4}} \cdot \epsilon^{n+1}$, and let $L := \langle b_0, \ldots, b_n \rangle_{\mathbb{Z}} \subseteq \mathbb{R}^{n+1}$, where the $b_i \in \mathbb{R}^{n+1}$ are the rows of the following matrix:

$$B := \begin{bmatrix} c & \beta_1 & \beta_2 & \ldots & \beta_n \\ . & -1 & . & \ldots & . \\ . & . & -1 & \ldots & . \\ \vdots & & & \ddots & \vdots \\ . & . & . & \ldots & -1 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}.$$

Let $\{\widehat{b}_0, \ldots, \widehat{b}_n\} \subseteq L$ be an LLL reduced $\mathbb{Z}$-basis, with respect to the parameter $\gamma = \frac{3}{4}$. Hence we $\|\widehat{b}_0\| \leq 2^{\frac{n}{4}} \cdot \det(L)^{\frac{1}{n+1}}$, and since $\det(L) = |\det(B)| = c$ we conclude $\|\widehat{b}_0\| \leq \epsilon < 1$. There are $q, p_1, \ldots, p_n \in \mathbb{Z}$ such that $\widehat{b}_0 = b_0 q + \sum_{i=1}^{n} b_i p_i = [qc, q\beta_1 - p_1, \ldots, q\beta_n - p_n] \in L$. We may assume that $q \geq 0$. Moreover, assume that $q = 0$, then $\|\widehat{b}_0\|^2 = \sum_{i=1}^{n} p_i^2 \geq 1$, a contradiction. Hence we have $q \geq 1$. Then we have $2^{-\frac{n(n+1)}{4}} \cdot \epsilon^{n+1} q = qc \leq \|\widehat{b}_0\| \leq \epsilon$, hence $q \leq 2^{\frac{n(n+1)}{4}} \cdot \epsilon^{-n}$, thus $\epsilon \leq 2^{\frac{n+1}{4}} \cdot q^{-\frac{1}{n}}$. This yields $|\beta_i - \frac{p_i}{q}| \leq \frac{1}{q} \cdot \|\widehat{b}_0\| \leq \frac{\epsilon}{q} \leq 2^{\frac{n+1}{4}} \cdot q^{-\frac{n+1}{n}}$. Note that this approximates the approximations β_i of the α_i, and is weaker than Dirichlet's bound by a factor of $2^{\frac{n+1}{4}}$.

E. g. we consider the musical scale: In the well-tempered scale the idea is to divide the octave into finitely many equal half tones, $q \in \mathbb{N}$ say, such that the natural intervals, with frequency ratio $r \in \mathbb{Q}$ say, are approximated well by an integral number of half tones, $p \in \mathbb{N}$ say, i. e. we would like to minimize $|2^{\frac{p}{q}} - r|$, which amounts to minimize $|\log_2(r) - \frac{p}{q}|$. The natural intervals considered are the octave itself, the fifth, the fourth, the major third, the minor third and major second, whose frequency ratios are given as follows, together with approximations of their binary logarithms up to 3 decimals; we also indicate the corresponding number of half tones in the well-tempered scale:

i	r_i	$\log_2(r_i) \sim$	p_i
1	$\frac{2}{1}$	1	12
2	$\frac{3}{2}$	$0,585$	7
3	$\frac{4}{3}$	$0,415$	5
4	$\frac{5}{4}$	$0,322$	4
5	$\frac{6}{5}$	$0,263$	3
6	$\frac{9}{8}$	$0,167$	2

As $\log_2(r_1) = 1$ anyway, the task is to find $q \in \mathbb{N}$ small and $p_2, \ldots, p_6 \in \mathbb{N}$ such that $|\log_2(r_i) - \frac{p_i}{q}|$ is minimized. Note that we have $\log_2(r_2) + \log_2(r_3) =$

$\log_2(r_2 r_3) = 1$, hence we could ignore r_2 as well. Anyway, let $n := 5$. To find the parameter $c = 2^{-\frac{n(n+1)}{4}} \cdot \epsilon^{n+1} \leq 2^{-\frac{n(n+1)}{4}} \cdot (2^{\frac{n+1}{4}} \cdot q^{-\frac{1}{n}})^{n+1} = 2^{\frac{n+1}{4}} \cdot q^{-\frac{n+1}{n}}$ we proceed as follows: We would like to have $1 \leq q \leq 100$, say. For $q = 100$ and $n = 5$ this inequality is fulfilled whenever $c \leq 0,0112$. Hence we choose $c := \frac{1}{100}$, and let

$$B := \begin{bmatrix} \frac{1}{100} & \frac{585}{1000} & \frac{415}{1000} & \frac{322}{1000} & \frac{263}{1000} & \frac{167}{1000} \\ . & -1 & . & . & . & . \\ . & . & -1 & . & . & . \\ . & . & . & -1 & . & . \\ . & . & . & . & -1 & . \\ . & . & . & . & . & -1 \end{bmatrix} \in \mathbb{Q}^{6\times 6}.$$

The LLL algorithm yields the following $\mathbb{Z}$-basis $\widehat{B} := \{\widehat{b}_0, \ldots, \widehat{b}_5\} \subseteq \mathbb{Q}^6$, which decomposes into the $\mathbb{Z}$-basis B as indicated by ${}_{\widehat{B}}\mathrm{id}_B \in \mathbb{Z}^{6\times 6}$:

$$\widehat{B} := \begin{bmatrix} \frac{3}{25} & \frac{1}{50} & \frac{-1}{50} & \frac{-17}{125} & \frac{39}{250} & \frac{1}{250} \\ \frac{19}{100} & \frac{23}{200} & \frac{-23}{200} & \frac{59}{500} & \frac{-3}{1000} & \frac{173}{1000} \\ \frac{-11}{50} & \frac{13}{100} & \frac{-13}{100} & \frac{-21}{250} & \frac{107}{500} & \frac{163}{500} \\ \frac{3}{20} & \frac{-9}{40} & \frac{9}{40} & \frac{-17}{100} & \frac{-11}{200} & \frac{101}{200} \\ \frac{17}{100} & \frac{-11}{200} & \frac{11}{200} & \frac{237}{500} & \frac{471}{1000} & \frac{-161}{1000} \\ \frac{-9}{50} & \frac{47}{100} & \frac{53}{100} & \frac{51}{250} & \frac{133}{500} & \frac{-3}{500} \end{bmatrix} \in \mathbb{Q}^{6\times 6},$$

$${}_{\widehat{B}}\mathrm{id}_B := \begin{bmatrix} 12 & 7 & 5 & 4 & 3 & 2 \\ 19 & 11 & 8 & 6 & 5 & 3 \\ -22 & -13 & -9 & -7 & -6 & -4 \\ 15 & 9 & 6 & 5 & 4 & 2 \\ 17 & 10 & 7 & 5 & 4 & 3 \\ -18 & -11 & -8 & -6 & -5 & -3 \end{bmatrix} \in \mathbb{Z}^{6\times 6}.$$

Hence good rational approximations, due to $\widehat{b}_0$ are found taking $q := 12$; and indeed in the well-tempered scale the octave is divided into 12 half tones. Hence the p_i found coincide with the number of half tones into which the natural intervals actually are divided. Note that the second vector $\widehat{b}_1$ also yields a good rational approximations, which mean a division of the octave into 19 third tones, and the indicated number of third tones for the natural intervals; even more, the third tone scale allows to distinguish intervals e. g. such as the minor third and augmented second, which have 5 and 4 third tones, respectively, but in the well-tempered scale both have 3 half tones.

6 Polynomial factorization over finite fields

(6.1) Algorithm: Squarefree factorization.
Let $p \in \mathbb{N}$ be a prime, let $q = p^f$ for some $f \in \mathbb{N}$, and let $0 \neq \Psi \in \mathbb{F}_q[X]$ be monic such that $\deg(\Psi) = n$. The ultimate aim is to find the **prime power factorization** $\Psi = \prod_{k=1}^{r} \Phi_k^{e_k}$, where $\Phi_1, \ldots, \Phi_r \in \mathbb{F}_q[X]$ are pairwise different irreducible monic polynomials, and where $e_k = e_{\Phi_k}(\Psi) \in \mathbb{N}$ is called the corresponding **multiplicity**; note that since $\mathbb{F}_q[X]$ is factorial the prime power factorization is uniquely defined, and the prime polynomials are precisely the irreducible ones. The polynomial Ψ is called **squarefree** if $e_k = 1$ for all $k \in \{1, \ldots, r\}$.

The **Frobenius map** $\varphi_p \colon \mathbb{F}_q \rightarrow \mathbb{F}_q \colon \alpha \mapsto \alpha^p$ is a field automorphism, where we even have $\mathrm{Gal}(\mathbb{F}_q/\mathbb{F}_p) = \langle \varphi_p \rangle$, which hence is a cyclic group of order f. We conclude that $\mathbb{F}_q$ is a **perfect field**, i. e. all finite field extensions of $\mathbb{F}_q$ are **separable**. Hence we conclude that $0 \neq \Psi \in \mathbb{F}_q[X]$ is squarefree if and only if $\Psi \in \overline{\mathbb{F}}[X]$ does not have multiple roots, where $\mathbb{F}_q \subseteq \overline{\mathbb{F}}$ is an algebraic closure of $\mathbb{F}_q$. The latter by (4.7) holds if and only if $0 \neq \mathrm{disc}(\Psi) \in \mathbb{F}_q$, which in turn holds if and only if $0 \neq \mathrm{res}(\Psi, \Psi') \in \mathbb{F}_q$, where $\Psi' = \frac{\partial \Psi}{\partial X} \in \mathbb{F}_q[X]$ denotes the formal derivative. The latter by (4.4) holds if and only if $\gcd(\Psi, \Psi') = 1 \in \mathbb{F}_q[X]$.

We have $\Psi = \prod_{e=1}^{n} \Psi_e^e$, where $\Psi_e := \prod_{k \in \mathcal{K}_e} \Phi_k \in \mathbb{F}_q[X]$ and $\mathcal{K}_e := \{l \in \{1, \ldots, r\}; e_l = e\}$; note that the Ψ_e are squarefree and pairwise coprime for $e \in \{1, \ldots, n\}$. The polynomials Ψ_e, for $e \in \{1, \ldots, n\}$, are found as follows:

Let $\Theta := \gcd(\Psi, \Psi') \in \mathbb{F}_q[X]$ monic, and let $\Phi \in \mathbb{F}_q[X]$ be monic irreducible such $\Phi \mid \Theta$. Hence there is a unique $e \in \{1, \ldots, n\}$ such that $\Phi \mid \Psi_e$, and $\Phi = \Phi_k$ for some $k \in \mathcal{K}_e$; note that hence $\Psi_e \neq 1$. We have $\Psi' = \sum_{e=1}^{n} \left(e\Psi_e' \Psi_e^{e-1} \cdot \prod_{d \in \{1, \ldots, n\} \setminus \{e\}} \Psi_d^d \right)$. Thus for $e \neq c \in \{1, \ldots, n\}$ we have the multiplicity $e_\Phi(\prod_{d \in \{1, \ldots, n\} \setminus \{c\}} \Psi_d^d) = e$, and hence for the corresponding summand of Ψ' we have $e_\Phi(c\Psi_c' \Psi_c^{c-1} \cdot \prod_{d \in \{1, \ldots, n\} \setminus \{c\}} \Psi_d^d) \geq e$. Moreover, for $p \mid e$ the e-th summand of Ψ' vanishes anyway, while for $p \not| \, e$ we have $e_\Phi(e\Psi_e' \Psi_e^{e-1} \cdot \prod_{d \in \{1, \ldots, n\} \setminus \{e\}} \Psi_d^d) = e - 1$; note that since Ψ_e is squarefree we have $\gcd(\Psi_e, \Psi_e') = 1$. Thus $e_\Phi(\Theta) = e - 1$ if $p \not| \, e$, and since $\Theta \mid \Psi$ we have $e_\Phi(\Theta) = e$ if $p \mid e$. This yields $\Theta = \prod_{e \in \{1, \ldots, n\}, p \not| e} \Psi_e^{e-1} \cdot \prod_{e \in \{1, \ldots, n\}, p \mid e} \Psi_e^e$.

To actually compute the squarefree factors Ψ_e, for $e \in \{1, \ldots, n\}$, we by induction define $\Theta_k \in \mathbb{F}_q[X]$ and $\Lambda_k \in \mathbb{F}_q[X]$ monic, for $k \in \mathbb{N}$, as follows: Let $\Theta_1 := \Theta = \gcd(\Psi, \Psi') \in \mathbb{F}_q[X]$ and $\Lambda_1 := \frac{\Psi}{\Theta} \in \mathbb{F}_q[X]$, and for $k \geq 1$ we let $\Lambda_{k+1} := \gcd(\Theta_k, \Lambda_k) \in \mathbb{F}_q[X]$ if $p \not| \, k$, and $\Lambda_{k+1} := \Lambda_k \in \mathbb{F}_q[X]$ if $p \mid k$, as well as $\Theta_{k+1} := \frac{\Theta_k}{\Lambda_{k+1}} \in \mathbb{F}_q(X) = Q(\mathbb{F}_q[X])$.

We by induction show that for $k \in \mathbb{N}$ we have $\Lambda_k = \prod_{e \in \{k, \ldots, n\}, p \not| e} \Psi_e$ and $\Theta_k = \prod_{e \in \{k, \ldots, n\}, p \not| e} \Psi_e^{e-k} \cdot \prod_{e \in \{1, \ldots, n\}, p \mid e} \Psi_e^e$; hence in particular we indeed have $\Theta_k \in \mathbb{F}_q[X]$: For $k = 1$ we have $\Lambda_1 = \frac{\Psi}{\Theta} = \prod_{e \in \{1, \ldots, n\}, p \not| e} \Psi_e$ and $\Theta_1 = \Theta = \prod_{e \in \{1, \ldots, n\}, p \not| e} \Psi_e^{e-1} \cdot \prod_{e \in \{1, \ldots, n\}, p \mid e} \Psi_e^e$. For $k \geq 1$ and $p \not| \, k$ we have $\Lambda_{k+1} = \gcd(\Theta_k, \Lambda_k)$, and since for $e \geq k$ we have $\Psi_e \mid \Theta_k$ if and only if $e \geq k + 1$, we conclude $\Lambda_{k+1} = \prod_{e \in \{k+1, \ldots, n\}, p \not| e} \Psi_e$; for $k \geq 1$ and

$p \mid k$ we have $\Lambda_{k+1} = \Lambda_k = \prod_{e\in\{k,\ldots,n\}, p\nmid e} \Psi_e = \prod_{e\in\{k+1,\ldots,n\}, p\nmid e} \Psi_e$; finally for $k \geq 1$ we have $\Theta_{k+1} = \frac{\Theta_k}{\Lambda_{k+1}} = \frac{\prod_{e\in\{k,\ldots,n\}, p\nmid e} \Psi_e^{e-k}}{\prod_{e\in\{k+1,\ldots,n\}, p\nmid e} \Psi_e} \cdot \prod_{e\in\{1,\ldots,n\}, p\,|\,e} \Psi_e^e = \prod_{e\in\{k+1,\ldots,n\}, p\nmid e} \Psi_e^{e-(k+1)} \cdot \prod_{e\in\{1,\ldots,n\}, p\,|\,e} \Psi_e^e$.

This yields $\Psi_e = \frac{\Lambda_e}{\Lambda_{e+1}}$ whenever $e \in \{1,\ldots,n\}$ such that $p \nmid e$. Hence we have computed those Ψ_e, and to obtain the Ψ_e for $p \mid e$ we proceed as follows: We have $\Lambda_k = 1$ for all $k > n$; and whenever we have $k \in \{1,\ldots,n\}$ such that $\Lambda_k = 1$, then we conclude $\Theta_k = \Theta_{k-1} = \prod_{e\in\{1,\ldots,n\}, p\,|\,e} \Psi_e^e = \left(\prod_{e\in\{1,\ldots,n\}, p\,|\,e} \Psi_e^{\frac{e}{p}}\right)^p$. Hence letting $\prod_{e\in\{1,\ldots,n\}, p\,|\,e} \Psi_e^{\frac{e}{p}} =: \widetilde{\Psi} = X^{\widetilde{n}} + \sum_{i=1}^{\widetilde{n}-1} \widetilde{\psi}_i X^i \in \mathbb{F}_q[X]$, where $\widetilde{n} \in \mathbb{N}_0$, we have $\Theta_k(X) = \widetilde{\Psi}(X)^p = X^{\widetilde{n}p} + \sum_{i=1}^{\widetilde{n}-1} \widetilde{\psi}_i^p X^{ip} \in \mathbb{F}_q[X]$, thus $\widetilde{\Psi}$ can be obtained from Θ_k by extracting p-th roots. Note that since the Frobenius map $\varphi_p \colon \mathbb{F}_q \to \mathbb{F}_q$ is a field automorphism, p-th roots always exist and are unique, and can be found using iterated application of φ_p, which has order f. Moreover, the squarefree factors of $\widetilde{\Psi}$ are $\widetilde{\Psi}_{\frac{e}{p}} = \Psi_e \in \mathbb{F}_q[X]$, where $e \in \{1,\ldots,n\}$ such that $p \mid e$. Hence the Ψ_e for $p \mid e$ can be computed from $\widetilde{\Psi}$ by recursion.

(6.2) Proposition. Let $p \in \mathbb{N}$ be a prime, let $q = p^f$ for some $f \in \mathbb{N}$, and for $n \in \mathbb{N}_0$ let $\mathcal{P}_{q,n} := \{0 \neq \Phi \in \mathbb{F}_q[X]; \Phi \text{ monic, irreducible, } \deg(\Phi) = n\}$. Then for all $d \in \mathbb{N}$ we have $X^{q^d} - X = \prod_{1\leq n\,|\,d} \prod_{\phi\in\mathcal{P}_{q,n}} \Phi \in \mathbb{F}_q[X]$. In particular, we have $\mathcal{P}_{q,n} \neq \emptyset$.

Proof. For $\alpha \in \mathbb{F}_{q^d}^*$ we have $\alpha^{q^d-1} = 1$, thus $X^{q^d} - X \in \mathbb{F}_{q^d}[X]$ has q^d pairwise distinct roots in $\mathbb{F}_{q^d}$, thus $X^{q^d} - X = \prod_{\alpha\in\mathbb{F}_{q^d}} (X - \alpha) \in \mathbb{F}_{q^d}[X]$. Hence in particular $X^{q^d} - X \in \mathbb{F}_q[X]$ is squarefree. Note that this shows that $\mathbb{F}_q \subseteq \mathbb{F}_{q^d}$ is a splitting field of the polynomial $X^{q^d} - X \in \mathbb{F}_q[X]$, proving existence and uniqueness of $\mathbb{F}_{q^d}$ whenever $\mathbb{F}_q$ exists; starting with the prime field $\mathbb{F}_p$ this shows existence and uniqueness of the finite fields $\mathbb{F}_{q^d}$.

Thus we have to show that for all $n \in \mathbb{N}$ and $\Phi \in \mathcal{P}_{q,n}$ we have $\Phi \mid X^{q^d} - X$ if and only if $n \mid d$: Let $\Phi \mid X^{q^d} - X$. Then there is $\alpha \in \mathbb{F}_{q^d}$ such that $\Phi(\alpha) = 0$, and since $\Phi \in \mathbb{F}_q[X]$ is irreducible we have $\mathbb{F}_{q^n} \cong \mathbb{F}_q[X]/\langle\Phi\rangle \cong \mathbb{F}_q(\alpha) \subseteq \mathbb{F}_{q^d}$. Since $\mathbb{F}_q \subseteq \mathbb{F}_{q^n}$ and $\mathbb{F}_q \subseteq F_{q^d}$ are field extensions of degree n and d, respectively, we conclude that $\mathbb{F}_{q^n} \subseteq F_{q^d}$ is a field extension of degree $\frac{d}{n}$, hence $n \mid d$.

Conversely, let $n \mid d$. Then $X^{q^d} - X = (X^{q^n} - X) \cdot \sum_{i=0}^{m-1} X^{i(q^n-1)} \in \mathbb{F}_q[X]$, where $m := \frac{q^d-1}{q^n-1} = \sum_{j=0}^{\frac{d}{n}-1} q^{jn} \in \mathbb{N}$. Moreover, for $\alpha := \overline{X} \in \mathbb{F}_q[X]/\langle\Phi\rangle \cong \mathbb{F}_{q^n}$ we have $\Phi(\alpha) = \Phi(\overline{X}) = \overline{\Phi(X)} = 0 \in \mathbb{F}_{q^n}$, hence $X - \alpha \mid \Phi \in \mathbb{F}_{q^n}[X]$. Since $\alpha \neq 0$ we have $\alpha^{q^n-1} = 1 \in \mathbb{F}_{q^n}$, and hence $X - \alpha \mid (X^{q^n} - X) \mid (X^{q^d} - X) \in \mathbb{F}_{q^n}[X]$ as well. Thus $X - \alpha \mid \gcd(\Phi, X^{q^d} - X) \in \mathbb{F}_{q^n}[X]$. Since both $\Phi, X^{q^d} - X \in \mathbb{F}_q[X]$, their monic gcd's in $\mathbb{F}_q[X]$ and in $\mathbb{F}_{q^n}[X]$ coincide, thus we have $\gcd(\Phi, X^{q^d} - X) \in \mathbb{F}_q[X]$ non-constant. Since $\Phi \in \mathbb{F}_q[X]$ is irreducible we conclude $\Phi \mid X^{q^d} - X \in \mathbb{F}_q[X]$. ♯

(6.3) Corollary. Let $q \in \mathbb{N}$ be a prime power and $n \in \mathbb{N}$. Then $0 \neq \Psi \in \mathbb{F}_q[X]$ monic such that $\deg(\Psi) = n$ is irreducible, if and only if $X^{q^n} \equiv X \bmod \Psi$ and $\gcd(X^{q^{\frac{n}{l}}} - X, \Psi) = 1 \in \mathbb{F}_q[X]$, for all primes $l \in \mathbb{N}$ such that $l \mid n$.

Note that $X^{q^n} \bmod \Psi$ is found using binary modular exponentiation in $\mathbb{F}_q[X]$.

Proof. If $\Psi \in \mathbb{F}_q[X]$ is irreducible, then we have $\Psi \mid X^{q^n} - X \in \mathbb{F}_q[X]$ and $\Psi \not| X^{q^{\frac{n}{l}}} - X \in \mathbb{F}_q[X]$, for all $l \mid n$. Conversely, if the above conditions are fulfilled, then for all $\Phi \in \mathbb{F}_q[X]$ irreducible such that $\Phi \mid \Psi \mid X^{q^n} - X \in \mathbb{F}_q[X]$ we have $\deg(\Phi) \mid n$; assume that $\deg(\Phi) < n$, then let $l \in \mathbb{N}$ be a prime such that $l \mid n$ and $\Phi \mid X^{q^{\frac{n}{l}}} - X \in \mathbb{F}_q[X]$, hence we have $\Phi \mid \gcd(X^{q^{\frac{n}{l}}} - X, \Psi) \in \mathbb{F}_q[X]$, a contradiction. ♯

(6.4) Algorithm: Distinct degree factorization, Zassenhaus (1969). Let $q \in \mathbb{N}$ be a prime power and let $0 \neq \Psi = \prod_{k=1}^r \Phi_k \in \mathbb{F}_q[X]$ be monic and squarefree, where $\Phi_1, \ldots, \Phi_r \in \mathbb{F}_q[X]$ are pairwise different irreducible monic polynomials, and thus $\Psi = \prod_{d=1}^n \Psi_d \in \mathbb{F}_q[X]$, where $n := \deg(\Psi)$ and $\Psi_d := \prod_{k \in \mathcal{K}_d} \Phi_k \in \mathbb{F}_q[X]$ and $\mathcal{K}_d := \{l \in \{1, \ldots, r\}; \deg(\Phi_l) = d\}$; note that the Ψ_d are pairwise coprime for $d \in \{1, \ldots, n\}$.

Letting $\widetilde{\Psi}_d := \frac{\Psi}{\prod_{c=1}^{d-1} \Psi_c} = \prod_{c=d}^n \Psi_c \in \mathbb{F}_q[X]$, for all $d \in \mathbb{N}$, we successively compute $\Psi_d = \gcd(X^{q^d} - X, \widetilde{\Psi}_d) \in \mathbb{F}_q[X]$. Moreover, since $\widetilde{\Psi}_d \mid \widetilde{\Psi}_{d-1}$ we have $(X^{q^d} \bmod \widetilde{\Psi}_d) \equiv \Big((X^{q^{d-1}} \bmod \widetilde{\Psi}_{d-1}) \cdot X^q\Big) \bmod \widetilde{\Psi}_d$.

(6.5) Algorithm: Cantor-Zassenhaus (1981). Let $q \in \mathbb{N}$ be a prime power, and let $0 \neq \Psi = \prod_{k=1}^r \Phi_k \in \mathbb{F}_q[X]$ be monic and squarefree, where $n := \deg(\Psi)$ and $\Phi_1, \ldots, \Phi_r \in \mathbb{F}_q[X]$ are pairwise different irreducible monic polynomials such that $\deg(\Phi_i) = d$ for all $i \in \{1, \ldots, r\}$, where d is known. Hence we have $n = dr$; note that we are done if $n = d$, and hence we may assume that $r \geq 2$.

Let $\Theta \in \mathbb{F}_q[X]$. From $X^{q^d} - X = \prod_{\alpha \in \mathbb{F}_{q^d}} (X - \alpha) \in \mathbb{F}_{q^d}[X]$, we get $\Theta^{q^d} - \Theta = \prod_{\alpha \in \mathbb{F}_{q^d}} (\Theta - \alpha) \in \mathbb{F}_{q^d}[X]$, hence any $\alpha \in \mathbb{F}_{q^d}$ is a root of $\Theta^{q^d} - \Theta \in \mathbb{F}_q[X] \subseteq \mathbb{F}_{q^d}[X]$. Thus $X^{q^d} - X \mid \Theta^{q^d} - \Theta \in \mathbb{F}_q[X]$. Since $\deg(\Phi_i) = d$ for all $i \in \{1, \ldots, r\}$ we have $\Phi_i \mid X^{q^d} - X \in \mathbb{F}_q[X]$, and thus $\Psi \mid \Theta^{q^d} - \Theta \in \mathbb{F}_q[X]$.

Since the Φ_i are irreducible such that $\deg(\Phi_i) = d$ and pairwise coprime, the Chinese remainder theorem yields $\mathbb{F}_q[X]/\langle\Psi\rangle \cong \bigoplus_{i=1}^r \mathbb{F}_q[X]/\langle\Phi_i\rangle \cong \bigoplus_{i=1}^r \mathbb{F}_{q^d}$, where for $i \in \{1, \ldots, r\}$ the corresponding projections are derived from the natural maps $\pi_i \colon \mathbb{F}_q[X] \to \mathbb{F}_q[X]/\langle\Phi_i\rangle$.

a) Let q be odd. Since we have $\Theta^{q^d} - \Theta = \Theta \cdot (\Theta^{\frac{q^d-1}{2}} - 1) \cdot (\Theta^{\frac{q^d-1}{2}} + 1) \in \mathbb{F}_q[X]$, where $\gcd(\Theta^{\frac{q^d-1}{2}} - 1, \Theta^{\frac{q^d-1}{2}} + 1) = \gcd(\Theta^{\frac{q^d-1}{2}} - 1, 2) = 1$ and $\gcd(\Theta^{\frac{q^d-1}{2}} \pm 1, \Theta) = \gcd(1, \Theta) = 1$, we have $\Psi = \gcd(\Psi, \Theta) \cdot \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) \cdot \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} + 1) \in \mathbb{F}_q[X]$, where the factors are pairwise coprime. Moreover, we have $\gcd(\Psi, \Theta) = 1$ if and only if $\Theta^{\pi_i} \in (\mathbb{F}_q[X]/\langle \Phi_i \rangle)^* \cong \mathbb{F}_{q^d}^*$ for all $i \in \{1, \ldots, r\}$. Thus if we choose $\Theta \in \mathbb{F}_q[X]_{<n}$ randomly, then the probability to have $\gcd(\Psi, \Theta) \neq 1$, which if $\Theta \neq 0$ yields a non-trivial factor, is given as $1 - (\frac{q^d-1}{q^d})^r \sim \frac{r}{q^d}$, hence is negligible.

Since $\mathbb{F}_{q^d}^*$ is a cyclic group of order $q^d - 1$, we have a group epimorphism $\mathbb{F}_{q^d}^* \to \{\pm 1\} \colon \alpha \mapsto \alpha^{\frac{q^d-1}{2}}$, where $\{\pm 1\} \leq \mathbb{F}_{q^d}^*$ is a cyclic group of order 2. Hence we have a partition $\mathbb{F}_{q^d}^* = \{\alpha \in \mathbb{F}_{q^d}^*; \alpha^{\frac{q^d-1}{2}} = 1\} \,\dot{\cup}\, \{\alpha \in \mathbb{F}_{q^d}^*; \alpha^{\frac{q^d-1}{2}} = -1\}$ into two subsets of cardinality $\frac{q^d-1}{2}$. Hence for $\Theta \in \mathbb{F}_q[X]_{<n}$ such that $\gcd(\Psi, \Theta) = 1$ and fixed $i \in \{1, \ldots, r\}$ we either have $(\Theta^{\frac{q^d-1}{2}})^{\pi_i} = 1 \in \mathbb{F}_q[X]/\langle \Phi_i \rangle$, i. e. $\Phi_i \mid \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) \in \mathbb{F}_q[X]$, or $(\Theta^{\frac{q^d-1}{2}})^{\pi_i} = -1 \in \mathbb{F}_q[X]/\langle \Phi_i \rangle$, i. e. $\Phi_i \mid \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} + 1) \in \mathbb{F}_q[X]$. If we choose $\Theta \in \mathbb{F}_q[X]_{<n}$ such that $\gcd(\Psi, \Theta) = 1$ randomly then either possibility occurs with probability $\frac{1}{2}$. As this happens independently for all $i \in \{1, \ldots, r\}$, both the probability to have $\gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) = 1$ and the probability to have $\gcd(\Psi, \Theta^{\frac{q^d-1}{2}} + 1) = 1$, i. e. $\gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) = \Psi$, are equal to $\frac{1}{2^r}$. Thus for these Θ the probability to have $1 \neq \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) \neq \Psi$ is given by $1 - \frac{1}{2^{r-1}}$.

Note that for constant polynomials $\Theta = \lambda \in \mathbb{F}_q^*$ we have $\lambda^{\frac{q^d-1}{2}} \in \{\pm 1\} \in \mathbb{F}_q^*$, independent of $i \in \{1, \ldots, r\}$, and hence $\gcd(\Psi, \lambda^{\frac{q^d-1}{2}} - 1) = 1$ or $\gcd(\Psi, \lambda^{\frac{q^d-1}{2}} - 1) = \Psi$; thus these need not be tested, which increases the success probability for non-constant test polynomials even further. Anyway, testing all non-constant polynomials $\Theta \in \mathbb{F}_q[X]_{<n}$ finally yields a complete factorization, where we are done as soon as r pairwise coprime factors have been found.

Note that the above probability analysis even shows the following: If we choose $\Theta \in \mathbb{F}_q[X]_{<2d}$ such that $\gcd(\Psi, \Theta) = 1$ randomly then still either of the above possibilities occurs with probability $\frac{1}{2}$ independently for 2 of the factors Φ_i, hence for these Θ the probability to have $1 \neq \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) \neq \Psi$ still is given by $1 - \frac{1}{2^{2-1}} = \frac{1}{2}$. Thus this decreases the success probability, but it still is large enough to allow for the following randomized algorithm: Choose $0 \neq \Theta \in \mathbb{F}_q[X]$ monic such that $1 \leq \deg(\Theta) < 2d$ randomly, and compute $\widetilde{\Psi} := \gcd(\Psi, \Theta^{\frac{q^d-1}{2}} - 1) \in \mathbb{F}_q[X]$; note that $\Theta^{q^d-1} \bmod \Psi$ is found using binary modular exponentiation in $\mathbb{F}_q[X]$. If $1 \neq \widetilde{\Psi} \neq \Psi$, then proceed with $\widetilde{\Psi} \in \mathbb{F}_q[X]$ and $\frac{\Psi}{\widetilde{\Psi}} \in \mathbb{F}_q[X]$ recursively.

b) Let q be even, hence $q = 2^f \in \mathbb{N}$, where $f \in \mathbb{N}$. Letting $T_f := \sum_{i=0}^{f-1} X^{2^i} \in \mathbb{F}_2[X]$ we have $T_f \cdot (1 + T_f) = (\sum_{i=0}^{f-1} X^{2^i}) \cdot (1 + \sum_{i=0}^{f-1} X^{2^i}) = (\sum_{i=0}^{f-1} X^{2^i}) +$

$(\sum_{i=0}^{f-1} X^{2^i})^2 = (\sum_{i=0}^{f-1} X^{2^i}) + (\sum_{i=0}^{f-1} X^{2^{i+1}}) = X + X^{2^f} \in \mathbb{F}_2[X]$.

Thus we also have $(\sum_{i=0}^{fd-1} \Theta^{2^i}) \cdot (1 + \sum_{i=0}^{fd-1} \Theta^{2^i}) = \Theta + \Theta^{2^{fd}} \in \mathbb{F}_q[X]$, where $\gcd(\sum_{i=0}^{fd-1} \Theta^{2^i}, 1 + \sum_{i=0}^{fd-1} \Theta^{2^i}) = \gcd(\sum_{i=0}^{fd-1} \Theta^{2^i}, 1) = 1$, hence we conclude $\Psi = \gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) \cdot \gcd(\Psi, 1 + \sum_{i=0}^{fd-1} \Theta^{2^i}) \in \mathbb{F}_q[X]$, where the factors are pairwise coprime.

Since $T_{fd} \cdot (1 + T_{fd}) = X + X^{2^{fd}} = \prod_{\alpha \in \mathbb{F}_{q^d}} (X - \alpha) \in \mathbb{F}_{q^d}[X]$, as well as $\deg(T_{fd}) = 2^{fd-1}$, we have a partition $\mathbb{F}_{q^d} = \{\alpha \in \mathbb{F}_{q^d}; T_{fd}(\alpha) = 0\} \,\dot{\cup}\, \{\alpha \in \mathbb{F}_{q^d}; T_{fd}(\alpha) = 1\}$ into two sets of cardinality $2^{fd-1} = \frac{q^d}{2}$. Hence for $\Theta \in \mathbb{F}_q[X]_{<n}$ and fixed $i \in \{1, \ldots, r\}$ we either have $\sum_{i=0}^{fd-1} \Theta^{2^i} = 0 \in \mathbb{F}_q[X]/\langle \Phi_i \rangle$, i. e. $\Phi_i \mid \gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) \in \mathbb{F}_q[X]$, or $\sum_{i=0}^{fd-1} \Theta^{2^i} = 1 \in \mathbb{F}_q[X]/\langle \Phi_i \rangle$, i. e. $\Phi_i \mid \gcd(\Psi, 1 + \sum_{i=0}^{fd-1} \Theta^{2^i}) \in \mathbb{F}_q[X]$. Thus, if we choose $\Theta \in \mathbb{F}_q[X]_{<n}$ randomly, then either possibility occurs with probability $\frac{1}{2}$. Hence again both the probability to have $\gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) = 1$ and the probability to have $\gcd(\Psi, 1 + \sum_{i=0}^{fd-1} \Theta^{2^i}) = 1$, i. e. $\gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) = \Psi$, are equal to $\frac{1}{2^r}$, and thus the probability to have $1 \neq \gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) \neq \Psi$ is given by $1 - \frac{1}{2^{r-1}}$.

Note that for constant polynomials $\Theta = \alpha \in \mathbb{F}_q^*$ we have the following: Let $T_{\mathbb{F}_q/\mathbb{F}_2} \colon \mathbb{F}_q \to \mathbb{F}_q \colon \alpha \mapsto \sum_{i=0}^{f-1} \alpha^{\sigma_2^i} = \sum_{i=0}^{f-1} \alpha^{2^i} = T_f(\alpha)$; note that we have $\mathrm{Gal}(\mathbb{F}_q/\mathbb{F}_2) = \langle \sigma_2 \rangle$, which is a cyclic group of order f, hence $T_{\mathbb{F}_q/\mathbb{F}_2}$ actually is the associated Galois trace map. Since $\sigma_2|_{\mathbb{F}_2} = \mathrm{id}_{\mathbb{F}_2}$, we conclude that $T_{\mathbb{F}_q/\mathbb{F}_2}$ is an $\mathbb{F}_2$-linear map, and since we have $\mathbb{F}_q = \{\alpha \in \mathbb{F}_q; T_f(\alpha) = 0\} \,\dot{\cup}\, \{\alpha \in \mathbb{F}_q; T_f(\alpha) = 1\}$, we conclude that actually $T_{\mathbb{F}_q/\mathbb{F}_2} \colon \mathbb{F}_q \to \mathbb{F}_2$ surjective. Anyway, since $\sigma_2^f|_{\mathbb{F}_q} = \mathrm{id}_{\mathbb{F}_q}$, for $\alpha \in \mathbb{F}_q^*$ we have $T_{fd}(\alpha) = \sum_{i=0}^{fd-1} \alpha^{2^i} = d \cdot \sum_{i=0}^{f-1} \alpha^{2^i} = d \cdot T_f(\alpha) \in \mathbb{F}_2 = \{0, 1\}$, independent of $i \in \{1, \ldots, r\}$, hence $\gcd(\Psi, \sum_{i=0}^{fd-1} \alpha^{2^i}) = 1$ or $\gcd(\Psi, \sum_{i=0}^{fd-1} \alpha^{2^i}) = \Psi$; thus constant polynomials need not be tested. Again, testing all non-constant polynomials $\Theta \in \mathbb{F}_q[X]_{<n}$ finally yields a complete factorization.

Note that again we may restrict ourselves to $\Theta \in \mathbb{F}_q[X]_{<2d}$, where still for these Θ the probability to have $1 \neq \gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) \neq \Psi$ is given by $\frac{1}{2}$. Thus we have the following randomized algorithm: Choose $0 \neq \Theta \in \mathbb{F}_q[X]$ monic such that $1 \leq \deg(\Theta) < 2d$ randomly, and compute $\widetilde{\Psi} := \gcd(\Psi, \sum_{i=0}^{fd-1} \Theta^{2^i}) \in \mathbb{F}_q[X]$; note that $\Theta^{2^i} \bmod \Psi$ is found using modular squaring in $\mathbb{F}_q[X]$. If $1 \neq \widetilde{\Psi} \neq \Psi$, then proceed with $\widetilde{\Psi} \in \mathbb{F}_q[X]$ and $\frac{\Psi}{\widetilde{\Psi}} \in \mathbb{F}_q[X]$ recursively.

We present another algorithm, the Berlekamp algorithm, for the final splitting, which is based on linear algebra techniques. Actually it does not require that the prime divisors of the polynomial all have the same degree, and hence distinct degree factorization can be avoided here. For further algorithms for polynomial factorization in $\mathbb{F}_q[X]$, see [3, Ch.14].

(6.6) Algorithm: Berlekamp (1970).
Let $q \in \mathbb{N}$ be a prime power and let $0 \neq \Psi = \prod_{i=1}^{r} \Phi_i \in \mathbb{F}_q[X]$ be monic and squarefree, where $n := \deg(\Psi)$ and $\Phi_1, \ldots, \Phi_r \in \mathbb{F}_q[X]$ are pairwise different irreducible monic polynomials. Note that the number r is not a priorly known; Ψ might even be irreducible, i. e. we might have $r = 1$.

Then for $\Theta \in \mathbb{F}_q[X]_{<n}$ we have $\Psi \mid \Theta^q - \Theta$ if and only if there are $\alpha_1, \ldots, \alpha_r \in \mathbb{F}_q$ such that $\Phi_i \mid \Theta - \alpha_i$ for all $i \in \{1, \ldots, r\}$: Since the Φ_i are irreducible and pairwise coprime, by the Chinese remainder theorem we have $\mathbb{F}_q[X]/\langle\Psi\rangle \cong \bigoplus_{i=1}^{r} \mathbb{F}_q[X]/\langle\Phi_i\rangle \cong \bigoplus_{i=1}^{r} \mathbb{F}_{q^{\deg(\Phi_i)}}$, where for $i \in \{1, \ldots, r\}$ the corresponding projections are derived from the natural maps $\pi_i \colon \mathbb{F}_q[X] \to \mathbb{F}_q[X]/\langle\Phi_i\rangle$. Hence, given $\alpha_1, \ldots, \alpha_r \in \mathbb{F}_q$ the polynomial $\Theta \in \mathbb{F}_q[X]_{<n}$ is uniquely determined by $\Phi_i \mid \Theta - \alpha_i$ for all $i \in \{1, \ldots, r\}$. Hence we have $(\Theta^q)^{\pi_i} = (\alpha_i^q)^{\pi_i} = \alpha_i^{\pi_i} = \Theta^{\pi_i}$, thus $\Phi_i \mid \Theta^q - \Theta$ for all $i \in \{1, \ldots, r\}$. Conversely, if $\Psi \mid \Theta^q - \Theta$, then from $X^q - X = \prod_{\alpha \in \mathbb{F}_q}(X - \alpha) \in \mathbb{F}_q[X]$ we get $\Phi_i \mid \Psi \mid \Theta^q - \Theta = \prod_{\alpha \in \mathbb{F}_q}(\Theta - \alpha) \in \mathbb{F}_q[X]$, for all $i \in \{1, \ldots, r\}$. Since Φ_i is irreducible, we conclude that there is $\alpha_i \in \mathbb{F}_q$ such that $\Phi_i \mid \Theta - \alpha_i$.

Hence, if $\Theta \in \mathbb{F}_q[X]_{<n}$ such that $\Psi \mid \Theta^q - \Theta$, then $\Psi = \prod_{\alpha \in \mathbb{F}_q} \gcd(\Psi, \Theta - \alpha) \in \mathbb{F}_q[X]$, where since $\gcd(\Theta - \alpha, \Theta - \alpha') = \gcd(\Theta - \alpha, \alpha - \alpha') = 1$ for $\alpha \neq \alpha' \in \mathbb{F}_q$, the factors are pairwise coprime. Thus the aim is to find suitable polynomials Θ, yielding non-trivial factors; we proceed as follows: Let $\mathcal{U}_\Psi := \{\Theta \in \mathbb{F}_q[X]_{<n}; \Psi \mid \Theta^q - \Theta\}$. Since for $\Theta \in \mathbb{F}_q[X]_{<n}$ we have $\Theta \in \mathcal{U}_\Psi$ if and only if $\Theta^{\pi_i} \in \mathbb{F}_q \subseteq \mathbb{F}_q[X]/\langle\Phi_i\rangle$, we conclude that $\mathcal{U}_\Psi \leq \mathbb{F}_q[X]_{<n}$, and that $\mathcal{U}_\Psi + \langle\Psi\rangle \subseteq \mathbb{F}_q[X]/\langle\Psi\rangle$ is an $\mathbb{F}_q$-subalgebra, called the **Berlekamp algebra**. In particular, $\mathcal{U}_\Psi$ encompasses the constant polynomials $\lambda \in \mathbb{F}_q[X]$, but since we $\gcd(\Psi, \lambda - \alpha) = 1$ for $\lambda \neq \alpha$, and $\gcd(\Psi, \lambda - \alpha) = \Psi$ for $\lambda = \alpha$, these need not be tested. Letting $\alpha_1, \ldots, \alpha_r \in \mathbb{F}_q$ vary, by the uniqueness statement in the Chinese remainder theorem we have $|\mathcal{U}_\Psi| = q^r$, hence we have $\dim_{\mathbb{F}_q}(\mathcal{U}_\Psi) = r$. To find $\mathcal{U}_\Psi$ first we proceed as follows:

Let $\Theta = \sum_{j=0}^{n-1} \vartheta_j X^j \in \mathbb{F}_q[X]_{<n}$, then $\Theta^q = \sum_{i=0}^{n-1} \vartheta_i^q X^{qi} = \sum_{i=0}^{n-1} \vartheta_i X^{qi} \in \mathbb{F}_q[X]$, and letting $X^{qi} \equiv \sum_{j=0}^{n-1} \xi_{ij} X^j \bmod \Psi$, for $\xi_{ij} \in \mathbb{F}_q$ and $i \in \{0, \ldots, n-1\}$, we get $\Theta^q \equiv \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \vartheta_i \xi_{ij} X^j \bmod \Psi$. Thus $\Theta^q \equiv \Theta \bmod \Psi$ is equivalent to $\sum_{j=0}^{n-1} \vartheta_j X^j \equiv \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \vartheta_i \xi_{ij} X^j \bmod \Psi$, which holds if and only if $\vartheta_j = \sum_{i=0}^{n-1} \vartheta_i \xi_{ij} \in \mathbb{F}_q$, for all $j \in \{0, \ldots, n-1\}$. Thus with respect to the $\mathbb{F}_q$-basis $\{1, X, \ldots, X^{n-1}\} \subseteq \mathbb{F}_q[X]_{<n}$, the polynomial $\Theta \in \mathbb{F}_q[X]_{<n}$ is described by $\vartheta := [\vartheta_0, \ldots, \vartheta_{n-1}] \in \mathbb{F}_q^n$, and the $\mathbb{F}_q$-linear map $\mathbb{F}_q[X]/\langle\Psi\rangle \to \mathbb{F}_q[X]/\langle\Psi\rangle$ induced by the Frobenius map $\mathbb{F}_q[X] \to \mathbb{F}_q[X] \colon \Theta \mapsto \Theta^q$ is described by the **Petr-Berlekamp matrix** $Q_\Psi := [\xi_{ij}] \in \mathbb{F}_q^{n \times n}$. Hence the $\Theta \in \mathcal{U}_\Psi$ searched for are precisely given as the solutions in $\mathbb{F}_q^n$ of the system of $\mathbb{F}_q$-linear equations $\vartheta \cdot Q_\Psi = \vartheta$, i. e. as the solutions of the eigenvalue problem $\vartheta \cdot (Q_\Psi - E_n) = 0 \in \mathbb{F}_q^n$.

Hence we have to compute the row kernel $\ker(Q_\Psi - E_n) \in \mathbb{F}_q^n$; note that an $\mathbb{F}_q$-basis can be computed using the Gauß algorithm over the finite field $\mathbb{F}_q$. Note that by the above we have $\dim_{\mathbb{F}_q}(\ker(Q_\Psi - E_n)) = r$, hence this yields the so far unknown number r of prime divisors of Ψ, in particular Ψ is irreducible if and only if $\mathrm{rk}_{\mathbb{F}_q}(Q_\Psi - E_n) = n-1$.

We finally have to pick $\Theta \in \mathcal{U}_\Psi$ yielding non-trivial factors: For $i \neq j \in \{1,\ldots,r\}$ there is $\Theta \in \mathcal{U}_\Psi$ such that $\alpha_i := \Theta^{\pi_i} \neq \Theta^{\pi_j} =: \alpha_j \in \mathbb{F}_q$, i. e. we have $\Phi_i \mid \gcd(\Psi, \Theta - \alpha_i)$ and $\Phi_j \not| \gcd(\Psi, \Theta - \alpha_i)$, as well as $\Phi_j \mid \gcd(\Psi, \Theta - \alpha_j)$ and $\Phi_i \not| \gcd(\Psi, \Theta - \alpha_j)$. Hence given an $\mathbb{F}_q$-basis $\{\Theta_0, \ldots, \Theta_{r-1}\} \subseteq \mathcal{U}_\Psi$, where we may assume $\Theta_0 = 1$, by $\mathbb{F}_q$-linearity there is an element Θ_k, for some $k \in \{1,\ldots,r-1\}$, having the same distinguishing property. Thus we have the following deterministic algorithm:

We successively for $k \in \{1,\ldots,r-1\}$ and $\alpha \in \mathbb{F}_q$ compute $\gcd(\widetilde{\Psi}, \Theta_k - \alpha) \in \mathbb{F}_q[X]$, where $\widetilde{\Psi} \in \mathbb{F}_q[X]$ runs through all the factors of Ψ found so far; here we initially have $\widetilde{\Psi} = \Psi$, and whenever $1 \neq \gcd(\widetilde{\Psi}, \Theta_k) \neq \widetilde{\Psi}$ we replace $\widetilde{\Psi}$ by the non-trivial factors found; we terminate as soon as a total of r factors has been found. Note that computing the Petr-Berlekamp matrix $Q_\Psi \in \mathbb{F}_q^{n\times n}$ and the kernel $\ker(Q_\Psi - E_n) \in \mathbb{F}_q^n$ are costly; hence it is better to proceed in the way described above, rather than to use the algorithm recursively whenever a non-trivial factor has been found. Moreover, note that the number of polynomials $\Theta - \alpha$ to be tested grows linearly with the field size q; thus the Cantor-Zassenhaus algorithm is superior for large q, whenever $q \geq 100$, say.

As an alternative for large q, we have the following randomized algorithm for $q \geq 3$ odd: We choose $\lambda_0, \ldots, \lambda_{r-1} \in \mathbb{F}_q$ randomly, let $\Theta := \sum_{k=0}^{r-1} \Theta_k \lambda_k \in \mathbb{F}_q[X]_{<n}$, and compute $\gcd(\widetilde{\Psi}, \Theta^{\frac{q-1}{2}} - 1) \in \mathbb{F}_q[X]$, where as above $\widetilde{\Psi} \in \mathbb{F}_q[X]$ runs through all the factors of Ψ found so far, and if due is replaced by newly found factors. The success probability is given as follows: For all $i \in \{1,\ldots,r\}$ we have $(\Theta^{\frac{q-1}{2}})^{\pi_i} \in \{0, \pm 1\}$, as in (6.5). The first case occurs with probability $\frac{1}{q}$, which is negligible for large q. The latter two cases occur with probability $\frac{q-1}{2q} \sim \frac{1}{2}$ each. Hence if $\widetilde{\Psi}$ is reducible, a non-trivial factor is found from $\gcd(\widetilde{\Psi}, \Theta^{\frac{q-1}{2}} - 1)$ with probability at least $2 \cdot \frac{q-1}{2} \cdot \frac{q+1}{2} \cdot \frac{1}{q^2} = \frac{1}{2} - \frac{1}{2q^2} \geq \frac{4}{9}$.

(6.7) Remark. We briefly comment on running times: The input length of $0 \neq \Psi \in \mathbb{F}_q[X]$ of degree $\deg(\Psi) = n$ is asymptotically $\sim n\ln(q)$. Squarefree factorization uses ring operations, quotient and remainder operations, and gcd computations in $\mathbb{F}_q[X]$, applied to polynomials whose degree is bounded by n, thus this needs a number of field operations in $\mathbb{F}_q$ which is a polynomial in n. Distinct degree factorization additionally uses binary modular exponentiation, where the exponents are in $O(q^n)$, which hence needs $O(\ln(q^n)) = O(n\ln(q))$ ring operations as well as quotient and remainder operations in $\mathbb{F}_q[X]$, thus this needs a number of field operations in $\mathbb{F}_q$ which is a polynomial in $n\ln(q)$. Note that squarefree factorization and distinct degree factorization are deterministic.

The randomized Cantor-Zassenhaus algorithm, where the success probability has been determined above, needs a number of field operations in $\mathbb{F}_q$ which is a polynomial in $n\ln(q)$. The deterministic version of the Cantor-Zassenhaus algorithm has to test $O(q^n) = O(e^{n\ln(q)})$ polynomials, hence runs in exponential time. Since linear algebra algorithms over $\mathbb{F}_q$ need a number of field operations in $\mathbb{F}_q$ which is a polynomial in n, the randomized Berlekamp algorithm, where the success probability has been determined above, needs a number of field operations in $\mathbb{F}_q$ which is a polynomial in $n\ln(q)$. The deterministic version of the Berlekamp algorithm has to test $O(nq) = O(ne^{\ln(q)})$ polynomials, hence runs in exponential time. Actually, it is an open problem whether polynomial factorization in $\mathbb{F}_q[X]$ can be performed in deterministic polynomial time.

7 Polynomial factorization over the integers

(7.1) Remark. Let $0 \neq \Psi \in \mathbb{Z}[X]$ be primitive; note that otherwise we have to deal with integer factorization as well. Again the aim is to find the factorization $\Psi = \prod_{k=1}^{r} \Phi_k^{e_k}$, where $\Phi_1, \ldots, \Phi_r \in \mathbb{Z}[X]$ are pairwise non-associate irreducible polynomials and $e_k = e_{\Phi_k}(\Psi) \in \mathbb{N}$; note that the $\Phi_k \in \mathbb{Z}[X]$ again are primitive. By Gauß's Theorem the $\Phi_k \in \mathbb{Q}[X]$ are irreducible as well; in this sense factorization in $\mathbb{Z}[X]$ and in $\mathbb{Q}[X]$ are equivalent.

By Gauß's Theorem again $\Psi \in \mathbb{Z}[X]$ is squarefree if and only if $\Psi \in \mathbb{Q}[X]$ is squarefree. Since $\mathbb{Q}$ is a perfect field, we by (4.7) conclude that the latter holds if and only if $\operatorname{disc}(\Psi) \neq 0 \in \mathbb{Z}$, which by (4.4) holds if and only if $\gcd(\Psi, \Psi') \in \mathbb{Z}[X]$ is constant, where $\Psi' = \frac{\partial \Psi}{\partial X} \in \mathbb{Z}[X]$ denotes the formal derivative. Moreover, similar to (6.1), we let $\Psi = \prod_{e=1}^{n} \Psi_e^e$, where $\Psi_e := \prod_{k \in \mathcal{K}_e} \Phi_k \in \mathbb{Z}[X]$ and $\mathcal{K}_e := \{l \in \{1, \ldots, r\}; e_l = e\}$, and now get $\gcd(\Psi, \Psi') = \prod_{e=1}^{n} \Psi_e^{e-1} \in \mathbb{Z}[X]$, thus $\frac{\Psi}{\gcd(\Psi,\Psi')} = \prod_{e=1}^{n} \Psi_e = \prod_{k=1}^{r} \Phi_k \in \mathbb{Z}[X]$. Note that an algorithm similar to the one in (6.1) actually yields the $\Psi_e \in \mathbb{Z}[X]$, see Exercise (8.38). Thus we may assume that $0 \neq \Psi \in \mathbb{Z}[X]$ is squarefree, hence $\operatorname{disc}(\Psi) \neq 0 \in \mathbb{Z}$.

We apply a modular technique: Let $p \in \mathbb{N}$ be a prime such that $p \not| \operatorname{lc}(\Psi)$, and let $\bar{\ }\colon \mathbb{Z} \rightarrow \mathbb{Z}/\langle p \rangle \cong \mathbb{F}_p$ denote the natural map; hence we have $\deg(\Psi) = \deg(\overline{\Psi})$ and $\overline{\Psi} = \prod_{k=1}^{r} \overline{\Phi_k} \in \mathbb{F}_p[X]$, where $p \not| \operatorname{lc}(\Phi_k)$. Hence in particular if $\overline{\Psi} \in \mathbb{F}_p[X]$ is irreducible, then $\Psi \in \mathbb{Z}[X]$ also is. The $\overline{\Phi_k} \in \mathbb{F}_p[X]$ in general are reducible: Actually there are irreducible monic $\Phi \in \mathbb{Z}[X]$, such that $\overline{\Phi} \in \mathbb{F}_p[X]$ is reducible for all primes p. Moreover, by the Chebotarev density theorem, see [3, Ch.15.3], the modular factorization pattern varies with the prime chosen, which shows that we cannot hope to find suitable primes with sufficiently high probability.

Hence after factorizing $\overline{\Psi} \in \mathbb{F}_p[X]$ for a fixed big prime p, we try to deduce the factorization of $\Psi \in \mathbb{Z}[X]$ by trial and error factor combination from the modular factorization pattern found. Note that by the variability in the modular factorization patterns, using several small primes and the Chinese remainder theorem would require to fit the various factorizations together, which is even harder than trying the possible factor combinations coming from a single big prime.

Subsequently, we present another modular technique, where instead of using a big prime, we use a small prime p, and successively improve the factorization mod p to factorizations modulo sufficiently high powers of p. Still, this does not save us from finally having to try all possible factor combinations to find the genuine divisors. Finally, although this does not yield a practical algorithm, we show how lattice base reduction can be used to solve the factorization problem in polynomial time; actually this was the original reason for the invention of LLL reduction.

(7.2) Example: A Swinnerton-Dyer polynomial.
Let $\Phi := 1 - 10X^2 + X^4 \in \mathbb{Z}[X]$. We have $\mathrm{disc}(\Phi) = 147\,456 = 2^{14} \cdot 3^2 \in \mathbb{Z}$, hence Φ is squarefree, and as we show below $\overline{\Phi}$ for $p \neq 2, 3$ also is. Indeed, for $p = 2$ we have $\overline{\Phi} = 1 + X^4 = (1 + X)^4 \in \mathbb{F}_2[X]$, and for $p = 3$ we have $\overline{\Phi} = 1 + 2X^2 + X^4 = (1 + X^2)^2 \in \mathbb{F}_3[X]$. Moreover, we have $\Phi = (X - \sqrt{2} - \sqrt{3})(X - \sqrt{2} + \sqrt{3})(X + \sqrt{2} - \sqrt{3})(X + \sqrt{2} + \sqrt{3}) = (-1 - 2\sqrt{2} + X^2)(-1 + 2\sqrt{2} + X^2) = (1 - 2\sqrt{3} + X^2)(1 + 2\sqrt{3} + X^2) = (-5 - 2\sqrt{6} + X^2)(-5 + 2\sqrt{6} + X^2) \in \mathbb{C}[X]$.

Thus for $p \geq 5$ we conclude as follows: Let $\mathbb{F}_p^{*2} := \{\alpha^2; \alpha \in \mathbb{F}_p^*\} \leq \mathbb{F}_p^*$. Hence if both $\overline{2}, \overline{3} \in \mathbb{F}_p^{*2}$, then $\overline{\Phi} \in \mathbb{F}_p[X]$ factors into 4 linear factors; if $\overline{2} \in \mathbb{F}_p^{*2}$ and $\overline{3} \in \mathbb{F}_p^* \setminus \mathbb{F}_p^{*2}$, then $\overline{\Phi} \in \mathbb{F}_p[X]$ factors into 2 irreducible factors of degree 2; and if $\overline{3} \in \mathbb{F}_p^{*2}$ and $\overline{2} \in \mathbb{F}_p^* \setminus \mathbb{F}_p^{*2}$, then $\overline{\Phi} \in \mathbb{F}_p[X]$ also factors into 2 irreducible factors of degree 2. Finally, if both $\overline{2}, \overline{3} \in \mathbb{F}_p^* \setminus \mathbb{F}_p^{*2}$, then since p is odd and hence $[\mathbb{F}_p^* \colon \mathbb{F}_p^{*2}] = 2$, we have $\overline{6} = \overline{2} \cdot \overline{3} \in \mathbb{F}_p^{*2}$, and again $\overline{\Phi} \in \mathbb{F}_p[X]$ also factors into 2 irreducible factors of degree 2.

Thus for all primes $p \in \mathbb{N}$ these modular factorization patterns are compatible either with $\Phi \in \mathbb{Z}[X]$ being irreducible or $\Phi \in \mathbb{Z}[X]$ having a factorization into 2 factors of degree 2. But letting $K_2 := \mathbb{Q}(\sqrt{2}) \subseteq \mathbb{C}$ and $K_3 := \mathbb{Q}(\sqrt{3}) \subseteq \mathbb{C}$, the splitting field of $\Phi \in \mathbb{Q}[X]$ is given as $K := K_2K_3 \subseteq \mathbb{C}$; note that $K_2 \cap K_3 = \mathbb{Q}$. Hence we have $[K \colon \mathbb{Q}] = 4$ and $\mathrm{Gal}(K/Q) = \langle \alpha_2, \alpha_3 \rangle \cong V_4$, the Klein 4-group, where $\alpha_2 \colon \sqrt{2} \mapsto -\sqrt{2}, \sqrt{3} \mapsto \sqrt{3}$ and $\alpha_3 \colon \sqrt{2} \mapsto \sqrt{2}, \sqrt{3} \mapsto -\sqrt{3}$. Hence $\mathrm{Gal}(K/Q)$ acts transitively on the roots of $\Phi \in K[X]$, and thus $\Phi \in \mathbb{Q}[X]$ is irreducible.

(7.3) Algorithm: Lifting factorizations, big prime version.
Let $0 \neq \Psi \in \mathbb{Z}[X]$ be primitive and squarefree, where $n := \deg(\Psi)$, and let $p \in \mathbb{N}$ be a prime such that $p \not| \, \mathrm{lc}(\Psi)$. The polynomial $\overline{\Psi} \in \mathbb{F}_p[X]$ is squarefree if and only if $\mathrm{disc}(\overline{\Psi}) \neq 0 \in \mathbb{F}_p$. Since by (4.16) we have $\mathrm{disc}(\overline{\Psi}) = \overline{\mathrm{disc}(\Psi)} \in \mathbb{F}_p$, the latter holds if and only if $p \not| \, \mathrm{disc}(\Psi) \in \mathbb{Z}$. Hence there are only finitely many primes $p \in \mathbb{N}$ such that $\overline{\Psi} \in \mathbb{F}_p[X]$ is not squarefree, and we may choose p suitably such that $\overline{\Psi}$ actually is squarefree.

If $\Theta \mid \Psi \in \mathbb{Z}[X]$, then Mignotte's inequality yields $\|\Theta\|_1 \cdot \|\frac{\Psi}{\Theta}\|_1 \leq 2^n \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty$. Hence letting $B_\Psi := |\mathrm{lc}(\Psi)| \cdot 2^n \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty \in \mathbb{N}$, where the reason for the additional factor $|\mathrm{lc}(\Psi)|$ becomes clear below, we additionally choose $p \in \mathbb{N}$ such that $p \geq 2B_\Psi$. Lifting the factors of $\overline{\Psi} \in \mathbb{F}_p[X]$ to $\mathbb{Z}[X]$, let $\Theta_1, \ldots, \Theta_s \in \mathbb{Z}[X]$ be monic such that $\|\Theta_l\|_\infty \leq \frac{p}{2}$, for all $l \in \{1, \ldots, s\}$, and $\overline{\Psi} = \overline{\mathrm{lc}(\Psi)} \cdot \prod_{l=1}^s \overline{\Theta_l} \in \mathbb{F}_p[X]$; note that hence the $\Theta_l \in \mathbb{Z}[X]$ are primitive, irreducible and pairwise coprime.

For a subset $\mathcal{I} \subseteq \{1, \ldots, s\}$, lifting the corresponding subproduct, let $\Theta, \widetilde{\Theta} \in \mathbb{Z}[X]$ such that $p \not| \operatorname{lc}(\Theta), \operatorname{lc}(\widetilde{\Theta})$ and $\|\Theta\|_\infty, \|\widetilde{\Theta}\|_\infty \leq \frac{p}{2}$ as well as $\overline{\Theta} = \overline{\operatorname{lc}(\Psi)} \cdot \prod_{l \in \mathcal{I}} \overline{\Theta_l} \in \mathbb{F}_p[X]$ and $\overline{\widetilde{\Theta}} = \overline{\operatorname{lc}(\Psi)} \cdot \prod_{l \in \{1,\ldots,s\} \setminus \mathcal{I}} \overline{\Theta_l} \in \mathbb{F}_p[X]$. We show that $\Theta \cdot \widetilde{\Theta} = \operatorname{lc}(\Psi) \cdot \Psi \in \mathbb{Z}[X]$ if and only if $\|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 \leq B_\Psi$:

If $\Theta \cdot \widetilde{\Theta} = \operatorname{lc}(\Psi) \cdot \Psi$, then Mignotte's inequality applied to $\operatorname{lc}(\Psi) \cdot \Psi$ yields $\|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 \leq B_\Psi$. Conversely, we have $\overline{\Theta} \cdot \overline{\widetilde{\Theta}} = \overline{\operatorname{lc}(\Psi)} \cdot \overline{\Psi} \in \mathbb{F}_p[X]$ anyway. Let $\Theta = \sum_{i=0}^n \vartheta_i X^i \in \mathbb{Z}[X]$ and $\widetilde{\Theta} = \sum_{j=0}^n \widetilde{\vartheta}_j X^j \in \mathbb{Z}[X]$, then $\|\Theta \cdot \widetilde{\Theta}\|_1 = \|\sum_{k=0}^n \sum_{l=0}^k (\vartheta_{k-l}\widetilde{\vartheta}_l) \cdot X^k\|_1 = \sum_{k=0}^n |\sum_{l=0}^k (\vartheta_{k-l}\widetilde{\vartheta}_l)| \leq \sum_{k=0}^n \sum_{l=0}^k |\vartheta_{k-l}| \cdot |\widetilde{\vartheta}_l| = (\sum_{k=0}^n |\vartheta_k|) \cdot (\sum_{l=0}^n |\widetilde{\vartheta}_l|) = \|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1$. Hence we have $\|\Theta \cdot \widetilde{\Theta}\|_\infty \leq \|\Theta \cdot \widetilde{\Theta}\|_1 \leq \|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 \leq B_\Psi$, and since $\|\operatorname{lc}(\Psi) \cdot \Psi\|_\infty = |\operatorname{lc}(\Psi)| \cdot \|\Psi\|_\infty \leq B_\Psi$ as well we conclude $\Theta \cdot \widetilde{\Theta} = \operatorname{lc}(\Psi) \cdot \Psi$. ♯

Note that by computing the constant coefficient $\vartheta_0 = \Theta(0) \in \mathbb{Z}$ first, we may exclude cases such that $\Theta(0) \not| \operatorname{lc}(\Psi) \cdot \Psi(0) \in \mathbb{Z}$ from consideration in advance. Then, the condition $\|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 \leq B_\Psi$ is easily checked, replacing trial division. If this condition holds then we have $\Psi = \frac{\Theta}{\gamma(\Theta)} \cdot \frac{\widetilde{\Theta}}{\gamma(\widetilde{\Theta})} \in \mathbb{Z}[X]$, where $\gamma(\cdot) \in \mathbb{Z}$ denoting the content, the factors are primitive; note that $\gamma(\operatorname{lc}(\Psi) \cdot \Psi) = \operatorname{lc}(\Psi)$. Thus we successively run through the subsets $\mathcal{I} \subseteq \{1, \ldots, s\}$ such that $1 \leq t := |\mathcal{I}| \leq \frac{s}{2}$, with t increasing, until we find $\Theta \mid \Psi \in \mathbb{Z}[X]$; hence $\Theta \in \mathbb{Z}[X]$ is irreducible; and we proceed with $\widetilde{\Theta} := \frac{\Theta}{\Phi} \in \mathbb{Z}[X]$ and index sets $\widetilde{\mathcal{I}} \subseteq \{1, \ldots, s\} \setminus \mathcal{I}$, where $t \leq \widetilde{\mathcal{I}} \leq \frac{s-t}{2}$.

E. g. let $\Psi := 1 + 6X - 7X^2 - 2X^3 - 6X^4 + X^6 \in \mathbb{Z}[X]$. Hence we have $n = \deg(\Psi) = 6$ and $\operatorname{lc}(\Psi) = 1$ as well as $\operatorname{disc}(\Psi) = -10\,930\,094\,080 = -2^{18} \cdot 5 \cdot 31 \cdot 269$, thus we choose $p \notin \{2, 5, 31, 269\}$. Moreover, we have $B_\Psi = 2^6 \cdot 7 \cdot \sqrt{7} \sim 1\,185,3$. Note that varying p we find several modular factorization patterns, with varying number $s \geq 2$ of modular prime divisors. We make the minimal possible choice $p := 2\,371 \geq 2\,370 = 2B_\Psi$, where s is not minimal possible, thus leaves something to do: Factorization of $\overline{\Psi} \in \mathbb{F}_p[X]$, and lifting the factors to $\mathbb{Z}[X]$, yields $s = 4$ and $\Theta_1 = 1\,130 + X \in \mathbb{Z}[X]$ and $\Theta_2 = 1\,133 + X \in \mathbb{Z}[X]$, as well as $\Theta_3 = -1\,068 - 1\,130X + X^2 \in \mathbb{Z}[X]$ and $\Theta_4 = 971 - 1\,133X + X^2 \in \mathbb{Z}[X]$.

For linear, quadratic or cubic factors $\Theta \mid \Psi$ Mignotte's inequality shows that we have $\|\Theta\|_\infty \leq 2 \cdot 7 \cdot \sqrt{7} \sim 37,0$ and $\|\Theta\|_\infty \leq 2^2 \cdot 7 \cdot \sqrt{7} \sim 74,1$ and $\|\Theta\|_\infty \leq 2^3 \cdot 7 \cdot \sqrt{7} \sim 148,2$, respectively. This or a consideration of constant coefficients excludes the case $t = 1$, i. e. the singleton subsets $\mathcal{I} \subseteq \{1, \ldots, 4\}$. For $t = 2$ and $\mathcal{I} = \{1, 2\}$ we get $\Theta = -50 - 108X + X^2 \in \mathbb{Z}[X]$ and $\widetilde{\Theta} = -901 - 994X - 147X^2 + 2\,224X^3 + X^4 \in \mathbb{Z}[X]$, thus $\|\Theta\|_\infty \cdot \|\widetilde{\Theta}\|_\infty = 108 \cdot 2\,224 > B_\Psi$ excludes this case. For $\mathcal{I} = \{1, 4\}$ we get $\Theta = -543 + 1\,021X - 3X^2 + X^3 \in \mathbb{Z}[X]$ and $\widetilde{\Theta} = -834 - 1\,018X + 3X^2 + X^3 \in \mathbb{Z}[X]$, thus $\|\Theta\|_\infty \cdot \|\widetilde{\Theta}\|_\infty = 1\,012 \cdot 1\,018 > B_\Psi$ excludes this case. Note that the latter cases can also be excluded directly using Mignotte's inequality or considering constant coefficients. Finally, for $\mathcal{I} = \{1, 3\}$ we get $\Theta = -1 + X + X^3 \in \mathbb{Z}[X]$ and $\widetilde{\Theta} = -1 - 7X + X^3 \in \mathbb{Z}[X]$, thus $\|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 = 3 \cdot 9 \leq B_\Psi$ implies that $\Theta \cdot \widetilde{\Theta} = \Psi \in \mathbb{Z}[X]$; note that the above analysis also shows that $\Theta, \widetilde{\Theta} \in \mathbb{Z}[X]$ are irreducible.

(7.4) Proposition: Hensel lifting.
Let R be an integral domain, let $\pi \in R$, and let $0 \neq f \in R[X]$. Moreover, let $0 \neq g, h \in R[X]$ such that $\deg(g) > 0$, as well as $\mathrm{lc}(g) \in (R/\langle \pi \rangle)^*$ and $\pi \nmid \mathrm{lc}(h)$, as well as $f \equiv gh \bmod \pi$; note that this implies $\pi \nmid \mathrm{lc}(f)$ and $\deg(g) + \deg(h) = \deg(f)$. Finally, let $s, t \in R[X]$ such that $\deg(s) < \deg(h)$ and $\pi \nmid \mathrm{lc}(s)$, as well as $\deg(t) < \deg(g)$ and $\pi \nmid \mathrm{lc}(t)$, and $sg + th \equiv 1 \bmod \pi$, i. e. g and h are **Bezout coprime** $\bmod\, \pi$.

Then there are $\widehat{g}, \widehat{h} \in R[X]$ such that $\deg(\widehat{g}) = \deg(g)$ and $\deg(\widehat{h}) = \deg(h)$ as well as $\widehat{g} \equiv g \bmod \pi$ and $\widehat{h} \equiv h \bmod \pi$, as well as $\mathrm{lc}(\widehat{g}) = \mathrm{lc}(g)$ and $f \equiv \widehat{g}\widehat{h} \bmod \pi^2$. Moreover, there are $\widehat{s}, \widehat{t} \in R[X]$ such that $\deg(\widehat{s}) < \deg(\widehat{h})$ and $\deg(\widehat{t}) < \deg(\widehat{g})$, as well as $\widehat{s} \equiv s \bmod \pi$ and $\widehat{t} \equiv t \bmod \pi$, as well as $\widehat{s}\widehat{g} + \widehat{t}\widehat{h} \equiv 1 \bmod \pi^2$; note that here we let $\deg(0) < 0$.

Hence $\widehat{g}, \widehat{h}, \widehat{s}, \widehat{t} \in R[X]$ fulfill the assumptions made for $g, h, s, t \in R[X]$, with $\pi \in R$ replaced by $\pi^2 \in R$; note that we have $\pi^2 \nmid \mathrm{lc}(h)$ anyway, and that from $\mathrm{lc}(g) \cdot \alpha - 1 = \beta\pi$, for some $\alpha, \beta \in R$, we get $\mathrm{lc}(g) \cdot \alpha(1 - \beta\pi) = (1 + \beta\pi)(1 - \beta\pi) = 1 - \beta^2\pi^2$, and thus $\mathrm{lc}(g) \in (R/\langle \pi^2 \rangle)^*$.

Proof. Let $\delta := \frac{1}{\pi} \cdot (f - gh) \in R[X]$. Since $\mathrm{lc}(g) \in (R/\langle \pi \rangle)^*$, by quotient and remainder in $R/\langle \pi \rangle[X]$ let $q, r \in R[X]$ such that $\deg(r) < \deg(g)$ and $\pi \nmid \mathrm{lc}(r)$, as well as $r \equiv t\delta - qg \bmod \pi$, and let $u \in R[X]$ such that $\pi \nmid \mathrm{lc}(u)$, and $u \equiv s\delta + qh \bmod \pi$. Let $\widehat{g} := g + r\pi \in R[X]$ and $\widehat{h} := h + u\pi \in R[X]$.

Then since $sg + th \equiv 1 \bmod \pi$ we have $f - \widehat{g}\widehat{h} \equiv f - (g + r\pi)(h + u\pi) \equiv \delta\pi - (ug + rh)\pi - ru\pi^2 \equiv \delta\pi - (s\delta + qh)g\pi - (t\delta - qg)h\pi \equiv \delta\pi - (sg + th)\delta\pi \equiv (1 - sg - th)\delta\pi \equiv 0 \bmod \pi^2$. We may assume $u, r \neq 0$. From $\deg(r) < \deg(g)$ we get $\deg(\widehat{g}) = \deg(g)$ and $\mathrm{lc}(\widehat{g}) = \mathrm{lc}(g)$ as well as $\widehat{g} \equiv g \bmod \pi$. Moreover, we have $rh + ug \equiv (t\delta - qg)h + (s\delta + qh)g \equiv (th + sg)\delta \equiv \delta \bmod \pi$. Since $\deg(\delta) \leq \deg(f)$, and $\deg(rh) = \deg(r) + \deg(h) < \deg(g) + \deg(h) = \deg(f)$ as well as $\pi \nmid \mathrm{lc}(u)$ and $\mathrm{lc}(g) \in (R/\langle \pi \rangle)^*$, we conclude that $\deg(ug) = \deg(u) + \deg(g) \leq \deg(f)$ and hence $\deg(u) \leq \deg(h)$. Thus $\deg(\widehat{h}) \leq \deg(h)$, and since $\widehat{h} \equiv h \bmod \pi$ and $\pi \nmid \mathrm{lc}(h)$ we conclude $\deg(\widehat{h}) = \deg(h)$.

Let $\epsilon := \frac{1}{\pi} \cdot (1 - s\widehat{g} - t\widehat{h}) \in R[X]$. Again by quotient and remainder in $R/\langle \pi \rangle[X]$ let $q, r \in R[X]$ such that $\deg(r) < \deg(\widehat{g})$ and $\pi \nmid \mathrm{lc}(r)$, as well as $r \equiv t\epsilon - q\widehat{g} \bmod \pi$, and let $u \in R[X]$ such that $\pi \nmid \mathrm{lc}(u)$, and $u \equiv s\epsilon + q\widehat{h} \bmod \pi$. Let $\widehat{s} := s + u\pi \in R[X]$ and $\widehat{t} := t + r\pi \in R[X]$.

Then we have $\widehat{s}\widehat{g} + \widehat{t}\widehat{h} \equiv (s + u\pi)\widehat{g} + (t + r\pi)\widehat{h} \equiv (1 - \epsilon\pi) + (u\widehat{g} + r\widehat{h})\pi + ru\widehat{g}\widehat{h}\pi^2 \equiv (1 - \epsilon\pi) + (s\epsilon + q\widehat{h})\widehat{g}\pi + (t\epsilon - q\widehat{g})\widehat{h}\pi \equiv (1 - \epsilon\pi) + (s\widehat{g} + t\widehat{h})\epsilon\pi \equiv (1 - \epsilon\pi) + (1 - \epsilon\pi)\epsilon\pi \equiv 1 - \epsilon^2\pi^2 \equiv 1 \bmod \pi^2$. Again we may assume $u, r \neq 0$. Since $\deg(r) < \deg(\widehat{g})$ and $\deg(t) < \deg(g) = \deg(\widehat{g})$ we conclude that $\deg(\widehat{t}) < \deg(\widehat{g})$ as well as $\widehat{t} \equiv t \bmod \pi$, similarly $\widehat{s} \equiv s \bmod \pi$. Finally, since $\pi \nmid \mathrm{lc}(s), \mathrm{lc}(u)$, thus $\pi \mid \mathrm{lc}(u\pi)$ and $\pi^2 \nmid \mathrm{lc}(u\pi)$, we have $\pi^2 \nmid \mathrm{lc}(\widehat{s})$, thus since $\mathrm{lc}(\widehat{g}) \in (R/\langle\pi^2\rangle)^*$ and $\deg(\widehat{t}\widehat{h}) = \deg(\widehat{t}) + \deg(\widehat{h}) < \deg(\widehat{g}) + \deg(\widehat{h}) = \deg(f)$, we conclude that $\deg(\widehat{s}\widehat{g}) = \deg(\widehat{s}) + \deg(\widehat{g}) < \deg(f)$, and hence $\deg(\widehat{s}) < \deg(\widehat{h})$. $\sharp$

(7.5) Algorithm: Lifting factorizations, Zassenhaus (1969).
Let $0 \neq \Psi \in \mathbb{Z}[X]$ be primitive and squarefree, where $n := \deg(\Psi) > 0$, and let $p \in \mathbb{N}$ be a prime such that $p \nmid \mathrm{lc}(\Psi)$ and that $\overline{\Psi} \in \mathbb{F}_p[X]$ is squarefree. Let again $\Theta_1, \ldots, \Theta_s \in \mathbb{Z}[X]$ be monic, irreducible and pairwise coprime, such that $\|\Theta_l\|_\infty \leq \frac{p}{2}$, for all $l \in \{1, \ldots, s\}$, and $\overline{\Psi} = \overline{\mathrm{lc}(\Psi)} \cdot \prod_{l=1}^s \overline{\Theta_l} \in \mathbb{F}_p[X]$, where the $\overline{\Theta_l} \in \mathbb{F}_p[X]$ are irreducible.

To apply Hensel lifting, for $R = \mathbb{Z}$ and modulus $p^{2^i} \in \mathbb{Z}$, where $i \in \mathbb{N}_0$, we consider the factorization $\Psi \equiv \Theta \cdot \widetilde{\Theta} \bmod p$, where $\Theta, \widetilde{\Theta} \in \mathbb{Z}[X]$ such that $p \nmid \mathrm{lc}(\Theta), \mathrm{lc}(\widetilde{\Theta})$ and $\|\Theta\|_\infty, \|\widetilde{\Theta}\|_\infty \leq \frac{p}{2}$ as well as $\Theta \equiv \mathrm{lc}(\Psi) \cdot \prod_{l=1}^k \Theta_l \bmod p$ and $\widetilde{\Theta} \equiv \prod_{l=k+1}^s \Theta_l \bmod p$, and $k := \lfloor \frac{s}{2} \rfloor$. Since the $\Theta_l \in \mathbb{Z}[X]$ are monic and $p \nmid \mathrm{lc}(\Psi)$, where $\mathbb{Z}/\langle p\rangle \cong \mathbb{F}_p$ is a field, the assumptions in (7.4) on leading coefficients are fulfilled, and since $\overline{\Psi}$ is squarefree the extended Euclidean algorithm in $\mathbb{F}_p[X]$ yields Bezout coefficients fulfilling the degree assumptions.

Letting $B_\Psi := |\mathrm{lc}(\Psi)| \cdot 2^n \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty \in \mathbb{N}$, and $e := \lceil \log_2(\log_p(2B_\Psi)) \rceil \in \mathbb{N}$ as well as $\pi := p^{2^e} \geq 2B_\Psi$, iterated application of Hensel lifting yields $\widehat{\Theta}, \widehat{\widetilde{\Theta}} \in \mathbb{Z}[X]$ such that $\|\widehat{\Theta}\|_\infty, \|\widehat{\widetilde{\Theta}}\|_\infty \leq \frac{\pi}{2}$ and $\Psi \equiv \widehat{\Theta} \cdot \widehat{\widetilde{\Theta}} \bmod \pi$. Applying this recursively to $\widehat{\Theta}, \widehat{\widetilde{\Theta}} \in \mathbb{Z}[X]$, we finally obtain $\Psi \equiv \mathrm{lc}(\Psi) \cdot \prod_{l=1}^s \widehat{\Theta_l} \bmod \pi$, where $\|\widehat{\Theta_l}\|_\infty \leq \frac{\pi}{2}$ and $\widehat{\Theta_l} \equiv \Theta_l \bmod p$, where the $\widehat{\Theta_l} \in \mathbb{Z}[X]$ are monic.

Now we proceed similar to (7.3): For a subset $\mathcal{I} \subseteq \{1, \ldots, s\}$ let $\Theta, \widetilde{\Theta} \in \mathbb{Z}[X]$ such that $p \nmid \mathrm{lc}(\Theta), \mathrm{lc}(\widetilde{\Theta})$ and $\|\Theta\|_\infty, \|\widetilde{\Theta}\|_\infty \leq \frac{\pi}{2}$ as well as $\Theta \equiv \mathrm{lc}(\Psi) \cdot \prod_{l \in \mathcal{I}} \widehat{\Theta_l} \bmod \pi$ and $\widetilde{\Theta} \equiv \mathrm{lc}(\Psi) \cdot \prod_{l \in \{1,\ldots,s\} \setminus \mathcal{I}} \widehat{\Theta_l} \bmod \pi$. Again we have $\Theta \cdot \widetilde{\Theta} = \mathrm{lc}(\Psi) \cdot \Psi \in \mathbb{Z}[X]$ if and only if $\|\Theta\|_1 \cdot \|\widetilde{\Theta}\|_1 \leq B_\Psi$; and in this case $\Psi = \frac{\Theta}{\gamma(\Theta)} \cdot \frac{\widetilde{\Theta}}{\gamma(\widetilde{\Theta})} \in \mathbb{Z}[X]$. Thus again we successively run through the subsets $\mathcal{I} \subseteq \{1, \ldots, s\}$ until we find $\Theta \mid \Psi \in \mathbb{Z}[X]$, and proceed recursively with $\widetilde{\Theta} := \frac{\Theta}{\Phi} \in \mathbb{Z}[X]$.

(7.6) Lemma. Let R be a principal ideal domain, let $p \in R$ be a prime, and let $\overline{}\colon R \to R/\langle p\rangle =: F$ be the natural map; note that F is a field. Let $0 \neq f, g, h \in R[X]$ such that $\deg(f) > 0$ and $p \nmid \mathrm{lc}(g), \mathrm{lc}(h)$ as well as $f \equiv gh \bmod p$ and $\gcd(\overline{g}, \overline{h}) = 1 \in F[X]$; note that hence g and h are Bezout coprime $\bmod\, p$.

Let $\widehat{g}, \widehat{h} \in R[X]$ be the associated Hensel lifts, with respect to p^e for some $e \in \mathbb{N}$. Let $\widetilde{g}, \widetilde{h} \in R[X]$ such that $\mathrm{lc}(\widetilde{g}) = \mathrm{lc}(\widehat{g})$ and $\mathrm{lc}(\widetilde{h}) = \mathrm{lc}(\widehat{h})$, as well as $\widetilde{g} \equiv g \bmod p$ and $\widetilde{h} \equiv h \bmod p$, and $f \equiv \widetilde{g}\widetilde{h} \bmod p^e$. Then $\widetilde{g} \equiv \widehat{g} \bmod p^e$ and $\widetilde{h} \equiv \widehat{h} \bmod p^e$.

Proof. Assume that $\widetilde{g} \not\equiv \widehat{g} \bmod p^e$ or $\widetilde{h} \not\equiv \widehat{h} \bmod p^e$. Let $1 \leq i < e$ be maximal such that both $\widetilde{g} \equiv \widehat{g} \bmod p^i$ and $\widetilde{h} \equiv \widehat{h} \bmod p^i$. Hence there are $u, v \in R[X]$ such that $\widetilde{g} - \widehat{g} = up^i$ and $\widetilde{h} - \widehat{h} = vp^i$, where $p \nmid u$ or $p \nmid v$. We may assume that $p \nmid u$. Hence from $0 \equiv \widetilde{g}\widetilde{h} - \widehat{g}\widehat{h} \equiv \widetilde{g}(\widetilde{h} - \widehat{h}) + \widehat{h}(\widetilde{g} - \widehat{g}) \equiv (\widetilde{g}v + \widehat{h}u)p^i \bmod p^e$ we conclude $p \mid p^{e-i} \mid \widetilde{g}v + \widehat{h}u \in R[X]$.

Letting $s, t \in R[X]$ such that $\deg(s) < \deg(h)$ and $\deg(t) < \deg(g)$, as well as $p \nmid \operatorname{lc}(s), \operatorname{lc}(t)$ and $sg + th \equiv 1 \bmod p$, we have $\overline{sg} + \overline{t}\overline{h} = 1 \in F[X]$. Thus we get $0 = \overline{t}(\overline{\widetilde{g}}\overline{v} + \overline{\widehat{h}}\overline{u}) = \overline{t}\overline{gv} + (1 - \overline{sg})\overline{u} = (\overline{t}\overline{v} - \overline{su})\overline{g} + \overline{u} \in F[X]$, which implies $\overline{g} \mid \overline{u} \in F[X]$. Since $\operatorname{lc}(\widetilde{g}) = \operatorname{lc}(\widehat{g})$ we have $\deg(\overline{u}) \leq \deg(u) < \deg(\widetilde{g}) = \deg(\widehat{g}) = \deg(g) = \deg(\overline{g})$. Thus we conclude $\overline{u} = 0$, a contradiction. ♯

(7.7) Lemma. Let R be a principal ideal domain, let $p \in R$ be a prime, and let $\overline{}\colon R \to R/\langle p \rangle =: F$ be the natural map. Let $0 \neq f \in R[X]$ such that $p \nmid \operatorname{lc}(f)$ and $\overline{f} \in F[X]$ is squarefree, let $g \in R[X]$ such that $g \mid f$, and let $h \in R[X]$ such that $p \nmid \operatorname{lc}(h)$ and $\deg(h) > 0$ as well as $f \equiv hu \bmod p^e$, for some $u \in R[X]$ such that $p \nmid \operatorname{lc}(u)$ and some $e \in \mathbb{N}$, and $g \equiv hv \bmod p$, for some $v \in R[X]$ such that $p \nmid \operatorname{lc}(v)$. Then we have $g \equiv h\widehat{v} \bmod p^e$, for some $\widehat{v} \in R[X]$.

Proof. We have $p \nmid \operatorname{lc}(g)$ and $\overline{g} \in F[X]$ is squarefree, hence $\gcd(\overline{h}, \overline{v}) = 1 \in F[X]$, thus h and v are Bezout coprime $\bmod p$. Hence Hensel lifting yields $\widehat{h}, \widehat{v} \in R[X]$ such that $\widehat{h} \equiv h \bmod p$ and $\widehat{v} \equiv v \bmod p$, as well as $\operatorname{lc}(\widehat{h}) = \operatorname{lc}(h)$ and $\deg(\widehat{v}) = \deg(v)$, as well as $g \equiv \widehat{h}\widehat{v} \bmod p^e$.

Letting $w \in R[X]$ such that $f = gw$, we have $p \nmid \operatorname{lc}(w)$ and $\widehat{h} \cdot (\widehat{v}w) \equiv gw \equiv f \equiv hu \bmod p^e$. Moreover, from $\overline{h}\overline{\widehat{v}w} = \overline{h}\overline{u} \in F[X]$ we get $\overline{\widehat{v}w} = \overline{vw} = \overline{u} \in F[X]$. Since $\operatorname{lc}(\widehat{h}) = \operatorname{lc}(h)$ and $\operatorname{lc}(\widehat{v}w) \equiv \operatorname{lc}(u) \bmod p$, by (7.6) there is $p \nmid \lambda \in R$ such that $\widehat{v}w \equiv u\lambda \bmod p^e$ and $\widehat{h} \equiv h \bmod p^e$. Thus $g \equiv \widehat{h}\widehat{v} \equiv h\widehat{v} \bmod p^e$. ♯

(7.8) Lemma. Let $0 \neq f, g \in \mathbb{Z}[X]$ such that $n := \deg(f) > 0$ and $m := \deg(g) > 0$. Moreover, let $\pi \in \mathbb{N}$ such that $\|f\|_2^m \cdot \|g\|_2^n < \pi$, and let $0 \neq h \in \mathbb{Z}[X]$ monic such that $\deg(h) > 0$ and $f \equiv hh' \bmod \pi$ as well as $g \equiv hh'' \bmod \pi$, for some $h', h'' \in \mathbb{Z}[X]$. Then $\gcd(f, g) \in \mathbb{Z}[X]$ is non-constant.

Proof. Assume that $\gcd(f, g) \in \mathbb{Z}[X]$ is constant. By (4.4) and (4.5) there are $s, t \in \mathbb{Z}[X]$ such that $sf + tg = \operatorname{res}(f, g) \neq 0 \in \mathbb{Z}$. Hence $\operatorname{res}(f, g) \equiv h(sh' + th'') \bmod \pi$, and since h is monic and $\deg(h) > 0$, we have $\pi \mid \operatorname{res}(f, g) \in \mathbb{Z}$. By Hadamard's inequality we have $|\operatorname{res}(f, g)| \leq \|f\|_2^m \cdot \|g\|_2^n < \pi$, a contradiction. ♯

(7.9) Algorithm: Factorization using lattice base reduction.
Let $0 \neq \Psi \in \mathbb{Z}[X]$ be primitive and squarefree, where $n := \deg(\Psi) > 0$, and let $p \in \mathbb{N}$ be a prime such that $p \nmid \operatorname{lc}(\Psi)$ and that $\overline{\Psi} \in \mathbb{F}_p[X]$ is squarefree. By Mignotte's inequality let $B_\Psi := 2^n \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty \in \mathbb{N}$, and let $e := \lceil \log_2(\log_p(2^{\frac{n^2}{2}} \cdot B_\Psi^{2n})) \rceil \in \mathbb{N}$, thus $\pi := p^{2^e} \geq 2^{\frac{n^2}{2}} \cdot B_\Psi^{2n}$.

By Hensel lifting let $\Theta_1, \ldots, \Theta_s \in \mathbb{Z}[X]$ be monic, irreducible and pairwise coprime, such that $\|\Theta_i\|_\infty \leq \frac{\pi}{2}$, for all $i \in \{1, \ldots, s\}$, and $\Psi \equiv \operatorname{lc}(\Psi) \cdot \prod_{i=1}^s \Theta_i \bmod \pi$, in particular we have $\overline{\Psi} = \overline{\operatorname{lc}(\Psi)} \cdot \prod_{i=1}^s \overline{\Theta_l} \in \mathbb{F}_p[X]$, where the $\overline{\Theta_i} \in \mathbb{F}_p[X]$ are irreducible. Let $\Theta \in \{\Theta_1, \ldots, \Theta_s\}$ and $l := \deg(\Theta) > 0$, and let $\sum_{j=0}^{\deg(\Phi)} \varphi_j X^j = \Phi \mid \Psi \in \mathbb{Z}[X]$ be irreducible such that $\overline{\Theta} \mid \overline{\Phi} \in \mathbb{F}_p[X]$. Then by (7.7) we have $\Phi \equiv \Theta\widetilde{\Theta} \bmod \pi$ for some $\widetilde{\Theta} \in \mathbb{Z}[X]$, where $\deg(\widetilde{\Theta}) = \deg(\Phi) - l \geq 0$.

For $l < m \leq n$ let $L \subseteq \mathbb{R}^m$ be the $\mathbb{Z}$-lattice generated by the coefficient tuples of the polynomials $\{\Theta, \Theta \cdot X, \ldots, \Theta \cdot X^{m-l-1}\} \subseteq \mathbb{Z}[X]$ and $\{\pi, \pi X, \ldots, \pi X^{l-1}\} \subseteq \mathbb{Z}[X]$, with respect to the $\mathbb{Z}$-basis $\{1, X, \ldots, X^{m-1}\} \subseteq \mathbb{Z}[X]_{<m}$, i. e. letting $\Theta = \sum_{k=0}^{l} \vartheta_k X^k$, the $\mathbb{Z}$-lattice L is generated by the rows of the following matrix, where the upper half consists of $m - l$ rows, and the lower half consists of l rows:

$$\left[\begin{array}{cccccccccc}
\vartheta_0 & \vartheta_1 & \ldots & \vartheta_{l-1} & 1 & & & & & \\
 & \vartheta_0 & \ldots & \vartheta_{l-2} & \vartheta_{l-1} & 1 & & & & \\
 & & \ddots & & & & & \ddots & & \\
 & & & \ddots & & & & & \ddots & \\
 & & & & \vartheta_0 & & \ldots & \ldots & \ldots & 1 \\
\hline
\pi & & & & & & & & & \\
 & \pi & & & & & & & & \\
 & & \ddots & & & & & & & \\
 & & & \pi & & & & & &
\end{array}\right] \in \mathbb{Z}^{m \times m}.$$

We have $[g_0, \ldots, g_{m-1}] \in L$ if and only if for $\Gamma := \sum_{j=0}^{m-1} g_j X^j \in \mathbb{Z}[X]_{<m}$ there are $q \in \mathbb{Z}[X]_{<m-l}$ and $r \in \mathbb{Z}[X]_{<l}$ such that $\Gamma = q\Theta + r\pi \in \mathbb{Z}[X]$, which holds if and only if there is $q \in \mathbb{Z}[X]_{<m-l}$ such that $\Gamma \equiv q\Theta \bmod \pi$: If $[g_0, \ldots, g_{m-1}] \in L$, then indeed we have $\Gamma \equiv q\Theta \bmod \pi$. If conversely $\Gamma := \sum_{j=0}^{m-1} g_j X^j \in \mathbb{Z}[X]_{<m}$ such that there is $q \in \mathbb{Z}[X]_{<m-l}$ such that $\Gamma \equiv q\Theta \bmod \pi$, i. e. there is $r \in \mathbb{Z}[X]_{<m}$ such that $\Gamma = q\Theta + r\pi \in \mathbb{Z}[X]$, then since $\Theta \in \mathbb{Z}[X]$ is monic there are $q' \in \mathbb{Z}[X]_{<m-l}$ and $r' \in \mathbb{Z}[X]_{<l}$ such that $r = q'\Theta + r' \in \mathbb{Z}[X]$. This yields $\Gamma = q\Theta + (q'\Theta + r')\pi = (q + q'\pi)\Theta + r'\pi \in \mathbb{Z}[X]$, and since $\deg(q + q'\pi) < m - l$ and $\deg(r') < l$ we have $[g_0, \ldots, g_{m-1}] \in L$.

Hence for $m > \deg(\Phi) \geq l$ we have $\deg(\widetilde{\Theta}) < m - l$, and from $\Phi \equiv \Theta\widetilde{\Theta} \bmod \pi$ we conclude $[\varphi_0, \ldots, \varphi_{m-1}] \in L$, where we let $\varphi_j := 0$ for $j > \deg(\Phi)$. Still letting $m > \deg(\Phi)$, let $[\gamma_0, \ldots, \gamma_{m-1}] \in L$ be the first element of an LLL reduced $\mathbb{Z}$-basis of L, with respect to parameter $\gamma = \frac{3}{4}$, and let $\Gamma := \sum_{j=0}^{m-1} \gamma_j X^j \in \mathbb{Z}[X]$. Since $\|\Phi\|_2 \leq \|\Phi\|_1 \leq B_\Psi$ by (5.7) we have $\|\Gamma\|_2 \leq 2^{\frac{m-1}{2}} \cdot \min(L) \leq 2^{\frac{n}{2}} \cdot \|\Phi\|_2 \leq 2^{\frac{n}{2}} \cdot B_\Psi$. Hence $\|\Gamma\|_2^{\deg(\Phi)} \cdot \|\Phi\|_2^{\deg(\Gamma)} \leq 2^{\frac{n^2}{2}} \cdot B_\Psi^{2n} \leq \pi$. Since $\Gamma \equiv q\Theta \bmod \pi$ and $\Phi \equiv \Theta\widetilde{\Theta} \bmod \pi$, by (7.8) we have $\gcd(\Gamma, \Phi) \in \mathbb{Z}[X]$ non-constant, thus since $\Phi \in \mathbb{Z}[X]$ is irreducible we have $\gcd(\Gamma, \Phi) \sim \Phi \in \mathbb{Z}[X]$. Hence for $m = \deg(\Phi) + 1$ we have $m - 1 = \deg(\Phi) \leq \deg(\Gamma) < m$, and thus $\Gamma \sim \Phi \in \mathbb{Z}[X]$.

Hence we successively let $m \in \{l+1, \ldots, n\}$, as above compute the first element Γ of an LLL reduced $\mathbb{Z}$-basis of L, and check whether $\Gamma \mid \Psi \in \mathbb{Z}[X]$. Note that we also can apply other checks, e. g. the one using 1-norms applied in the Zassenhaus algorithm.

(7.10) Remark. We briefly comment on running times: The input length of $0 \neq \Psi \in \mathbb{Z}[X]$ of degree $\deg(\Psi) = n$ is asymptotically $\sim n \ln(\|\Psi\|_\infty)$.

Squarefree factorization needs a gcd computation $\gcd(\Psi, \Psi') \in \mathbb{Z}[X]$, where $\Psi' := \frac{\partial \Psi}{\partial X} \in \mathbb{Z}[X]$ is the formal derivative; since $\|\Psi'\|_\infty \leq n \cdot \|\Psi\|_\infty$ the bit lengths of the coefficients of the polynomials occurring in the extended Euclidean algorithm are by (4.12) in $O(n \cdot \ln(n^2 \cdot \|\Psi\|_\infty))$, hence this needs a polynomial number of bit operations. Moreover, a division $\frac{\Psi}{\gcd(\Psi,\Psi')} \in \mathbb{Z}[X]$ is needed, which as well needs a polynomial number of bit operations.

In the big prime version to lift factorizations we choose $p \geq |\mathrm{lc}(\Psi)| \cdot 2^{n+1} \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty$, where we have to avoid prime divisors of $\mathrm{disc}(\Psi) \in \mathbb{Z}$. As by the proof of (4.12) we have $|\mathrm{disc}(\Psi)| = |\mathrm{res}(\Psi, \Psi')| \leq (n+1)^n \cdot n^{2n} \cdot \|\Psi\|_\infty^{2n}$, we additionally choose $p \geq |\mathrm{disc}(\Psi)|$, thus the bit length $\ln(p)$ is a polynomial in $n \ln(\|\Psi\|_\infty)$. Note that by **Bertrand's postulate** there always is a prime at most twice as large as a given positive integer. Polynomial factorization in $\mathbb{F}_p[X]$ deterministically needs at least $O(p)$ finite field operations, hence has exponential running time, while a randomized version needs a number of finite field operations which is a polynomial in $n \ln(p)$. The coefficients of the integer polynomials occurring are bounded by p, hence a polynomial number of bit operations for the ring operations in $\mathbb{Z}[X]$ is needed. Alone 2^{s-1} subsets of the index set $\{1, \ldots, s\}$ have to be checked, which needs exponential running time.

In the Zassenhaus algorithm we choose a small prime $p \not| \mathrm{lc}(\Psi)$, where still we have to avoid prime divisors of $\mathrm{disc}(\Psi) \in \mathbb{Z}$. Using the **prime number theorem**, see [3, Ch.18.4], a prime p fulfilling these requirements and such that $p \sim n \cdot \ln(n \cdot \|\Psi\|_\infty)$ can be found deterministically needing a polynomial number of bit operations, see [3, Cor.18.12]. Thus polynomial factorization in $\mathbb{F}_p[X]$, deterministically needing a number of finite field operations which is a polynomial in $n \ln(p)$, needs a polynomial number of bit operations. The number of Hensel lifting steps e is polynomial, and the coefficients of the integer polynomials occurring are bounded by $\pi = p^{2^e} \sim |\mathrm{lc}(\Psi)| \cdot 2^{n+1} \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty$, hence a polynomial number of bit operations for the ring operations in $\mathbb{Z}[X]$ is needed. Still, alone 2^{s-1} subsets of the index set $\{1, \ldots, s\}$ have to be checked, which needs exponential running time.

Factorization using lattice base reduction avoids the factor combination step in the Zassenhaus algorithm. The elements, b say, of the $\mathbb{Z}$-bases defining the relevant $\mathbb{Z}$-lattices fulfill $\|b\| \leq \pi \sim 2^{\frac{n^2}{2}} \cdot (2^n \cdot \sqrt{n+1} \cdot \|\Psi\|_\infty)^{2n}$, thus by (5.9) the LLL algorithm needs $O(n^4 \cdot \ln(\pi))$ ring operations in $\mathbb{Q}$, where the numerators and denominators occurring have bit length in $O(n \cdot \ln(\pi))$, which amounts to a polynomial number of bit operations; moreover the LLL algorithm is performed polynomially many times. Although the running time still is dominated by the lattice base reduction step, we thus get an overall polynomial running time.

8 Exercises (in German)

(8.1) Aufgabe: Turing-Maschinen.
Man gebe eine Turing-Maschine über dem Alphabet $\mathcal{X} = \{0,1\}$ an, die für $n \in \mathbb{N}_0$ in Binärdarstellung als Eingabe den Nachfolger $n+1 \in \mathbb{N}$ in Binärdarstellung ausgibt.

Beweis. Siehe [13, Ex.2.2]. $\sharp$

(8.2) Aufgabe: k-Band-Turing-Maschinen.
a) Für $k \in \mathbb{N}$ gebe man eine Definition einer **k-Band-Turing-Maschine** über dem Alphabet $\mathcal{X}$ mit Transitionsfunktion

$$\tau\colon \left(\mathcal{X} \,\dot{\cup}\, \mathcal{Y}\right)^k \times (\mathcal{S} \setminus \{s_\infty\}) \longrightarrow \left((\mathcal{X} \,\dot{\cup}\, \mathcal{Y}) \times \{\leftarrow, \uparrow, \rightarrow\}\right)^k \times \mathcal{S}$$

an, und definiere Eingaben, Ausgaben und Konfigurationen.
b) Man zeige: Wird die Sprache $\mathcal{L}$ durch eine k-Band-Turing-Maschine mit Laufzeit f akzeptiert, so wird $\mathcal{L}$ von einer Turing-Maschine mit Laufzeit in $O(f^2)$ akzeptiert.

Beweis. Siehe [13, Ch.2.3] oder [1, La.10.1]. $\sharp$

(8.3) Aufgabe: Nichtdeterministische Turing-Maschinen.
Man zeige: Wird die Sprache $\mathcal{L}$ durch die nichtdeterministische Turing-Maschine $\mathcal{T}$ mit Laufzeit f entschieden, so wird $\mathcal{L}$ von einer 3-Band-Turing-Maschine mit Laufzeit in $O(n \mapsto c_{\mathcal{T}}^{f(n)})$ entschieden, wobei $c_{\mathcal{T}} > 1$ eine von $\mathcal{T}$ abhängige Konstante ist.

Beweis. Siehe [13, Ch.2.3]. $\sharp$

(8.4) Aufgabe: Chernoff-Schranke.
Es seien $X_1, \ldots, X_k$ unabhängige Zufallsvariablen mit Wertebereich $\{0,1\}$ und $P[X_i = 1] = \epsilon$, für $0 < \epsilon < 1$, sowie $X := \sum_{i=1}^k X_i$. Man zeige: Für $0 < \vartheta \leq 1$ gilt $P[X \geq (1+\vartheta)\epsilon k] \leq e^{-\frac{1}{2}\vartheta^2 \epsilon k}$.

Hinweis. Man betrachte die Zufallsvariable e^{tX}, für $t \in \mathbb{R}$, und verwende $P[X \geq sE(X)] \leq \frac{1}{s}$, für $s > 0$, und die Konvexität der Exponentialfunktion.

Beweis. Siehe [13, La.11.9]. $\sharp$

(8.5) Aufgabe: Asymptotisches Verhalten.
a) Man zeige ohne Benutzung der Stirling-Formel: Es gelten $\ln(n!) \in O(n\ln(n))$ und $n\ln(n) \in O(\ln(n!))$.
b) Für $k \in \mathbb{N}$ zeige man: Es gilt $\sum_{i=1}^n i^k \sim \frac{n^{k+1}}{k+1}$.

c) Man betrachte die **Fibonacci-Zahlen** $F_n := F_{n-1} + F_{n-2} \in \mathbb{N}$, für $n \geq 3$, wobei $F_2 = F_1 := 1$. Man gebe eine einfache Funktion g mit $F_n \sim g(n)$ an.
d) Man betrachte die folgenden Funktionen $\mathbb{N} \setminus \{1\} \to \mathbb{R}_{>0}$, wobei $0 < \epsilon < 1 < c$:

$$1 < \ln(\ln(n)) < \ln(n) < e^{(\ln(n))^{\frac{1}{2}} \cdot (\ln(\ln(n)))^{\frac{1}{2}}} < n^\epsilon < n^c < n^{\ln(n)} < c^n < n^n < c^{c^n}$$

Man zeige, daß für je zwei dieser Funktionen mit $f < g$ auch $f \in o(g)$ gilt.

Beweis. **a)** Siehe [10, Ex.2.2.2, Exc.2.2.4]. **b)** Siehe [10, Ex.2.1.3].
c) Siehe [10, Exc.2.2.2]. **d)** Siehe [12, Ex.2.58]. ♯

(8.6) Aufgabe: Laufzeitabschätzungen.
a) Man zeige: Für $n \in \mathbb{N}$ kann man $n!$ mit $O(n^2 \cdot \ln^2(n))$ Bitoperationen berechnen. Wieviele Bitoperationen braucht man zur Berechnung von n^n?
b) Man zeige: Für $n \in \mathbb{N}$ gilt $\sum_{i=1}^n i^2 = \frac{n(n+1)(2n+1)}{6}$. Wieviele Bitoperationen braucht man zur Berechnung der linken bzw. der rechten Seite dieser Gleichung?
c) Für $i \in \mathbb{N}$ sei $F_i \in \mathbb{N}$ die zugehörige Fibonacci-Zahl. Wieviele Bitoperationen braucht man zur Berechnung von $\sum_{i=1}^n F_i$ bzw. $\prod_{i=1}^n F_i$, für $n \in \mathbb{N}$?
d) Für $1 \neq z \in \mathbb{N}$ und $n \in \mathbb{N}$ seien $\mathcal{P}_{z,n} := \{p \in \mathbb{N} \text{ prim}; b_z(p) \leq n\}$. Wieviele Bitoperationen braucht man zur Berechnung von $\sum \mathcal{P}_{z,n}$ bzw. $\prod \mathcal{P}_{z,n}$?

Beweis. **a)** Siehe [10, Ex.2.3.3] und [10, Exc.2.3.1]. **b)** Siehe [10, Exc.2.3.3].
c) Siehe [10, Exc.2.3.5]. **d)** Siehe [10, Exc.2.3.6]. ♯

(8.7) Aufgabe: Subtraktion.
Man gebe einen Algorithmus zur Subtraktion zweier Zahlen $n, m \in \mathbb{N}$ an. Wie entscheidet man algorithmisch, ob $n \geq m$ gilt?

Beweis. Siehe [3, Exc.2.3]. ♯

(8.8) Aufgabe: Matrixmultiplikation.
Für $k, m, n \in \mathbb{N}$ seien $A \in \mathbb{Z}^{k \times m}$ und $B \in \mathbb{Z}^{m \times n}$. Wieviele Ringoperationen braucht man zur klassischen Berechnung des Matrixprodukts $AB \in \mathbb{Z}^{k \times n}$?

Beweis. Siehe [3, Exc.2.11]. ♯

(8.9) Aufgabe: Euklidischer Algorithmus.
Für $q, m, n \in \mathbb{N}$, $q \neq 1$ zeige man: Es gilt $\text{ggT}(q^m - 1, q^n - 1) = q^{\text{ggT}(m,n)} - 1$.

(8.10) Aufgabe: Satz von Lamé.
Es seien $m \geq n \in \mathbb{N}$. Man zeige, daß der erweiterte Euklidische Algorithmus höchstens $l = \lceil \frac{\ln(\sqrt{5} \cdot n)}{\ln(\frac{1+\sqrt{5}}{2})} \rceil - 2$ Schritte benötigt.

Beweis. Siehe [2, Thm.1.3.2]. ♯

(8.11) Aufgabe: Binärer ggT-Algorithmus.
Es seien $m, n \in \mathbb{N}$.

```
1. $k \leftarrow 0$.
2. while $0 \equiv m \bmod 2$ and $0 \equiv n \bmod 2$ do
     $m \leftarrow \frac{m}{2}$
     $n \leftarrow \frac{n}{2}$
     $k \leftarrow k + 1$
3. while $0 \equiv m \bmod 2$ do $m \leftarrow \frac{m}{2}$
4. while $0 \equiv n \bmod 2$ do $n \leftarrow \frac{n}{2}$
5. repeat
     $t \leftarrow \frac{m-n}{2}$.
     if $t \neq 0$ then while $0 \equiv t \bmod 2$ do $t \leftarrow \frac{t}{2}$
     if $t > 0$ then $m \leftarrow t$
     if $t < 0$ then $n \leftarrow -t$
     until $t = 0$
6. return $2^k \cdot m$
```

Man zeige, daß dieser Algorithmus $\mathrm{ggT}(m, n)$ berechnet, und gebe unter Verwendung von $\max\{b_2(m), b_2(n)\}$ eine Abschätzung für die benötigte Anzahl von Bitoperationen an. Welche Vorteile besitzt dieser Algorithmus gegenüber dem erweiterten Euklidischen Algorithmus, wenn es nur auf die Berechnung von $\mathrm{ggT}(m, n)$ ankommt?

Beweis. Siehe [2, Alg.1.3.5]. ♯

(8.12) Aufgabe: Arithmetik in $\mathbb{Z}$.
Man implementiere die folgenden Algorithmen zur Arithmetik in $\mathbb{Z}$, unter Benutzung eines Computeralgebra-Systems wie MAPLE, und vergleiche Laufzeiten und asymptotisches Verhalten:
a) Klassischer und Karatsuba-Algorithmus zur Multiplikation.
b) Klassischer und binärer Algorithmus zum modularen Potenzieren.
c) Erweiterter Euklidischer und binärer Algorithmus zur ggT-Berechnung.

(8.13) Aufgabe: Modulare Inversion.
a) Man gebe einen Algorithmus an, der für $m \in \mathbb{N}$ und $k \in \{0, \ldots, m-1\}$ mit $\overline{k} \in (\mathbb{Z}/\langle m \rangle)^*$ das **modulare Inverse** $l \in \{0, \ldots, m-1\}$ mit $\overline{kl} = \overline{1} \in \mathbb{Z}/\langle m \rangle$ berechnet. Man gebe eine Laufzeitabschätzung an.
b) Ist $m \in \mathbb{N}$ eine Primzahl, so gebe man einen alternativen Algorithmus zur Berechnung von modularen Inversen an, der den Satz von Fermat benutzt. Man gebe eine Laufzeitabschätzung an.

Beweis. a) Siehe [3, Ch.4.2]. **b)** Siehe [3, Ch.4.4]. ♯

(8.14) Aufgabe: Lineare diophantische Gleichungen.
Es seien $a, b, c \in \mathbb{Z}$ mit $[a, b] \neq [0, 0]$. Man zeige: Die **lineare diophantische Gleichung** $ax + by = c$ hat genau dann eine Lösung $[x, y] \in \mathbb{Z}^2$, wenn $\mathrm{ggT}(a, b) \mid c$ gilt. Wie sieht in diesem Fall die Lösungsgesamtheit aus? Was bedeutet das für die Entscheidbarkeit des Lösungsproblems für lineare diophantische Gleichungen?

Beweis. Siehe [3, Ch.4.5]. $\sharp$

(8.15) Aufgabe: Chinesischer Restsatz.
Es sei $k \in \mathbb{N}$. Man gebe einen Algorithmus an, der für $n_1, \ldots, n_k \in \mathbb{N}$ mit $\mathrm{ggT}(n_i, n_j) = 1$, für alle $i, j \in \{1, \ldots, k\}$, sowie $r_i \in \{0, \ldots, n_i - 1\}$ die **simultanen Kongruenzen** $r \equiv r_i \bmod n_i$ löst, und die eindeutig bestimmte Lösung $r \in \{0, \ldots, n-1\}$, wobei $n := \prod_{i=1}^{k} n_i \in \mathbb{N}$, berechnet. Man zeige, daß dazu höchstens $O(\ln^2(n))$ Bitoperationen benötigt werden.

Beweis. Siehe [3, Ch.5.4]. $\sharp$

(8.16) Aufgabe: Polynomarithmetik.
Es seien R ein kommutativer Ring, F ein Körper, und $R[X]$ sowie $F[X]$ die zugehörigen Polynomringe.
a) Man formuliere den Karatsuba-Algorithmus zur Multiplikation der Polynome $0 \neq f, g \in R[X]$, und zeige für $\deg(f) \geq \deg(g)$, daß hierzu $O(\deg(f)^{\log_2 3})$ Ringoperationen im Ring R benötigt werden.
b) Man formuliere den erweiterten Euklidischen Algorithmus für Polynome $0 \neq f, g \in F[X]$, und zeige, daß hierzu $O(\deg(f) \cdot \deg(g))$ Ringoperationen im Ring F benötigt werden.
c) Man gebe einen Algorithmus zum Lösen simultaner Kongruenzen über dem Polynomring $F[X]$, zusammen mit einer Laufzeitabschätzung, an.

(8.17) Aufgabe: Primitive Einheitswurzeln.
Es seien F ein Körper und $1 \neq n \in \mathbb{N}$ mit $\mathrm{char}(F) \not| \, n$. Ist $\overline{X} \in F[X]/\langle X^n - 1 \rangle$ eine primitive n-te Einheitswurzel?

Beweis. Siehe [3, Exc.8.28]. $\sharp$

(8.18) Aufgabe: FFT-Algorithmus.
Es seien R ein kommutativer Ring, $l \in \mathbb{N}$ und $n := 2^l$, sowie $\omega \in R$ eine primitive n-te Einheitswurzel. Man gebe einen FFT-Algorithmus zur Berechnung der diskreten Fourier-Transformation δ_ω für $f \in R[X]_{<n}$ an, der auf der Zerlegung $f = f_0(X^2) + X \cdot f_1(X^2) \in R[X]$, für geeignete $f_0, f_1 \in R[X]_{<\frac{n}{2}}$, beruht. Man gebe eine Laufzeitabschätzung an.

Beweis. Siehe [3, Exc.8.25]. $\sharp$

(8.19) Aufgabe: 3-adischer FFT-Algorithmus.
Es seien R ein kommutativer Ring, $n = 3^l$ für ein $l \in \mathbb{N}$, und $\omega \in R$ eine primitive n-te Einheitswurzel. Man gebe einen FFT-Algorithmus zur Berechnung der diskreten Fourier-Transformation $\delta_\omega \colon R^n \to R^n$ an. Man zeige, daß dazu höchstens $O(n \ln(n))$ Ringoperationen benötigt werden.

Hinweis. Für $f \in R[X]_{<n}$ betrachte man f mod $(X^{\frac{n}{3}} - \omega^{\frac{jn}{3}})$ für $j \in \{0, \ldots, 2\}$.

Beweis. Siehe [3, Exc.8.26]. ♯

(8.20) Aufgabe: Schönhage-Algorithmus.
Es seien R ein kommutativer Ring mit $3 \in R^*$ und $2n = 2 \cdot 3^l$ für ein $l \in \mathbb{N}$. Man gebe einen zum Schönhage-Strassen-Algorithmus analogen Algorithmus an, der für $f, g \in R[X]_{<2n}$ die **kubische Konvolution** $h \in R[X]_{<2n}$ mit $h \equiv fg$ mod $(X^{2n} + X^n + 1)$ berechnet. Man zeige, daß dazu höchstens $O(n \ln(n) \ln(\ln(n)))$ Ringoperationen benötigt werden.

Hinweis. Es seien $m := 3^{\lceil \frac{l}{2} \rceil}$ und $t := 3^{\lfloor \frac{l}{2} \rfloor}$, sowie $\omega \in R[X]/\langle X^{2m} + X^m + 1 \rangle$ eine primitive $3t$-te Einheitswurzel. Man schreibe $f = f'(X, X^m)$ und $g = g'(X, X^m)$ für $f', g' \in R[X, Y]$, und für $j \in \{1, 2\}$ seien $h'_j \in R[X, Y]$ mit $\overline{f'}(\omega^j Y)\overline{g'}(\omega^j Y) \equiv \overline{h'_j}(\omega^j Y)$ mod $(Y^t - 1)$ in $(R[X, Y]/\langle X^{2m} + X^m + 1 \rangle)[Y]$. Man setze $h' := \frac{2\omega^t+1}{3} \cdot (Y^t(h'_2 - h'_1) + \omega^{2t} h'_1 - \omega^t h'_2) \in R[X, Y]$. Zur Berechnung der $\overline{h'_j}$ verwende man den 3-adischen FFT-Algorithmus und Rekursion.

Beweis. Siehe [3, Exc.8.30]. ♯

(8.21) Aufgabe: Sylvester-Matrix.
Es seien F ein Körper und $0 \neq f, g \in F[X]$ mit $\deg(f) + \deg(g) \geq 1$. Wie kann man mittels des Gauß-Algorithmus, angewendet auf die Sylvester-Matrix $S(f, g)$, den $\mathrm{ggT}(f, g) \in F[X]$ bestimmen?

(8.22) Aufgabe: Sylvester-Matrix.
Es seien F ein Körper und $0 \neq f, g \in F[X]$ mit $\deg(f) + \deg(g) \geq 1$. Man zeige: Es gilt $\dim_F(\ker(S(f, g))) = \deg(\mathrm{ggT}(f, g))$.

Hinweis. Es gibt $s \in F[X]_{\deg(g)-k}$ und $t \in F[X]_{\deg(f)-k}$ mit $0 \neq [s, t] \in \ker(\varphi(f, g))$ genau dann, wenn $k \in \{1, \ldots, \deg(\mathrm{ggT}(f, g))\}$ ist.

Beweis. Siehe [3, Exc.6.16]. ♯

(8.23) Aufgabe: Resultanten.
Es seien R ein faktorieller Ring und $f, g, h \in R[X]$. Man zeige:
a) Für $\lambda \in R$ gilt $\mathrm{ggT}(f(\lambda), g(\lambda)) \mid \mathrm{res}(f, g) \in R$.
b) Es gilt $\mathrm{res}(f, gh) = \mathrm{res}(f, g) \cdot \mathrm{res}(f, h) \in R$.
c) Ist $0 \leq k \leq \min\{\deg(f), \deg(g)\}$, so gilt $\mathrm{ggT}(\mathrm{lc}(f), \mathrm{lc}(g)) \mid \mathrm{res}_k(f, g) \in R$.

Beweis. **a)** Siehe [3, Exc.6.10]. **b)** Siehe [3, Exc.6.12]. **c)** Siehe [3, Exc.6.41]. ♯

(8.24) Aufgabe: Fundamentalsatz über Subresultanten.
Es seien F ein Körper und $0 \neq f, g \in F[X]$. Für $i \in \{0, \ldots, l\}$ seien $\lambda_i \in F^*$ die Leitkoeffizienten und $n_i \in \mathbb{N}_0$ die Restgrade im normierten Euklidischen Algorithmus für f und g. Für $i \in \{1, \ldots, l\}$ zeige man: Es gilt $\mathrm{res}_{n_i}(f, g) = (-1)^{\sum_{j=1}^{i-1}(n_{j-1}-n_i)(n_j-n_i)} \cdot \lambda_0^{n_1-n_i} \cdot \prod_{j=1}^{i} \lambda_j^{n_{j-1}-n_i}$.

Hinweis. Siehe Aufgabe (8.21).

(8.25) Aufgabe: Collins-Algorithmus.
Es seien R ein faktorieller Ring, $0 \neq f, g \in R[X]$ primitiv mit $\deg(f) \geq \deg(g)$.

```
$r_0 \leftarrow f$, $r_1 \leftarrow g$
$n_0 \leftarrow \deg(f)$, $\lambda_0 \leftarrow 1$, $\eta_0 \leftarrow 1$
$i \leftarrow 1$
while $r_i \neq 0$ do
    $n_i \leftarrow \deg(r_i)$
    $\delta_i \leftarrow n_{i-1} - n_i$
    $\lambda_i \leftarrow \mathsf{lc}(r_i)$
    $\eta_i \leftarrow \eta_{i-1}^{1-\delta_i} \cdot \lambda_i^{\delta_i}$
    $\widehat{r}_{i+1} \leftarrow (\lambda_i^{\delta_i+1} \cdot r_{i-1}) \bmod r_i$        # Pseudo-Division
    $r_{i+1} \leftarrow \frac{1}{\lambda_{i-1} \cdot \eta_{i-1}^{\delta_i}} \cdot \widehat{r}_{i+1}$
    $i \leftarrow i + 1$
return $r_{i-1}$        # $i = l + 1$
```

a) Man zeige: Für $i \in \{1, \ldots, l\}$ gelten $\eta_i = \pm\mathrm{res}_{n_i}(f, g) \in R$ und $r_i \in R[X]$.
b) Man zeige: Sind $R = \mathbb{Z}$ und $\|f\|_\infty, \|g\|_\infty \leq B$, für ein $B > 0$, so gilt für $i \in \{1, \ldots, l\}$ auch $\|r_i\|_\infty \leq (n+1)^{\frac{m}{2}} \cdot (m+1)^{\frac{n}{2}} \cdot B^{n+m}$.
c) Was ist der Vorteil des Collins-Algorithmus im Vergleich zum primitiven Euklidischen Algorithmus und zum normierten Euklidischen Algorithmus über $\mathrm{Quot}(R)[X]$? Wie kann man die Resultante $\mathrm{res}(f, g)$ mit ihm berechnen?

Hinweis zu a). Was bedeutet Pseudo-Division für die verallgemeinerte Sylvester-Matrix $S_{n_i}(f, g)$? Außerdem betrachte man geeignet vergrößerte verallgemeinerte Sylvester-Matrizen $\widetilde{S}_{n_{i-1}}(f, g)$, um die Koeffizienten von r_i als Determinanten zu beschreiben.

Beweis. Siehe [8, p.429ff., Exc.4.6.1.24] und [2, Alg.3.3.7]. ♯

(8.26) Aufgabe: Bivariate Polynome.
Es seien F ein Körper und $0 \neq f, g \in F[X]$ mit $\deg_X(f), \deg_X(g) \leq n$ und $\deg_Y(f), \deg_Y(g) \leq d$, für $n, d \in \mathbb{N}_0$, sowie $\mathrm{lc}_X(f) = \mathrm{lc}_X(g) = 1$. Man zeige: Ist $\mathrm{ggT}(f(X, \lambda), g(X, \lambda)) \in F[X]$ nicht kontant für mindestens $2nd + 1$ paarweise verschiedene $\lambda \in F$, so ist $\deg_X(\mathrm{ggT}(f, g)) > 0$.

Beweis. Siehe [3, Exc.6.20]. $\sharp$

(8.27) Aufgabe: Ebene Kurven.
Es seien $f := (Y^2+6)(X-1) - Y(X^2+1) \in \mathbb{Z}[X,Y]$ und $g(X,Y) := f(Y,X)$, sowie $\mathcal{X} := \{[x,y] \in \mathbb{C}^2; f(x,y) = 0\}$ und $\mathcal{Y} := \{[x,y] \in \mathbb{C}^2; g(x,y) = 0\}$ die zugehörigen ebenen Kurven. Man zeichne die $\mathbb{R}$-rationalen Punkte $\mathcal{X} \cap \mathbb{R}^2$ und $\mathcal{Y} \cap \mathbb{R}^2$, und berechne $\mathcal{X} \cap \mathcal{Y}$.

Beweis. Siehe [3, Ex.6.41]. $\sharp$

(8.28) Aufgabe: Minimalpolynome.
Es seien $K \subseteq L$ eine algebraische Körpererweiterung, und $0 \neq \alpha \in L$ mit K-Minimalpolynom $f \in K[X]$.
a) Es seien $0 \neq \beta \in L$ mit K-Minimalpolynom $g \in K[X]$, und $h \in K[X]$ das K-Minimalpolynom von $\alpha + \beta \in L$. Man zeige: Es gilt $h \mid \mathrm{res}_Y(f(X-Y), g(Y)) \in K[X]$. Für $a, b \in K$ gebe man eine Verallgemeinerung für $a\alpha + b\beta \in L$ an.
b) Man zeige, daß $\widetilde{f} := f_0^{-1} \cdot X^n \cdot f(X^{-1}) \in K[X]$ das K-Minimalpolynom von $\alpha^{-1} \in L$ ist, wobei $n = \deg(f)$ und $f_0 \in K$ den konstanten Koeffizienten von f bezeichne. Damit zeige man: Sind $0 \neq \beta \in L$ mit K-Minimalpolynom $g \in K[X]$, und $h \in K[X]$ das K-Minimalpolynom von $\alpha\beta \in L$, so gilt $h \mid \mathrm{res}_Y(\widetilde{f}(Y), g(XY)) \in K[X]$. Man gebe eine analoge Formel für $\frac{\alpha}{\beta} \in L$ an.
c) Man zeige: Sind $0 \neq g \in K[X]$ mit $\deg(g) < \deg(f)$ und $h \in K[X]$ das K-Minimalpolynom von $g(\alpha) \in L$, so gilt $h \mid \mathrm{res}_Y(f(Y), X - g(Y)) \in K[X]$. Man bestimme $\deg_X(\mathrm{res}_Y(f(Y), X - g(Y)))$.
d) Man berechne die $\mathbb{Q}$-Minimalpolynome der Elemente $\alpha_1 := \sqrt{2} + \sqrt{3}$ und $\alpha_2 := \sqrt{2} - 2\sqrt{3}$, sowie $\alpha_3 := \sqrt{2} \cdot \sqrt[3]{3}$ und $\alpha_4 := 1 + \sqrt{3}$.

Beweis. Siehe [3, Ch.6.8, Ex.6.35, Ex.6.36]. $\sharp$

(8.29) Aufgabe: Rekonstruktion rationaler Zahlen.
Es seien $r, t \in \mathbb{Z}$ mit $t > 0$ und $\mathrm{ggT}(r,t) = 1$. Ziel ist es, zu zeigen, daß $\frac{r}{t} \in \mathbb{Q}$ aus einer modularen Reduktion $n \bmod m$ zurückgewonnen werden kann, wenn m genügend groß ist.

Dazu seien also $m \in \mathbb{N}$ mit $\mathrm{ggT}(t,m) = 1$ und $n \in \mathbb{N}_0$ mit $n < m$ und $r \equiv nt \bmod m$. Weiter gebe es $k \in \mathbb{N}$ mit $|r| < k$ und $kt \leq m$. Schließlich seien $i \in \{1, \ldots, l+1\}$ im erweiterten Euklidischen Algorithmus für m und n minimal mit $k > r_i$, und $a \in \mathbb{N}$ minimal mit $k > r_{i-1} - ar_i$.

Man zeige: Es gilt $[r,t] = \pm[r_i, t_i]$ oder $[r,t] = \pm[r_{i-1} - ar_i, t_{i-1} - at_i]$.

Beweis. Siehe [3, Ch.5.10]. $\sharp$

(8.30) Aufgabe: Satz von Sturm.
Es sei $0 \neq f \in \mathbb{R}[X]$ mit paarweise verschiedenen Nullstellen in $\mathbb{C}[X]$. Man modifiziere den Euklidischen Algorithmus für $r_0 := f$ und $r_1 := \frac{\partial f}{\partial X}$ durch Verwenden der Polynomdivision $r_{i-1} = q_i r_i - r_{i+1}$, für $i \in \{1, \ldots, l-1\}$. Für $a \in \mathbb{R}$ sei $\nu_a(f) \in \mathbb{N}_0$ die Anzahl der Vorzeichenwechsel in der Folge $[\mathrm{sgn}(r_0(a)), \ldots, \mathrm{sgn}(r_l(a))]$, wobei Einträge 0 ignoriert werden. Man zeige: Für $a < b$ gilt $|\{a < x \leq b; f(x) = 0\}| = \nu_a(f) - \nu_b(f)$.

Hinweis. Gewisse Vorzeichenfolgen können nicht vorkommen.

Beweis. Siehe [8, p.434, Exc.4.6.1.22]. $\sharp$

(8.31) Aufgabe: Charakteristisches Polynom.
Für $M = [m_{ij}] \in \mathbb{C}^{n \times n}$ seien $\mathcal{P}_M := E_n \cdot X - M \in \mathbb{C}[X]^{n \times n}$ die zugehörige charakteristische Matrix und $\chi_M := \det(\mathcal{P}_M) = X^n + \sum_{k=0}^{n-1} c_k X^k \in \mathbb{C}[X]$ das zugehörige charakteristische Polynom.
a) Man zeige: Ist $B > 0$, so daß $|m_{ij}| \leq B$, für alle $i, j \in \{1, \ldots, n\}$, so gilt $|c_{n-k}| \leq \binom{n}{k} \cdot k^{\frac{k}{2}} \cdot B^k$, für alle $k \in \{0, \ldots, n-1\}$.
b) Man zeige: Es gilt $\chi_M = \sum_{k=0}^{n} \left(\det(\mathcal{P}_M(k)) \cdot \prod_{j \in \{0,\ldots,n\} \setminus \{k\}} \frac{X-j}{k-j}\right)$.
c) Man zeige: Ist $\mathcal{A} \in \mathbb{C}[X]^{n \times n}$ die adjungierte Matrix zu $\mathcal{P}_M$, so gilt $\mathrm{Spur}(\mathcal{A}) = \frac{\partial \chi_M}{\partial X} \in \mathbb{C}[X]$, wobei $\frac{\partial \chi_M}{\partial X}$ die formale Ableitung bezeichne.
d) Man gebe einen Algorithmus zur Berechnung des charakteristischen Polynoms quadratischer komplexer Matrizen an, der das Ergebnis aus c) benutzt.

Beweis. **a)** Siehe [2, Prop.2.2.10]. **b)** Siehe [2, Ch.2.2.4].
c) Siehe [2, La.2.2.8]. **d)** Siehe [2, Alg.2.2.7]. $\sharp$

(8.32) Aufgabe: Mignotte-Ungleichung.
Es seien $0 \neq f = \sum_{i=0}^{n} f_i X^i \in \mathbb{C}[X]$ und $g = \sum_{j=0}^{m} g_j X^j \in \mathbb{C}[X]$ ein Teiler von f, wobei $n = \deg(f)$ und $m = \deg(g)$. Man zeige: Für alle $j \in \{0, \ldots, m\}$ gilt $|g_j| \leq \binom{m-1}{j} \cdot \|f\|_2 + \binom{m-1}{j-1} \cdot |f_n|$.

Beweis. Siehe [2, Thm.3.5.1] und [8, Exc.4.6.2.20]. $\sharp$

(8.33) Aufgabe: Monagan-Test.
Für $0 \neq f = \sum_{i=0}^{n} f_i X^i \in \mathbb{C}[X]$, mit $n = \deg(f)$, sei die zugehörige **Cauchy-Schranke** definiert als $B_f := 1 + \max\{|\frac{f_i}{f_n}|; i \in \{0, \ldots, n\}\} \in \mathbb{R}_{>0}$.
a) Man zeige: Ist $\alpha \in \mathbb{C}$ mit $f(\alpha) = 0$, so gilt $|\alpha| < B_f$.
b) Nun sei $f \in \mathbb{Z}[X]$ primitiv mit $f_0 \neq 0$. Man zeige: Gibt es ein $k \in \mathbb{N}$, so daß $f(B_f + k) \in \mathbb{Z}$ oder $f(-B_f - k) \in \mathbb{Z}$ prim ist, so ist f irreduzibel. Wie kann man daraus einen randomisierten Irreduzibilitätstest gewinnen?

Beweis. Siehe [11, p.46]. $\sharp$

(8.34) Aufgabe: Paar-Reduktion.
Es sei $L := \{a, b\}_Z \subseteq \mathbb{R}^n$, für $n \geq 2$, wobei $\{a, b\} \subseteq \mathbb{R}^n$ $\mathbb{R}$-linear unabhängig sei.

```
1. if $\|b\|^2 < \|a\|^2$ then $b \leftrightarrow a$
2. $c \leftarrow b - a \cdot \lceil \frac{\langle b,a \rangle}{\|a\|^2} \rfloor$
3. while $\|c\|^2 < \|a\|^2$ do
       $b \leftarrow a$
       $a \leftarrow c$
       $c \leftarrow b - a \cdot \lceil \frac{\langle b,a \rangle}{\|a\|^2} \rfloor$
4. return $B := [a, b]$
```

Man zeige: Es ist $B \subseteq L$ eine LLL-reduzierte $\mathbb{Z}$-Basis, für jeden Parameter $\frac{3}{4} < \gamma \leq 1$, und $a \in L$ ist ein minimaler Vektor.

Beweis. Siehe [2, Alg.1.3.14] und [15, Exc.3.3.3]. ♯

(8.35) Aufgabe: LLL-Reduktion.
Es seien $L \subseteq \mathbb{R}^n$ ein $\mathbb{Z}$-Gitter mit LLL-reduzierter $\mathbb{Z}$-Basis $B = \{b_1, \ldots, b_n\}$, sowie $\{v_1, \ldots, v_s\} \subseteq L$ eine $\mathbb{Z}$-linear unabhängige Teilmenge. Man zeige: Für $j \in \{1, \ldots, s\}$ gilt $\|b_j\| \leq 2^{\frac{n-1}{2}} \cdot \max\{\|v_1\|, \ldots, \|v_s\|\}$.

Beweis. Siehe [2, Thm.2.6.2.(5)]. ♯

(8.36) Aufgabe: LLL-Reduktion.
Man zeige, daß man die Lovasz-Bedingung $\|b'_k\|^2 \geq (\gamma - \mu^2_{k,k-1} \cdot \|b'_{k-1}\|^2$ der LLL-Reduktion für geeignete Wahlen für $\frac{1}{4} < \gamma \leq 1$ durch die **Siegel-Bedingung** $\|b'_k\|^2 \geq \frac{1}{2} \cdot \|b'_{k-1}\|^2$ ersetzen kann.

Beweis. Siehe [2, Rem.2.6.1.(5)]. ♯

(8.37) Aufgabe: LLL-Algorithmus.
Es sei $Q \in \mathbb{R}^{n \times n}$ eine positiv definite symmetrische Matrix. Man gebe eine Variante des LLL-Algorithmus an, der Q als Eingabe und eine LLL-reduzierte Basis in Form einer als Ausgabe hat. Man gebe eine Laufzeitabschätzung an.

Beweis. Siehe [2, Rem.2.6.1.(2)]. ♯

(8.38) Aufgabe: Quadratfreie Faktorisierung.
Es seien K ein Körper der Charakteristik 0 und $0 \neq \Psi \in K[X]$ normiert mit $\deg(\Psi) = n$. Weiter seien $\Psi_1, \ldots, \Psi_n \in K[X]$ normiert, quadratfrei und paarweise teilerfremd mit $\Psi = \prod_{e=1}^{n} \Psi_e^e$. Man gebe einen Algorithmus an, der mit Ψ als Eingabe die Polynome $\Psi_1, \ldots, \Psi_n$ berechnet.

Beweis. Siehe auch [3, Ch.14.6]. ♯

(8.39) Aufgabe: Quadratfreie Polynome.
Es sei $q \in \mathbb{N}$ eine Primzahlpotenz. Man zeige: Die Wahrscheinlichkeit, daß ein zufällig gewähltes normiertes Polynom in $\mathbb{F}_q[X]$ vom Grad $n \in \mathbb{N}_0$ quadratfrei ist, ist unabhängig von n und beträgt $1 - \frac{1}{q}$.

Hinweis. Für die Anzahl $s_n \in \mathbb{N}_0$ der normierten quadratfreien Polynome vom Grad n zeige man die Rekursionsformel $q^n = \sum_{0 \leq 2k \leq n} q^k \cdot s_{n-2k}$, und gewinne daraus eine geschlossene Formel für s_n.

Beweis. Siehe [3, Exc.14.32]. $\sharp$

(8.40) Aufgabe: Irreduzible Polynome.
Es seien $q \in \mathbb{N}$ eine Primzahlpotenz und $n \in \mathbb{N}_0$ sowie $\mathcal{P}_{q,n} := \{0 \neq \Phi \in \mathbb{F}_q[X]; \Phi \text{ normiert, irreduzibel, } \deg(\Phi) = n\}$. Ziel dieser Aufgabe ist es, eine Formel für die Mächtigkeit $|\mathcal{P}_{q,n}| \in \mathbb{N}_0$ zu finden.
a) Die **Möbius-Funktion** $\mu\colon \mathbb{N} \to \mathbb{Z}$ sei definiert durch $\mu(1) := 1$, und $\mu(n) := 0$ falls $n > 1$ nicht quadratfrei ist, sowie $\mu(n) := (-1)^k$ falls $n > 1$ quadratfrei ist und genau $k \in \mathbb{N}$ paarweise nicht-assoziierte Primteiler hat. Man zeige: Die Funktion μ ist **multiplikativ**, d. h. für alle $m, n \in \mathbb{N}$ teilerfremd gilt $\mu(mn) = \mu(m)\mu(n)$, und es gilt $\sum_{d \mid n} \mu(d) = 0$ für alle $n > 1$.
b) Es seien R ein kommutativer Ring, $f\colon \mathbb{N} \to R$ eine Funktion und $g\colon \mathbb{N} \to R\colon n \mapsto \sum_{d \mid n} f(d)$, für alle $n \in \mathbb{N}$. Man zeige: Es gilt **Möbius-Inversion** $f(n) = \sum_{d \mid n} \mu(\frac{n}{d}) g(d) \in R$, für alle $n \in \mathbb{N}$. Man gebe eine analoge Inversionsformel für die Funktion $h\colon \mathbb{N} \to R\colon n \mapsto \prod_{d \mid n} f(d)$ an.
c) Man zeige: Es gilt $|\mathcal{P}_{q,n}| = \frac{1}{n} \cdot \sum_{d \mid n} \mu(\frac{n}{d}) \cdot q^d$. Man berechne $|\mathcal{P}_{q,n}| \in \mathbb{N}_0$ für $q \leq 9$ und $n \leq 10$.
d) Wie groß ist asymptotisch für $q^n \gg 0$ die Wahrscheinlichkeit, daß ein zufällig gewähltes normiertes Polynom in $\mathbb{F}_q[X]$ vom Grad n irreduzibel ist? Man gebe einen randomisierten Algorithmus an, der bei Eingabe von q und n ein zufälliges irreduzibles normiertes Polynom in $\mathbb{F}_q[X]$ vom Grad n ausgibt.

Beweis. Siehe [3, Ch.14.9, Exc.14.46]. $\sharp$

(8.41) Aufgabe: Kreisteilungspolynome.
Es seien $n \in \mathbb{N}$ und $\zeta_n := e^{\frac{2\pi\sqrt{-1}}{n}} \in \mathbb{C}$, sowie $\Phi_n := \prod_{k \in (\mathbb{Z}/\langle n \rangle)^*} (X - \zeta_n^k) \in \mathbb{C}[X]$ das n-te **Kreisteilungspolynom**. Insbesondere ist also $\deg(\Phi_n) = \varphi(n)$, wobei $\varphi\colon \mathbb{N} \to \mathbb{N}$ die **Eulersche φ-Funktion** sei. Man zeige:
a) Es sind $\{\zeta_n^k \in \mathbb{C}; k \in (\mathbb{Z}/\langle n \rangle)^*\}$ genau die primitiven n-ten Einheitswurzeln in $\mathbb{C}$, und es gelten $X^n - 1 = \prod_{d|n} \Phi_d \in \mathbb{C}[X]$ sowie $\Phi_n = \prod_{d|n} (X^d - 1)^{\mu(\frac{n}{d})} \in \mathbb{C}(X)$, wobei $\mu\colon \mathbb{N} \to \mathbb{Z}$ die Möbius-Funktion aus Aufgabe (8.40) sei. Man folgere daraus, daß $\Phi_n \in \mathbb{Z}[X]$ gilt.
b) Für Primzahlen $p \in \mathbb{N}$ gilt $\Phi_p = \sum_{i=0}^{p-1} X^i \in \mathbb{Z}[X]$; für $n \geq 3$ ungerade gilt $\Phi_{2n}(X) = \Phi_n(-X)$; für Primzahlen $p \in \mathbb{N}$ mit $p \not| n$ gilt $\Phi_{pn}(X) \cdot \Phi_n(X) = \Phi_n(X^p)$; und falls jeder Primteiler von $k \in \mathbb{N}$ auch Primteiler von n ist, so gilt $\Phi_{kn}(X) = \Phi_n(X^k)$.

c) Man gebe einen Algorithmus an, der bei Eingabe von n und seinen Primteilern das Polynom Φ_n als Ausgabe hat, und berechne Φ_n für $n \leq 100$.

Beweis. Siehe [3, Ch.14.10]. $\sharp$

(8.42) Aufgabe: Kreisteilungspolynome.
Es seien $p \in \mathbb{N}$ eine Primzahl, $f \in \mathbb{N}$ und $q := p^f \in \mathbb{N}$, sowie $\mathbb{F}_q$ der endliche Körper mit q Elementen. Weiter sei $n = p^e \cdot m \in \mathbb{N}$, wobei $e \in \mathbb{N}_0$ und $p \not| \, m$. Man zeige: Das n-te Kreisteilungspolynom $\overline{\Phi_n} \in \mathbb{F}_q[X]$ zerfällt in $\frac{\varphi(m)}{d}$ paarweise nicht-assoziierte irreduzible Polynome vom Grad d, wobei $d \in \mathbb{N}$ die Ordnung von $\overline{q} \in (\mathbb{Z}/\langle m \rangle)^*$ sei.

Beweis. Siehe auch [3, La.14.50]. $\sharp$

(8.43) Aufgabe: Faktorisierung in $\mathbb{F}_q[X]$.
Man gebe genaue Abschätzungen für die benötigten Körperoperationen in $\mathbb{F}_q$, in Abhängigkeit vom Grad n des Eingabepolynoms in $\mathbb{F}_q[X]$ und der Ordnung q des Grundkörpers $\mathbb{F}_q$, für die quadratfreie Faktorisierung, die Distinct-Degree-Faktorisierung, sowie die deterministischen und randomisierten Versionen des Cantor-Zassenhaus-Algorithmus und des Berlekamp-Algorithmus an.

Beweis. Siehe [3, Ch.14]. $\sharp$

9 References

[1] A. Aho, J. Hopcroft, J. Ullman: The design and analysis of computer algorithms, second printing, Addison-Wesley Series in Computer Science and Information Processing, 1975.

[2] H. Cohen: A course in computational algebraic number theory, Graduate Texts in Mathematics 138, Springer, 1993.

[3] J. von zur Gathen, J. Gerhard: Modern computer algebra, second edition, Cambridge University Press, 2003.

[4] G. Hardy, E. Wright: An introduction to the theory of numbers, 5. edition, Oxford University Press, 1979.

[5] H. Heuser: Lehrbuch der Analysis, Teil 1, Teubner, 1980.

[6] D. Hilbert: Mathematische Probleme, Vortrag, gehalten auf dem internationalen Mathematiker-Kongreß zu Paris 1900.

[7] D. Knuth: The art of computer programming, vol. 1: fundamental algorithms, 2. printing of the 2. edition, Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley, 1975.

[8] D. Knuth: The art of computer programming, vol. 2: seminumerical algorithms, 2. edition, Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley, 1981.

[9] D. Knuth: The art of computer programming, vol. 3: sorting and searching, Addison-Wesley Series in Computer Science and Information Processing, Addison-Wesley, 1973.

[10] N. Koblitz: Algebraic aspects of cryptography, Algorithms and Computation in Mathematics 3, Springer, 1998.

[11] B. Matzat: Vorlesung 'Computeralgebra', IWR, Universität Heidelberg, private Mitschrift, 1995.

[12] A. Menezes, P. van Oorschot, S. Vanstone: Handbook of applied cryptography, CRC Press Series on Discrete Mathematics and its Applications, 1997.

[13] C. Papadimitriou: Computational complexity, Addison-Wesley, 1995.

[14] M. Pohst: Computational algebraic number theory, DMV Seminar Bd. 21, Birkhäuser, 1993.

[15] M. Pohst, H. Zassenhaus: Algorithmic algebraic number theory, revised reprint of the 1989 original, Encyclopedia of Mathematics and its Applications 30, Cambridge University Press, 1997.

[16] A. Turing: On computable numbers, with an application to the Entscheidungsproblem, Proc. London Math. Soc. Ser. 2, 42, 1936, 230–265.