Zahlentheoretische Grundlagen der Public-Key Kryptographie und deren Behandlung im Mathematikunterricht

Erik Einhaus

Schriftliche Hausarbeit im Fach Mathematik

Referent: Prof. Dr. Michael Hortmann

Korreferent: Prof. Dr. Eberhard Oeljeklaus

Studentisches Mitglied: Lutz Fischer

Vorwort

Die vorliegende Arbeit entstand vor dem Hintergrund, zwei im Schulunterricht wenig beachtete Teilbereiche der Mathematik, die Zahlentheorie und die diskrete Mathematik, anschaulich anhand eines attraktiven Rahmenthemas für den Unterricht in einem Mathematik-Leistungskurs der 12. Jahrgangsstufe aufzubereiten. Gerade bei einem für Schüler auf den ersten Blick eher „theoretisch“ oder „trocken “erscheinendem Thema, wie der Zahlentheorie bietet die Einbettung dieser Thematik in einen praktischen Anwendungszusammenhang - in diesem Fall die Kryptographie - eine geeignete Motivationsgrundlage.

Hauptziel dieser Arbeit ist es, interessierten Mathematiklehrern, welche die Kryptographie im Kontext der diskreten Mathematik im Unterricht behandeln wollen, die notwendigen theoretischen und didaktischen Grundlagen aufzuzeigen, zumal die diskrete Mathematik jüngst in mehreren Bundesländern in die Lehrpläne für die Sekundarstufe II eingegliedert wurde. Die zugehörigen Arbeitsblätter finden sich im Anhang dieser Arbeit.

Die Arbeit gliedert sich in zwei Haupteile, einen fachlichen, als mathematische Grundlage dienenden, und einen didaktisch orientierten Teil.
In ersten Teil werden zunächst einige grundlegende Definitionen, Sätze und Algorithmen der Zahlentheorie, welche für die Public-Key-Kryptographie von Bedeutung sind, aufgeführt. Allen voran der euklidische Algorithmus, mit dessen Hilfe dann die Eindeutigkeit der Zerlegung in Primelemente von Elementen euklidischer Ringe bewiesen wird. Es wird der Begriff der Restklasse eingeführt und gezeigt, dass die Menge aller Restklassen einen Ring und unter besonderer Voraussetzung sogar einen Körper bildet. In diesem Kontext werden der chinesische Restsatz und der kleine Satz von Fermat betrachtet, welcher eine Grundlage für das RSA-Verfahren bildet. Hiernach werden Algorithmen zur Faktorisierung großer Zahlen und zum Finden von diskreten Logarithmen betrachtet, welche für Anfriffe auf Public-Key-Kryptosysteme von Bedeutung sind. Weiterhin werden die Begriffe des Kryptosystems und der perfekten Sicherheit eines Kryptosystems definiert, und gezeigt, unter welchen Voraussetzungen ein Kryptosystem aus stochastischer Sicht als perfekt sicher gilt. Im letzten Abschnitt des ersten Teils wird dann das Public-Key-Chiffre definiert und als Beispiele das RSA- sowie das ElGamal-Verfahren diskutiert. Am Beispiel des RSA-Verfahrens werden einige gängige Angriffe auf Public-Key-Systeme erläutert und Gefahren, die bei falschen Gebrauch des Verfahrens auftreten können, erläutert.

Der zweite Teil dieser Arbeit stellt eine Unterrichtseinheit zum Thema *Zahlentheorie und Kryptographie im Mathematikuntericht dar*. Der Unterricht gliedert sich im Wesentlichen in vier Phasen, in denen ausgehend von der klassischen Kryptographie eine Einführung in die Zahlentheorie motiviert wird, da erst mit Hilfe der Zahlentheorie ein Verständnis der modernen Kryptographie möglich ist. Für den Unterricht ist eine Zeit von etwa 4 Wochen geplant, wobei von einem Leistungskurs der 12. Jahrgangsstufe

mit 5 Wochenstunden ausgegangen wird. Neben dem ausführlich geplanten Unterricht werden zusätzlich viele Anregungen gegeben, verschiedene Themengebiete zu vertiefen, oder mit Hilfe der Kryptographie in neue Gebiete der Mathematik einzuführen.

Inhaltsverzeichnis

1 Zahlentheoretische Grundlagen der Public-Key Kryptographie und deren Anwendungen

1.1 Der euklidische Algorithmus

1.1.1 Teilbarkeit in Integritätsbereichen

Ein Integritätsbereich R ist ein nullteilerfreier, kommutativer Ring mit Einselement. Das für das RSA-Verfahren wichtigste Beispiel eines Integritätsbereiches sind die ganzen Zahlen $\mathbb{Z}$.

Definition 1.1.1 *Seien $x, y \in R$. Man sagt x teilt y und schreibt $x|y$, wenn ein $q \in R$ existiert mit $y = qx$. Ist dies nicht der Fall, so schreibt man $x \nmid y$*

Definition 1.1.2 *Ein Element $u \in R$ heißt Einheit, wenn es ein $v \in R$ gibt, so dass $uv = 1$. Die Menge aller Einheiten in R bezeichnen wir mit R^* Zwei Elemente $x, y \in R$ heißen assoziiert, falls es eine Einheit u in$R$ gibt, mit $x = uy$*

Bemerkungen.

1. Für alle $x \in R$ gilt $x|0$
2. R^* ist eine multiplikative Gruppe
3. Es gilt $\mathbb{Z}^* = \{1, -1\}$

Satz 1.1.3 *Seien $x, y \in R$ zwei verschiedene, von 0 verschiedene Elemente. Dann gilt*

$$x|y \text{ und } y|x \Rightarrow x = uy \text{ mit } u \in R^*$$

Beweis. Es gilt also $x = q_1 y$ und $y = q_2 x$ mit $q_1, q_2 \in R$. Daraus folgt $x = q_1 q_2 x$, also $0 = (q_1 q_2 - 1)x$. Da $x \neq 0$ nach Voraussetzung und R nullteilerfrei, folgt daraus, dass $q_1 q_2 = 1$ gelten muss, also $q_1, q_2 \in R^*$. □

Definition 1.1.4 *Seien $x, y \in R$. Ein Element $d \in R$ heißt größter gemeinsamer Teiler von x und y, wenn gilt*

1. *$d|x$ und $d|y$*
2. *Für jeden gemeinsamen Teiler t von x und y gilt: $t|d$*

Bemerkung. Für $x = y = 0$ ist 0 der eindeutig bestimmte größte gemeinsame Teiler.

Satz 1.1.5 *Besitzen zwei Elemente $x, y \in R$ einen größten gemeinsamen Teiler, so ist dieser bis auf Einheiten eindeutig bestimmt.*

Beweis. Seien d_1 und d_2 zwei größte gemeinsame Teiler von x und y. Dann gilt nach Voraussetzung $d_1|d_2$ und $d_2|d_1$, also $d_1 = ud_2$ mit $u \in R^*$. □

Definition 1.1.6 *Ein Intergritätsbereich R heißt euklidischer Ring, falls eine Abbildung $\gamma : R \to \mathbb{N}$ mit den folgenden Eigenschaften existiert:*

1. *Zu jeden Elementen $x, y \in R$ mit $y \neq 0$ existieren $q, r \in R$ mit $r = 0$ oder $\gamma(r) < \gamma(y)$ mit*
$$x = qy + r$$
2. *$\gamma(xy) \geq \gamma(x)$ für alle $x, y \neq 0$ und $\gamma(xy) > \gamma(x)$, wenn $y \notin R^*$.*

Bemerkungen.

1. γ wird auch *Gradfunktion* auf R genannt. Für $d \in R$ wollen $\gamma(d)$ den *euklidschen Betrag* von d nennen.
2. Auf $\mathbb{Z}$ ist z.B. durch die Betragsfunktion $\gamma(x) = |x|$ eine Gradfunktion definiert.

Damit können wir nun Regeln für die Teilbarkeit zusammenstellen.
Seien $a, b, c, d, x, y \in R$, dann gilt

1. $d|d$
2. $d|a \Rightarrow d|ab$
3. $d|c$ und $c|a \Rightarrow d|a$
4. $d|c$ und $d|b \Rightarrow d|xa + yb$
5. $d|c \Rightarrow c = 0$ oder $\gamma(d) \leq \gamma(c)$
6. $d|c$ und $c|d \Leftrightarrow d = ec$ mit einer Einheit e

Satz 1.1.7 *In einem euklidischen Ring R besitzen je zwei Elemente $x, y \in R$ einen größten gemeinsamen Teiler*

Beweis. Falls $y = 0$ so ist x der nach Satz 1.1.5 eindeutig bestimmte größte gemeinsame Teiler. Sei nun also $y \neq 0$ vorausgesetzt und $\gamma : R \to \mathbb{N}$ eine Gradfunktion auf R.
Wir konstruieren den größten gemeinsamen Teiler durch sukzessives Teilen mit Rest. Im ersten Schritt ergibt sich
$$x = q_1 y + r_2 \text{ mit } r_2 = 0 \text{ oder } \gamma(r_2) < \gamma(y)$$
Ist $r_1 = 0$, so ist y der größte gemeinsame Teiler. Andernfalls ist aber der größte gemeinsame Teiler von x und y gleich dem größten gemeinsamen Teiler von y und r_2, da jeder Teiler von x und y ein Teiler von y und r_2 ist und umgekehrt. Man fährt nun fort mit Dividieren mit Rest und erhält eine Kette von Gleichungen
$$y = q_2 r_2 + r_3, \ \ldots, \ r_{n-2} = q_{n-1} r_{n-1} + r_n, \ r_{n-1} = q_n r_n.$$
Wegen $\gamma(r_k) < \gamma(r_{k+1})$ bricht dieses Verfahren stets nach endlich vielen Schritten ab. Das konstruierte r_n ist dann der größte gemeinsame Teiler, da $r_n|r_k$ für alle $k < n$, also insbesondere $r_n|a$ und $r_n|b$, und für jedes d mit $d|a$ und $d|b$ gilt $d|r_n$. □

Der nach Satz 1.1.5 und Satz 1.1.7 bis auf Einheiten eindeutig existierende größte gemeinsame Teiler zweier Elemente $x, y \in R$ wird im folgenden mit $\mathrm{ggT}(x, y)$ bezeichnet.

1.1.2 Der euklidische Algorithmus

Die Idee des Beweises zu Satz 1.1.7 liefert gleichzeitig einen Algorithmus zur Bestimmung des größten gemeinsamen Teilers von $a, b \in R$ durch sukzessives Teilen mit Rest, den euklidischen Algorithmus. Setze $r_0 := a$ und $r_1 := b$ und berechne

$$\begin{aligned} r_0 &= q_1 r_1 + r_2 \text{ mit } 0 < r_2 < b \\ r_1 &= q_2 r_2 + r_3 \text{ mit } 0 < r_3 < r_2 \\ &\vdots \\ r_{n-2} &= q_{n-1} r_{n-1} + r_n \text{ mit } 0 < r_n < r_{n-1} \\ r_{n-1} &= q_n r_n \end{aligned}$$

In dieser Divisionskette ist n dadurch bestimmt, dass r_n der letzte Divisionsrest ungleich 0 ist. Nach dem Beweis von Satz 1.1.7 ist der letzte von 0 verschiedene Rest r_n der eindeutig bestimmte größte gemeinsame Teiler von a und b.

Satz 1.1.8 *Es seien R ein euklidischer Ring mit der Gradfunktion γ, sowie $a_1, a_2, ..., a_n \in R$ und $A = \{x_1 a_1 + x_2 a_2 + \cdots + x_n a_n \mid x_1, x_2, \ldots, x_n \in R\}$. Dann gibt es ein $d \in R$ mit $A = \{xd \mid x \in R\}$.*

Beweis. Die Menge der euklidischen Beträge $\{\gamma(a) | a \in A\} \subset \mathbb{N}$ besitzt ein kleinstes Element. Sei $d \in A$ ein Element mit minimalem Wert von γ. Dann besitzt $u \in A$ eine Darstellung $u = qd + r$ mit $q, r \in R$ und $\gamma(r) < \gamma(d)$ oder $r = 0$. Da $r = u - qd \in A$ folgt wegen der Minimalität von $\gamma(d)$ $r = 0$ und damit $u = qd$, also $u \in \{xd | x \in R\}$. □

Satz 1.1.9 *In einem euklidischen Ring R sind äquivalent:*

1. $A = \{x_1 a_1 + x_2 a_2 + \cdots + x_n a_n \mid x_1, x_2, \ldots, x_n \in R\} = \{xd \mid x \in R\}$
2. $d = ggT(a_1, a_2, \ldots, a_n)$

Beweis.
„⇒“. Sei also $A = \{xd | x \in R\}$. Da insbesondere die $a_i \in R$, gilt $d | a_i$ für $i = 1, 2, \ldots, n$. Ist andererseits c ein Teiler aller a_i, dann teilt c jedes Element aus $\{xd | x \in R\}$, also inbesondere $c | d$. Damit ist $d = \text{ggT}(a_1, \ldots, a_n)$.
„⇐“. Sei nun $d = \text{ggT}(a_1, a_2, \ldots, a_n)$. Nach Satz 1.1.8 gibt es ein $\tilde{d} \in R$, so dass $A = \{x\tilde{d} | x \in R\}$. Dann gilt $d | \tilde{d}$ und nach der Hinrichtung auch $\tilde{d} | d$. Also gilt $d = e\tilde{d}$ mit einer Einheit e und damit $\{x\tilde{d} | x \in R\} = \{xd | x \in R\}$ □

Folgerung. Seien $x_1, \ldots, x_n \in R$ und d der größte gemeinsame Teiler der x_i. Dann gibt es Elemente $a_1, \ldots, a_n \in R$ mit $d = a_1 x_1 + \cdots + a_n x_n$. Insbesondere ist d die vom euklidischen Betrag kleinste mögliche Linearkombination.

Für zwei Elemente $a, b \in R$ erhält man die Darstellung des größten gemeinsamen Teilers leicht mit Hilfe des euklidischen Algorithmus:

$$\begin{aligned} r_n &= r_{n-2} + (-q_{n-1}) r_{n-1} \\ &= r_{n-2} + (-q_{n-1})(r_{n-3} + (-q_{n-2}) r_{n-2}) \\ &= r_{n-4} + (-q_{n-3}) r_{n-3} + (-q_{n-1})(r_{n-3} + (-q_{n-2})(r_{n-4} + (-q_{n-3}) r_{n-3})) \\ &\vdots \end{aligned}$$

Dieses Verfahren kann allerdings bei großen Zahlen sehr unübersichtlich werden. Eine bessere Möglichkeit bietet der folgende Algorithmus nach Berlekamp.

Satz 1.1.10 *Seien $a, b \in R$ und r_i, q_i wie beim euklidischen Algorithmus, gelte also $r_{i+1} = r_{i-1} - q_i r_i$, sowie $r_0 = a, r_1 = b$. Seien die Folgen $(s_k)_{k\in\mathbb{N}}$ und $(t_k)_{k\in\mathbb{N}}$ definiert durch*

$$\begin{aligned} s_0 = 1,\ \ s_1 = 0,\ \ s_{k+1} = s_{k-1} - q_k s_k \ \ \textit{für}\ \ k \geq 2, \\ t_0 = 0,\ \ t_1 = 1,\ \ t_{k+1} = t_{k-1} - q_k t_k \ \ \textit{für}\ \ k \geq 2, \end{aligned}$$

dann gilt $r_k = s_k a + t_k b$.

Beweis. Induktion nach k:
Induktionsanfang: $r_0 = 1 \cdot a + 0 \cdot b = a$, $r_1 = 0 \cdot a + 1 \cdot b = b$
Induktionsschritt:

$$\begin{aligned} r_{k+1} &= r_{k-1} - q_k r_k = s_{k-1}a + t_{k-1}b - q_k(s_k a + t_k b) \\ &= (\underbrace{s_{k-1} - q_k s_k}_{s_{k+1}})a + (\underbrace{t_{k-1} - q_k t_k}_{t_{k+1}})b \end{aligned}$$

□

Folgerung. Ist r_n der letzte Divisionsrest größer 0, dann gilt

$$\mathrm{ggT}(a, b) = r_n = s_n a + t_n b$$

Bemerkung. Eine übersichtliche Möglichkeit, den erweiterten euklidischen Algorithmus per Hand anzuwenden, bietet die Darstellung in einer Tabelle der Form

k	0	1	2	...	n
r_k	a	b	r_2	...	r_n
q_k		q_1	q_2	...	q_n
x_k	1	0	x_2	...	x_n
y_k	0	1	y_3	...	y_n

Der erweiterte euklidische Algorithmus liefert weiterhin eine schöne Möglichkeit, lineare diophantische Gleichungen zu lösen.

Satz 1.1.11 *Seien $a, b, c \in \mathbb{Z}$. Die diophantische Gleichung*

$$ax + by = c$$

besitzt genau dann eine ganzzahlige Lösung, wenn gilt $ggT(a, b)|c$.

Beweis. Sei $d = \mathrm{ggT}(a, b)$ und $A = \{ax + by | x, y \in Z\}$. Nehmen wir zunächst an, $ax+by = c$ besäße eine Lösung, aber es gelte $d \not| c$. Da $\mathbb{Z}$ euklidisch gilt $A = \{xd | x \in \mathbb{Z}\}$. Also gibt es ein $\tilde{x} \in \mathbb{Z}$, mit $ax + by = c = \tilde{x}d$, also folgt $d|c$ ↯.

Gelte nun $d|c$. Dann gibt es ein $q \in \mathbb{Z}$ mit $c = qd$. Da $\mathbb{Z}$ euklidisch gibt es weiterhin $\tilde{x}, \tilde{y} \in \mathbb{Z}$ mit

$$\tilde{x}a + \tilde{y}b = d.$$

Multiplikation mit q ergibt $(\tilde{x}q)a + (\tilde{y}q)b = qd = c$ □

Sei nun (x_0, y_0) eine spezielle Lösung der diophantischen Gleichung $ax+by = c$. Dann ist, wie man leicht nachrechnet, die allgemeine Lösung gegeben durch

$$(x, y) = (x_0 + x_h, y_0 + y_h),$$

wobei (x_h, y_h) die Lösung der homogenen diophantischen Gleichung $ax+by = 0$ ist. Weiterhin besitzt jede Lösung der diophantischen Gleichung diese Form, denn sei (x_1, y_1) eine weitere spezielle Lösung, so gilt $a(x_1 - x_0) + b(x_1 - x_0) = 0$, es ist $(x_1 - x_0, y_1 - y_0)$ also eine Lösung der homogenen diphantischen Gleichung und man erhält

$$(x_1, y_1) = (x_0, y_0) + (x_1 - x_0, y_1 - y_0).$$

Sei nun $d = \text{ggT}(a, b)$, dann ist, wie man sofort sieht, $\{(x, y) | x = t\frac{b}{d}, y = -t\frac{a}{d}, t \in \mathbb{Z}\}$ die Lösung der homogenen diophantischen Gleichung $ax + by = 0$. Dass dieses auch alle Lösungen sind, wird im nächsten Abschnitt gezeigt.

1.2 Primfaktor-Zerlegung

Definition 1.2.1 *Sei R ein Integritätsbereich und R^* die Gruppe seiner Einheiten. Ein Element $r \in R \setminus (R^* \cup \{0\})$ heißt irreduzibel, wenn es keine Zerlegung $r = xy$ mit $x, y \in R \setminus R^*$ gibt.*
Ein Element $p \in R \setminus (R^* \cup \{0\})$ heißt prim, wenn für alle $a, b \in R \setminus \{0\}$ gilt

$$p|ab \Rightarrow p|a \vee p|b$$

Satz 1.2.2 *Sei R ein Integritätsbereich. Dann ist jedes Primelement $p \in R \setminus (R^* \cap \{0\})$ irreduzibel.*

Beweis. Angenommen p hätte eine Zerlegung $p = xy$ mit $x, y \in R \setminus R^*$. Da p prim und $p|xy$ folgt $p|x$ oder $p|y$. Gelte oBdA $p|x$. Da aber auch gilt $x|p$, folgt, dass p und x assoziiert sind. Dann folgt aber $y \in R^*$. ↯ □

Satz 1.2.3 *Sei R ein euklidischer Ring. Dann ist jedes irreduzible Element $r \in R \setminus (R^* \cap \{0\})$ prim.*

Beweis. Seien $x, y \in R \setminus \{0\}$ und $a \in R \setminus (R^* \cap \{0\})$, a irreduzibel mit $a|xy$ aber $a \nmid x$. Dann ist zu zeigen $a|y$. Da R ein euklidischer Ring, existiert $d = \text{ggT}(a, x) \in R^*$, da a irreduzibel. Also gibt es $u, v \in R$ mit $ua + vx = 1$. Also ist $y = uay + vxy$ und da $a|xy$ nach Voraussetzung folgt $a|y$. □

Bemerkung. Im euklidischen Ring $\mathbb{Z}$ der ganzen Zahlen fallen die Begriffe prim und irreduzibel zusammen. Die Primelemente in $\mathbb{Z}$ sind also genau die (Prim)Zahlen $p \in \pm\mathbb{P} := \{\pm 2, \pm 3, \pm 5, \pm 7, \pm 11, ...\}$, welche irreduzibel in $\mathbb{Z}$ sind.

Satz 1.2.4 *Sei R ein euklidischer Ring mit Gradfunktion γ. Dann besitzt jedes Element $r \in R \setminus (R^* \cap \{0\})$ eine bis auf die Reihenfolge eindeutige Darstellung als Produkt endlich vieler Primelemente.*

Beweis. Wie wir oben gezeigt haben, sind Primelemente in einem euklidischen Ring dasselbe wie irreduzible Elemente. Falls $r \in R \setminus (R^* \cap \{0\})$ schon irreduzibel ist, sind wir also fertig. Andernfalls besitzt es eine Zerlegung $r = r_1 r_2$ mit $r_i \in R \setminus (R^* \cap \{0\})$. Sind r_1, r_2 irreduzibel, sind wir fertig. Andernfalls besitzen diese Zerlegungen $r_i = r_{i_1} r_{i_2}$, welche selbst entweder irreduzibel sind oder eine Zerlegung besitzen. Man erhält für r also eine Teilerkette $(r_1, r_2, r_{1_1}, r_{1_2}, r_{2_1}, r_{2_2}, \dots)$. Da aus $a_2 | a_1$ folgt $\gamma(a_2) < \gamma(a_1)$ für $a_1, a_2 \in R \setminus (R^* \cap \{0\})$, bricht diese Teilerkette nach endlich vielen Schritten ab, und man erhält eine Zerlegung von r in Primelemente.

Nehmen wir nun an, $r \in R \setminus (R^* \cap \{0\})$ besitzte zwei Zerlegungen

$$r = p_1 p_2 p_3 \dots p_r = q_1 q_2 q_3 \dots q_s$$

in irreduzible Elemente. Da die p_i prim sind und $p_i | a$ gilt $p_i | q_j$ für $1 \leq i, j \leq s$. Da die q_j ebenfalls irreduzibel sind, gilt $\mathrm{ggT}(p_i, q_j) = 1$ für alle $1 \leq i, j \leq s$. Bei geeigneter Nummerierung gelte $p_1 | q_1$, dann sind p_1 und q_1 assoziiert und es gilt $q_1 = e p_1$ mit einer Einheit e. Sei o.B.d.A $r \leq s$. Dann können wir diese Überlegung noch $(r-1)$-mal durchführen und erhalten $p_i = e_k q_i$ mit Einheit e_k. Einsetzten in die obige Gleichung ergibt

$$1 = e q_{r+1} \dots q_s$$

mit einer Einheit e. Wäre nun $s > r$, dann teilt q_s die 1, ist also eine Einheit. ↯
Es gilt als $r = s$. □

Folgerung. Betrachten wir die homogene lineare diophantische Gleichung $ax + by = 0$ und sei $d = \mathrm{ggT}(a, b)$. Dann folgt wegen der eindeutigen Primfaktorzerlegung

$$\begin{aligned} d(\tilde{a}x + \tilde{b}y) = 0 \quad &\Rightarrow \quad -\tilde{a}x = \tilde{b}y \text{ mit } \mathrm{ggT}(\tilde{a}, \tilde{b}) = 1 \\ &\Rightarrow \quad \tilde{b} | x \Rightarrow x = t\tilde{b} \text{ mit } t \in \mathbb{Z} \\ &\Rightarrow \quad y = -t\tilde{a} \end{aligned}$$

und damit haben wir gezeigt, dass jede Lösung der homogenen diophantischen Gleichung von der Form $\{(x, y) | x = t\frac{b}{d}, y = -t\frac{a}{d}, t \in \mathbb{Z}\}$ ist. □

Satz 1.2.5 *In $\mathbb{Z}$ gibt es unendlich viele Primzahlen.*

Beweis(Euklid). Wir nehmen an, es gäbe nur endlich viele Primzahlen $p_1, p_2, \dots, p_k$. Dann ist die Zahl $n := p_1 \cdot p_2 \cdot \dots \cdot p_k + 1 \neq \pm 1$ keine Einheit. Nach Satz 10 müsste n also durch mindestens eine Primzahl $p_i,\ i \in \{1, \dots, k\}$ teilbar sein. Dies ist aber nicht möglich, weil sonst p_i selbst eine Einheit wäre. □

Theoretisch ist mit den obigen Sätzen also eine eindeutige Zerlegung der ganzen Zahlen in Primzahlen gesichert. In der Praxis ist diese allerdings nicht immer leicht zu bestimmen, bei sehr großen Zahlen ist dieses sogar praktisch unmöglich.

Im folgenden werden nun zwei einfache Möglichkeiten gezeigt, wie man entscheiden kann, ob eine Zahl eine Primzahl ist, bzw. eine Zahl, welche nicht prim ist, als Produkt von Primzahlen zu schreiben.

Satz 1.2.6 *Sei $n \in \mathbb{N}, p \in \mathbb{P}$.*

1. *Besitzt n keinen Primteiler $p \leq \sqrt{n}$, dann ist n eine Primzahl.*
2. *Sei $n = p\tilde{n}$ eine Zerlegung von n und p der kleinste Primteiler von n und $\sqrt[3]{n} < p$. Dann ist $\tilde{n}$ eine Primzahl.*

Beweis. (1) Nehmen wir an, n sei nicht prim. Dann besitzt n eine Zerlegung $n = xy$ mit $1 < x \leq y < n$. Dann ist $x^2 \leq xy \leq n$ und damit $x \leq \sqrt{n}$. Also besitzt n einen Teiler $x \leq \sqrt{n}$ und damit insbesondere einen Primteiler $p \leq \sqrt{n}$.
(2) Wäre $\tilde{n}$ nicht prim, dann gäbe es eine Zerlegung $\tilde{n} = xy$ mit $1 < p < x \leq y < \tilde{n}$. Dann folgt $\tilde{n} > p^2$ und damit $n > p^3$. $\lightning$, da $n < p^3$ nach Voraussetztung. □

Eine weitere Möglichkeit, Faktoren einer ganzen Zahl n zu finden, basiert auf der folgenden Überlegung:
Sei n eine postitive ungerade ganze Zahl und $n = ab$ eine Zerlegung von n mit $a \geq b > 0$. Seien weiterhin t, s definiert durch $t = \frac{a+b}{2}$ und $s = \frac{a-b}{2}$, dann gilt $n = t^2 - s^2$.
Dieses ist sofort klar, da $n = ab = (t + s)(t - s) = t^2 - s^2$.

Diese Tatsache wird bei der *Fermat-Faktoriesierung* ausgenutzt, indem man ganze Zahlen s, t sucht mit der Eigenschaft $t^2 - n = s^2$. Dieses Verfahren ist besonders dann geeignet, einen Faktor von $n = ab$ zu finden, wenn die Faktoren a und b eng zusammen liegen. Denn dann ist $s = \frac{a-b}{2}$ klein und damit t nur etwas größer als $\sqrt{n}$. Man beginnt also mit $t = [\sqrt{n}] + 1$ und erhöht t sukzessive um 1, bis man ein Paar (t, s) gefunden hat, welches der Bedingung $t^2 - n = s^2$ genügt.
Wir werden diese Methode später mit Hilfe von Faktor-Basen verallgemeinern.

1.3 Der Restklassenring $\mathbb{Z}/n\mathbb{Z}$

Definition 1.3.1 *Seien $n \in \mathbb{N}$, $x, y \in \mathbb{Z}$. Dann heißt $x \equiv y \mod n$, geprochen „x kongruent y modulo n“, wenn gilt $n|x - y$.*

Notation. Gilt $x \equiv y \mod n$ und $x \equiv y \mod m$, so schreiben wir dafür auch

$$x \equiv y \mod (n, m).$$

Wie man leicht nachrechnet gilt der folgende Hilfssatz:

Lemma 1.3.2 *Seien $m, n \in \mathbb{N}$ und $x, y \in \mathbb{Z}$. Dann gilt*

$$x \equiv y \mod (m, n) \Leftrightarrow x \equiv y \mod kgV(m, n),$$

wobei mit $kgV(m, n)$ das kleinste gemeinsame Vielfache von m und n bezeichnet ist. □

Eine anschauliche Charakterisierung der Kongruenz folgt aus dem folgenden Satz:

Satz 1.3.3 *$x \equiv y \mod n \Leftrightarrow x$ und y lassen beim Teilen durch n denselben Rest.*

Beweis.
„⇒“ : Nehmen wir an, y läßt bei Division mit Rest durch n den Rest r, es gibt also ein $\tilde{k} \in \mathbb{Z}$ mit $y = \tilde{k}n + r$ und $r < n$. Da $n|(x - y)$ gibt es ein $k \in \mathbb{Z}$ so dass $x = y + kn$.

Dann folgt $x = (k + \tilde{k})n + r$.
„$\Leftarrow$“ : Es gibt also $u, v, r \in \mathbb{Z}$ mit $x = un + r$ und $b = vn + r$. Subtraktion ergibt $x - y = (u - v)n$ also $n|(x - y)$ □

Bemerkungen. Wie man leicht nachrechnet, ist die Kongruenz $x \equiv y \mod n$ eine Äquivalenzrelation, welche eine Klasseneinteilung der Menge der ganzen Zahlen in die so genannten *Restklassen* modulo n induziert. Die Menge der Restklassen modulo n bezeichnet man auch mit $\mathbb{Z}/n\mathbb{Z} = \{[0]_n, [1]_n, \ldots, [n-1]_n\}$, wobei
$[x]_n = \{y \in \mathbb{Z} | b \equiv a \mod n\}$.

Satz 1.3.4 *Auf der Menge $\mathbb{Z}/n\mathbb{Z}$ sind die Verknüpfungen $(+, \cdot)$ wie folgt definiert:*

$$[x]_n + [y]_n := [x + y]_n \text{ und } [x]_n \cdot [y]_n := [x \cdot y]_n$$

Dies Verknüpfungen sind wohldefiniert.

Beweis. Seien $[\tilde{x}]_n = [x]_n$ und $[\tilde{y}]_n = [y]_n$, also $\tilde{x} \equiv x \mod n$ bzw. $\tilde{y} \equiv y \mod n$. Also gilt zunächst

$$m|(\tilde{x} - x) + (\tilde{y} - y) = (\tilde{x} + \tilde{y}) - (x + y)$$

und daraus folgt $\tilde{x} + \tilde{y} \equiv x + y \mod n$, also $[\tilde{x} + \tilde{y}]_n = [x + y]_n$ und weiterhin

$$m|(\tilde{x} - x) \cdot \tilde{y} + (\tilde{y} - y) \cdot \tilde{y} = (\tilde{x} \cdot \tilde{y}) \cdot (x \cdot y).$$

Daraus folgt nun wie gewünscht $\tilde{x} \cdot \tilde{y} \equiv x \cdot y \mod n$, also $[\tilde{x} \cdot \tilde{y}]_n = [x \cdot y]_n$. □

Wie man leicht nachrechnen kann, vererben sich die Assoziativität,Kommutativität und Distributivität der ganzen Zahlen in die Menge der Restklassen. Zusammenfassend erhalten wir den folgenden Satz:

Satz 1.3.5 *Die Menge $(\mathbb{Z}/n\mathbb{Z}, +, \cdot)$ der Restklassen modulo n, versehen mit den oben definierten Verknüpfungen, bildet einen kommutativen Ring mit Einselement $[1]_n$.*
Das neutrale Element bezüglich der Addition ist $[0]_n$, das additive Inverse zu $[a]_n$ ist $-[a]_n := [-a]_n$. □

Satz 1.3.6 *Seien $a, b \in Z$ und $n \in \mathbb{N}$. Dann ist die lineare Kongruenz $ax \equiv b \mod n$ ist genau dann lösbar, wenn gilt $ggT(a, n)|b$. Modulo n gibt es dann genau $ggT(a, n)$ Lösungen.*

Beweis. Die Kongruenz $ax \equiv b \mod n$ ist äquivalent zu der diophantischen Gleichung $ax + kn = b$ mit $k \in \mathbb{Z}$ und diese ist genau dann lösbar, wenn gilt $d = \text{ggT}(a, n)|b$. Dann ist mit x_0 auch $x_t = x_0 + t \cdot \frac{n}{d}$, $t \in Z$ eine Lösung der lineraren Kongruenz. Wie man leicht nachrechnet, gilt dann $x_{d+k} \equiv x_k \mod n$, $k \in Z$. Also sind $x_0, x_1, \ldots, x_{d-1}$ d verschiedene Lösungen modulo n. □

Die Gruppe $(\mathbb{Z}/n\mathbb{Z})^*$ und der chinesische Restsatz

Satz 1.3.7 *Seien $x, y \in \mathbb{Z}$ und $n \in \mathbb{N}$ mit $x \equiv y \mod n$. Dann gilt*

$$ggT(x, n) = ggT(y, n)$$

Insbesondere folgt dann mit $ggT(x, n) = 1$ für alle $y \equiv x \mod n$ $ggT(y, n) = 1$.

Beweis. Seien $\mathrm{ggT}(x,n) = d = ux + vn$ und $\mathrm{ggT}(y,n) = \tilde{d} = \tilde{u}x + \tilde{v}n$ mit ihren Darstellungen als Linearkombinationen. Nach Voraussetzung gibt es ein $k \in \mathbb{Z}$ mit $x = y + kn$. Daraus folgt

$$d = uy + n(v - uk) \;\Rightarrow\; d - \tilde{d} = y(u - \tilde{u}) + n(v - uk - \tilde{v}) \;\Rightarrow\; \tilde{d}|d$$

Genauso läßt sich zeigen $d|\tilde{d}$, und daraus folgt, das $d = \tilde{d}$ ist. □

Definition 1.3.8 *Seien $x \in \mathbb{Z}$ und $n \in \mathbb{N}$ und gelte $ggT(x,n) = 1$. Dann heißt die Restklasse $[x]_n \in \mathbb{Z}/n\mathbb{Z}$ prime Restklasse modulo n. Die Menge aller primen Restklassen modulo n bezeichnen wir mit $(\mathbb{Z}/n\mathbb{Z})^* \subset \mathbb{Z}/nZ$.*

Satz 1.3.9 *Die Restklasse $[x]_n$ ist invertierbar $\Leftrightarrow$ $ggT(x,n) = 1$.*

Beweis.
„$\Rightarrow$“. Sei also $[x]_n$ invertierbar. Dann gibt es eine Restklasse $[y]_n$ mit $[x]_n[y]_n = [1]_n$, es gilt also $xy \equiv 1 \mod n \;\Leftrightarrow\; xy + kn = 1$, wobei $k \in Z$. Daraus folgt $\mathrm{ggT}(x,n) = 1$.
„$\Leftarrow$“. Sei nun $\mathrm{ggT}(x,n) = 1$. Dann gibt es $k, y \in \mathbb{Z}$ mit $xy + kn = 1 \Leftrightarrow xy = 1 + (-k)n$. Damit gilt $xy \equiv 1 \mod n$, also ist $[y]_n$ das Inverse zu $[x]_n$. □

Bemerkung und Folgerungen.

1. Die Menge $(\mathbb{Z}/n\mathbb{Z})^*$ ist eine Gruppe bezüglich der Multiplikation.
2. Die Menge $\mathbb{Z}/p\mathbb{Z}$ mit p prim ist ein Körper.
3. Der erweiterte euklidische Algorithmus ist ein effizientes Verfahren zur Berechnung des Inversen eines Elementes aus $(\mathbb{Z}/n\mathbb{Z})^*$.

Satz 1.3.10 (Chinesischer Restsatz) *Seien $n_1, n_2, \ldots, n_k \in \mathbb{N}$, paarweise teilerfremd, und $c_1, c_2, \ldots, c_k \in \mathbb{Z}$. Dann existiert genau eine Restklasse $[x]_{n=n_1 \cdot \cdots \cdot n_k}$ mit $x \equiv c_i \mod n_i$, $i = 1, \ldots, k$.*

Beweis. Es sei $N_i := \frac{n}{n_i}$. Da $\mathrm{ggT}(N_i, n_i) = 1$, gibt es ein M_i mit $N_i M_i \equiv 1 \mod n_i$. Dann gilt für die ganze Zahl

$$x = \sum_{i=1}^{k} c_i N_i M_i \equiv c_i \mod n_i$$

da $N_j \equiv 0 \mod n_i$ für $i \neq j$.
Ist $y \in \mathbb{Z}$ eine weitere Lösung mit $y \equiv c_i \mod n_i$, $i = 1, \ldots, k$, dann ist $x \equiv y \mod n_i$, $i = 1, \ldots, k$ und damit $x \equiv y \mod n$. □

Folgerungen. Seien $n_1, n_2, \ldots, n_k$ paarweise teilerfremde natürliche Zahlen.

1. Dann ist die Abbildung

$$\begin{aligned} \varphi : \mathbb{Z}/(n_1 n_2 \cdot \cdots \cdot n_k)\mathbb{Z} &\to \mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z} \\ [x]_n &\mapsto ([x]_{n_1}, [x]_{n_2}, \ldots, [x]_{n_k}) \end{aligned}$$

ein Ring-Isomorphismus, wobei unter $\mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z}$ das direkte Produkt der Ringe $\mathbb{Z}/n_1\mathbb{Z}, \ldots, \mathbb{Z}/n_k\mathbb{Z}$ mit komponentenweiser Addition und Multiplikation zu verstehen ist.

2. Dann ist die Abbildung

$$\begin{aligned} \psi : (\mathbb{Z}/(n_1 n_2 \cdot \cdots \cdot n_k)\mathbb{Z})^* &\to (\mathbb{Z}/n_1\mathbb{Z})^* \times \cdots \times (\mathbb{Z}/n_k\mathbb{Z})^* \\ [x]_n &\mapsto ([x]_{n_1}, [x]_{n_2}, \ldots, [x]_{n_k}) \end{aligned}$$

ein Gruppen-Isomorphismus.

Beweis. (1). Da $\#(\mathbb{Z}/(n_1 n_2 \cdot \cdots \cdot n_k)\mathbb{Z}) = \#(\mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z}) = n$ reicht es die Surjektivität von φ zu zeigen, und das haben wir oben im Beweis getan, da wir für jedes Tupel $(x_1, x_2, \ldots, x_n) \in \mathbb{Z}/n_1\mathbb{Z} \times \cdots \times \mathbb{Z}/n_k\mathbb{Z}$ ein Urbild $x \in \mathbb{Z}/(n_1 n_2 \cdot \cdots \cdot n_k)\mathbb{Z}$ konstruiert haben.
(2). Wir definieren $\psi := \varphi|_{(\mathbb{Z}/n\mathbb{Z})^*}$. Dann erhalten wir im Bildψ genau die Elemente der multiplikativen Untergruppen und daraus folgt die Behauptung. □

Die Eulersche φ-Funktion und der Satz von Fermat

Definition 1.3.11 *Sei $n \in \mathbb{N}$. Dann ist $\varphi(n) := \#\{\ t \mid 0 < t < n,\ ggT(n,t) = 1\}$, die Anzahl der zu n teilerfremden Zahlen $0 < t < n$. $\varphi(n)$ wird auch die Eulersche φ-Funktion genannt.*

Bemerkungen. Wie man leicht nachrechnet gilt

1. $\#(\mathbb{Z}/n\mathbb{Z})^* = \varphi(n)$
2. Für p, q prim gilt $\varphi(p) = p - 1$ und $\varphi(pq) = (p-1)(q-1)$.

Satz 1.3.12 (Euler) *Sei $n \in \mathbb{N}$. Dann gilt für alle $a \in (\mathbb{Z}/n\mathbb{Z})^*$*

$$a^{\varphi(n)} \equiv 1 \mod n$$

Beweis. Da mit $\mathrm{ggT}(a,n) = 1 = \mathrm{ggT}(x,n)$ auch $\mathrm{ggT}(ax,n) = 1$ ist, können wir die Abbildung

$$\begin{aligned} \vartheta : (\mathbb{Z}/n\mathbb{Z})^* &\to (\mathbb{Z}/n\mathbb{Z})^* \\ x &\mapsto ax \mod n \end{aligned}$$

mit $a \in (\mathbb{Z}/n\mathbb{Z})^*$ betrachten. Diese ist injektiv, da aus $ax_1 \equiv ax_2 \mod n$ folgt $x_1 \equiv x_2 \mod n$. Seien nun die Elemente von $(\mathbb{Z}/n\mathbb{Z})^*$ dargestellt durch $b_1, b_2, \ldots, b_{\varphi(n)}$. Dann folgt

$$\vartheta(b_1)\vartheta(b_2) \cdot \cdots \cdot \vartheta(b_{\varphi(n)}) \equiv a^{\varphi(n)} b_1 b_2 \cdot \cdots \cdot b_{\varphi(n)} \mod n.$$

Andererseits ist wegen der Injektivität von ϑ

$$\vartheta(b_1)\vartheta(b_2) \cdot \cdots \cdot \vartheta(b_{\varphi(n)}) \equiv b_1 b_2 \cdot \cdots \cdot b_{\varphi(n)} \mod n.$$

Insgesamt erhalten wir also

$$b_1 b_2 \cdot \cdots \cdot b_{\varphi(n)} \equiv a^{\varphi(n)} b_1 b_2 \cdot \cdots \cdot b_{\varphi(n)} \mod n.$$

und daraus folgt die Behauptung. □

Satz 1.3.13 (Fermat) *Sei p prim und $a \in \mathbb{Z}$. Dann gilt $a^p \equiv a \mod p$. Ist zudem $ggT(a,p) = 1$ dann gilt $a^{p-1} \equiv 1 \mod p$.*

Beweis. Dieser Satz ist eine einfache Folgerung aus dem Satz von Euler, da $\varphi(p) = p-1$. Ein elementarer Beweis mit vollständiger Induktion findet sich im zweiten Teil dieser Arbeit. □

Folgerungen.

1. $x^{-1} \equiv x^{p-2} \mod p$ für $x \not\equiv 0 \mod p$
2. Sei p prim und $a, b \in \mathbb{Z}$ mit $a \equiv b \mod (p-1)$. Dann gilt $x^a \equiv x^b \mod p$.
3. Seien p, q prim, $n = pq$ und $a, b \in \mathbb{Z}_{>0}$ mit $a \equiv b \mod (p-1, q-1)$. Dann gilt $x^a \equiv x^b \mod n$.

Beweis.
1. Folgt direkt aus dem Satz von Fermat durch Multiplikation mit x^{p-2}.
2. Ist $x \equiv 0 \mod p$, dann gilt trivialerweise $0^a \equiv 0^b \mod p$.
Wir können also $x \not\equiv 0 \mod p$ vorraussetzten. Gelte also $a \equiv b \mod (p-1)$, dann gibt es ein $k \in \mathbb{Z}$ mit $a = b + k(p-1)$ und es folgt

$$x^a = x^{b+k(p-1)} = x^b \cdot x^{(p-1)^k} \equiv x^b \mod p$$

3. Für $x \equiv 0 \mod n$ gilt die Behauptung trivialerweise, also können wir im folgenden $x \not\equiv 0 \mod n$ vorraussetzten. Nach dem Chinesischen Restsatz gibt es einen natürlichen Isomophismus

$$\varphi : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/q\mathbb{Z}$$

und damit

$$x^a \mod n \overset{\varphi}{\longmapsto} (x^a \mod p,\ x^a \mod q)$$

Da $a \equiv b \mod (p-1, q-1)$ ist nach 1. $x^a \equiv x^b \mod p$ bzw. $x^a \equiv x^b \mod q$. Dann folgt

$$(x^a \equiv x^b \mod p,\ x^a \equiv x^b \mod q) \overset{\varphi^{-1}}{\longmapsto} x^b \mod n$$

und damit $x^a \equiv x^b \mod n$. □

Definition 1.3.14 *Sei $n \in \mathbb{N}$. Dann heißt für jedes $a \in (\mathbb{Z}/n\mathbb{Z})^*$ die kleinste natürliche Zahl k mit $a^k \equiv 1 \mod n$ Ordung von a modulo n. Man schreibt dann $k = ord_n a$.*

Satz 1.3.15 *Sei $n \in \mathbb{N}$. Dann gilt für alle $a \in (\mathbb{Z}/n\mathbb{Z})^*$ $ord_n a \mid \varphi(n)$.*

Beweis. Sei $k := \mathrm{ord}_n a$. Da $\mathbb{Z}$ euklidisch besitzt $\varphi(n)$ eine Darstellung $\varphi(n) = qk + r$ mit $q \in \mathbb{N}$ und $0 \leq r < k$. Dann gilt

$$a^{\varphi(n)} \equiv a^{qk+r} \equiv (a^k)^q a^r \equiv a^r \mod n$$

Es folgt also $a^r \equiv 1 \mod n$. Da aber $r < k$ folgt $r = 0$ und damit die Behauptung. □

In der Kryptographie müssen häufig Potenzen der Form $x^d \mod n$ mit sehr großen Werten für d und n berechnet werden. Eine geschicktere Möglichkeit, als diese Potenzen durch wiederholte Multiplikation zu bestimmen, basiert auf wiederholtem Quadrieren. Zunächst stellen wir die Potenz $d = \sum_{i=0}^{k-1} a_1 2^i$, $a_i \in \{0,1\}$ im Binärsystem dar. Dann gilt

$$x^d \equiv x^{\sum a_i 2^i} \equiv \prod_{i=0}^{k-1} (x^{2^i})^{a_i} \mod n.$$

Es reicht also, zunächst durch sukzessives Quadrieren, die Werte x^{2^i} für alle i mit $a_i \neq 0$, und danach deren Produkt zu bestimmen. Man kann zeigen [Kob94], dass mit diesem Algorithmus das Potenzieren modulo n in polynomialer Zeit möglich ist.

1.3.1 Primitivwurzeln und diskreter Logarithmus

Die Primitivwurzeln in $(\mathbb{Z}/p\mathbb{Z})^*$

Als nächstes wollen wir die Existenz von erzeugenden Elementen in $(\mathbb{Z}/p\mathbb{Z})^*$ zeigen. Dazu benötigen wir die folgenden Hilfssätzte:

Lemma 1.3.16 *Für $n \in \mathbb{N}$ gilt $\sum_{d|n} \varphi(d) = n$.*

Beweis. Wir bilden für alle Teiler t von n die Menge

$$A_t = \left\{ x \in \{1, 2, \ldots, n\} |\ \mathrm{ggT}\left(\frac{x}{t}, \frac{n}{t}\right) = 1 \right\}.$$

Dann ist $|A_t| = \varphi(\frac{n}{t})$. Andererseits ist $A_t = \{x \in \{1, 2, \ldots, n\} |\ \mathrm{ggT}(x, n) = t\}$. Und da jedes $x \in \{1, 2, \ldots, n\}$ genau einen größten gemeinsamen Teiler mit n besitzt, gilt $\sum_{t|n} |A_t| = n$, also $\sum_{t|n} \varphi(\frac{n}{t}) = n$. Da $\sum_{t|n} \varphi(\frac{n}{t}) = \sum_{t|n} \varphi(n)$ folgt die Behauptung. □

Lemma 1.3.17 *Ist K ein Körper und $f = \sum_{i=0}^{n} a_i X^i \in K[X]$ mit grad$(f) = n$ und $a_n \neq 0$. Dann besitzt f höchstens n Nullstellen in K.*

Beweis. Beweis mit Hilfe vollständiger Induktion über grad$f = n$:
Für $n = 1$ besitzt das lineare Polynom $f(X) = a_1 X + a_0$ genau eine Nullstelle. Sei nun die Behauptung für $n-1$ vorausgesetzt und sei $x \in K$ eine Nullstelle von f, dann gibt es nach Division mit Rest in $K[X]$ $g, r \in K[X]$, so dass

$$f(X) = g(X)(X - x) + r(X), \text{ mit } \mathrm{grad}(r) < \mathrm{grad}(X - x)$$

Es ist also grad$(r(X)) = 0$, also $r(X) = r_0 \in K$ und damit $0 = f(x) = g(x)(x - x) + r_0$ und damit $f(X) = g(X)(X - x)$ mit grad(g) <grad(f). Nach Induktionsvoraussetzung besitzt g höchstens $n - 1$ Nullstelle, also besitzt f höchsten n Nullstellen. □

Lemma 1.3.18 *Sei $a \in (\mathbb{Z}/p\mathbb{Z})^*$ mit ord$_p(a) = k$, $h \in \mathbb{N}$ und ggT$(h, k) = d$. Dann sind die Potenzen $\{a^j \mod p | 0 \leq j < k\}$ paarweise verschieden und es gilt ord$_p(a^h) = \frac{k}{d}$.*

Beweis. Wäre $a^i \equiv a^j \mod p$, $0 \leq i, j < k$, dann würde folgen $a^{i-j} \equiv 1 \mod p$ und damit $k|(i - j)$. ↯, da $|i - j| < k$.
Sei nun $h = dh_1$ und $k = dk_1$ mit ggT$(h_1, k_1) = 1$.
Dann ist $(a^h)^{k_1} \equiv a^{dh_1 k_1} \equiv (a^k)^{h_1} \equiv 1 \mod p$. Es gilt also ord$_p(a^h)|k_1$. Ist andererseits $r =$ord$_p(a^h)$, dann folgt $(a^h)^r \equiv a^{hr} \equiv 1 \mod p$ und damit $k|hr$, also auch $k_1|h_1 r$. Da ggT$(k_1, h_1) = 1$ folgt daraus $k_1|r =$ord$_p(a^h)$. Damit folgt ord$_p(a^h) = k_1 = \frac{d}{k}$. □

Satz 1.3.19 *Sei p prim und $d \in \mathbb{N}$ mit $d|p-1$. Dann gibt es in $(\mathbb{Z}/p\mathbb{Z})^*$ $\varphi(d)$ Restklassen mit der Ordnung d.*

Beweis. Sei mit $\psi(d)$ die Anzahl der Restklassen der Ordnung d in $(\mathbb{Z}/p\mathbb{Z})^*$ bezeichnet. Wir zeigen, dass unter der Voraussetzung $\psi(d) > 0$, $\psi(d) = \varphi(d)$ gilt.
Sei dazu $a \in (\mathbb{Z}/p\mathbb{Z})^*$ mit $\mathrm{ord}_p(a) = d$ gegeben, und da $\varphi(p) = p-1$ gilt $d|p-1$. Mit a sind auch alle Potenzen a^c mit $0 \leq c < d$ Nullstellen des Polynoms
$f(X) = X^d - 1 \in (\mathbb{Z}/p\mathbb{Z})[X]$, da

$$(a^c)^d \equiv a^{cd} \equiv 1^c \mod p.$$

Da f im Körper $\mathbb{Z}/p\mathbb{Z}$ höchstens d Nullstellen besitzt, sind die Zahlen
$\{a^c \in (\mathbb{Z}/p\mathbb{Z})^* | 0 \leq c < d\}$ auch alle Nullstellen von f. Da aber auch jedes Element der Ordnung d eine Nullstelle von f ist, ist es als Potenz von a darstellbar. Nach Lemma 1.3.18 ist $\mathrm{ord}_p(a^j) = \frac{d}{\mathrm{ggT}\ (j,d)}$, also gilt $\mathrm{ord}_p(a^j) = d$ genau dann, wenn $\mathrm{ggT}(j,d) = 1$ für $j \in \{0, 1, \ldots, d-1\}$, also für $\varphi(d)$ Elemente. Es folgt also $\psi(d) = \varphi(d)$.

Es bleibt zu zeigen, dass der Fall $\psi(d) = 0$ nicht eintreten kann. Da jede Restklasse eine Ordnung besitzt, gilt $\sum_{d|p-1} \psi(d) = p-1$. Ist nun $\psi(d) = 0$ für ein d, so folgt

$$p - 1 = \sum_{d|p-1} \psi(d) < \sum_{d|p-1} \varphi(d) = p - 1 \ \lightning.$$

□

Folgerung. Es gibt in $(\mathbb{Z}/p\mathbb{Z})^*$ also insbesondere $\varphi(p-1)$ Restklassen mit maximaler Ordung $p-1$. Solche Restklassen werder *Primitivwurzeln* genannt, da ihre Potenzen die gesamte multiplikative Gruppe $(\mathbb{Z}/p\mathbb{Z})^*$ erzeugen, welche also zyklisch ist.

Bemerkung. Man kann zeigen, dass genau dann eine Primitivwurzel modulo einer natürlichen Zahl n existiert, wenn gilt $n = 2$, $n = 4$, $n = p^k$ oder $n = 2p^k$, wobei p eine ungerade Primzahl und $k \in \mathbb{N}$ ist. [RU87]

Satz 1.3.20 *Sei $n \in \mathbb{N}$ und $g \in (\mathbb{Z}/nZ)^*$ eine Primitivwurzel. Dann ist*

$$\{g^k \mid 1 \leq k \leq \varphi(n),\ ggT(k, \varphi(n)) = 1\}$$

die Menge aller Primitivwurzeln modulo n.

Beweis. Da g primitiv ist, gilt $\{g^k \mid 1 \leq k \leq \varphi(n)\} = (\mathbb{Z}/nZ)^*$. Nach Lemma 1.3.18 ist $\mathrm{ord}_n(g^k) = \mathrm{ord}_n(g) = \varphi(n)$ genau dann, wenn $\mathrm{ggT}(k, \varphi(n)) = 1$. □

Wir sehen also, dass man aus der Kenntnis einer Primitivwurzel sämtliche Primitivwurzeln bestimmen kann. Es gibt allerdings kein elegantes Verfahren, zur Bestimmung einer Primitivwurzel. Eine Methode durch systematisches Probieren in $(\mathbb{Z}/p\mathbb{Z})^*$ geht auf Gauss zurück:

1. Wir bestimmen $t_2 = \mathrm{ord}_p(2)$. Ist $t_2 = p-1$ sind wir fertig.
2. Ist $t_2 \neq p-1$, so wählen wir ein $b \in (\mathbb{Z}/p\mathbb{Z})^* \setminus \{2^k \mid 0 \leq k < t_2\}$ und bestimmen $t_b = \mathrm{ord}_p(b)$. Ist $t_b = p-1$ sind wir fertig.

3. Andernfalls wiederholen wir den 2. Schritt mit einem
   $c \in (\mathbb{Z}/p\mathbb{Z})^* \setminus \{\{2^k \mid 0 \leq k < t_2\} \cap \{b^k \mid 0 \leq k < t_b\}\}$ usw.

Gauss merkt zu diesem Verfahren an:

> „Der Geübte wird wissen, dass man die Weitläufigkeit des Verfahrens durch mannigfache Kunstgriffe abkürzen kann; doch lernt man diese viel schneller durch praktische Übung als durch theoretische Vorschriften kennen."

In der folgenden Tabelle, sind die kleinsten Primitivwurzeln g zu alle Primzahlen $p \leq 59$ gegeben:

p	2	3	5	7	11	13	17	19	23	29	31	37	41	43	47	53	59
$\varphi(p-1)$	1	1	2	2	4	4	8	6	10	12	8	12	16	12	22	24	28
g	1	2	2	3	2	2	3	2	5	2	3	3	6	3	5	2	2

Der diskrete Logarithmus

Mit Hilfe der Primitivwurzeln können wir den Begriff des Logarithmus in endliche Körper $\mathbb{Z}/p\mathbb{Z}$ übertragen. Wir betrachten dazu die folgende Abbildung.

Satz 1.3.21 *Sei g eine Primitivwurzel von $(\mathbb{Z}/p\mathbb{Z})^*$. Dann ist die Abbildung*

$$\begin{aligned} \Phi : \mathbb{Z}/(p-1)\mathbb{Z} &\to (\mathbb{Z}/p\mathbb{Z})^* \\ n &\mapsto g^n \mod p, \end{aligned}$$

ein Isomorphismus von der additiven Gruppe $\mathbb{Z}/(p-1)\mathbb{Z}$ in die multplikative Gruppe $(\mathbb{Z}/p\mathbb{Z})^*$.

Beweis. Φ ist offensichtlich ein Homomorphismus. Da $\#\mathbb{Z}/(p-1)\mathbb{Z} = \#(\mathbb{Z}/p\mathbb{Z})^*$ reicht es, die Injektivität von Φ zu zeigen. Gelte also

$$\Phi(a) = \Phi(b) \Leftrightarrow g^a \equiv g^b \mod p \Leftrightarrow g^{a-b} \equiv 1 \mod p,$$

dann ist $a \equiv b \mod (p-1)$. Also ist Φ ein Isomorphismus. □

Folgerung. Die Umkehrabbildung

$$\begin{aligned} \mathrm{ind}_g : (\mathbb{Z}/p\mathbb{Z})^* &\to \mathbb{Z}/(p-1)\mathbb{Z} \\ a = g^n &\mapsto n \end{aligned}$$

ist damit ebenfalls ein Isomorphismuns und wird als *Index* oder *diskreter Logarithmus* zur Basis g bezeichnet. Für den Index gelten die folgenden Rechenregeln:

1. $\mathrm{ind}_g(xy) = \mathrm{ind}_g(x) + \mathrm{ind}_g(y)$
2. $\mathrm{ind}_g(x^k) = k \cdot \mathrm{ind}_g(x)$, $k \in \mathbb{N}$
3. $\mathrm{ind}_g(a) \cdot \mathrm{ind}_a(b) \equiv \mathrm{ind}_g(b) \mod (p-1)$

Beweis. (1) ist die Homomorphie-Eigenschaft von ind_g und (2) folgt aus (1) mit Induktion. (3) rechnen wir nach:

$$g^{\mathrm{ind}_g(a)\cdot\mathrm{ind}_a(b)+k(p-1)} \equiv a^{\mathrm{ind}_a(b)} \left(g^{(p-1)}\right)^k \equiv b \mod p$$

□

Neben der Faktorisierung großer Zahlen stellt die Bestimmung des diskreten Logarithmus zu einer gegeben Basis g ein zweites Problem der Zahlentheorie dar, welches für große Zahlen praktisch unlösbar ist.

Das Baby-Step - Giant-Step Verfahren

Ein Verfahren, diskrete Logarithmen durch systematisches Probieren zu bestimmen, ist der *Baby-Step - Giant-Step* Algorithmus, welcher auf D. Shanks zurückgeht.

Sei $x = \mathrm{ind}_g(a)$ in $(\mathbb{Z}/p\mathbb{Z})^*$ und g eine Primitvwurzel. Wir setzten nun $m := \lfloor\sqrt{p}\rfloor$. Dann besitzt x eine Darstellung $x = qm + r$ mir $0 \leq q, r \leq m$ und es gilt

$$g^{qm} \equiv ag^{-r} \mod p.$$

Wir berechnen nun zunächst die Menge der *Baby-Steps* $B = \{ag^{-r} | 0 \leq r < m\}$. Finden wir in dieser Menge das Element 1, so ist das dazugehörige r offensichtlich der gesuchte diskrete Logarithmus. Ist dies nicht der Fall, so berechnen wir die Menge der *Giant-Steps* $G = \{(g^m)^q | q \in \mathbb{N}\}$, solange bis $G \cap B \neq \varnothing$. Für ein Element im Schnitt gilt dann

$$g^{qm+r} \equiv \mod p,$$

also ist $x = gm + r$ der gesuchte diskrete Logarithmus. Ein anderes Verfahren, diskrete Logarithmen zu berechnen, ist das *Silver-Pohlig-Hellman Verfahren*. Es basiert darauf, dass die Primfaktorzerlegung von $p - 1$ bekannt ist [Kob94]. Damit kommen wir zum nächsten Abschnitt, den Faktorisierungsalgorithmen.

1.4 Primzahltests und Faktorisierungsalgorithmen

1.4.1 Der Miller-Rabin Primzahltest

Ein naiver Primzahltest geht auf den Satz von Fermat zurück.

Fermat-Test Sei $n \in \mathbb{N}$, $b \in (\mathbb{Z}/n\mathbb{Z})^*$ und gelte

$$b^n \not\equiv 1 \mod n,$$

dann ist n keine Primzahl.

Dieses folgt unmittelbar aus dem Satz von Fermat. Es läßt sich mit diesem Test allerdings nicht mit Sicherheit zeigen, dass n eine Primzahl ist, selbst wenn n die Kongruenz für alle Basen $b \in (\mathbb{Z}/n\mathbb{Z})^*$ erfüllt.

Definition 1.4.1 *Sei n eine ungerade ganze zusammengesetzte Zahl und $b \in (\mathbb{Z}/n\mathbb{Z})^*$. Erfüllt n die Kongruenz*

$$b^n \equiv 1 \mod n,$$

heißt n Pseudoprimzahl zur Basis b. Erfüllt n die Kongruenz sogar für alle $b \in (\mathbb{Z}/n\mathbb{Z})^*$, so heißt n Carmichael Zahl.

Bemerkung. Man kann zeigen, dass eine quadratfreie ganze Zahl n genau dann eine Carmichael Zahl ist, wenn für jeden Primteiler p von n gilt $p-1|n-1$. [Kob94]

Wir wollen den obigen *Fermat-Test* nun mit einer zusätzlichen Bedingung versehen, die es uns erlauben wird, eine Primzahl mit einer höheren Wahrscheinlichkeit als solche zu erkennen. Dafür benötigen wir die folgenden Hilfssätze.

Lemma 1.4.2 *Sei p prim. Dann gibt es in $\mathbb{Z}/p\mathbb{Z}$ genau zwei quadratische Einheitswurzeln, nämlich ± 1.*

Beweis. Es ist klar, dass ± 1 quadratische Einheitswurzeln sind. Es bleibt also zu zeigen, dass es nicht mehr als diese zwei gibt. Wir suchen eine Lösung der Kongruenz

$$n^2 \equiv 1 \mod p.$$

Sei nun g eine Primitivwurzel von $(\mathbb{Z}/p\mathbb{Z})^*$ und $n \not\equiv 0 \mod p$, dann folgt

$$2\ \mathrm{ind}_g(n) \equiv \mathrm{ind}_g(1) \equiv 0 \mod (p-1) \Leftrightarrow \mathrm{ind}_g(n) = k \cdot \frac{p-1}{2},\ k \in \mathbb{Z}$$

Die Kongruenz hat also genau zwei Lösungen $\mathrm{ind}_g(n) = 0$ und $\mathrm{ind}_g(n) = \frac{p-1}{2}$. Wir haben gezeigt, dass es in $(\mathbb{Z}/p\mathbb{Z})^*$ genau 2 quadratische Einheitswurzeln gibt, und damit auch in $\mathbb{Z}/p\mathbb{Z}$, da $0^2 \not\equiv 1 \mod p$. □

Folgerung. Sei g eine Primitivwurzel von $(\mathbb{Z}/p\mathbb{Z})^*$. Dann gilt $g^{\frac{p-1}{2}} \equiv -1 \mod p$.

Lemma 1.4.3 *Sei p prim und $p-1 = 2^s d$ eine Darstellung von $p-1$ mit $s > 0$ und t ungerade. Sei $a := b^d \mod p$ mit $b \in (\mathbb{Z}/p\mathbb{Z})^*$. Dann gilt entweder $a \equiv 1 \mod p$ oder es existiert ein $r \in \{1, 2, \ldots, s-2\}$, so dass $a^{2^r} \equiv -1 \mod p$.*

Beweis. Wir betrachten die Ordnung von b in $(\mathbb{Z}/p\mathbb{Z})^*$. Da $d|p-1$ ist der Fall $a \equiv 1 \mod p$ möglich. Sei nun $a \not\equiv 1 \mod p$. Da $\mathrm{ord}_p(b)|p-1$, muss $\mathrm{ord}_p(b^d)$ eine Zweierpotenz sein. Wir nehmen an, diese ist 2^{r_0}. Dann ist aber $a^{2^{r_0-1}}$ eine quadratische Einheitswurzel modulo p und da $a^{2^{r_0-1}} \not\equiv 1 \mod p$, da $\mathrm{ord}_p(a) = 2^{r_0}$ und es nur die quadratischen Einheitswurzeln ± 1 modulo p gibt, folgt $a^{2^{r_0-1}} \equiv -1 \mod p$ mit $r_{0-1} \in \{1, 2, \ldots, s-2\}$. □

Bemerkungen.

1. Eine zusammengesetzte ungerade ganze Zahl n, welche die Bedingungen von Lemma 1.4.2 erfüllt, wird *starke Pseudoprimzahl* zur Basis b genannt.

2. Man kann zeigen, dass die Wahrscheinlichkeit, dass eine zusammengesetzte ungerade ganze Zahl n die Bedingungen aus Lemma 1.4.2 für ein zufälliges $b \in (\mathbb{Z}/n\mathbb{Z})^*$ erfüllt, kleiner als $\frac{1}{4}$ ist. [Kob94][Buc99]

Der **Miller-Rabin-Primzahltest** besteht nun aus den folgenden Schritten:

1. Sei n eine postitive ungerade ganze Zahl, welche wir auf ihre Zerlegbarkeit testen wollen. Wir berechnen zunächst eine Zerlegung von $n-1 = 2^s d$ mit d ungerade, indem wir den Faktor 2 mit maximaler Potenz s herausziehen.

2. Wir wählen zufällig eine ganze Zahl b aus $\mathbb{Z}/n\mathbb{Z}$ und berechnen mit dem euklidischen Algorithmus $\mathrm{ggT}(b, n)$. Ist dieser ungleich 1, so haben wir einen Faktor von n gefunden und sind fertig.

3. Ist $\mathrm{ggT}(b, n) = 1$, berechnen wir $a \equiv b^d \mod n$. Erhalten wir den Wert $a = 1$ so n hat den Test bestanden und wir wiederholen den Test mit einem neuen b.

4. Bekommen wir für a einen Wert ungleich 1, berechnen wir $a^{2^r} \mod n$ für $r \in \{1, 2, \ldots, n-2\}$. Erhalten wir den Wert -1, so hat n den Test bestanden und wir wiederholen den Test mit einem neuen b. Erhalten wir dagegen den Wert 1 für $r = r_0$, aber nicht den Wert -1 für $r = r_0 - 1$, dann hat n den Test nicht bestanden und wir wissen nach Lemma 1.4.2, dass n zusammengesetzt ist.

Besteht eine Zahl n diesen Test für k verschiedene zufällige Werte von b, so ist n mit Wahrscheinlichkeit von höchstens $\frac{1}{4^k}$ keine Primzahl.

1.4.2 Faktorisierung mit Hilfe von Faktorbasen

Lemma 1.4.4 *Seien $n \in \mathbb{N}$ und $s, t \in \mathbb{Z}$ mit $t^2 \equiv s^2 \mod n$ und $t \not\equiv \pm s \mod n$. Dann ist $d = ggT(n, t+s)$ (oder $\tilde{d} = ggT(n, t-s)$) ein Teiler von n.*

Beweis. Es gilt also $n|(t-s)(t+s)$, aber $n \not| (t \pm s)$. Wäre o.B.d.A $\mathrm{ggT}(n, t+s) = 1$, dann würde gelten $n|(t-s)$ ↯. Also ist $1 < \mathrm{ggT}(n, t \pm s) \leq n$ ein Teiler von n. □

Es stellt sich nun die Frage, wie man geeignete s, t mit obigen Eigenschaften finden kann. Wir stellen nun einen Algorithmus dar, der mit Hilfe sogenannter Faktorbasen arbeitet.

Definition 1.4.5 *Eine Faktor-Basis ist eine Menge $\mathcal{B} = \{p_1, p_2, \ldots, p_k\}$ von verschiedenen Primzahlen (wobei p_1 auch -1 sein darf).*

Definition 1.4.6 *Eine Zahl $b \in \mathbb{Z}$ heißt $\mathcal{B}_n$-Zahl (für ein gegebenes n), wenn ihr kleinster absoluter Rest $b^2 \mod n$ als Produkt der Elemente von $\mathcal{B}$ geschrieben werden kann.*

Bemerkung. Um mit kleineren Zahlen zu hantieren, betrachten wir in diesem Abschnitt die kleinsten absoluten Reste $\{-\frac{n-1}{2}, \ldots, \frac{n-1}{2}\}$ (wir können n als ungerade vorraussetzten) als Repräsentanten der Elemente aus $\mathbb{Z}/n\mathbb{Z}$.

Wir betrachten nun die Abbildung ψ, die jeder $\mathcal{B}_n$-Zahl einen Vektor $v \in (\mathbb{Z}/2\mathbb{Z})^h$ zuordnet, wobei h die Anzahl der Elemente der Faktor-Basis $\mathcal{B}$ ist. Ist nun b eine $\mathcal{B}$-Zahl und $\prod_{i=1}^{n} p_i^{r_i}$ die Primfaktorzerlegung des kleinsten absoluten Restes $b_i^2 \mod n$, dann ist ψ definiert durch:

$$b \overset{\psi}{\mapsto} (r_1 \mod 2,\ r_2 \mod 2, \ldots,\ r_h \mod 2) \in (\mathbb{Z}/2\mathbb{Z})^h$$

Satz 1.4.7 *Seien $b_1, b_2, \ldots, b_k$ $\mathcal{B}_n$-Zahlen mit $\psi(b_1)+\psi(b_2)+\cdots+\psi(b_k) = \vec{0} \in (\mathbb{Z}/2\mathbb{Z})^h$ und seien mit $c_i = \prod_{j=1}^{h} p_j^{r_{ij}}$ die kleinsten absoluten Reste $b_i^2 \mod n$ bezeichnet, wobei $p_j \in \mathcal{B}$. Dann ist $\prod_{i=1}^{k} c_i = \prod_{j=1}^{h} p_j^{\sum_{i=1}^{k} r_{ij}}$ eine Quadratzahl und es gilt*

$$\left(\prod_{i=1}^{k} b_i\right)^2 \equiv \prod_{i=1}^{k} c_i \mod n.$$

Beweis. Da $\psi(b_1)+\psi(b_2)+\cdots+\psi(b_k) = \vec{0}$ ist auch $\sum_{i=1}^{k} r_{ij} \equiv 0 \mod 2$ für $j = 1, \ldots, h$, also ist $\prod_{i=1}^{k} c_i = \prod_{j=1}^{h} p_j^{\sum_{i=1}^{k} r_{ij}}$ eine Quadratzahl. Die Gleichheit der Produkte ist klar, da $c_i = b_i^2 \mod n$. □

Bemerkung. Da aus $b^2 \equiv c^2 \mod n$ nicht notwendig folgt $b \equiv c \mod n$ ist es möglich ein Paar (b, c) zu finden, so dass $\mathrm{ggT}(b + c, n) \neq 1$ und damit ein Teiler von n ist.

Es stellt sich nun die Frage, wie wir geeignete b_i bzw. eine geeignete Faktor-Basis $\mathcal{B}$ finden. Eine Möglichkeit ist es mit einer Faktor-Basis $\mathcal{B} = \{-1, 2, 3, 5, \ldots, p_{h-1}\}$ bestehend aus den ersten $h-1$ Primzahlen zu beginnen und zufällig mehrere b_i wählen, bis wir eine Menge von $\mathcal{B}$-Zahlen gefunden hat. Eine andere Möglichkeit ist es, mit einer einer Menge zufällig gewählter ganzer Zahlen b_i zu beginnen, für die der Wert des kleinsten absoluten Restes $c_i := b_i^2 \mod n$ ein Produkt kleiner Primzahlen ist. Hierzu ist es sinnvoll, die b_i in der Nähe von $\sqrt{kn}$, $k = 1, 2, \ldots$ zu wählen, etwa $b_i = \lfloor\sqrt{kn}\rfloor + 1$. Nun wählen wir die Faktor-Basis $\mathcal{B}$ so, dass die b_i $\mathcal{B}$-Zahlen sind.

In jedem Falle reicht es zu einer gegeben Faktor-Basis $\mathcal{B}$, bestehend aus h Elementen, $h + 1$ $\mathcal{B}$-Zahlen b_i zu suchen, da $h + 1$ Vektoren $v_i \in (\mathbb{Z}/2\mathbb{Z})^h$ im Vektorraum $(\mathbb{Z}/2\mathbb{Z})^h$ linear abhängig sind, es also eine nicht triviale Darstellung des Nullvektors gibt.

Bemerkungen.

1. Eine weitere Methode, eine Faktorbasis mit geeigneten Zahlen b_i zu konstruieren wird in [Kob94] beschrieben. Diese basiert auf Kettenbrüchen und ihr Vorteil ist, dass die kleinsten absoluten Reste $b_i^2 \mod n$ klein und damit Produkte kleiner Primzahlen sind.

2. Man kann zeigen, dass der Faktorbasis-Algorithmus zur Faktorisierung einer natürlichen Zahl n maximal $\mathcal{O}\left(e^{C\sqrt{\ln(n)\ln(\ln(n))}}\right)$ Bit-Operationen benötigt, mit einer Konstanten C. Man kann weiterhin zeigen, dass die Konstante C für schnelle Algorithmen von der Form $1 + \epsilon$ mit beliebig kleinem $\epsilon > 0$ ist. [Kob94] Nach [Sem02] können mit heutigen Rechnern mit diesen Algorithmen Zahlen mit bis zu 120 Dezimalstellen in vernünftiger faktorisiert werden.

3. Der schnellste zur Zeit bekannte Faktorisierungsalgorithmus ist das so genannte Zahlkörpersieb. Hier werden Lösungen der Kongruenz $x^2 \equiv y^2 \mod n$ mit Hilfe der algebraischen Zahlentheorie gesucht. Die Laufzeit zur Faktosisierung einer natürlichen Zahl n beträgt $\mathcal{O}\left(e^{\sqrt{(\ln(n))^{\frac{1}{3}}(\ln(\ln(n)))^{\frac{2}{3}}}}\right)$. Nach [Sem02] können mit dem Zahlkörpersieb Zahlen mit deutlich mehr als 120 Dezimalstellen in vernünftiger Zeit faktorisiert werden.

Eine andere Faktoriersierungsmethode, für welche theoretische Algorithmen auf Quantencomputern existieren, basieren auf dem Finden von Ordnungen modulo n:

Lemma 1.4.8 *Sei $n \in \mathbb{N}$, $x \in (\mathbb{Z}/n\mathbb{Z})^*$ und $r = 2k$ mit $k \in \mathbb{N}$ die Ordnung von $x \mod n$. Gilt $x^{\frac{r}{2}} \not\equiv -1 \mod n$ dann folgt $ggT(x^{\frac{r}{2}} \pm 1, n) \neq 1$ und damit ein echter Teiler von n.*

Beweis. Es gilt also $x^r \equiv 1 \mod n$ und damit $n | x^r - 1 = (x^{\frac{r}{2}} + 1)(x^{\frac{r}{2}} - 1)$. Weiterhin gilt $n \not| x^{\frac{r}{2}} + 1$ nach Vorraussetzung und $n \not| x^{\frac{r}{2}} - 1$, da sonst $x^{\frac{r}{2}} \equiv 1 \mod n$ und dies wäre ein Widerspruch, da $\text{ord}_n(x) = r$. Also ist $\text{ggT}(x^{\frac{r}{2}} \pm 1, n) \neq 1$. □

1.5 Krypto-Systeme

1.5.1 Definitionen und Schreibweisen

Zunächst sollen einige Sprechweisen eingeführt werden, wie sie in der Kryptographie üblich sind:

Definition 1.5.1 *Ein Kryptosystem ist ein Quintupel $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ von endlichen Mengen, wobei für jedes $k \in \mathcal{K}$ ein Paar $(e_k, d_k) \in \mathcal{E} \times \mathcal{D}$ existiert, so dass*

$$\begin{aligned} e_k &: \mathcal{P} \to \mathcal{C} \\ d_k &: \mathcal{C} \to \mathcal{D} \\ d_k \circ e_k(P) &= P \text{ für alle } P \in \mathcal{P} \end{aligned}$$

Mit $\mathcal{P}$ wird üblicherweise die Menge Klartexte, mit $\mathcal{C}$ die Menge der Chiffretexe, mit $\mathcal{K}$ die Menge der möglichen Schlüssel und mit $\mathcal{V}$ bzw. $\mathcal{E}$ die Mengen der Ver- bzw. Entschlüsselungsfunktionen, welche durch die Schlüssel k parametrisiert sind, bezeichnet.

Beispiel. Ein einfaches Beispiel für eine Kryptosystem ist das *Caesar-Chiffre*. In der einfachsten Form gilt $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}/26\mathbb{Z}$, wobei jeder Buchstabe $\{A, B, ...Z\}$ mit einer Zahl entsprechend seiner Position im Alphabet identifiziert wird. Die Verschlüsselung v_k hat hierbei die Form

$$\begin{aligned} v_k &: \mathbb{Z}/26\mathbb{Z} \to \mathbb{Z}/26\mathbb{Z} \\ v_k &: P \longmapsto P + k \mod 26, \end{aligned}$$

mit $P \in \mathcal{P} = \{0, 1, ..., 26\}$. Die Entschlüsselungsfunktion e_k hat nun einfach die Form

$$\begin{aligned} e_k &: \mathbb{Z}/26\mathbb{Z} \to \mathbb{Z}/26\mathbb{Z} \\ e_k &: P \longmapsto P - k \mod 26 \end{aligned}$$

1.5.2 Verschlüsseln in Blöcken

Zum Verschlüsseln mit modernen Verschlüsselungsalgorithmen ist es sinnvoll, jedem Zeichen ein Element aus $(\mathbb{Z}/n\mathbb{Z})$ zuzuordnen, da die Verschlüsselungen häufig in diesen Ringen stattfindet. Wir werden sehen, dass die Sicherheit eines Kryptosystems zunimmt, wenn nicht nur einzelne Buchstaben sondern „Wörter“ von mehreren Buchstaben einem Element aus $(\mathbb{Z}/n\mathbb{Z})$ zugeordnet werden.

Als *Wort* der Länge n wollen wir im folgenden Folgen von n Buchstaben oder Zeichen verstehen. Da die Übersicht des Klartextes stark von Leer- und Satzzeichen abhängt, ist es ein Vorteil, wenn der Klartextraum diese enthält.

Codieren der Zeichen

Im folgenden soll kurz betrachtet werden, auf welche verschiedenen Weisen Texte in numerische Werte umgewandelt oder codiert werden können. Dazu unterscheiden wir das Alphabet der Zeichen

$$\overline{\Sigma}_N := \{\underbrace{A, B, ..., Z, ..., 0, 1, ..., 9, ..., ., ?, !}_{N}\}$$

und das dazu numerische Äquivalent $\Sigma_N = \mathbb{Z}/N\mathbb{Z}$.

Da $|\overline{\Sigma}_N| = |\Sigma_N|$, existiert eine bijektive Abbildung $c : \overline{\Sigma}_N \to \Sigma_N$, die jedem Zeichen sein numerisches Äquivalent zuordnet.

Seien im folgenden $\overline{p}_i \in \overline{\Sigma}_N$ und $p_i \in \Sigma_N$. Im Falle von Wörtern der Länge 1, ist eine Codierungsabbildung gegeben durch

$$\begin{array}{rcl} c : \overline{\Sigma}_N & \to & \Sigma_N \\ \overline{p}_i & \longmapsto & p_i \end{array}$$

Betrachten wir nun Wörter $\overline{w}_n = \overline{p}_1 ... \overline{p}_n \in \prod_n \overline{\Sigma}_N$ der Länge n. Für die numerische Darstellung eines Wortes, sind im wesentlichen zwei Möglichkeiten vorhanden:

1. Numerische Darstellung als eine Zahl aus $\{0, ..., N^n - 1\}$
Die Codierungsabbildung c hat hierbei die Form

$$\begin{array}{rcl} c : \prod\limits_n \overline{\Sigma} & \to & \mathbb{Z}/N^n\mathbb{Z} \\ \overline{w}_n & \longmapsto & p_0 + p_1 \cdot N + p_2 \cdot N^2 + \cdots + p_n \cdot N^n, \end{array}$$

wobei $p_i \in \Sigma_N$ die numerischen Äquivalente der $\overline{p_i} \in \overline{\Sigma}_N$ sind.

Die Rücktransformation geschieht durch sukzessives Teilen und ist eindeutig, da jede Zahl eine eindeutige endliche g-adische Zerlegung besitzt. [Wal99]

Bemerkung
Bei der Verschlüsselung muss nicht notwendiger Weise die Länge eines Wortes des Klartextes mit der Länge eines Wortes im Schlüsseltext übereinstimmen.

2. Darstellung der Wörter als Vektoren aus dem $(\mathbb{Z}/N\mathbb{Z})^n$
Die Codierungsabbildung c hat hierbei die Form

$$\begin{array}{rcl} c : \prod\limits_n \overline{\Sigma} & \to & (\mathbb{Z}/N\mathbb{Z})^n \\ \overline{w}_n & \longmapsto & (p_n, \ldots, p_0) \end{array}$$

wobei $p_i \in \Sigma_N$ die numerischen Äquivalente der $\overline{p_i} \in \overline{\Sigma}_N$ sind.

1.5.3 Sicherheit von Kryptosystemen

Wir untersuchen nun die Frage, ob es Kryptosysteme gibt, die „perfekt sicher“ sind. Wir werden sehen, dass es solche Systeme gibt, diese aber für die Anwendung nicht praktikabel sind.

Zunächst definieren wir, was wir unter perfekter Sicherheit eines Kryptosystems verstehen wollen. Hierzu benutzen wir einige Tatsachen aus der Stochastik:

Wir gehen davon aus, dass die Wahrscheinlichkeit der Klartexte durch eine Wahrscheinlichkeitsverteilung $P_{\mathcal{P}} : \mathcal{P} \to [0,1]$ gegeben ist. Weiterhin gehen wir davon aus, dass für die Schlüssel eine Wahrscheinlichkeitsverteilung $P_{\mathcal{K}} : \mathcal{K} \to [0,1]$ vorliegt und dass diese Verteilungen voneinander unabhängig sind. Ist dies der Fall, existiert auf dem Produktraum $\mathcal{K} \times \mathcal{P}$ eine gemeinsame Verteilung $P : \mathcal{K} \times \mathcal{P} \to [0,1]$, die gleich der Produktwahrscheinlichkeit von $P_{\mathcal{P}}$ und $P_{\mathcal{K}}$ ist:

$$P(p,k) = P_{\mathcal{P}}(p) \cdot P_{\mathcal{K}}(k)$$

Aus der gemeinsamen Verteilung P können auch die Wahrscheinlichkeiten für bestimmte Klartexte oder Schlüssel bestimmt werden: $P(p) = \sum_{k \in \mathcal{K}} P(p,k)$

Definition 1.5.2 *Ein Kryptosytem heißt perfekt sicher, wenn die Ereignisse, dass ein bestimmter Chiffretext auftritt und dass ein bestimmter Klartext vorliegt, voneinander unabhängig sind.*

Bemerkungen.

1. Ein perfekt sicheres System kann also ohne weitere Kenntnisse über den Klartext oder den Schlüssel nicht geknackt werden.
2. In Formeln bedeutet dies: $P(p|c) = P(p)$. Die bedingte Wahrscheinlichkeit eines bestimmten Klartextes p unter der Vorraussetzung eines bestimmten Chiffretextes c ist gleich der Wahrscheinlichkeit des Klartextes p unabhängig vom Chiffretext c.

Damit kann nun der Satz von Shannon formuliert und bewiesen werden.

Satz 1.5.3 *Sei $|\mathcal{C}| = |\mathcal{K}|$ und sei $P(p) > 0$ für jeden Klartext p. Ein Kryptosystem ist genau dann perfekt sicher, wenn es für jeden Klartext $p \in \mathcal{P}$ und jeden Chiffretext $c \in \mathcal{C}$ genau einen Schlüssel k gibt, so dass $v_k(p) = c$ gilt und die Wahrscheinlichkeitsverteilung auf dem Schlüsselraum $\mathcal{K}$ die Gleichverteilung ist.*

Beweis.
„$\Rightarrow$“ : Sei das Kryptosystem perfekt sicher und p ein Klartext. Gäbe es einen Chiffretext c, für den es keinen Schlüssel k gibt mit $v_k(p) = c$, dann wäre $P(p|c) = 0 \neq P(p)$, was einen Widerspruch zur perfekten Sicherheit wäre. Damit ist die Existenz eines Schlüssels gezeigt.

Da die Anzahl der Elemente von $\mathcal{C}$ gleich der Anzahl der Elemente von $\mathcal{K}$ ist, ist jede surjektive Abbildung $f : \mathcal{C} \to \mathcal{K}$ auch injektiv und damit bijektiv. Damit gibt es zu jedem Chiffretext $c \in \mathcal{C}$ genau einen Schlüssel $k \in \mathcal{K}$ mit $v_k(p) = c$.

Zeige nun, dass die Wahrscheinlichkeitsverteilung auf dem Schlüsselraum $\mathcal{K}$ die Gleichverteilung ist. Fixiere dazu einen Chiffretext $c \in \mathcal{C}$ mit $c = v_{k_p}(p)$. Dann gilt

$$\begin{aligned} P(c|p) = P(v_k(p)|p) &= \sum_{k \in \mathcal{K}} P(v_k(p)|p,k) = \sum_{k \in \mathcal{K}} \underbrace{P_{\mathcal{P}}(v_k(p)|p)}_{=\begin{cases} 1 & \text{für } k = k_p \\ 0 & \text{für } k \neq k_p \end{cases}} \cdot P_{\mathcal{K}}(k) \\ &= P_{\mathcal{K}}(k_p) = \sum_{\tilde{p} \in \mathcal{P}} P_{\mathcal{K}}(k_p) \cdot P_{\mathcal{P}}(\tilde{p}) = P(k_p) \\ \Rightarrow P(c|p) &= P(k_p) \end{aligned}$$

Im allgemeinen ist also die Wahrscheinlichkeit des richtigen Schlüssels sowohl abhängig vom Chiffretext als auch vom Klartext. Wenn das Kryptosystem aber perfekt sicher ist, gilt für jeden Klartext

$$\begin{aligned} P(p|c) = \frac{P(c|p) \cdot P(p)}{P(c)} &= \frac{P(k_P) \cdot P(p)}{P(c)} = P(p) \\ \Rightarrow P(k_p) &= P(c) \end{aligned}$$

Das heißt die Wahrscheinlichkeit für den Schlüssel k_p ist unabhängig vom Klartext p und damit für jeden Schlüssel gleich. Und das heißt $P(k) = \frac{1}{|\mathcal{K}|}$ für alle $k \in \mathcal{K}$.

„$\Leftarrow$“ : Sei nun die Wahrscheinlichkeitsverteilung auf dem Schlüsselraum $\mathcal{K}$ die Gleichverteilung und existiere für jeden Klartext p und jeden Chiffretext c genau ein Schlüssel $k = k_{p,c}$ mit $v_k(p) = c$. Zu zeigen ist $P(p|c) = P(p)$:

$$P(p|c) = \frac{P(p)P(c|p)}{P(c)} = \frac{P(p)P(k_{p,c})}{\sum_{\tilde{p} \in \mathcal{P}} P(\tilde{p})P(k_{\tilde{p},c})} = \frac{P(p)\frac{1}{|\mathcal{K}|}}{\sum_{\tilde{p} \in \mathcal{P}} P(\tilde{p})\frac{1}{|\mathcal{K}|}} = P(p)$$

□

1.6 Public-Key Kryptographie

Grundlegende Ideen und Definitionen

In allen Kryptosystemen, die in in der Zeit vor etwa 20 Jahren benutzt wurden (das Vernam-Chiffre einmal ausgenommen), war es möglich, aus dem Schlüssel zum Verschlüsseln den Schlüssel zum Entschlüsseln zu berechnen und umgekehrt. Zusätzlich ergab sich das Problem, dass je zwei Parteien, die miteinander kommunizieren wollten, sich auf einen Schlüssel (e_k) einigen und diesen Schlüssel austauschen mussten. In einem Netz, in dem n Teilnehmer miteinander kommunizieren, werden also $\frac{n(n-1)}{2}$ Schlüssel benötigt. In einem vergleichsweise kleinen Netz mit $n = 1000$ Teilnehmern kommen wir so aber schon auf 499500 Schlüssel!

Im Jahre 1976 kam W. Diffie und M. Hellman die Idee der Public-Key-Kryptographie, in welcher geheime Kommunikation möglich ist, ohne vorher einen geheimen Schlüssel austauschen zu müssen. Die Ideen von Diffie und Hellman werden im zweiten Teil dieser Arbeit näher erläutert, an dieser Stelle wollen wir uns auf die Mathematik beschränken.

Definition 1.6.1 *Seien $\mathcal{P}$ und $\mathcal{C}$ Mengen. Eine Funktion $f : \mathcal{P} \to \mathcal{C}$ heißt Falltür-Funktion, wenn ihre Umkehrfunktion $f^{-1} : \mathcal{C} \to \mathcal{P}$ ohne zusätzliche Informationen praktisch nicht zu berechnen ist.*

Bemerkung. Praktisch nicht zu berechnen bedeutet in diesem Zusammenhang, dass die Berechnung zwar theoretisch möglich ist, aber nicht in einer vernünftigen Zeit (z.B. einem Menschenleben). Vor allem durch den Fortschritt in der Rechengeschwindigkeit der Computer ist nicht vorherzusagen, ob eine heutige Falltür-Funktion auch morgen noch eine ist.

Eine Idee von Diffie und Hellmann war es nun, zur Verschlüsselung eine Falltür-Funktion zu verwenden, wobei der Schlüssel zum Entschlüsseln die Zusatzinformation zum Berechnen der Umkehrfunktion sein sollte.

Definition 1.6.2 *Ein Public-Key-Kryptosystem ist ein Kryptosystem* $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ *mit den folgenden Eigenschaften:*

1. *Die Menge* $\mathcal{K}$ *der Schlüssel zerfällt in zwei Mengen,* $\mathcal{K} = \mathcal{K}_{Pu} \cap \mathcal{K}_{Pr}$*, die Menge der öffentlichen und die Menge der privaten Schlüssel.*

2. *Die Menge* $\mathcal{E}$ *der Verschlüsselungsfunktionen besteht aus Falltür-Funktionen, welche nach den öffentlichen Schlüsseln* $k_{Pu} \in \mathcal{K}_{Pu}$ *parametrisiert sind.*

3. *Die Menge* $\mathcal{D}$ *der Entschlüsselungsfunktionen besteht aus den Umkehrfunktionen der* $e_{k_{Pu}} \in \mathcal{E}$*, welche nur mit Wissen des zugehörigen privaten Schlüssels* $k_{P_r} \in \mathcal{K}_{Pr}$ *bestimmt werden können.*

Bemerkung. Public-Key-Kryptosysteme werden auch *assymetrische* Kryptosysteme genannt, im Gegensatz zu den klassischen symmetrischen Verfahren, in denen aus der Kenntnis der Verschlüsselung die Fähigkeit der Entschlüsselung folgte.

Die Verschlüsselung in einem Public-Key-Kryptosystem verläuft dann nach dem folgenden Schema:
Möchte A eine verschlüsselte Nachricht an B schicken, so besorgt sich A den öffentlichen Schlüssel $k_{Pu,B}$ von B und verschlüsselt die Nachricht P mit der zugehörigen Funktion $e_{k_{Pu,B}}$. Das Ergebnis $e_{k_{Pu,B}}(P)$ sendet er an B. Da B den privaten Schlüssel $k_{Pr,B}$ besitzt, kann B die Entschlüsselungsfunktion $d_{k_{Pr,B}}$ und damit $P = (d_{k_{Pr,B}} \circ e_{k_{Pu,B}})(P)$ berechnen.

Da außer B niemand den privaten Schlüssel $k_{Pr,B}$ besitzt, ist außer B niemand in der Lage die verschlüsselte Nachricht zu entschlüsseln, da $e_{k_{Pu,B}}$ eine trapdoor-Funktion ist, und $d_{k_{Pr,B}} = e^{-1}_{k_{Pu,B}}$ nur mit Hilfe von $k_{Pr,B}$ zu berechnen ist.

In einem Public-Key-Kryptosystem werden also für jeden Teilnehmer nur 2 Schlüssel, von denen je einer, der öffentliche Schlüssel zum Verschlüsseln, allen Teilnehmern zugänglich sein muss, benötigt. In unserem obigen Netz mit 1000 Teilnehmern werden nun also lediglich 1000 Schlüsselpaare, also 2000 Schlüssel benötigt.

Das RSA-Verfahren

Eines der bekanntesten Public-Key-Systeme ist das RSA-Verfahren. Beim RSA-Verfahren sind die Mengen $\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D}$ gegeben durch:

- $\mathcal{P}, \mathcal{C} = \mathbb{Z}/(pq)\mathbb{Z}$, p, q prim
- $\mathcal{K}_{Pu} = \{(e, n) \mid 3 \leq e \leq \varphi(n),\ \mathrm{ggT}(e, \varphi(n)) = 1,\ n = pq\}$
- $\mathcal{K}_{Pr} = \{(d, n) \mid de \equiv 1 \mod (p-1, q-1)$ wobei e erste Komponente eines Elementes $(e, n) \in \mathcal{K}_{Pu},\ n = pq\}$
- $\mathcal{E} = \{x \mapsto x^e \mod n \mid (e, n) \in \mathcal{K}_{Pu}\}$

- $\mathcal{D} = \{x \mapsto x^d \mod n \mid (d,n) \in \mathcal{K}_{Pr}\}$

Der öffentliche Schlüssel besteht also aus dem Exponenten und dem dem Modul $n = pq$ der Funktion $x \mapsto x^e \mod n$. Der private Schlüssel besteht aus dem Exponenten d, mit $ed \equiv 1 \mod (p-1, q-1)$ zusammen mit dem Modul n. Die Verschlüsselung verläuft nun nach dem folgendem Schema:

1. A besorgt sich den öffentlichen Schlüssel (e_B, n) von B und verschlüsselt den Klartext $P \mapsto P^{e_B} \mod n$ und sendet das Ergebnis an B

2. B entschlüsselt den von A erhaltenen Text mit seinem privaten Schlüssel (d_B, n): $(P^{e_B})^{d_B} \equiv P^{e_B d_B} \equiv P \mod n$

Satz 1.6.3 *Das RSA-Verfahren ist ein Public-Key-Kryptosystem.*

Beweis. Wir müssen zunächst zeigen, dass für jedes Paar $(f_k, g_k) \in \mathcal{E} \times \mathcal{D}$ gilt $(f_k \circ g_k)(P) = P$. Seien also $f_k = x^{e_k} \mod (pq)$, mit $e_k \in (\mathbb{Z}/(p-1)(q-1)\mathbb{Z})^*$ und $g_k = x^{d_k} \mod (pq)$, mit $e_k d_k \equiv 1 \mod (p-1, q-1)$. Ein solches d_k existiert, da $\mathrm{ggT}(e_k, p-1) = \mathrm{ggT}(e_k, q-1) = 1$. Dann ist

$$(g_k \circ f_k)(P) = P^{ed} \equiv P^1 \mod (pq)$$

wie wir oben in der Folgerung 3 zum Satz von Fermat (1.3.6) gezeigt haben.
Wir müssen nun noch zeigen, dass dies wirklich ein Public-Key-System ist, also der Entschlüsselungsexponent d nicht aus dem Verschlüsselungsexponenten e zu berechnen ist. Um d zu berechnen, müssen die Gleichungen $ed \equiv 1 \mod (p-1)$ und $ed \equiv 1 \mod (q-1)$ gelöst werden. Dies ist aber bei großen Zahlen n praktisch unmöglich, da die beiden Primfaktoren p, q nicht öffentlich sind, sondern nur ihr Produkt n. Die Schwierigkeit das RSA-System zu knacken, hängt also von der Schwierigkeit große Zahlen zu faktorisieren ab. □

Bemerkung. Das RSA-Verfahren ist, wie alle Public-Key-Kryptosysteme, natürlich nicht perfekt sicher im Sinne der obigen Definition, da der Schlüssel e_k bekannt ist, und man alle Klartexte durchprobieren könnte.

Untersuchung des RSA-Verfahrens

Die RSA-Entschlüsselung kann mit Hilfe des chinesischen Restsatztes beschleunigt werden. Der Empfänger, welcher in Besitz der Primzahlen p und q ist, bestimmt dazu einmalig mit Hilfe des erweiterten euklidischen Algorithmus $h, k \in \mathbb{Z}$, so dass

$$hp + kq = 1.$$

Zum Entschlüsseln einer Nachricht C berechnet er nun statt $P \equiv C^d \mod n$

$$\begin{aligned} P_1 &\equiv C^d \mod p \\ P_2 &\equiv C^d \mod q. \end{aligned}$$

Dann gilt $P \equiv (P_1 kq + P_2 hp) \mod n$ nach dem chinesischem Restsatz.

Wie wir oben gesehen haben, hängt die Sicherheit des RSA-Verfahrens von der Schwierigkeit des Faktorisierens großer Zahlen ab. Man geht heute davon aus, dass die Sicherheit gewährleistet ist, wenn die Primzahlen p und q in der Größenordnung 512-1024 Bit (etwa 150 Dezimalstellen) liegen. Dieser Wert hängt zukünftig natürlich von der Entwicklung der Faktorisierungsalgorithmen und der Rechner, auf denen diese implementiert sind, ab. Weiterhin sollten p und q nicht zu dicht zusammen liegen (die Zahl der Dezimalstellen sollte sich um mehr als 2 unterscheiden), da sonst eine schnelle Faktorisierung von $n = pq$ mit Hilfe der Fermat-Faktorisation möglich ist.

Neben der Wahl der Primzahlen p und q gibt es zusätzliche Gefahren, die bei einer falschen Anwedung des RSA-Verfahrens auftreten können. So könnte man auf die Idee kommen, ein Modul n für alle Benutzer eines Kommunikationsnetztes zu fixieren und lediglich jedem Benutzer i ein eigenes Schlüsselpaar (e_i, n), (d_i, n) bereitzustellen, um möglichst wenig Module n erzeugen zu müssen.

Hiervon ist allerdings abzuraten, wie der nächste Satz nach [MOV96] zeigt.

Satz 1.6.4 *Aus der Kenntnis des öffentlichen Schlüssels (e, n) und des privaten Schlüssels (d, n) des RSA-Verfahrens lässt sich die Primfaktorzerlegung des Moduls n berechnen.*

Beweis. Nach dem chinesischen Restsatz gibt es in $(\mathbb{Z}/n\mathbb{Z})^* \cong (\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^*$ genau 4 quadratische Einheitswurzeln:

$$\zeta_1 = (1,1),\ \zeta_2 = (-1,-1),\ \zeta_3 = (1,-1),\ \zeta_4 = (-1,1).$$

Für ζ_3 gilt also $p|(\zeta_3 - 1)$ und $q|(\zeta_3 + 1)$. Damit folgt $\mathrm{ggT}(\zeta_3 + 1, n) = q$ bzw. $\mathrm{ggT}(\zeta_3 - 1, n) = p$. Für ζ_4 gelten, wie man sofort sieht, dieselben Beziehungen mit umgedrehten Vorzeichen.
Wir zeigen nun, dass wir mit Kenntnis der beiden Schlüssel des RSA-Verfahrens in der Lage sind, eine quadratische Einheitswurzel $\neq \pm 1$ modulo n zu berechnen.

Nach dem chinesischem Restsatz gilt $(\mathbb{Z}/n\mathbb{Z})^* \cong (\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^*$. Sei $p - 1 = 2^i y$ sowie $q - 1 = 2^j w$ mit $i, j > 0$ und y, w ungerade.

Nehmen wir nun an, es gibt ein $g \in (\mathbb{Z}/n\mathbb{Z})^*$ mit $\mathrm{ord}_p(g) = 2^r u$ bzw. $\mathrm{ord}_q(g) = 2^s v$, wobei $r > s \geq 0$ und u, v ungerade. Da $\mathrm{ord}_p(g)|p - 1$ bzw. $\mathrm{ord}_q(g)|q - 1$, folgt $r \leq i$, $s \leq j$, $u|y$ sowie $v|w$.

Setze nun $k := ed - 1$. Dann existiert, da $ed \equiv 1 \mod \mathrm{kgV}(p-1, q-1)$, ein $a \in \mathbb{Z}$, $a = 2^\alpha x$ mit $\alpha \geq 0$, x ungerade, so dass

$$k = \mathrm{kgV}((p-1)(q-1))a = 2^{\max\{i,j\}+\alpha}\tilde{y}x,$$

$\tilde{y} = \mathrm{kgV}(y, w)$, also $\tilde{y}$ ungerade. Setzte weiterhin $r_0 := \max\{i, j\} + \alpha - r + 1$, sowie $l := \frac{k}{2^{r_0}}$, dann gilt

$$g^l \equiv g^{2^{r-1}\tilde{y}x} \not\equiv 1 \mod p \text{ und } g^l \equiv g^{2^{r-1}\tilde{y}x} \equiv 1 \mod q,$$

da $s \leq r - 1$. Weiterhin gilt

$$(g^l)^2 \equiv g^{2^r \tilde{y}x} \equiv 1 \mod p \text{ und } (g^l)^2 \equiv 1 \mod q.$$

Also ist g^l eine quadratische Einheitswurzel mod p ungleich 1. Dann folgt, wie wir oben (Lemma 1.4.2) gezeigt haben $g^l \equiv -1 \mod p$. Wir haben also, unter der Annahme der

Existenz eines $g \in (\mathbb{Z}/n\mathbb{Z})^*$ mit obigen Eigenschaften, eine quadratische Einheitswurzel $\neq \pm 1 \bmod n$ gefunden, welche eine Faktorisierung von n liefert. Es ist aber klar, dass es solche Elemente $g \in (\mathbb{Z}/n\mathbb{Z})^*$ gibt, da es, wie wir oben gezeigt haben, zu jedem Teiler d_p, d_q von $p-1$, bzw. $q-1$ Elemente der Ordung d_p, d_q gibt. Die Frage ist nur, mit welcher Wahrscheinlichkeit wir im folgenden Algorithmus ein solches g erwischen.

1. Wir berechnen zunächst $k := de-1$. Dann gilt $g^k \equiv 1 \mod n$ für alle $g \in (\mathbb{Z}/n\mathbb{Z})^*$.

2. Wir bilden nun solange sukzessive $a_t := g^{\frac{k}{2^t}}$, $t = 0, 1, 2, 3, \ldots$ bis $a_{t_0} \not\equiv 1 \mod n$, oder $\frac{k}{2^{t_0}} \notin \mathbb{N}$. Hatte g die obigen Eigenschaften, so ist $a_{t_0} = \zeta_3$ oder $a_{t_0} = \zeta_4$. Dann sind wir fertig und bilden $\mathrm{ggT}(a_{t_0}, n)$. Hatte g die obigen Eigenschaften nicht, so ist $a_{t_0} = \zeta_1$ oder $a_{t_0} = \zeta_2$ und wir müssen mit einem neuen g beginnen.

Da es nach obigen Überlegungen reicht, ein g zu erwischen, mit einer geraden Ordnung mod p und einer ungeraden Ordnung mod q (oder umgekehrt), ist, unter der Annahme, dass die Warscheinlichkeit einer geraden bzw. ungeraden Ordnung eines Elementes ungefähr gleich ist, die Wahrscheinlichkeit, mit einem beliebig gewähltem g Erfolg zu haben ungefähr $\frac{1}{2}$. (siehe auch [Bon]) □

Mit der Kenntnis der Faktorisierung von n wäre nun jeder Benutzter des Netzwerkes in der Lage, die privaten Schlüssel der anderen Teilnehmer aus den öffentlichen Schlüsseln zu berechnen.

Man kann weiterhin zeigen, dass auch zu kleine Exponenten e und d ein Unsicherheitsfaktor sind. Eine *Low Private Exponent* Attacke geht auf M. Wiener, eine der effizientesten Attacken gegen einen kleinen öffentlichen Schlüssel geht auf D. Coppersmith zurück. [Bon]

Eine weitere Unsicherheit bilden die Fixpunkte, also solche Mitteilungen $x \in \mathbb{Z}/n\mathbb{Z}$, welche durch die Verschlüsselung in sich selber übergehen. Wir werden zeigen, dass es solche Fixpunkte immer gibt.

Satz 1.6.5 *Die Anzahl der Fixpunkte x_F mit $x_F^e \equiv x_F \mod (pq)$ beim RSA-Verfahren ist* $(\mathrm{ggT}(e-1, p-1)+1)(\mathrm{ggT}(e-1, q-1)+1) \geq 4$.

Beweis. Wir bestimmen zunächst die Anzahl der Fixpunkte modulo einer Primzahl p. Sei g eine primitive Wurzel in $(\mathbb{Z}/p\mathbb{Z})^*$. Gelte nun

$$\begin{aligned} x^e \equiv x \mod p \quad &\Rightarrow \quad e \cdot \mathrm{ind}_g(x) \equiv \mathrm{ind}_g(x) \mod (p-1) \\ &\Rightarrow \quad \mathrm{ind}_g(x)(e-1) \equiv 0 \mod (p-1) \end{aligned}$$

Die Anzahl der Lösungen dieser lineare Kongruenz ist dann $\mathrm{ggT}(e-1, p-1) > 1$, da $\mathrm{ggT}(e, p-1) = 1$. Da 0 ebenfalls ein Fixpunkt ist, ist die Anzahl der Fixpunkte in $\mathbb{Z}/p\mathbb{Z}$ also $\mathrm{ggT}(e-1, p-1)+1$. Die Behauptung folgt dann mit Hilfe des chinesischen Restsatzes. □

Als letzte Unsicherheit betrachten wir die Tatsache, dass die Klartexte x nicht ausschließlich Elemente der Multiplikativen Gruppe $(\mathbb{Z}/n\mathbb{Z})^*$ sind. Solche Elemente werden zwar auch korrekt verschlüsselt, aber da gilt $\mathrm{ggT}(x, n) > 1$ und $n = pq$ nur die Teiler p und q besitzt, ist es mit Hilfe eines solchen Klartextes möglich, die Faktoren von n zu bestimmen. Wir werden aber sehen, dass die Wahrscheinlichkeit, dass ein zufälliger Klartext x nicht in der Multiplikiven Gruppe liegt, bei großen p, q sehr klein ist:

Satz 1.6.6 *Sei $n = pq$ und $x \in \mathbb{Z}/n\mathbb{Z}$ ein zufällig gewähltes Element. Dann ist die Wahrscheinlichkeit, dass gilt $ggT(x, n) \neq 1$ kleiner als $\frac{1}{p} + \frac{1}{q}$.*

Beweis. Da $n = pq$ folgt $\mathrm{ggT}(x, n) = p$ oder $\mathrm{ggT}(x, n) = q$ oder $\mathrm{ggT}(x, n) = n$. Im letzten Fall gibt es nur die Möglichkeit $x = 0$. Im ersten Fall gibt es für x die $(q-1)$ Möglichkeiten $x = p, 2p, 3p, \ldots, (q-1)p$. Analog gibt es im zweiten Fall $(p-1)$ Möglichkeiten. Insgesamt gibt es also $1 + (q-1) + (p-1) = p + q - 1$ Möglichkeiten, dass gilt $\mathrm{ggT}(x, n) \neq 1$. Da x zufällig gewählt war, ist die Wahrscheinlichkeit P, dass gilt $\mathrm{ggT}(x, n) \neq 1$:

$$P = \frac{p+q-1}{n} < \frac{p+q}{pq} = \frac{1}{p} + \frac{1}{q}$$

□

Das ElGamal-Verfahren

Beim ELGamal-Verfahren sind die Mengen $\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D}$ gegeben durch:

- $\mathcal{P} = \mathbb{Z}/p\mathbb{Z}$, p prim, fest und öffentlich
- $\mathcal{C} = \mathbb{Z}/p\mathbb{Z} \times (\mathbb{Z}/p\mathbb{Z})^*$
- $\mathcal{K}_{Pu} = \{(g, g^a \mod p) \mid g \text{ Primitivwurzel von } (\mathbb{Z}/p\mathbb{Z})^*, a \in \mathcal{K}_{Pr}\}$
- $\mathcal{K}_{Pr} = \{a \mid 1 < a < p-1\}$
- $\mathcal{E} = \{x \mapsto (x(g^a)^k \mod p,\ g^k \mod p) \mid (g, g^a) \in \mathcal{K}_{Pu}, k \in \mathbb{N}\}$
- $\mathcal{D} = \{(x, y) \mapsto xy^{-a} \mod p \mid a \in \mathcal{K}_{Pr}\}$

Der private Schlüssel besteht hier also aus einem Element $a \in (\mathbb{Z}/p\mathbb{Z})^*$, aus welchem dann der öffentliche Schlüssel, das Tupel $(g, g^a \mod p)$, wobei g eine Primitivwurzel von $(\mathbb{Z}/p\mathbb{Z})^*$ ist, berechnet wird. Die Verschlüsselung verläuft nun nach dem folgenden Schema:

1. A besorgt sich den öffentlichen Schlüssel $(g, g^{a_B} \mod p)$ von B, wählt zufällig eine natürliche Zahl k und verschlüsselt den Klartext
   $P \mapsto (P(g^{a_B})^k \mod p,\ g^k \mod p)$. Das Ergebnis sendet er an B.
2. B entschlüsselt nun die Botschaft, indem er die zweite Komponente des von A erhaltenen Tupels invertiert und in die a_B-te Potenz erhebt und das Ergebnis mit der ersten Komponente des Tupels multipliziert:

$$P(g^{a_B})^k (g^k)^{-a_B} \equiv P g^{a_B k - a_B k} \equiv P \equiv P \mod p$$

Satz 1.6.7 *Das ElGamal-Verfahren ist ein Public-Key-Chiffre.*

Beweis. Wir haben bereits oben nachgerechnet, dass für jedes Paar $(f_k, g_k) \in \mathcal{E} \times \mathcal{D}$ gilt $(f_k \circ g_k)(P) = P$. Ein Angreifer, der in der Lage ist, diskrete Logarithmen in $(\mathbb{Z}/p\mathbb{Z})^*$ zu lösen, kann damit das ElGamal-System brechen, da er aus den Komponenten des öffentlichen Schlüssels den geheimen Schlüssel berechnen könnte. Gäbe es einen Weg, um mit dem Wissen von g^k und g^a auch g^{ka} berechnen zu können, könnte man das

ElGamal-Verfahren ebenfalls brechen, indem man $(g^{ak})^{-1}$ bildet und $Pg^{ak}g^{-ak} \equiv P \mod p$ bestimmt, ohne diskrete Logarithmen bestimmen zu können. Es wird aber vermutet, dass es hierzu kein Verfahren gibt. Da es bei großen Zahlen praktisch unmöglich ist, diskrete Logarithmen zu bestimmen, ist eine ElGamal-Verschlüsselung nicht ohne Wissen des privaten Schlüssels zu entschlüsseln. Das ElGamal-Verfahren ist also ein Public-Key-Chiffre. □

Das Massey-Omura-Verfahren

Beim Massey-Omura-Verfahren sind die Mengen $\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D}$ gegeben durch:

- $\mathcal{P}, \mathcal{C} = \mathbb{Z}/p\mathbb{Z}$, p prim und öffentlich.
- $\mathcal{K} = \{e \mid 1 < e < p-1, \text{ggT}(e, p-1) = 1\}$
- $\mathcal{E} = \{x \mapsto x^e \mod p \mid e \in \mathcal{K}\}$
- $\mathcal{D} = \{x \mapsto x^d \mod p \mid d \in \mathcal{K}\}$

Das Massey-Omura-Verfahren ist kein Public-Key-Verfahren im obigen Sinne, da in diesem Verfahren jeder Teilnehmer ein privates Schlüsselpaar $(e, d) \in \mathcal{K} \times \mathcal{K}$ mit $ed \equiv 1 \mod (p-1)$ besitzt. Die Verschlüsselung verläuft dann nach dem folgenden Schema:

1. A verschlüsselt $P \in \mathcal{P}$ durch $P \mapsto P^{e_a} \mod p$ und sendet das Ergebnis an B.
2. B verschlüsselt nun das von A Erhaltene durch $P^{e_a} \mapsto (P^{e_a})^{e_b} \mod p$ und sendet das Ergebnis an A.
3. A entschlüsselt nun das von B Erhaltene mit seinem Schlüssel

   $$d_a : \ (P^{e_a})^{e_b} \mapsto (P^{e_a e_b})^{d_A} \equiv (P^{e_b})^{e_a d_a} \equiv P^{e_b} \mod p$$

   und sendet das Ergebnis an B.
4. B kann das Ergebnis nun mit seinem Schlüssel d_b entschlüsseln und erhält den Klartext P.

Eine Unsicherheit dieses Verfahrens ist, dass A sich nicht sicher sein kann, wirklich mit B und nicht mit einem Fremden zu kommunizieren. So könnte ein dritter die Kommunikation abhören und die Nachricht mit seinem Schlüssel versehen an A zurücksenden. A würde dann seinen Schlüssel entfernen und der dritte könnte die Botschaft lesen. Um diesem vorzubeugen benötigt man ein *digitales Signaturschema* und darauf soll im nächsten Abschnitt kurz eingegangen werden.

1.6.1 Digitale Signatur

Eine digitale Signatur ist so etwas wie eine persönliche Unterschrift einer digitalen Mitteilung, mit deren Hilfe sich der Empfänger der Mitteilung über den Absender sicher sein kann. Eine Möglichkeit der digitalen Signatur bieten die Public-Key-Kryptosysteme:

Seien $e_{k_{Pu,A}}$, $e_{k_{Pu,B}}$ die Entschlüsselungsfunktionen zu den öffentlichen Schlüsseln $k_{Pu,A}$, $k_{Pu,B}$ eines Public-Key-Kryptosystems der Teilnehmer A und B sowie $k_{Pr,A}$,

$k_{Pr,B}$ und $d_{k_{Pr,A}}$, $d_{k_{Pr,B}}$ die dazugehörigen privaten Schlüssel bzw. Verschlüsselungsfunktionen. Möchte A nun eine Nachricht x an B senden, die zwar jeder lesen darf, aber B sicher sein soll, dass diese Nachricht wirklich von A stammt, dann berechnet A $d_{k_{Pr,A}}(x)$ und sendet das Ergebnis zusammen mit der Nachricht x an B. B kann nun die Unterschrift von A überprüfen, indem er $(e_{k_{Pu,A}} \circ d_{k_{Pr,A}})(x)$ berechnet und das Ergebnis mit x vergleicht. Stimmen die Ergebnisse überein, so kann sich B sicher sein, dass die Nachricht von A stammt, da nur A mit Hilfe des privaten Schlüssels $k_{Pr,A}$ in der Lage ist, eine Funktion $d_{k_{Pr,A}}$ zu bestimmen, so dass $e_{k_{Pu,A}} \circ d_{k_{Pr,A}} = \text{ id}$ gilt.

Verschlüsseln und Signieren verläuft dann nach dem folgenden Schema.

1. A berechnet $c_1 = (d_{k_{Pr,A}} \circ e_{k_{Pu,B}})(x)$ sowie $c_2 = e_{k_{Pu,B}}(x)$ sendet beides an B.
2. B berechnet nun $(d_{k_{Pr,B}} \circ e_{k_{Pu,A}})(c_1) = x$ und $d_{k_{Pr,B}}(c_2) = x$. Stimmen die beiden Ergebnisse überein, so kann er sich sicher sein, dass die verschlüsselte Nachricht von A stammt.

Dieses Verfahren ist allerdings sehr rechenaufwendig. In der Praxis werden sogenannte *Hashfunktionen* benutzt. Das sind Funktionen, die Strings beliebiger Länge auf Strings vorgegebener Länge abbilden. Ein einfaches Beispiel einer Hasfunktion ist

$$\begin{aligned} h : \mathbb{N} &\to \mathbb{Z}/2\mathbb{Z} \\ n &\mapsto n_0 + n_1 + \cdots + n_{k-1} \mod 2, \end{aligned}$$

wobei $\sum_{i=0}^{k-1} n_i 2^i$ die Binärdarstellung der natürlichen Zahl n ist. Bei Hashfunktionen, die in der Kryptographie verwendet werden, sollte es zudem praktisch unmöglich sein, zu einem Hashwert $h(x)$ das Urbild x bestimmen zu können. Da die Bildmenge einer Hashfunktion sehr viel kleiner ist, als der Definitionsbereich, kann diese nicht injektiv sein. Trotzdem wird von einer Hashfunktion erwartet, dass es praktisch unmöglich ist, zwei Elemente x_1, x_2 zu finden, deren Hashwerte $h(x_1), h(x_2)$ gleich sind. Statt die komplette Nachricht mit seinem privaten Schlüssel zu signieren, reicht es nun, den Hashwert der Nachricht zu signieren. A sendet also neben $e_{k_{Pu,B}}(x)$ den Wert $d_{k_{Pr,A}}(h(x))$ an B, wobei h eine öffentliche Hashfunktion ist. B kann nun seinerseits den Hashwert $h(x)$ aus der Nachricht bestimmen und diesen mit dem Hashwert von A vergleichen. Da es praktisch unmöglich sein sollte, zwei Elemente zu finden, deren Hashwerte gleich sind, kann sich B bei Übereinstimmung der Werte sicher sein, dass die Nachricht von A stammt.

1.6.2 Der Diffie-Hellman Schlüsselaustausch

Da die Public-Key-Verfahren im Vergleich zu den klassischen Verfahren sehr rechenaufwendig sind, werden diese in der Praxis häufig nur zu der Übermittlung des Schlüssels eines klassischen symmetrischen Verfahrens, welcher in der Regel kürzer als die eigentlichen Nachrichten ist, eingesetzt. Eine weitere Möglichkeit, den Schlüssel des klassischen Verfahrens zu erzeugen und beiden Teilnehmern zugänglich zu machen, ist der Diffie-Hellman-Schlüsselaustausch:

1. A und B einigen sich auf eine Primzahl p und eine Primitivwurzel g von $(\mathbb{Z}/p\mathbb{Z})^*$.
2. A und B wählen nun a bzw. $b \in (\mathbb{Z}/p\mathbb{Z})^*$ und berechnen $K_a = g^a \mod p$ bzw. $K_b = g^b \mod p$ und senden ihr Ergebnis an der anderen.

3. A und B berechnen nun $K = (K_b)^a \equiv g^{ab} \mod p$ bzw. $K = (K_a)^b \equiv g^{ab} \mod p$. Der gemeinsame Schlüssel ist also K.

Die Sicherheit dieses Verfahrens liegt wie bei der ElGamal-Verschlüsselung in der Schwierigkeit diskrete Logarithmen, also den Wert von a bzw. b aus $g^a \mod p$ bzw. $g^b \mod p$ zu berechnen.

2 Zahlentheorie und Kryptographie im Mathematikunterricht

2.1 Einleitung

„Die Mathematik ist die Königin der Wissenschaften,
die Zahlentheorie ist die Königin der Mathematik.“
C.F. Gauss

Neben der Analysis, der linearen Algebra, der analytischen Geometrie und der Stochastik geht die Zahlentheorie im Mathematikunterricht insbesondere in der Sekundarstufe II ein wenig unter. In der Primarstufe und der Sekundarstufe I werden den Schülern die Regeln der Teilbarkeit und einige Merkregeln hierzu beigebracht, welche dann in der Bruchrechnung ihre erste und einzige Anwendung finden.

Diese Arbeit zeigt auf, wie sich Zahlentheorie in Form eines abgeschlossenen Themas in der Sekundarstufe II unterrichten läßt. Es wird eine Unterrichtseinheit über 10 Doppelstunden beschrieben, welche als Teil dieser Arbeit mit großem Erfolg in einem Leistungskurs der Jahrgangsstufe 12 durchgeführt worden ist.

Diese Unterrichtseinheit kann auch im Rahmen eines Kurses zur *diskreten Mathematik*, wie er jüngst in mehreren Bundesländern im Lehrplan eingegliedert wurde, mit einem Schwerpunkt auf die Verschlüsselung und Codierung eingesetzt werden.

Thema der Unterrichtseinheit ist die Kryptographie vom antiken Caesar-Chiffre bis zum aktuellen RSA-Verfahren. Erfahrungsgemäß (siehe hierzu auch [Puh98], [Som98], [Sch94]) ist bei diesem Thema eine hohe intrinsische Motivation seitens der Schüler zu beobachten, die vermutlich mit dem Bezug auf aktuelle Probleme aus der Lebenswelt der Schüler (Sicherheit im Internet, bei der drahtlosen Kommunikation,etc.) zu erklären ist.

Das Thema eignet sich gerade durch den historisch-genetischen Zugang sehr gut für einen *produktiven* Mathematikunterricht (siehe hierzu [Jah00]) in welchem die Schüler[1] mit Hilfe von produktiven Aufgaben, die sowohl inner- als auch außermathematisch relevant sind, selbstentdeckend lernen können.

Im Verlauf des Unterrichts wird darauf Wert gelegt, dass die Schüler, angeleitet durch Arbeitsblätter, selbstständig Vermutungen aufstellen und diese dann anhand paradigmatischer Beispiele, also solchen Beispielen, an denen bereits das Allgemeine erkannt werden kann, verifizieren und zu begründen versuchen. In vielen Fällen erübrigt sich damit ein strenger Beweis, welcher den Schülern keine neuen Erkentnisse, bis auf die Beweisstrategien und -methoden selber, welche aber hier nicht das Thema sind, liefern würde.

[1] Im Folgenden wird stets die allgemeine, männliche Form geschlechterneutral verwendet

2.2 Darstellung der Unterrichtseinheit

2.2.1 Das Caesar-Chiffre und Rechnen mit Restklassen

Inhalt

Nach einem Brainstorming zum Thema Kryptographie, bei dem die Bedeutung der Kryptographie in der Geschichte (Hinrichtung von Maria Stuart, Enigma im 2.Weltkrieg, Entscheidungen über Leben und Tod) und heute (e-commerce, Kommunikation im Internet, mobile Kommunikation) gegenübergestellt wird, werden die Schüler mit einem einfachen cäsar-verschlüsselten Text

„Nrpph khxwh xp guhl “

konfrontiert, welchen die Schüler zu entschlüsseln versuchen sollen. Dieser Text sollte nicht zu einfach zu erraten sein, damit die Schüler auf eine Häufigkeitsanalyse angewiesen sind. Unter der Annahme, das bei der Verschlüsselung jeder Buchstabe um dieselbe Stellenzahl verschoben wurde, gibt die Häufigkeitsanalyse H $\leftrightarrow$ E und damit G $\leftrightarrow$ D, F $\leftrightarrow$ C usw. Als *Schlüssel* wollen wir den Buchstaben bezeichnen, auf welchen das „A“ durch die Verschlüsselung abgebildet wird, also in diesem Fall das „D“ , da D $\leftrightarrow$ A.

Es stellt sich nun die Frage, wie man eine solche Verschlüsselungsfunktion einfach aufschreiben kann. Bezeichnen wir die Menge der Buchstaben des Alphabets mit $\Sigma_{26} := \{A,B,C,D,E,F,...,Z\}$, dann kann man die Verschlüsselungsfunktion

$$f : \Sigma_{26} \longrightarrow \Sigma_{26}$$

einfach angeben als Tabelle:

A	B	C	D	E	F	G	H	I	J	K	L	...
D	E	F	H	H	I	J	K	L	M	N	O	...

Die nächste Aufgabe ist es, eine geschlossene Darstellung dieser Funktion der Art $f(x) = ...$ zu finden, mit welcher man zum Beispiel auch einen Computer programmieren könnte. Hierbei sollen die Schüler in den Begriff der Restklasse eingeführt werden und lernen mit diesen zu rechnen. Im Vergleich mit anderen Vorschlägen zur Behandlung dieses Themas [Puh98], [Som98], [Sch94] wird in dieser Arbeit ein größerer Wert auf den Begriff der Restklasse gelegt, da dieser eine wichtige Grundlage zum Verständnis der modernen Algorithmen in der Kryptographie ist und auch in anderen Bereichen der Mathematik aber auch im täglichen Leben wie z.B. in der Musik [Joh01] oder im Kalender eine Rolle spielt.

Um die Verschlüsselungsfunktion f geschlossen darzustellen, ist es nötig, die Buchstaben in Zahlen zu codieren. Eine mögliche sinnvolle Codierung wäre z.B. A $\mapsto$ 0, B $\mapsto$ 1, C $\mapsto$ 2, usw.

An dieser Stelle wollen wir auf den Unterschied zwischen *Verschlüsseln* und *Codieren* hinweisen. Während sich die Kryptographie mit der Geheimhaltung von Daten beschäftigt, ist die Aufgabe der Codierung zunächst einmal die Daten für die weitere Verarbeitung vorzubereiten, z.B. unsere Codierung (Texte $\leftrightarrow$ Zahlen), der ASCII-Code als Beispiel für eine Binär-Codierung, der EAN-Code oder der ISBN-Code. Eine gute Codierung bietet zudem häufig Sicherheit gegen Verabeitungsfehler, die z.B. durch Störungen in der Übertragung zustande kommen können. Beispiele hierfür sind der

oben genannte EAN- oder der ISBN-Code, bei denen die letzte Ziffer, die sogenannte Prüfziffer, aus den restlichen Ziffern berechnet wird. Um dies zu verdeutlichen, bietet sich an dieser Stelle ein etwa einstündiger Exkurs zum EAN-Code an. Siehe z.B. [Her00]

Mit Hilfe unserer Codierung können wir die obige Verschlüsselungsfunktion geschlossen darstellen:

$$f(x) = x + 3,$$

da D $\mapsto 3$, wobei $x \in \mathbb{Z}, x \leq 25$. Wir wollen uns an dieser Stelle noch keine weiteren Gedanken über den Definitions- bzw. den Wertebereich dieser Funktion machen, sondern erst einmal den Beispielsatz „Schluss mit lustig“ mit dem Schlüssel „R“ verschlüsseln. Die Schüler bekommen hierzu ein Arbeitsblatt, auf dem sie den Satz zunächst mit Hilfe der Tabelle, und danach mit Hilfe der geschlossenen Form der Verschlüsselungsfunktion verschlüsseln sollen. Sie erhalten damit das folgende Schema:

$$\begin{array}{ccc} \text{SCHLUSS MIT LUSTIG} & \overset{codieren}{\longrightarrow} & \text{18 02 07 11 20 18 18 12...} \\ \downarrow f_{Tabelle} & & \downarrow f_{geschlossen} \\ \text{JTYCLJJ DZK CLJZKX} & & \text{35 19 24 28 37 35 35 29...} \end{array}$$

Die Schüler erhalten also zunächst zwei Chiffretexte - den codierten und den nichtcodierten. Diese sollten bei demselben Schlüssel übereinstimmen, also das Codewort 35 19 24 28 37 35 35 29... für den Chiffretext JTYCLJJ DZK CLJZKX stehen.
An dieser Stelle stellen die Schüler fest, dass ein Buchstabe durch mehrere Zahlen repräsentiert werden kann, und erstellen eine Tabelle, in welcher zu jedem Buchstaben die dazugehörigen Zahlen aufgeführt sind:

A	B	C	D	...	Z
0, 26, 52, ...	1, 27, ...	2, 28, ...	3, 29, ...	...	25, 51, ...

Die Schüler beschäftigen sich weiterhin mit den Fragen

1. Wie viele Zahlen einen Buchstaben repräsentieren? - *Unendlich viele*

2. Repräsentiert jede positive ganze Zahl einen Buchstaben? - *Ja*

3. Wie kann man herausfinden, welchen Buchstaben eine ganze positive Zahl repräsentiert? - *Mit Teilen durch 26 mit Rest.*

4. Wie kann man alle Zahlen erfassen, die den Buchstaben „B“ repräsentieren? - $1 + 26n$ *mit* $n \in \mathbb{Z}$*, denn diese Zahlen haben alle den Rest 1.*

Diese Fragen sollten die Schüler beantworten können und damit zu der ersten Definition kommen:

Definition 2.2.1 (spezielle Definition von Klasse) *Die Menge aller Zahlen, die einen Buchstaben repräsentieren nennen wir „Klasse“ des Buchstaben. Z.B. Klasse A* $= \{0, 26, 52, ..., \}$*. Statt Klasse A schreiben wir auch* $[A]_{26}$*, wobei der Index 26 bedeutet, dass es insgesamt 26 Klassen gibt.*

Mit Hilfe des eben Erarbeiteten können wir nun schreiben: $[\mathrm{A}]_{26} = \{26n |\ n \in \mathbb{Z}\}$, $[\mathrm{B}]_{26} = \{1 + 26n |\ n \in \mathbb{Z}\}$, usw. Die Menge aller Klassen nennen wir

$$\mathbb{Z}/26\mathbb{Z} = \{[\mathrm{A}]_{26}, [\mathrm{B}]_{26}, \ldots, [\mathrm{Z}]_{26}\}.$$

Die Schüler haben festgestellt, dass die Elemente einer Klasse durch Teilen mit Rest entstehen. Deshalb wollen wir diese Klassen auch als *Restklassen* bezeichnen. Weiterhin wollen wir festhalten, dass jede Restklasse unendlich viele Zahlen enthält.

Als ein anderes Beispiel für Klassen betrachten wir nun die rationalen Zahlen. Hier besitzt jeder Bruch unendlich viele Darstellungen, welche durch Erweitern entstehen. Z.B. besteht die Klasse [2] aus allen Brüchen, die den Wert 2 besitzen. $[2] = \{\frac{2}{1}, \frac{4}{2}, \frac{6}{3}, ...\}$.

Wir haben nun eine Einteilung der positiven ganzen Zahlen in 26 Klassen gefunden. $\mathbb{Z}^+ = [\mathrm{A}]_{26} \cap [\mathrm{B}]_{26} \cap \cdots \cap [\mathrm{Z}]_{26}$. (Diese Tatsache ist den Schülern sofort klar. Jeder Beweis in die Richtung, dass Äquivalenzklassen auf einer Menge M eine Partition von M implizieren, würde diese einfache Tatsache verschleiern). Allerdings hängen die Namen der Klassen von der Codierung ab. Wählt man z.B. eine Codierung A $\mapsto$ 1, B $\mapsto$ 2, usw. dann ist $[\mathrm{A}]_{26} = \{1, 27, 53, ...\}$. Eine Idee (auf die auch die Schüler kommen) wäre es, die Klassen nach ihren kleinsten Elementen (also den kleinsten positiven Resten, die nach dem Teilen mit Rest durch 26 entstehen) zu bezeichnen: $[0]_{26} = \{0, 26, 52, \ldots\}$, $[1]_{26} = \{1, 27, 53, \ldots\}$, $\ldots$, $[25]_{26} = \{25, 51, 76 \ldots\}$. Wir können nun eine allgemeinere Definition der Restklasse geben:

Definition 2.2.2 (Restklasse) *Mit $[a]_n$ bezeichnen wir die Menge aller Zahlen, die beim Teilen mit Rest durch n denselben ganzzahligen Rest $\tilde{a}$ lassen, wobei $\tilde{a}$ das kleinste positive Element dieser Klasse ist, und nennen diese Menge Restklasse modulo n.*

Die Schüler sollen diese Gedankengänge nun selbstständig für die Mengen $\mathbb{Z}/7\mathbb{Z}$ und $\mathbb{Z}/7\mathbb{Z}$ durchführen, Anwendungen für diese Mengen im täglichen Leben suchen und sich überlegen, in welche Klassen die negativen ganzen Zahlen gehören.

Anwendungen findet man für $\mathbb{Z}/7\mathbb{Z}$ z.B. in der Uhr, den Monaten oder der Musik [Joh01], für $\mathbb{Z}/7\mathbb{Z}$ in den Wochentagen. Als Beispiel kann man den Schülern die folgende Aufgabe stellen, die sich mit Hilfe der Restklassen ohne zu große Gehirnakrobatik lösen lässt und eine Motivation in das Thema Rechnen mit Restklassen ist.

Was war einen Tag vor vorgestern, wenn zwei Tage nach übermorgen Donnerstag ist?

(aus „Das Quiz“)
Lösung: Sei $[x]_7$ der heutige Tag. Gesucht ist dann $[y]_7 = [x]_7 - 1 - 2 = [x]_7 - 3 \Leftrightarrow [x]_7 = [y]_7 + 3$. Zwei Tage nach übermorgen ist Donnerstag:

$$[x]_7 + 2 + 2 \overset{(1)}{=} [x]_7 + 4 \overset{(2)}{=} [y]_7 + 7 \overset{(3)}{=} [y]_7 \overset{!}{=} [3]_7$$

Der gesuchte Tag ist also Donnerstag und heute ist demnach Sonntag.

Von den Gleichheitszeichen der Gleichungskette ist vor allem das (3) erklärungsbedürftig. Allgemein stellt sich die Frage: Wie rechnet man mit Restklassen?

Die Schüler bekommen vermischte Aufgaben, in denen sie Restklassen mit Ganzen Zahlen addieren und multiplizieren sollen, indem sie verschiedene Elemente der Klassen mit ganzen Zahlen addieren und subtrahieren und prüfen, in welcher Klasse jeweils das Ergebnis liegt. So bekommen sie heraus, dass das Ergebnis unabhängig vom gewählten

Repräsentanten der Klasse ist. Weiterhin sollen sich die Schüler anhand dieses Ergebnisses überlegen, dass man zwei Restklassen miteinander addieren bzw. multiplizieren kann, indem man je einen Komponenten jeder Klasse mit dem einer anderen Klasse addiert oder multipliziert.

Diese Ergebnisse werden nun mit den Schülern noch einmal gemeinsam an paradigmatischen Beispielen nachgerechnet, da die Repräsentantenunabhängigkeit der Grundrechenarten eine wichtige Grundlage der Kongruenzrechnung ist. Stellen wir uns den Aufgabe $[2]_7 + 10$ und $[2]_7 \cdot 3$. Zum Berechnen dieser Aufgabe nehmen wir uns ein beliebiges Element $x \in [2]_7$. Da alle Elemente der Restklasse $[2]_7$ gemeinsam haben, dass sie nach Teilen mit Rest durch 7 den Rest 2 lassen, ist x von der Form $x = 2 + 7k$ mit $k \in \mathbb{Z}$. Wir können also schreiben:

$$\begin{aligned} x + 10 &= 2 + 7k + 10 = (2 + 10) + 7k = 12 + 7k \\ &= 2 + (3 + 7) + 7k = (2 + 3) + 7 \cdot (k + 1) = 5 + 7\tilde{k} \\ x \cdot 3 &= (2 + 7k) \cdot 3 = 6 + 21k \end{aligned}$$

Das Ergebnis ist in beiden Fällen unabhängig vom gewählten Element x!
Weiterhin liefert uns die erste Zeile ein Ergebnis von der Form $12 + 7k$. Dieses lässt nach der zweiten Zeile beim Teilen mit Rest durch 7 denselben Rest wie $5 + 7\tilde{k}$. Es liegt also in der Restklasse $[5]_7$. Wir haben damit zum einen gezeigt, dass

$$[2]_7 + 10 = [2 + 10]_7 \text{ und } [2]_7 \cdot 3 = [2 \cdot 3]_7$$

(wir hätten für x also am einfachsten das kleinste positive Element der Klasse, 2, wählen sollen) und zum anderen nachgerechnet, dass $[12]_7 = [5]_7$ gilt, also die Klassen zweier Elemente gleich sind, wenn sich die Elemente selbst jeweils in der Klasse des anderen Elements befinden.

Weiterhin fällt auf, dass $[2]_7 + 10$ dasselbe wie $[2]_7 + 3$ ist. Damit liegt die Vermutung nahe, dass auch $[2]_7 + 17$ dasselbe Ergebnis liefert.

$$[2]_7 + 17 = [2 + 17]_7 = [19]_7 = [5]_7$$

Da $3, 10, 17 \in [3]_7$ kommt man auf die Idee, dass $[2]_7 + [3]_7 = [2 + 3]_7 = [5]_7$ gilt.

Dieses Ergebnis kann nun mit den Schülern ebenfalls nach demselben Schema wie oben nachgerechnet werden.

Für den allgemeinen Fall stellen wir also die folgende Vermutung auf:

Vermutung 1 *Seien $a, b, c, n \in \mathbb{Z}$. Dann gilt*

1. $[a]_n = [b]_n \Leftrightarrow (a \in [b]_n$ *und* $b \in [a]_n)$
2. $[a]_n + c = [a + c]_n$ *und* $[a]_n \cdot c = [a \cdot c]_n$
3. $[a]_n + [b]_n = [a + b]_n$ *und* $[a]_n \cdot [b]_n = [a \cdot b]_n$

Diesen Aussagen werden die Schüler nach den gerechneten paradigmatischen Beispielen sofort zustimmen, da wir keine spezifische Eigenschaft der Zahlen bei den Beispielrechnungen verwendet haben. Der allgemeine Beweis ist also aus der Sicht der Schüler nicht notwendig, da sie von der Richtigkeit der Aussage bereits überzeugt sind. Da der Beweis keine neuen Zusammenhänge aufzeigt, hätte ein Vorführen der analogen allgemeinen

Rechnung lediglich einen Selbstzweck und soll an dieser Stelle weggelassen werden, damit wir uns wieder den Anwendungen in der Kryptographie zuwenden können.

Am Ende dieses Kapitels definieren wir nun die Kongruenz.

Definition 2.2.3 *Zwei Zahlen $a, b \in \mathbb{Z}$ heißen kongruent modulo n*

$$a \equiv b \mod n,$$

wenn sie sich in derselben Restklasse $[x]_n$ befinden, also wenn sie beim Teilen mit Rest durch n denselben Rest lassen.

Es bleibt die Frage, in welche Klassen die negativen ganzen Zahlen eingeordnet werden. Division mit Rest ist in diesem Fall für die Schüler nicht befriedigend. Logisches Auffüllen von Restklassentabellen lässt vermuten, dass der folgende Zusammenhang gilt:

Vermutung 2 $-a \in [x]_n \Leftrightarrow n|x-(-a) = x+a$

Diese Tatsache kann zusammen mit den Schülern nachgerechnet werden, und liefert die Äquivalenz zu der Definition der Kongruenz, die man in den meisten Zahlentheoriebüchern findet:

Satz 2.2.4 $a \equiv b \mod n \Leftrightarrow n|a-b$

Mit Hilfe der Kongruenzrechnung können wir die obige Verschlüsselungsfunktion schreiben als

$$\begin{array}{rcll} f & : & \mathbb{Z}/26\mathbb{Z} \longrightarrow \mathbb{Z}/26\mathbb{Z} & \\ [x]_{26} & \mapsto & [x]_{26} + a & \text{oder} \\ x & \mapsto & x + a \mod 26, & \end{array}$$

wobei a der Schlüssel ist.

Methoden, Ziele und Erläuterungen

In diesem ersten Abschnitt der Unterrichtseinheit lernen die Schüler in der Kongruenzrechnung eine für sie neue Art der Mathematik kennen. Aus diesem Grund sollte ihnen hier viel Zeit zum Üben der unbekannten Rechentechniken eingeräumt werden. Insgesamt sind für diesen Abschnitt etwa 5-6 Unterrichtsstunden vorgesehen, wobei diese Zahl in Abhängigkeit von den Vorkenntnissen und dem Leistungsvermögen der Schüler etwas variieren kann.

Der Weg zunächst den Begriff der Restklasse und dann den Begriff der Kongruenz einzuführen, wurde unter anderem auch deshalb gewählt, weil einige Schüler bereits Vorkenntnisse zur „Modulo-“ Rechnung haben, welche sich aber zumeist auf das Rechnen mit Rest beziehen und hier eher verwirrend als hilfreich sind. Zudem ist die Motivation durch die Kryptographie gegeben und leicht nachzuvollziehen.

Falls die Schüler es gewohnt sind, eigenständig zu arbeiten, können mit Hilfe geeigneter Aufgaben nahezu die gesamten Inhalte selbstständig bis zur Vermutung 1 erarbeitet werden. Sind die Schüler das selbstständige Arbeiten nicht gewohnt, oder kommen sie

nicht alleine zu den Ergebnissen, sollten diese Schüler umso mehr Aufgaben zum Rechnen bekommen, um sich das Rechnen mit Restklassen anzueignen.

Nach dieser Phase sollen die Schüler wissen, was man unter dem Begriff Kryptographie versteht und das einfache Caesar-Chiffre zum Verschlüsseln benutzen und es mit Hilfe einer Häufigkeitsanalyse entschlüsseln können. Weiterhin sollen die Schüler den Begriff der Restklasse verstanden haben und mit Restklassen rechnen können.

2.2.2 Verschlüsseln durch Multiplikation und das multiplikative Inverse modulo n

Die Schüler haben bis jetzt Verschlüsselungsfunktionen der Form $f(x) \equiv x + a \mod n$ kennen gelernt, die so genannten *Verschiebechiffre*. Da diese sehr einfach zu entschlüsseln sind, betrachten wir nun einen anderen Typ Verschlüsselungsfunktion:

$$\begin{aligned} f_{mult} &: \mathbb{Z}/26\mathbb{Z} \to \mathbb{Z}/26\mathbb{Z} \\ f_{mult}([x]_{26}) &= a \cdot [x]_{26} \end{aligned}$$

Zur Motivation bekommen die Schüler die folgende Aufgabe:
Verschlüssele den Satz „Anna kommt um drei“ mit der Verschlüsselungsfunktion

$$f([x]_{26}) = 3 \cdot [x]_{26}.$$

Anna kommt um drei	$\overset{codieren}{\longrightarrow}$	00 13 13 00 10 14 12 12 19 20 12 03 17 04 08
		$\downarrow f_{mult}$
ANNA EQKKF IK JZMY	$\overset{decodieren}{\longleftarrow}$	00 13 13 00 04 16 10 10 05 08 10 09 25 12 24

An dieser Stelle treten zwei neue Probleme auf:

- ANNA wird durch das Verschlüsseln nicht verändert
- Wie sieht die Entschlüsselungsfunktion aus?

Zum ersten Punkt können wir bis jetzt nur sagen, dass eine Verschlüsselungsfunktion möglichst keinen Buchstaben auf sich selbst abbilden sollte, also keine Fixpunkte haben sollte.

Die Entschlüsselungsfunktion ist, naiv hingeschrieben, von der Form

$$g([x]_{26}) = \frac{[x]_{26}}{3} = \left[\frac{x}{3}\right]_{26}$$

Es stellt sich also die Aufgabe, ein beliebiges Element $[x] \in \mathbb{Z}_{26}$ durch 3 zu teilen. Offensichtlich ist dies bei den Vielfachen von 3 kein Problem. In den anderen Fällen suchen wir eine Lösung der Gleichung

$$[z]_{26} = \frac{[x]_{26}}{3} \Leftrightarrow 3 \cdot [z]_{26} = [x]_{26} \Leftrightarrow 26k = 3z - x$$

Für $x = [4]_{26}$ suchen wir z.B. $k, z \in \mathbb{Z}$, so dass die Gleichung $26k = 3z - 4$ erfüllt ist. Probieren ergibt als Lösung: $k = 1$ und $z = 10$. Also ist $3 \cdot [10]_{26} = [4]_{26} \Leftrightarrow [10]_{26} = [\frac{4}{3}]_{26}$.
Die Schüler bekommen nun ein Arbeitsblatt, auf dem sie einen kurzen Text entschlüsseln

und dabei diese Überlegungen noch einmal wiederholen sollen. Zusätzlich bekommen sie Aufgaben, in denen Gleichungen von der Form

$$a \cdot x \equiv 1 \mod n \text{ bzw. } [a]_n[b]_n = [1]_n$$

gelöst, und die Lösbarkeit dieser Aufgabe mit dem $\mathrm{ggT}(a, n)$ verglichen werden sollen. Damit kommen wir zunächst zu der

Definition 2.2.5 *Ein Element $[b]_n \in Z/n\mathbb{Z}$ heißt multiplikatives Inverses oder Kehrwert von $[a]_n \in Z/n\mathbb{Z}$, wenn gilt $[b]_n \cdot [a]_n = [1]_n$. Statt $[b]_n$ schreibt man auch $[\frac{1}{a}]_n$ oder $[a]_n^{-1}$.*

Diese Defintion ist den Schülern bereits von den rationalen Zahlen bekannt. Im Falle der rationalen Zahlen besitzt jede Zahl ungleich null einen Kehrwert. Die Aufgaben des letzten Arbeitsblattes lassen uns das Folgende vermuten:

Vermutung 3 *Das* multiplikative Inverse *oder der* Kehrwert *eines Elementes $[a]_n \in \mathbb{Z}/n\mathbb{Z}$ existiert genau dann, wenn a und n teilerfremd sind.*

Auf einen formalen Beweis und die Unabhängigkeit vom Repräsentanten a wollen wir an dieser Stelle verzichten, aber eine geometrische Plausibilitätsbetrachtung könnte mit den Schülern durchgeführt werden: Die Forderung $[a]_n[b]_n = [1]_n$ ist äquivalent zu der Gleichung $a \cdot b + k \cdot n = 1$, $k \in \mathbb{Z}$. Stellen wir uns nun a und n als Strecken der Länge a bzw. n vor und versuchen, Vielfache dieser Strecken aneinander zu legen, so dass sich die Einheitsstrecke ergibt. (Dieses kann und sollte von den Schülern ausprobiert werden.) Die Schüler werden erkennen, dass dies nur funktionieren kann, wenn a und n teilerfremd sind, und dass die kleinste Strecke, die man auf diese Weise legen kann die Länge $\mathrm{ggT}(a, n)$ hat.

Für das Multiplikationschiffre bedeutet diese Ergebnis, dass jede Verschlüsselungsfunktion der Form $f([x]_n) = [a]_n[x]_n$ mit $\mathrm{ggT}(a, n) = 1$ eine Entschlüsselungsfunktion der Form $g([x]_n) = [a]_n^{-1}[x]_n$ mit $[a]_n[a]_n^{-1} = [1]_n$ besitzt. Denn dann gilt

$$g(f([x]_n)) = [a]_n^{-1} \cdot [a]_n \cdot [x]_n = [x]_n$$

Statt zu teilen, multiplizieren wir hier mit dem Kehrwert. Es gibt also auch an dieser Stelle wieder eine Analogie zum Rechnen mit rationalen Zahlen.

Die Schüler bekommen nun die Aufgabe, obigen Text mit der Verschlüsselungsfunktion $f([x]_{26}) = 4[x]_{26}$ zu verschlüsseln. Sie werden feststellen, dass diese nicht eindeutig ist, da z.B. A $\mapsto$ A und N $\mapsto$ A. Wir halten also fest, dass eine Verschlüsselungsfunktion eindeutig oder *injektiv* sein sollte, da der Empfänger der Nachricht diese sonst nicht eindeutig entschlüsseln könnte. Von einer formalen Definition der Injektivität wollen wir absehen, da diese für die Schüler wenig erhellend ist. Allerdings könnten einige Beispiele bekannter reeller Funktionen die Vorstellung eindeutiger Funktionen festigen.

Die Schüler werden beim Legen der Strecken festgestellt haben, dass es im Fall $\mathrm{ggT}(a, n) \neq 1$ immer möglich ist, die Nullstrecke zu legen. Das heißt es gibt ein $k \in \mathbb{Z}$ so dass $ab + kn = 0 \Leftrightarrow ab \equiv 0 \mod n$, oder $[a]_n[b]_n = 0$. Ist dies der Fall, so nennt man $[a]_n$ und $[b]_n$ *Nullteiler*. Der in der Schule häufig vorkommende Schluss $[a]_n[b]_n = [0]_n \Rightarrow [a]_n = [0]_n \vee [b]_n = [0]_n$ ist hier also nicht möglich, wenn $\mathrm{ggT}(a, n) \neq 1$ ist.

Weiterhin ist eine Verschlüsselungsfunktion $f([x]_n) = a[x]_n$ nicht eindeutig, wenn $\text{ggT}(a, n) = t > 1$ ist. Denn dann wird das Element $[\tilde{n}]_n := \left[\frac{n}{t}\right]_n$ auf $[0]_n$ abgebildet:

$$a \cdot [\tilde{n}]_n = \left[a \cdot \frac{n}{t}\right]_n = \frac{a}{t} \cdot [0]_n = [0]_n$$

Wir haben also herausgefunden, dass es Verschlüsselungsfunktion der Form $f([x]_n) = a[x]_n$ nicht eindeutig ist, wenn $\text{ggT}(a, n) \neq 1$ ist. Gilt hingegen $\text{ggT}(a, n) = 1$, dann ist f umkehrbar, mit der Umkehrfunktion $g([x]_n) = [a]_n^{-1}[x]_n$, wobei $[a]_n^{-1}$ das multiplikative Inverse modulo n von $[a]_n$ ist. In diesem Fall ist f auch eindeutig.

Als nächstes beschäftigen wir uns mit der Frage, ob ein multplikatives Chiffre sicherer als ein Verschiebechiffre ist. Die Schüler bekommen hierzu einen Text, der mit einem Multipliaktionschiffre verschlüsselt wurde, und den sie ohne Wissen des Schlüssels entschlüsseln sollen. Dieses geschieht in drei Schritten:

1. Häufigkeitsanalyse zur Ermittlung des Schlüssels a
2. Berechnung des multiplikativen Inversen $[a]_{26}^{-1}$ von $[a]_{26}$ und Aufstellen der Entschlüsselungsfunktion $g([x]_{26}) = [a]_{26}^{-1} \cdot [x]_{26}$
3. Entschlüsseln des Textes mit Hilfe der Entschlüsselungsfunktion g

Wir erkennen, dass diese Verschlüsselung rechnerisch nicht so einfach zu entschlüsseln ist, wie das Verschiebechiffre. Sicher ist sie aber trotzdem nicht, da man sie mit Hilfe einer kompletten Häufigkeitsanalyse (alle Buchstaben werden durch ihre Häufigkeiten und Probieren entschlüsselt) immer noch mit kleinem Aufwand entschlüsseln kann.

Allgemein wollen wir jetzt den Begriff des *Permutationschiffre* einführen:

Definition 2.2.6 *Eine eineindeutige Funktion von einer Menge mit n Elementen in sich selbst heißt Permutation.*

Beispiele :

- Die beiden bekannten Funktionen des Verschiebe- und Multiplikationschiffre :
  $f : \mathbb{Z}/26\mathbb{Z} \longrightarrow \mathbb{Z}/26\mathbb{Z},\ f([x]_{26}) = a + [x]_{26}$
  $f : \mathbb{Z}/26\mathbb{Z} \longrightarrow \mathbb{Z}/26\mathbb{Z},\ f([x]_{26}) = a \cdot [x]_{26}$, mit $\text{ggT}(a, 26)=1$
- Aber auch Funktionen, die nicht geschlossen darstellbar sind, wie

$$f : \mathbb{Z}/26\mathbb{Z} \longrightarrow \mathbb{Z}/26\mathbb{Z},$$

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	...
E	R	I	K	N	H	A	U	S	Z	Y	X	W	V	T	Q	...

  Der Schlüssel zu diesem Chiffre ist „ErikEinhaus“ , genauso könnten die Buchstaben aber auch wild durcheinander gewürfelt sein.

  Wie wir oben schon festgestellt haben, sollte bei einer Verschlüsselung durch eine Permutation darauf geachtet werden, dass ein Buchstabe nicht unverändert bleibt.

  Bei einer Häufigkeitsanalyse reicht es nun nicht mehr aus, nur einen Buchstaben herauszubekommen, sondern man muss alle Buchstaben nach ihren Häufigkeiten finden.

Ergebnis: Jedes Permutationschiffre lässt sich mit Hilfe einer Häufigkeitsanalyse ohne Wissen des Schlüssels entschlüsseln.

Methoden, Ziele und Erläuterungen

Kern dieses zweiten Abschnittes ist die Einführung des Begriffes des multiplikativen Inversen eines Elementes $a \in \mathbb{Z}/n\mathbb{Z}$ und die Überlegung, wann dieses existiert. Für diesen Teil sind etwa drei Unterrichtsstunde vorgesehen, in denen die Schüler zum großen Teil selbstständig Aufgaben bearbeiten sollen. Die Erfahrung zeigt, dass die Schüler nicht von selbst auf die Idee kommen, den größten gemeinsamen Teiler als Hilfsinstrument für die Existenz des Inversen zu betrachten, sondern andere meistens kompliziertere Regeln aufstellen. Da Betrachtungen mit Hilfe des größten gemeinsames Teilers im Folgenden häufiger auftauchen, sollten die Schüler schon an dieser Stelle daran gewöhnt werden. Wenn nötig, muss dabei der Begriff des größten gemeinsamen Teilers noch einmal wiederholt werden. Es zeigte sich jedoch, dass die Schüler ein intuitiv richtiges Gefühl für Teilbarkeit und den größten gemeinsamen Teiler haben, vor allem wenn es darum geht, zu entscheiden, ob dieser gleich oder ungleich eins ist, so dass an dieser Stelle keine Probleme mit mangelnden Vorkenntnissen auftreten sollten.

Diese Phase des Unterrichts kann größtenteils Schüler orientiert, angeleitet durch Arbeitsblätter ablaufen, so wie oben dargestellt. Die Grundideen sollten allerdings in Frontalphasen erörtert werden, da Denkweisen dieser Art den meisten Schülern fremd sein werden. Ist ein Kurs hingegen freies Arbeiten gewöhnt, so bietet sich das Eingangsproblem (finde die Entschlüsselungsfunktion zu $f([x]_{26}) = 3[x]_{26}$) hierzu ebenfalls gut an. Durch Computereinsatz (z.B. Maple, Derive, Excel) kann die Vermutung 3 zusätzlich für große n gestützt werden.

Nach dem jetzigen Stand des Unterrichts sollen die Schüler den Begriff des *multipliaktiven Inversen* kennen, wissen wann dieses existiert und dieses mehr oder weniger systematisch berechnen können. Sie sollen verstanden haben, warum eine Verschlüsselungsfunktion eindeutig sein und keine Fixpunkte haben sollte und den Begriff des Permutationschiffres kennen. Zudem sollen sie die Unsicherheit eines Permutationschiffres verstanden haben und in der Lage sein, ein solches mit Hilfe einer Häufigkeitsanalyse zu entschlüsseln.

2.2.3 Verschlüsseln in Blöcken und das Paradoxon der klassischen Kryptographie

Wir haben festgestellt, dass sich jedes Permutationschiffre durch eine Häufigkeitsanalyse entschlüsseln lässt. Wie kann man sich also gegen eine Häufigkeitsanalyse wehren?

Anhand einiger Aufgaben, welche die Schüler in Kleingruppen bearbeiten, werden sie mit dem Vigenère-Chiffre, nach Blaise de Vigenère(1523-1596), einem französischen Diplomaten, der dieses Verfahren 1586 der Öffentlichkeit zugänglich machte, vertraut gemacht:
Auf dem Arbeitsblatt wird zunächst der Text „Treffen morgen um drei“ in Blöcken von je zwei Buchstaben mit dem Schlüsselwort „GN“ verschlüsselt:

(T	R)	(E	F)	(F	E)	(N	M)	(O	R)	(G	E)	(N	U)	(M	D)	(R	E)	(I)
(G	N)	(G	N)	(G	N)	(G	N)	(G	N)	(G	N)	(G	N)	(G	N)	(G	N)	(G)
									↓									
(Z	E)	(K	S)	(L	R)	(S	Z)	(U	E)	(M	R)	(T	H)	(S	Q)	(X	K)	(O

Die wesentlichen Unterschiede zu einem Permutationschiffre sind zum einen, dass ein Buchstabe des Klartextes im Chiffretext durch zwei verschiedene Buchstaben darge-

stellt wird, und zum anderem, dass ein Buchstabe im Ciffretext für zwei verschiedene Klartextbuchstaben stehen kann, die Entschlüsselung aber trotzdem eindeutig ist. Für eine Häufigkeitsanalyse wird nun die Häufigkeit der *Bigramme*, also den Paaren zweier aufeinander folgender Buchstaben, benötigt [Beu02]. Die Schüler werden aber wahrscheinlich auf eine andere Methode der Häufigkeitsanalyse kommen, indem sie einfach jeden zweiten Buchstaben verwenden. In diesem Fall sind zum Entschlüsseln zwei Häufigkeitsanalysen nötig, jede mit einer Hälfte des Textes.

Für den Fall, dass die Schüler mit der Vektorrechnung vertraut sind, sollen sie nun überlegen, wie man die zu diesem Chiffre gehörige Verschlüsselungsfunktion geschlossen darstellen könnte.

$$\begin{array}{rcl} f & : & (\mathbb{Z}/26\mathbb{Z})^2 \to (\mathbb{Z}/26\mathbb{Z})^2 \\ \begin{pmatrix}[x]_{26}\\[y]_{26}\end{pmatrix} & \mapsto & \begin{pmatrix}[x]_{26}+6\\[y]_{26}+13\end{pmatrix} \end{array}$$

Man könnte dieses Chiffre also auch als zweidimensionales Verschiebechiffre betrachten.

Verschlüsselungsverfahren wie das Vigenère-Chiffre, in welchen einem Klartextbuchstaben mehrere Chiffretextbuchstaben zugeordnet sind, werden auch *polyalphabetisch* genannt, im Gegensatz zu den *monoalphabetischen* Chiffren, die wir im letzten Kapitel kennengelernt haben.

Die Schüler werden nun aufgefordert, den obigen Text zunächst mit dem Schlüsselwört „AUS“, also in Dreierblöcken, und danach mit dem Schlüsselwort

„EINSZWEIDREIVIERFUE“

zu verschlüsseln. Die Schüler werden erkennen, dass die Sicherheit der Verschlüsselung mit der Länge des Schlüsselwortes zunimmt. Das führt uns auf die nächste

Vermutung 4 *Ein Vigenère-Chiffre ist nicht mehr mit Hilfe einer statistischen Häufigkeitsanalyse zu entschlüsseln, wenn das Schlüsselwort genauso lang wie der zu verschlüsselnde Text ist.*

An dieser Stelle bietet sich ein Exkurs über die *Sicherheit* von Kryptosystemen an. Diese wäre allerdings ein neuer, eigenständiger Abschnitt, welcher eine gute Stochastikkenntnis voraussetzt und über den Rahmen dieser Arbeit hinaus geht.[Buc99], [Beu02]

Das Ergebnis dieses Exkurses, dass ein Vigenère-Chiffre perfekt sicher gegen statistische Analysen ist, wenn das Schlüsselwort die gleiche Länge wie der Klartext hat und aus zufällig gewählten Buchstaben besteht, können die Schüler auch ohne Beweis einsehen. Die zusätzliche Bedingung des aus zufällig gewählten Buchstaben bestehenden Schlüsselwortes lässt sich damit erklären, dass sich sonst eine statistische Analyse anhand der Sprache des Schlüsselwortes ansetzten ließe.

Ein Vigenère-Chiffre mit diesen Eigenschaften nennt man auch *Vernam–Chiffre* nach Gilbert Vernam (1890 - 1960), der dieses im Jahr 1917 herausfand, oder *One-Time-Pad*.

Die Schüler sollen nun den zweiten Namen des Vefahrens diskutieren und dabei erläutern, warum derselbe Schlüssel nur einmal benutzt werden sollte. Hier stellt sich nun die Frage nach der Praktikabilität des Verfahrens und warum sich dieses Verfahren, obwohl es die optimale Sicherheit bietet nicht durchgesetzt hat. Eine Schwierigkeit liegt in der Notwendigkeit für jede Nachricht einen neuen zufälligen Schlüssel derselben

Länge der Nachricht zu erstellen. Darauf könnten die Schüler entgegnen, dass es mit Hilfe von Computern kein Problem sei, lange Zufallsstrings zu erzeugen. Um einer Diskussion über Zufalls- und Pseudozufallszahlen von Computern zu entgehen, betrachten wir lieber die nächste Schwierigkeit, die auch für die Schüler eine echte Schwierigkeit darstellt: Der zufällig erzeugte Schlüssel muss dem Kommunikationspartner mitgeteilt werden, und zwar für jede Nachricht neu. Da der Schlüssel dieselbe Länge wie die Nachricht hat, stellt die Übermittlung des Schlüssels dasselbe Problem dar, wie das Ursprüngliche, das Übermitteln der Nachricht. Diese Tatsache nennt man auch das

Paradoxon der klassischen Kryptographie *Will man ein Geheimnis austauschen, muss man vorher schon ein Geheimnis ausgetauscht haben*

Die Schüler werden an dieser Stelle viele Ideen haben, das Verfahren zu retten. Nahe liegend sind hier Schlüsselbücher oder ein frei verfügbares literarisches Werk als Schlüssel (auf dieses könnte man sich einmal im Jahr einigen). Die zweite Möglichkeit verstößt aber gegen die Forderung, dass der Schlüssel eine zufällige Zeichenkombination darstellen sollte. Die Benutzung eines Schlüsselbuches stellt dagegen vor allem in einem Netz aus vielen Kommunikationspartnern einen großen Anspruch an die Logistik. Zudem stellt sie ein großes Risiko dar, falls ein Schlüsselbuch unbemerkt in die falschen Hände geraten sollte. Ein Beispiel hierfür ist die Entschlüsselung der Enigma im zweiten Weltkrieg, welche ebenfalls bei korrekter Durchführung ein nahezu unknackbares System darstellte.

An dieser Stelle würde sich auch ein Schülerreferat zum Thema Enigma und ihrer Entschlüsselung anbieten. Es könnte dabei auch diskutiert werden, warum es gefährlich ist immer dieselben Floskeln in den verschlüsselten Nachrichten zu verwenden.

Methoden, Ziele, Erläuterungen

Da in diesem dritten Abschnitt im Wesentlichen keine neuen Mathematischen Inhalte eingeführt werden, bietet sich eine an Aufgaben orientierte Schülerarbeit an. So können die Schüler bis zu der Vermutung 4 nahezu selbstständig in Kleingruppen arbeiten und sollten zu den erwarteten Erbgebnissen gelangen.

Sind die Schüler mit der Vektorgeometrie vertraut, werden sie vermutlich trotzdem einen Hinweis benötigen, um das Vigenère-Chiffre als geschlossene Verschlüsselungsfunktion zu schreiben.

Andererseits bietet sich hier sogar die Möglichkeit, anschließend mit dieser Frage nach der geschlossenen Darstellung eine Einführung in das Gebiet der Vektorgeometrie zu geben. Denn die Schüler sehen an dieser Stelle, wie die Erfahrung zeigt, sehr schnell, dass die Verschlüsselungsfunktion der Beispielverschlüsselung als Tupel geschrieben werden könnte:

$$f(x, y) = (x + 6 \mod 26,\ y + 13 \mod 26)$$

Dieses könnte nun geometrisch interpretiert werden. Jeder Buchstabe ist dann ein Punkt in der Ebene, und jede Verschlüsselung eine Verschiebung. Eine genaue Ausführung dieser Idee wäre aber sicherlich Thema einer eigenen Arbeit sein und wird deswegen nicht weiter ausgeführt.

Weiterhin bietet sich in dieser Phase ein ausführlicher Exkurs in die Stochastik und Statistik an. Ausgehend vom Vigenère- und Vernamchiffre könnte die statistische Si-

cherheit von Kryptosystemen unterucht werden. Zusätzlich könnten statistische Analysen untersucht werden, mit deren Hilfe man feststellen, ob eine mono- oder polyalphabetsiche Verschlüsselung vorliegt, oder die Schlüssellänge eines Vigenère-Chiffre bestimmen kann. (Kasinski- und Friedmann-Test [Beu02])

Da diese Überlegungen uns aber dem Public-Key-Chiffre, dem Thema dieser Arbeit, nicht näher bringen, werden diese hier nicht vertieft.

Für diese dritte Phase sind (ohne die Exkurse zur Stochastik oder Vektorgeometrie) etwa zwei Unterrichtsstunden vorgesehen.

2.2.4 Moderne Kryptographie

In der letzten Stunde haben die Schüler das Paradoxon der klassischen Kryptographie kennengelernt. Dieses galt bis in die siebziger Jahre des letzten Jahrhunderts als unlösbares Problem in der Kryptographie: Wollen zwei Parteien geheim miteinander kommunizieren, dann müssen diese vorher einen geheimen Schlüssel vereinbart haben. Erst in den Jahren 1975-77 gelang es Ronald Rivest, Adi Shamir und Leonard Adleman das Paradoxon der klassischen Kryptographie zu widerlegen.

Die Schüler lernen nun also eine Mathematik kennen, die neuer ist, als alles andere, was in der Schulmathematik gelehrt wird!

Da die Grundideen der Public-Key-Kryptographie sehr spannend und wichtig sind, bietet sich eine Frontalphase mit übersichtlichem Tafelbild an, auch damit eventuelle Fehlvorstellungen seitens der Schüler sofort in der Diskussion ausgeräumt werden können. Die Erfahrung zeigt, dass die Schüler an dieser Stelle sehr gut mitarbeiten und motiviert sind, die Ideen zu verstehen. Dieses lässt sich vermutlich damit erklären, dass sie das Paradoxon der klassischen Kryptographie verstanden und als echtes Problem anerkannt haben, für welches es nun eine Lösung zu finden gilt.

Wir wollen die Grundideen der Public-Key-Kryptographie nun an den folgenden praktischen Beispielen erläutern, und uns dazu das „Geheimnis“ in einer abschließbaren Kiste vorstellen.

Das Verschlüsseln in der klassischen Krytographie, wie etwa mit dem Verschiebe- oder dem Vigenère-Chiffre, beschreibt die folgende Abbildung:

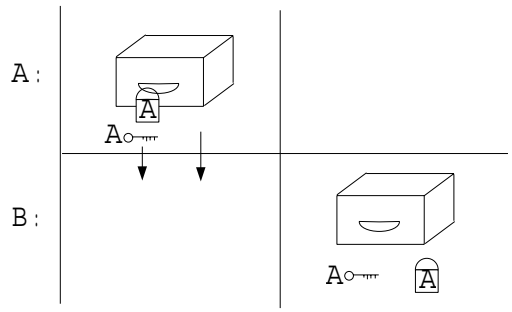

Abbildung 2.1: Klassische Verschlüsselung

A verschlüsselt das Geheimnis mit seinem Schlüssel A. Daraufhin muss er beides (das verschlüsselte Geheimnis und den Schlüssel) an B übermitteln. Um den Schlüssel sicher zu übermitteln, müsste er diesen wieder verschlüsseln usw.

Anfang der 70-iger Jahre hatten die Mathematiker Whitfield Diffie und Martin Hellman zwei Ideen, wie man sich geheime Nachrichten schicken könnte, ohne vorher einen Schlüssel austauschen zu müssen!

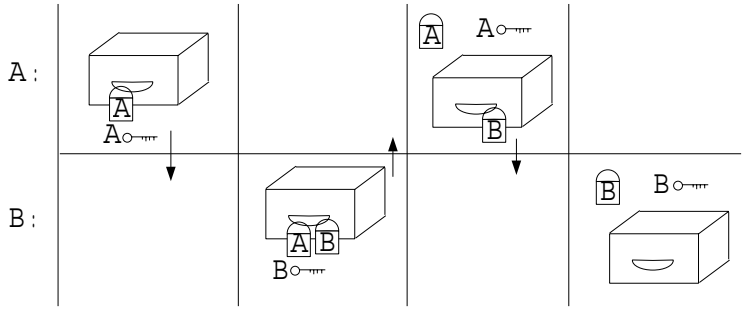

Abbildung 2.2: Verschlüsselung ohne Schlüsselaustausch

1. Im ersten Schritt verschlüsselt A mit seinem Schlüssel
2. Im zweiten Schritt verschlüsselt B noch einmal mit seinem Schlüssel
3. Nun entschlüsselt A mit seinem Schlüssel
4. Nun kann auch B mit seinem Schlüssel entschlüsseln und hat das Geheimnis entschlüsselt

Dieses Verfahren kann man am Beispiel des Vernam-Chiffres verdeutlichen, indem sich je zwei Schüler eine kurze Nachricht „schicken“. Allerdings ist dieses Verfahren mit dem Vernam-Chiffre keinesfalls sicher, da ein Angreifer aus den verschlüsselten Texten die verwendeten Schlüssel berechnen kann. Die Schüler haben nun die Gelegenheit, in die Rolle des Bösewichts zu schlüpfen, welcher versucht, das Kryptosystem zu knacken. Wie allgemein in der Kryptographie nehmen wir an, dass das Verschlüsselungsverfahren nicht geheim gehalten werden kann, also jeder Angreifer das Verfahren genau kennt und somit der verwendete Schlüssel die einzige Sicherheit bietet.

Wie dieses Beispiel zeigt, haben wir noch kein sicheres System gewonnen, da auch unter Verwendung eines multiplikativen Chiffre, wie die Schüler es oben kennengelernt haben, der Angreifer die verwendeten Schlüssel errechnen kann. (Erst wenn man Potenzfunktionen benutzt, wird aus dieser Idee ein sicheres Chiffre, das *Massey-Omura-Chiffre.*) Aber Diffie und Hellmann forschten weiter und schon ein paar Jahre später, im Mai 1977, folgte in Zusammenarbeit mit Adi Shamir ihre nächste Idee, welche den Durchbruch bringen sollte. (Abbildung 2.4)

Voraussetzung für dieses Verfahren ist, dass nur B einen Schlüssel zum Entschlüsseln besitzt. Der Schlüssel zum Verschlüsseln ist dagegen allen zugänglich (hier: ein Laden, der Schlösser für bestimmt Personen verkauft.) Wichtig ist vor allem, dass man aus der Kenntnis der Verschlüsselung nichts über das Verfahren der Entschlüsselung herausbekommen kann. Ein solches Verfahren nennt man *Public-Key-Chiffre*

Der theoretische Durchbruch war geschafft, er musste nun aber noch mathematisch umgesetzt werden: Wir suchen also eine Verschlüsselungsfunktion f, die zwar umkehrbar ist, (d.h. es gibt eine Funktion g so dass $g(f(x)) = x$) von der die Umkehrfunktion

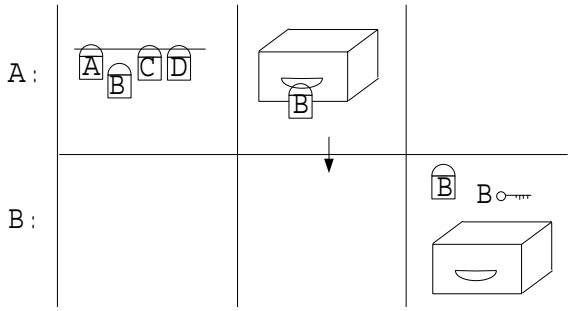

Abbildung 2.3: Idee des RSA-Verfahrens

g aber trotz Kenntnis von f praktisch nicht (ohne Zusatzinformationen) berechenbar ist!

Die Funktionen, die wir bisher kennen gelernt haben, fallen hier weg, da wir subtrahieren und teilen können, wir also aus dem Schlüssel zum Verschlüsseln den Schlüssel zum Entschlüsseln berechnen können. (Allerdings können die Schüler das multiplikative Inverse, welches wie zum Teilen benötigen, nur durch Probieren finden. Bei großen Zahlen ist dies also praktisch unmöglich. Der öffentliche Schlüssel wäre also das Tupel (a, n), wobei a und n zu der Funktion

$$x \mapsto a \cdot x \mod n$$

gehören. Der private Schlüssel wäre ein b mit $ab \equiv 1 \mod n$, welches sich nicht aus dem Schlüssel (a, n) berechnen ließe. Der große Nachteil ist aber, dass auch der Empfänger keine Chance hat, den privaten Schlüssel zu berechnen. In der Klasse könnte der Lehrer die Rolle eines Trust-Centers spielen, und die geheimen Schlüssel an die Schüler verteilen. Dieses wird aber an späterer Stelle noch einmal mit den Schüler probiert und sollte hier höchstens erwähnt werden.)

Bisher haben wir eine Klasse von Funktionen außer Acht gelassen: Potenzfunktionen. Zunächst sollen die Schüler Funktionen der Art $f(x) = x^a \mod n$ untersuchen. (In der Literatur ist an dieser Stelle die Potenz zumeist mit e bezeichnet. Dieses sollte in der Schule aber vermieden werden, da sonst die Gefahr einer Verwechselung mit der Eulerschen Zahl vorliegt.)

Um zu vermeiden, dass Schüler durch fehlende Vorkenntnisse den Anschluss verlieren sollten an dieser Stelle die Potenzgesetze kurz wiederholt werden.

Die Schüler sollen nun in Kleingruppen Funktionen der Form $f(x) = x^a \mod p$ zu verschiedenen Modulen p betrachten, und untersuchen, für welche Potenzen a diese eindeutig sind. (Da wir uns in den letzten Stunden überlegt haben, dass dieses sinnvoll ist.) Hierbei könnte eine Tabelle der Form

a	0	1	...	$p-1$	ggT(a, p)	ggT$(a, p-1)$
1						
2						
⋮						

hilfreich sein. Die Schüler werden durch ihre Ergebnisse zu der folgenden Vermutung kommen:

Vermutung 5 *1. Die Funktion $x \mapsto x^a \mod p$ ist eindeutig, wenn gilt $ggT(a, p-1) = 1$*

2. $x^{p-1} \equiv 1 \mod p$ für $x \neq 0$ bzw. $x^p \equiv x \mod p$

3. Man darf im Exponenten nicht modulo p rechnen, (z.B. $6 \equiv 1 \mod 5$, aber $x^6 \neq x^1 \mod 5$) sondern es gilt die Regel

$$a \equiv b \mod (p-1) \rightarrow x^a \equiv x^b \mod p$$

Die Vermutungen 1 und 2 werden die Schüler mit ziemlicher Sicherheit herausfinden. Die Vermutung 3 sollten dann in der Diskussion thematisiert werden und zunächst anhand der Tabellen, die die Schüler erstellt haben überprüft werden.

Die Vermutung 5.1 wollen wir an dieser Stelle nicht beweisen, da wir für die Eindeutigkeit einer Funktion keine formalen Kriterien eingeführt haben.

Der kleine Satz von Fermat kann mit Hilfe des binomischen Lehrsatzes und vollständiger Induktion bewiesen werden:

1.Schritt:

Es gilt $(x+y)^p \equiv x^p + y^p \mod p$ für alle $x, y \in \mathbb{Z}$ und p prim.

$$\begin{aligned}(x+y)^p &= x^p + \binom{p}{1}x^{p-1}y + \cdots + \binom{p}{k}x^{p-k}y^k + \cdots + \binom{p}{p-1}xy^{p-1} + y^p \\ &\equiv x^p + 0 \cdot x^{p-1}y + \cdots + 0 \cdot x^{p-k}y^k + \cdots + 0 \cdot xy^{p-1} + y^p \mod p \\ &\equiv x^p + y^p \mod p\end{aligned}$$

2. Schritt:

Es gilt $2^p = (1+1)^p \equiv 1^p + 1^p \equiv 1 + 1 \equiv 2 \mod p$

sowie $3^p = (1+2)^p \equiv 1^p + 2^p \equiv 1 + 2 \equiv 3 \mod p$

...

3. Schritt: Vollständige Induktion

Induktions Annahme: Gelte also $a^p \equiv a \mod p$

Induktions Schritt $a \rightarrow a+1$:

$$(a+1)^p = a^p + 1^p \equiv a + 1 \mod p$$

Es stellt sich nun die Frage, inwieweit dieser Beweis mit den Schülern ausgearbeitet werden sollte. Ist den Schülern der binomische Lehrsatz unbekannt, ist der Beweis für die Schüler sinnlos, denn dann müssten sie den ersten Schritt hinnehmen und könnten auch gleich den kleinen Satz von Fermat hinnehmen.

Ist der binomische Lehrsatz bekannt, müssen die Schüler zunächst nachrechnen, dass gilt $\binom{p}{k} = \frac{p(p-1)\cdot\cdots\cdot(p-k+1)}{k(k-1)\cdot\cdots\cdot 2\cdot 1} \equiv 0 \mod p$. Dies glaubt man (und auch die Schüler) naiv sehr schnell, da der Faktor p im Zähler des Binomialkoeffizienten auftaucht. Zusätzlich müsste man sich jedoch überlegen, dass p auch wirklich in der Primfaktorzerlegung des Binomialkoeffizienten auftritt. Da $k < p$ und p prim, kann aber p nicht gekürzt werden und tritt somit als Primfaktor in der Zerlegung des Binomialkoeffizienten auf.

Den zweiten Schritt werden die Schüler nachvollziehen können, und er sollte ihnen als Plausibilitätsbetrachtung für eine beliebige natürliche Zahl genügen. Der kleine Satz von Fermat gilt aber auch für negative ganze Zahlen. Ob dieser Beweis auch für negative Zahlen anwendbar ist, oder ob es hierfür ein anderes Argument gibt, könnte man den Schülern als freiwillige Knobelaufgabe stellen. Formal kann man nachrechnen, dass gilt $(-1)^p = (1-2)^p \equiv 1^p - 2^p \equiv 1 - 2 \equiv -1 \mod p$.

Die formale vollständige Induktion sollte nur durchgeführt werden, wenn die Schüler mit dieser gut vertraut sind, da den Schülern ein formales Aufschreiben keine neuen inhaltlichen Erkenntnisse bringt, abgesehen von der Anwendung der vollständigen Induktion als Beweisinstrument.

Nun können wir die Vermutung 5.2 formal nachrechnen:

$$x^a \overset{a\equiv b}{\underset{\mod (p-1)}{\equiv}} x^{b+k(p-1)} \equiv x^b \cdot (x^{p-1})^k \equiv x^b \mod p$$

Beispiele

1. Wir betrachten nun die Funktion $f(x) = x^3 \mod 5$. Diese kommt als Verschlüsselungsfunktion in Frage, da $\mathrm{ggT}(3,4) = 1$, und sie damit eindeutig ist. Wie sieht nun aber die Entschlüsselungsfunktion aus?

   Im Reellen hätte die Umkehrfunktion die Form $g(x) = x^{\frac{1}{3}}$, ist also auch eine Potenzfunktion. Versuchen wir hier also auch den Ansatz $g(x) = x^d \mod p$. Für die Entschlüsselungsfunktion sollte gelten $g(f(x)) = x$, also

   $$(x^3)^d = x^{3d} \equiv x^1 \mod 5$$

   Mit Vermutung 5.2 wissen wir, dass dies der Fall ist, wenn $3d \equiv 1 \mod 4$ ist. Wir suchen also das multiplikative Inverse von $[3]_4$. Dieses existiert, da $\mathrm{ggT}(3,4) = 1$ ist. Wir sehen hier also, dass unsere Vorraussetzung für Eindeutigkeit uns gleichzeitig die Existenz des Inversen, also der Potenz der Entschlüsselungsfunktion sichert. Durch Probieren bekommen wir heraus $b = 3$. Damit haben wir als Verschlüsselungsfunktion gefunden $g(x) = x^3 \mod 5$ und es gilt tatsächlich

   $$g(f(x)) = (x^3)^3 = x^{3\cdot 3} = x^9 = x^5 \cdot x^4 \equiv x \cdot 1 \mod 5$$

2. Die Schüler bearbeiten nun die folgende Aufgabe in Kleingruppen:
   Welche ganze Zahlen kommen in $\mathbb{Z}/11\mathbb{Z}$ als Exponenten für eine Entschlüsselungsfunktion Frage und wie sehen die dazugehörigen Entschlüsselungsfunktionen aus?

Wir haben also ein Kryptosystem gefunden:

- Verschlüsselungsfunktion $f(x) = x^a \mod p$, p prim, $\mathrm{ggT}(a, p-1)$=1
- Entschlüsselungsfunktion $g(x) = x^d \mod p$, mit d so dass $ad \equiv 1 \mod (p-1)$

Der öffentliche Schlüssel ist in diesem Fall der Exponent a sowie der Modul p der Verschlüsselungsfunktion. Der private Schlüssel ist das Paar (d, p), bestehend aus dem Exponenten und dem Modul der Entschlüsselungsfunktion.

Für die Schüler ist dieses eine Art Public-Key-Chiffre, da sie bei großen Zahlen nicht in der Lage sind, d aus a zu berechnen, und es bietet sich an, ein Verzeichnis mit

öffentlichen Schlüsseln zu erstellen, so dass die Schüler sich gegenseitig Nachrichten schicken können. (Der Lehrer kann nun als „Trust-Center“ auftreten und den Schülern ihre privaten Schlüssel berechnen). Zum Codieren der Nachrichten (und dem Sichern gegen eine Häufigkeitsanalyse) werden wir uns später noch Gedanken machen müssen.

Wie die Schüler feststellen, ist dies kein Public-Key-Chiffre, da der Lehrer offensichtlich in der Lage ist, zu jedem öffentlichen Schlüssel ohne Zusatzinformationen den privaten Schlüssel zu bestimmen. Wir müssen also weiter suchen und betrachten nun die Potenzfunktionen im nächst-einfachen Fall:

$$f(x) = x^a \mod (pq)$$

mit p, q prim. Die Schüler untersuchen wieder die Fragen:

1. Für welche Exponenten a ist eine solche Funktion eindeutig?
2. Unter welchen Bedingungen gilt $x^a \equiv x^b \mod (pq)$?

Die Erfahrung zeigt, dass die Schüler sofort die folgenden Vermutungen aufstellen:

Vermutung 6 *1. Die Funktion $f(x) = x^a \mod (pq)$ ist eindeutig, wenn $ggT(a, p-1)=1$ und $ggT(a, q-1) = 1$*

2. Es gilt $x^a \equiv x^b \mod (pq)$ wenn $a \equiv b \mod (p-1)$ und $a \equiv b \mod (q-1)$

Diese Vermutungen können die Schüler zunächst an kleinen Beispielen verifizieren. Die zweite Vermutung kann man auch einfach nachrechnen:
Gelte also $a \equiv b \mod (p-1)$, dann folgt $x^a \equiv x^b \mod p$ und damit $p|(x^a - x^b)$. Gelte weiterhin $a \equiv b \mod (q-1)$, dann folgt weiter $x^a \equiv x^b \mod q$ und damit $q|(x^a - x^b)$. Insgesamt folgt dann, da p und q teilerfremd sind, $pq|(x^a - x^b)$ und damit
$x^a \equiv x^b \mod (pq)$.

Die Schüler werden nun aufgefordert mit Hilfe dieser Vermutung ein Chiffre mit der Funktion $f(x) = x^a \mod (pq)$ zu erstellen:

- Verschlüsselungsfunktion: $f(x) = x^a \mod (pq)$, p, q prim und $\mathrm{ggT}(a, p-1)=1$ und $\mathrm{ggT}(a, q-1) = 1$.
- Entschlüsselungsfunktion: $g(x) = x^d \mod (pq)$, mit $ad \equiv 1 \mod (p-1)$ und $ad \equiv 1 \mod (q-1)$.

Dann gilt wirklich $g(f(x)) = x^{ed} \equiv x \mod (pq)$ nach der Vermutung 6.2.

Beispiele:

1. Wir betrachten die Funktion $f(x) = x^7 \mod 15$. Diese ist nach unseren Kriterien eine Verschlüsselungsfunktion, da $\mathrm{ggT}(7, 2) = 1$ und $\mathrm{ggT}(7, 4) = 1$. Zum Aufstellen der Entschlüsselungsfunktion müssen die beiden Kongruenzen $7d \equiv 1 \mod 2$ und $7d \equiv 1 \mod 4$ gelöst werden. Durch Probieren kommt man schnell auf die Lösung $d = 3$ und damit auf die Entschlüsselungsfunktion $g(x) = x^{21} \mod 15$.

   Es stellt sich nun die Frage, ob dieses Chiffre ein Public-Key-Chiffre ist. Zum Verschlüsseln benötigt man die Potenz a sowie das Modul $n = pq$, diese müssten

also öffentlich sein. Zum Entschlüsseln benötigt man neben dem Modul n nun die Potenz d. Ist es möglich, aus dem öffentlichen Schlüssel (a, n) die Potenz d zu berechnen? Für die Schüler erscheint diese Frage zunächst sinnlos, da sie selbiges eben getan haben. Deshalb sollen sie nun das nächste Beispiel rechnen:

2. Der öffentliche Schlüssel ist das Paar $(a, n) = (127, 178729)$. Wie lautet der private Schlüssel?

Es wird für die Schüler ohne weitere Hilfe nahezu unmöglich sein, die Faktorisierung von $178729 = 367 \cdot 487$ zu finden. Auch wenn die Schüler diese herausfinden sollten, dürfte es für sie unmöglich sein, die Kongruenzen

$$\begin{aligned} 127d &\equiv 1 \mod 366 \\ 127d &\equiv 1 \mod 486 \end{aligned}$$

zu lösen.

Deshalb lernen die Schüler nun den *euklidischen Algorithmus* kennen. Die Erfahrung zeigt, dass die Schüler an dieser Stelle stark motiviert sind, ein Verfahren zu lernen, mit dem sie in der Lage sind, die Kehrwerte modulo n zu bestimmen. Zudem bietet die Kenntnis der Kongruenzrechnung einen natürlichen Einstieg zum euklidischen Algorithmus.

Die Schüler sollen hierzu noch einmal die Restklassen modulo 6 bzw. 12 und betrachten, und jeweils die gemeinsamen Teiler der Elemente der Klassen mit 6 bzw. mit 12 bestimmen. So werden sie zu der folgenden Vermutung kommen:

Vermutung 7 *Alle Elemente einer Klasse $[a]_n$ besitzen mit n dieselben gemeinsamen Teiler. Insbesondere ist dann der $ggT(a, n)$ für alle $a \in [a]_n$ derselbe.*

Diese zunächst vielleicht verblüffende Tatsache kann man leicht nachrechnen:
Seien $b, c \in [a]_n$, dann ist $b = c + k \cdot n$, mit $k \in \mathbb{Z}$. Also ist jeder gemeinsame Teiler von c und n auch ein Teiler von b. Andererseits ist $c = b - k \cdot n$, also ist auch jeder gemeinsame Teiler von b und n ein Teiler von c. Die gemeinsamen Teiler der Elemente einer Klasse modulo n mit n sind also nur von der Klasse abhängig.

Diese Tatsache kann man nun zur Bestimmung des größten gemeinsamen Teilers benutzen:

$$\begin{aligned} \mathrm{ggT}(76, 23) &= \mathrm{ggT}(7, 23) \text{ da } 7 \in [76]_{23} \\ &= \mathrm{ggT}(7, 2) \text{ da } 2 \in [23]_7 \\ &= \mathrm{ggT}(1, 2) \text{ da } 1 \in [7]_2 \\ &= 1 \text{ da 1 der einzige Teiler von 1 ist.} \end{aligned}$$

Nachdem die Schüler ein paar Aufgaben nach diesem Schema gerechnet haben, werden sie aufgefordert, ihre Rechnungen etwas genauer aufzuschreiben. Für das obige Beispiel bedeutet dies:

$$\begin{aligned} 76 &= 3 \cdot 23 + 7 \\ 23 &= 3 \cdot 7 + 2 \\ 7 &= 3 \cdot 2 + 1 \end{aligned}$$

Durch sukzessives Einsetzten der erste und zweiten Zeile in die dritte erhält man:

$$1 = 10 \cdot 76 - 33 \cdot 23$$

Diese Rechnung wird den Schülern zunächst wenig sinnvoll erscheinen. Nach der folgenden Umformung sollten einige Schüler erkennen, dass wir ein Inverses von $[76]_{23}$ gefunden haben.

$$10 \cdot 76 = 1 + 33 \cdot 23 \Leftrightarrow 10 \cdot 76 \equiv 1 \mod 23$$

Nachdem die Schüler einige Aufgaben zur Übung gerechnet haben, kommen wir wieder zu unserem Verschlüsselungsverfahren. Gesucht war die Entschlüsselungsfunktion zum Schlüssel $(a, n) = (127, 178729)$. Wie hatten zunächst die Zerlegung $178729 = 367 \cdot 478$ gefunden und mussten zur Entschlüsselung die Kongruenzen

$$\begin{aligned} 127d &\equiv 1 \mod 366 \ (\Leftrightarrow d \equiv 49 \mod 366 \Leftrightarrow d = 49 + 366k) \\ 127d &\equiv 1 \mod 486 \ (\Leftrightarrow d \equiv 199 \mod 486 \Leftrightarrow d = 199 + 486l) \end{aligned}$$

lösen. Die Schüler werden nun beide Kongruenzen für sich lösen können. Ein neues Problem ist nun, ein d zu finden, welches beide Kongruenzen löst. Versuchen die Schüler dieses Problem rechnerisch durch Gleichsetzten zu lösen, so erhalten sie die diophantische Gleichung

$$366k + 486l = 150$$

bzw. die Kongruenz $366k \equiv 150 \mod 486$. Diese lässt sich ebenfalls mit Hilfe des euklidischen Algorithmus lösen:

$$\begin{aligned} \mathrm{ggT}(486, 366) &= 6 = 4 \cdot 366 - 3 \cdot 486 \\ &\Rightarrow 25 \cdot 6 = (25 \cdot 4) \cdot 366 - (25 \cdot 4) \cdot 486 \\ &\Leftrightarrow 366 \cdot 100 \equiv 150 \mod 486 \Rightarrow k = 100 \end{aligned}$$

Einsetzen ergibt $d = 49 + 366 \cdot 100 = 36649$. Wir haben also eine Entschlüssellungsfunktion $g(x) = x^{36649} \mod 178729$ gefunden.

Ein anderer möglicher Weg, den die Schüler entdecken könnten wäre es, die Kongruenz $127d \equiv 1 \mod (366 \cdot 486)$ zu lösen. Der euklidische Algorithmus gibt

$$1 = 7002 \cdot 127 - 5 \cdot (366 \cdot 486)$$

als Lösung $d = 7002$ einen kleineren Entschlüsselungsexponenten. Eine Probe zeigt, dass dieser wirklich ein Entschlüsselungsexponent zu der gegebenen Verschlüsselungsfunktion ist. Dieses Beispiel bringt uns auf die Vermutung:

Vermutung 8 *Gilt* $a \cdot d \equiv 1 \mod ((p-1)(q-1))$ *dann folgt*

$$\begin{aligned} a \cdot d &\equiv 1 \mod (p-1) \\ a \cdot d &\equiv 1 \mod (q-1) \end{aligned}$$

für p, q *prim und* $a, d \in \mathbb{Z}$.

Den Beweis könnte man den Schülern überlassen, oder als Zusatzaufgabe in einer Klausur stellen, da hier ein Grundverständnis über die Definition der Kongruenz abgefragt wird.

An welcher Stelle liegt aber die Sicherheit des Verfahrens? Sie muss nun in der Faktorisierung großer Zahlen liegen. Um den Schülern einen Eindruck dieses Problems zu vermittlen, sollen sie als Hausaufgabe mit Hilfe eines Computeralgebrasystems auf einem Computer oder Taschenrechner die Zahl

$$467673387707425404025773855039363318814216499363 85 = (2^{149} - 1) \cdot (2^{16} - 1)$$

faktorisieren, bzw. nach Primzahlen in der Größenordnung 2^{50} suchen.

Sie werden feststellen, dass diese Aufgaben auch mit schnellen Heimcomputern nicht ohne weiteres zu lösen sind.

Das RSA-Verfahren

Wir haben nun ein Verschlüsselungsverfahren kennen gelernt, welches nach seinen Entdeckern Ronald Rivest, Adi Shamir und Leonard Adleman „RSA-Verfahren“ genannt wird.

1. Wähle zwei Primzahlen p, q und bilde deren Produkt $n = p \cdot q$.
2. Wähle ein l mit $\text{ggT}(l, p-1) = 1 = \text{ggT}(l, q-1)$
3. Bestimme d, so das $l \cdot d \equiv 1 \mod (p-1) \cdot (q-1)$
4. Der öffentliche Schlüssel ist das Paar (l, n), der private Schlüssel das Paar (d, n). Die Verschlüsselungsfunktion ist die Funktion $f(x) = x^l \mod n$, die Entschlüsselungsfunktion die Funktion $g(x) = x^d \mod n$.

Dies ist eine Public-Key-Verschlüsselung, da es nicht möglich ist, aus dem öffentlichen Schlüssel den privaten zu bestimmen. Denn um d zu berechnen müsste man die Primfaktorzerlegung von n kennen, welche aber geheim ist.

Als letztes Problem muss nun einer Häufigkeitsanalyse vorgebeugt werden. Wir haben oben gesehen, dass dieses am besten durch ein Verschlüsseln in Blöcken geschieht. Wir haben eine Möglichkeit kennengelernt, Texte in Blöcken zu codieren und diese dann zu verschlüsseln. Wir wollen nun eine zweite Möglichkeit betrachten, einen Text in Blöcken zu codieren. Die Schüler bekommen hierzu ein Arbeitsblatt, auf dem zunächst das Dezimal- und das Dualsystem wiederholt werden. Hiernach sollen die Schüler sich überlegen, wie ein Zahlensystem zur Basis 26 aussehen würde:
Jede Zahl hat eine Darstellung

$$x_{26} = a_n \cdot 26^n + \cdots + a_1 \cdot 26 + a_0$$

mit $a_i \in \{0, 1, \ldots, 25\}$ Für die Zahlen $\{0, 1, \ldots, 25\}$ können wir nun nach unserer Anfangscodierung die Buchstaben {A,B,...,Z} schreiben.

Beispiele:

1. Das Wort (oder besser die Zahl) „TREFFEN_{26}“ steht dann für

   $$19 \cdot 26^6 + 17 \cdot 26^5 + 4 \cdot 26^4 + 5 \cdot 26^3 + 5 \cdot 26^2 + 4 \cdot 26 + 13$$

   und hat den dezimalen Wert 6073302417.

2. Verschlüssele „Treffen um drei“ in Zweierblöcken mit dem öffentlichen Schlüssel $(n, l) = (671, 131)$.

Methoden, Ziele, Erläuterungen

Für diesen Hauptteil der UE sind 5-6 Unterrichtsstunden vorgesehen. In den ersten beiden Stunden werden im Unterrichtsgespräch zunächst die Ideen von Diffie und Hellmann besprochen. Erfahrungsgemäß sind diese für die Schüler schnell einsichtig und die Schüler arbeiten gut mit. Um den Unterricht handlungsorientierter zu gestalten, könnte der Lehrer kleine abschließbare Kisten (oder Briefkästen) mitbringen, um das Public-Key-Prinzip zu verdeutlichen.

Nachdem die Schüler die Grundideen verinnerlicht haben, gilt es Funktionen zu finden, mit denen diese Ideen umsetzbar sind. Hierzu unteruchen die Schüler zunächst Potenzfunktionen modulo n mit Hilfe des vorgegebenen Schemas, ohne welches sie wenig Chancen hätten, die Regelmäßigkeiten zu erkennen und kommen auf die Vermutung 5. Diese Vermutung ist die Grundlage für das RSA-Verfahren und sollte deshalb soweit es geht bewiesen werden.

In diesem Abschnitt wird nun von der Restklassenschreibweise zu Kongruenzschreibweise übergegangen, da diese die Literatur beherrscht und ohne verwirrende Indizes und Klammern auskommt. Die Schüler sollten aber nun mit dem Rechnen mit Restklassen vertraut genug sein, und dieses ohne Probleme akzeptieren.

Die Vermutung 5 liefert ein Verschlüsselungsverfahren, welches allerdings kein Public-Key-Verfahren ist. Für die Schüler ist der Schritt zu Vermutung 6 durchaus natürlich und sie sehen keinen Anlass diese zusätzlich zu beweisen. Man sollte aber darauf hinweisen, dass diese nur deshalb richtig ist, weil p und q Primzahlen sind. Die Schüler könnten sich als Hausaufgabe ein Gegenbeispiel überlegen für den Fall, dass p und q keine Primzahlen sind.

Weiterhin bekommen wir durch den kleinen Satz von Fermat eine Möglichkeit, die Inversen modulo p zu bestimmen, da für p prim gilt:

$$a^{-1} \equiv a^{p-2} \mod p, \text{ für } a \neq 0 \mod p$$

Mit der Vermutung 6 kann man sogar zeigen, dass für die Inversen modulo pq gilt:

$$a^{-1} \equiv a^{n-p-q} \mod (pq)$$

Es stellt sich allerdings die Frage, ob man dieses den Schülern an dieser Stelle mitteilen sollte, da hierdurch die Motivation, den euklidischen Algorithmus kennen zu lernen, gesenkt werden könnte. Kommen die Schüler allerdings von selbst auf diese Formel, dann kann ein Vergleich der Rechenzeit für sehr große n mit dem euklidischen Algorithmus angestrebt werden.

Durch die Vermutung 6 haben die Schüler nun ein neues Verschlüsselungsverfahren, das auf den ersten Blick genauso aussieht wie das vorherige. Beim Rechnen des Beispiels merken sie dann aber, dass mehr Schwierigkeiten auf sie zukommen, allerdings ohne die Schwierigkeit des Faktorisierens von n zu betrachten. Deshalb wird auf das Problem der Faktorisierung großer Zahlen später genauer eingegangen.

Die Erfahrung zeigt, dass die Schüler nun eine hohe Motivation aufweisen, den euklidischen Algorithmus kennen zu lernen, was mit dem bisher gelernten auch keine großes Problem ist.

Nach diesem Unterricht sollten die Schüler Lösungen für das Paradoxon der klassischen Kryptographie kennen, die Grundideen des Public-Key-Chiffre verstanden haben und den RSA-Algorithmus anwenden können. Weiterhin sollen sie den Satz von Fermat mit seinen Folgerungen, sowie den euklidischen Algorithmus kennen und letzteren anwenden können.

2.2.5 Ausblicke

Für die gesamte Unterrichtseinheit, wie sie in den Abschnitte zuvor dargestellt worden ist, sind etwa 15-20 Unterrichtsstunden geplant. Nach diesem Unterricht verfügen die Schüler über Grundkenntnisse der Kryptographie, sowie im Rechnen mit Restklassen. An dieser Stelle folgen nun einige Anregungen, wie der Unterricht nach dieser Unterrichtseinheit fortgesetzt werden kann.

1. Möchte man noch mehr Zeit mit den modernen kryptographischen Verfahren verbringen, könnte man zunächst auf die Rechenintensivität des RSA-Verfahren hinweisen. Die Schüler könnten auf die Idee kommen, dass es reichen würde, den Schlüssel eines klassichen Verfahrens mit RSA sicher zu übermitteln, da dieser in der Regel nicht so lang ist. In diesem Kontext könnte auch eine vereinfachte Version des DES von den Schülern erarbeitet und das schon angesprochene Schema von Diffie-Hellmann betrachtet werden. Mathematisch führt dieses auf die Untersuchung von Funktionen der Form $x \mapsto a^x \mod n$ und auf das Problem des diskreten Logarithmus.

2. Neben der Verschlüsselung ist die *digitale Signatur* ein wichtiges Gebiet der modernen Kryptographie. Es könnte zunächst das Massey-Omura-Verfahren besprochen werden, bei welchem, wie bei jedem Public-Key-Verfahren, eine Signatur der Nachrichten nötig ist um sicherzustellen, dass die Kommunikation wirklich mit dem gewünschten Gesprächspartner stattfindet. Hiernach können mit den Schülern verschiedene Signatur-Verfahren besprochen werden.

3. In jedem Fall bietet es sich an diese UE im Informatik-Untericht zu begleiten und die besprochenen Algorithmen zu programmieren. Nach der UE könnten die Schüler dann mit einer PGP-Version experimentieren.

4. Sind die Schüler von der Zahlentheorie fasziniert, bietet es sich anschließend an, Diophantische Gleichungen zu lösen oder die Existenz und Eindeutigkeit der Primfaktorzerlegung zu betrachten, welche wir implizit in der gesamten UE bereits benutzt haben.

5. Möchte man nun zum klassischen Analysis-Unterricht übergehen, könnte man als Übergang das *Secret Sharing* betrachten. Je 5 Schüler bekommen ein Tupel (x, y), von dem jeder Schüler weiß, dass der Punkt (x, y) auf dem Graphen einer Funktion 5. Grades liegt. Die Aufgabe der 5 Schüler ist es nun, die Funktionsgleichung zu bestimmen. In einem Kurs mit ca. 15-20 Schülern ist ein Schnittpunkt der so von den einzelnen Gruppen bestimmten Kurven (wenn möglich besitzen alle Kurven genau einen Schnittpunkt) der gesuchte Punkt. (Man könnte ein Koordinatensystem in einen Stadtplan integrieren, so dass dieser Punkt einen bestimmten Ort angibt.) Die Schüler erkennen nun, dass nur alle gemeinsam diesen Punkt finden

konnten.

Beispielaufgabe.
f_1: $(-1,0),(0,0),(1,1),(3,36),(4,100)$
f_2: $(1,0),(-2,9),(3,12)$ & Extremwerte an den Stellen $x=0$ und $x=\frac{1}{2}\sqrt{30}$
f_3: $(0,0),(3,4),(4,0),(5,5),(-3+2\sqrt{13},0)$
Gesucht ist der Schnittpunkt aller Graphen der Funktionen.

2.3 Evaluation des durchgeführten Unterrichts

Im Rahmen dieser Arbeit wurde die oben dargestellte Unterrichtseinheit in einem Leistungskurs der 12. Jahrgangsstufe mit Erfolg durchgeführt. Am Ende der Unterrichtseinheit wurden die Schüler gebeten, einen Fragebogen zum Unterricht auszufüllen. Das Ergebnis dieser Befragung soll hier kurz wiedergegeben werden.
Der Fragebogen gliederte sich in drei Bereiche: (1) Fragen allgemein zum Unterricht, (2) Fragen zum Thema und (3) Fragen zu der Person des Lehrers. Für diese Arbeit sind insbesondere die Fragen der 2. Kategorie von Interesse:

1.	Das Thema hat mich	*mehr*	□□□□□	*weniger*	interessiert als der übliche Unterricht
2.	Das Thema bringt mir für mein späteres Leben	*mehr*	□□□□□	*weniger*	als der übliche Mathematikunterricht
3.	Ich würde gerne	*mehr*	□□□□□	*nicht noch mehr*	über die Zahlentheorie erfahren
4.	Ich würde gerne	*mehr*	□□□□□	*nicht noch mehr*	über die Kryptographie erfahren
5.	Im Unterricht haben wir zu viel	*Zahlentheorie*	□□□□□	*Kryptographie*	behandelt
6.	Ich hatte das Gefühl, mein Vorwissen war	*ausreichend*	□□□□□	*nicht ausreichend*	
7.	Das Thema hat mir	*Spaß*	□□□□□	*keinen Spaß*	gemacht.

Die Auswertung der Fragebögen der insgesamt 12 befragten Schüler ergab das folgende Ergebnis:

Frage		□	□	□	□	□	
1.	*mehr*	3	5	4	0	0	*weniger*
		25%	41,7%	33,3%	0	0	
2.	*mehr*	2	1	5	4	0	*weniger*
		16,7%	8,3%	41,7%	33,3%	0	
3.	*mehr*	3	4	1	2	2	*nicht noch mehr*
		25%	33,3%	8,3%	16,7%	16,7%	
4.	*mehr*	3	1	6	2	0	*nicht noch mehr*
		25%	8,3%	50%	16,7%	0	
5.	*Zahlentheorie*	1	2	9	0	0	*Kryptographie*
		8,3%	16,7%	75%	0	0	
6.	*ausreichend*	5	4	2	0	1	*nicht ausreichend*
		41,7%	33,3%	16,7%	0	8,3%	
7.	*Spaß*	5	6	1	0	0	*keinen Spaß*
		41,7%	50%	8,3%	0	0	

Die Ergebnisse dieser Befragung sind überwiegend nicht überraschend, sondern spiegeln vielmehr das Bild, welches die Schüler während des Unterrichts abgegeben haben. Es

zeigte sich häufig, dass die Schüler mit höherer Motivation und größerem Interesse am Unterricht beteiligt waren, als im normalen Unterricht. Dies wurde auch dadurch deutlich, dass sich mehrere Schüler zu Hause mit verschiedenen Verschlüsselungsverfahren beschäftigten und mit ihren Ergebnissen den Unterricht bereicherten. Die Fragen 3 und 4 zeigen auch, dass die Schüler durchaus motiviert sind, sich weiter mit diesem Thema auseinander zu setzten. Es wird allerdings auch deutlich, dass die Schüler ein größeres Interesse an der Kryptographie als an der Zahlentheorie haben. Dies wurde im Unterricht in der ersten längeren Phase, in der der Begriff der Restklasse eingeführt wurde, deutlich, in der die Schüler den Eindruck machten, als wäre ihnen nicht klar, wozu diese ganze Theorie nötig ist. Hier könnte ein begleitender Informatikkurs, in welchem mit Hilfe der neuen Theorien neue Algorithmen zur Verschlüsselung programmiert werden, Abhilfe schaffen.

Die zweite Frage nach der Relevanz des Themas im späteren Leben ist natürlich etwas unfair, da die Schüler im allgemeinen noch nicht wissen, womit sie sich in ihrem späteren Leben beschäftigen werden. Die überwiegend positiven Antworten der Schüler können vermutlich mit den Anwendungen der Kryptographie in der digitalen Kommunikation erklärt werden.

Auch die Frage nach den Vorkenntnissen wurde von den Schülern überwiegend positiv beantwortet. Als Vorkenntnisse sind neben der sicheren Beherrschung der Grundrechenarten auch ein funktionales Verständnis hilfreich, um Funktionen über endlichen Mengen betrachten und deren Eigenschaften, verstehen zu können. Ein solches Verständnis wird vor allem in der 10. und 11. Jahrgangsstufe geprägt, deshalb sollte diese UE auch nicht vorher durchgeführt werden.

Die letzte Frage zeigt noch einmal, dass die UE den Schülern, wie auch dem Autor, wie auch dem Lehrer des Kurses, der sich gerade in der zweiten Hälfte der UE in den Unterricht einschaltete und die Aufgaben mit den Schülern bearbeitete, sehr viel Spaß gemacht hat.

Im ersten Teil des Fragebogens wurde nun auf den Unterricht im Allgemeinen eingegangen.

1.	Der Unterricht war	*anders*	□□□□□	*genauso*	als/wie der übliche Unterricht
2.	Der Unterricht war	*leichter*	□□□□□	*schwerer*	als der übliche Mathematikunterricht
3.	Der Unterricht war	*langsamer*	□□□□□	*schneller*	als der übliche Mathematikunterricht
4.	Der Unterricht war	*zu schnell*	□□□□□	*zu langsam*	
5.	Ich hätte gerne	*mehr*	□□□□□	*weniger*	Beweise gesehen
6.	Ich hätte gerne	*mehr*	□□□□□	*weniger*	Beweise selbst gemacht
7.	Ich hätte gerne	*mehr*	□□□□□	*weniger*	in Gruppen gearbeitet
8.	Ich hätte gerne	*mehr*	□□□□□	*weniger*	Aufgabe zur Übung gerechnet
9.	Ich hätte gerne	*viel*	□□□□□	*gar nicht*	am Computer gearbeitet

Die Fragen 1-4 zielen auf subjektive Wahrnehmungen der Schüler und sind lediglich für den Unterrichtenden Lehrer interessant. Die Fragen 5-9 dagegen sind über diese UE hinaus für den Mathematikunterricht im Allgemeinen von Bedeutung. Die Auswertung der Fragebögen brachte das folgende Ergebnis:

Frage		□	□	□	□	□	
1.	*anders*	1	8	3	0	0	*genauso*
		8,3%	66,7%	25%	0	0	
2.	*leichter*	0	6	2	4	0	*schwerer*
		0%	50%	16,7%	33,3%	0	
3.	*schneller*	1	3	6	2	0	*langsamer*
		8,3%	25%	50%	16,7%	0%	
4.	*zu schnell*	0	2	7	3	0	*zu langsam*
		0%	16,7%	58,3%	25%	0	
5.	*mehr*	0	2	5	4	1	*weniger*
		0%	16,7%	41,7%	33,3%	8,3%	
6.	*mehr*	0	3	2	3	3	*weniger*
		0%	27,3%	18,2%	27,3%	27,3%	
7.	*mehr*	2	1	7	2	0	*weniger*
		16,7%	8,3%	58,3%	16,7	0%	
8.	*mehr*	2	6	1	1	2	*weniger*
		16,7%	50%	8,3%	8,3	16,7%	
9.	*viel*	3	7	0	1	1	*gar nicht*
		25%	58,3%	0%	8,3	8,3%	

Die Ergebnisse der ersten vier Fragen zeigen, dass die Schüler den Schwierigkeitsgrad und die Geschwindigkeit des Unterrichts überwiegend als normal im Vergleich zu ihrem anderem Unterricht empfunden haben. Die Frage 4 zeigt weiterhin, dass der Großteil der Schüler mit der Geschwindigkeit des Unterrichts einverstanden war.

Als nächstes betrachten wir die Fragen 5,6 und 9. Hier wird deutlich, dass die meisten Schüler lieber mit dem Computer arbeiten wollen, als sich mit den Beweisen der Theorien zu beschäftigen. Es verwundert jedoch etwas, dass die Schüler bei Beweisen einen Lehrervortrag den eigenen Versuchen vorziehen. Wahrscheinlich sind die Schüler durch ihre Erfahrungen mit Beweisen, die sie vermutlich selten wirklich durchschauen, abgeschreckt, eigenständige Beweisversuche zu starten. Hier bietet sich aber auch die Chance, die Schüler am Computer anhand vieler pragmatischer Beispiele Vermutungen aufstellen und diese dann, vielleicht auch mit Hilfe des Computers, begründen zu lassen. Die Gefahr dabei ist allerdings, dass die Schüler dann die Notwendigkeit eines formalen Beweises nicht mehr sehen, da sie durch die vielen Beispiele, die sie am Computer probiert haben, von der Richtigkeit der Aussage überzeugt sind.

Zur Frage 7 muss gesagt werden, dass bei der Durchführung der UE nicht so viel in Gruppen gearbeitet wurde, wie in der Planung vorgesehen, da die Schüler es vermutlich nicht gewohnt waren, über längere Zeit selbstständig zu arbeiten und häufig auf explizitere Arbeitsaufträge warteten. Die Auswertung dieser Frage bekräftigt diese Vermutung, da die Mehrheit der Schüler mit dem Anteil der Gruppenarbeit zufrieden war. Wahrscheinlich hätten mehr Gruppenarbeitsphasen, in denen zusätzliche Übungsaufgaben gerechnet werden, wie in Frage 8 mehrheitlich gefordert, zu einem besseren Verständnis des Themas beigetragen. Frage 8 zeigt aber auch, dass nicht alle Schüler mehr Übungsaufgaben für nötig hielten. Um diese Schüler nicht zu langweilen, bietet es sich an, ihnen weitergehende Fragen zu stellen, mit denen sie sich beschäftigen können. Hier hätte man solche Schüler z.B. auf das Problem der digitalen Signatur hinweisen können, woran sie dann ihre Kenntnis hätten wiederholen und erweitern können.

Literaturverzeichnis

[Beu02] Albrecht Beutelspacher. *Kryptologie, 6.,überarbeitete Aulage*. New-York; Berlin; Heidelberg. Springer-Verlag, 1994

[Buc99] Johannes Buchmann. *Einführung in die Kryptographie*. Braunschweig; Wiesbaden. Vieweg, 2002

[Bon] Dan Boneh. *Twenty Years of Attacks on the RSA Cryptosystem*.

[Her00] Wilfried Herget. *Europäische Artikel Nummer (EAN)*.
MUP[2] IV.Quartal 2000. 31-33

[Jah00] Thomas Jahnke. *Normaler, produktiver Mathematikunterricht*.
PM[3] 1/42 (2000)

[Joh01] Craig M. Johnson. *Functions of Number Theory in Music*. Mathematics Teacher Vol.94, No.8 (2001). 700-707

[Kob94] Neal Koblitz. *A Course in Number Therory and Cryptography, 2.Auflage*. New-York; Berlin; Heidelberg. Springer-Verlag, 1994

[LiN86] Rudolph Lidl, Harald Niederreiter. *Introduction to finite fields and their applications*. Cambridge. University Press, 1986

[MOV96] A.Menezes, P.van Oorschot, S.Vanstone. *Handbook of Applied Cryptography*. Boca Raton; New-York; London. CRC Press, 1996

[Puh98] Hermann Puhlmann. *Kryptographie verstehen - Ein schülergerechter Zugang zum RSA-Verfahren*. 1998

[RU87] Reinhold Remmert, Peter Ullrich. *Elemtare Zahlentheorie*. Basel; Boston. Birkhäuser-Verlag, 1987

[Sem02] Egmont Semmler. *Schnelle Faktorisierung*. Vortrag im Rahmen des Seminars *IT-Sicherheit*, Ruhr-Universität Bochum, 2002

[Som98] Heike Sommer. *Zahlentheorie in der Schule? - Der RSA-Algorithmus als ihre hochaktuelle Anwendung*. **PM** 4/40 (1998). 159-162

[Sch94] Ralph-Hardo Schulz. *Primzahlen in öffentlichen Chiffrierverfahren*.
mathematik **lehren** Heft 61 (1994). 46-54

[Wal99] Wolfgang Walter. *Analysis 1, 5.Auflage*. New-York; Berlin; Heidelberg. Springer-Verlag, 1999

[2] Mathematische Unterrichts Praxis
[3] Praxis der Mathematik

Anhang

Arbeitsblatt zum Thema
Formalisierung des Verschiebechiffre I

1. **Codierung**
   Die Menge der Buchstaben des Alphabets wollen wir mit Σ bezeichnen:

$$\Sigma = \{A, B, C, D, E, ..., Z\}$$

Um mit den Buchstaben „rechnen“ zu können, ist es sinnvoll, jedem Buchstaben eine Zahl zuzuordnen. Kannst du eine sinnvolle Zuordnung angeben?

A $\mapsto$ _____	B $\mapsto$ _____	C $\mapsto$ _____	D $\mapsto$ _____	E $\mapsto$ _____	F $\mapsto$ _____
G $\mapsto$ _____	H $\mapsto$ _____	I $\mapsto$ _____	J $\mapsto$ _____	K $\mapsto$ _____	L $\mapsto$ _____
M $\mapsto$ _____	N $\mapsto$ _____	O $\mapsto$ _____	P $\mapsto$ _____	Q $\mapsto$ _____	R $\mapsto$ _____
S $\mapsto$ _____	T $\mapsto$ _____	U $\mapsto$ _____	V $\mapsto$ _____	W $\mapsto$ _____	X $\mapsto$ _____
y $\mapsto$ _____	Z $\mapsto$ _____				

Codiere nun mit deiner Zuordnung den Satz „Schluss mit lustig“ und verschlüssele ihn mit dem Schlüssel „N“:

„Schluss mit lustig“	$\xrightarrow{codieren}$	_____
		_____
		$verschlüsseln\ mit\ f_{geschlossen}$ $\downarrow$
_____		_____
	$\xleftarrow{decodieren}$	
_____		_____

- Was hast du mit dem Leerzeichen gemacht? Könnte man dieses auch verschlüsseln? Wie?
- Wenn du eine Möglichkeit gefunden hast, verschlüssele den Satz mit dem Leerzeichen nocheinmal!
- Welche Informationen musst du deinem Gegenüber mitteilen, damit dieser deinen Text sinnvoll entschlüsseln kann?
- Kennst du andere Arten der Codierung aus dem täglichen Leben? Wie arbeiten Computer?

Arbeitsblatt zum Thema Codierungsverfahren

1. **Buchstaben- und Präfix-Codes**

   (a) Wir haben bisher einen Code kennengelernt, der die Buchstaben des Alphabets auf Zahlen zwischen 0 und 25 abbildet:

A $\mapsto$ 00	B $\mapsto$ 01	C $\mapsto$ 02	D $\mapsto$ 03	E $\mapsto$ 04	F $\mapsto$ 05
G $\mapsto$ 06	H $\mapsto$ 07	I $\mapsto$ 08	J $\mapsto$ 09	K $\mapsto$ 10	L $\mapsto$ 11
M $\mapsto$ 12	N $\mapsto$ 13	O $\mapsto$ 14	P $\mapsto$ 15	Q $\mapsto$ 16	R $\mapsto$ 17
S $\mapsto$ 18	T $\mapsto$ 19	U $\mapsto$ 20	V $\mapsto$ 21	W $\mapsto$ 22	X $\mapsto$ 23
y $\mapsto$ 24	Z $\mapsto$ 25				

   (b) Eine weitere Möglichkeit ist die Binäre ASCII-Codierung:

A $\mapsto$ 01000001	B $\mapsto$ 01000010	C $\mapsto$ 01000011	D $\mapsto$ 01000100	E $\mapsto$ 01000101
F $\mapsto$ 01000110	G $\mapsto$ 01000111	H $\mapsto$ 01001000	I $\mapsto$ 01001001	J $\mapsto$ 01001010
K $\mapsto$ 01001011	L $\mapsto$ 01001100	M $\mapsto$ 01001101	N $\mapsto$ 01001110	O $\mapsto$ 01001111
P $\mapsto$ 01010000	Q $\mapsto$ 01010001	R $\mapsto$ 01010010	S $\mapsto$ 01010011	T $\mapsto$ 01010100
U $\mapsto$ 01010101	V $\mapsto$ 01010110	W $\mapsto$ 01010111	X $\mapsto$ 01011000	y $\mapsto$ 01011001
Z $\mapsto$ 01011010				

      i. Was sind Vor- bzw. Nachteile des ASCII-Codes?
      ii. Rechne ein Paar der Binär-Codes in Dezimalzahlen um! Was fällt dabei auf?

   (c) Betrachte den folgenden Code:

$$\text{a} \mapsto 0,\ \text{b} \mapsto 10,\ \text{c} \mapsto 110,\ \text{d} \mapsto 1110$$

      i. Was unterscheidet diesen Code vom ASCII-Code?
      ii. Für welches Wort steht die Codierte Zeichenkette 110010?
      iii. Wie sieht die Codierung des Wortes „dab“aus?
      iv. Ist die Codierung eindeutig?
      v. Ist der Code a $\mapsto$ 0, b $\mapsto$ 10, c $\mapsto$ 1100, d $\mapsto$ 1101, e $\mapsto$ 1110, f $\mapsto$ 1111 eindeutig?
      vi. Ist der Code a $\mapsto$ 0, b $\mapsto$ 01, c $\mapsto$ 010, d $\mapsto$ 0110, e $\mapsto$ 01111, f $\mapsto$ 11111 eindeutig?
      vii. Welche Eigenschaften muss ein Code haben, um eindeutig zu sein?
      viii. Entwirf einen eindeutigen Code für die ersten 10 Buchstaben des Alphabets!

Arbeitsblatt zum Thema
EAN und ISBN Code

1. **EAN Code (Europäische Artikel Nummerierung)**
   Du kennst bestimmt die Strichcodes, die sich auf den meisten Lebensmitteln befinden:

EAN 13

EAN 8

Der Standard Produkt-Code EAN besteht aus 13 Ziffern. Für Produkte mit kleinerer Vielfalt gibt es eine kürzere Version, den EAN 8.

- Die ersten zwei oder drei Stellen des EAN Codes stehen meistens für das Land in dem der Arktikel verpackt wurde. Ausnahmen sind Bücher und Waren die direkt im Laden verpackt werden wie Wurst und Käse.

00-09	:	USA	20-29	:	Im Laden verpackt
30-37	:	Frankreich	40-43	:	Deutschland
45+49	:	Japan	54	:	Belgien
977	:	Zeitschriften	978,979	:	Bücher (gefolgt von ISBN-Nummer)

  (Eine vollständige Liste der Ländercodes findet sich unter
  http://www.evz.de/meldungen/ean-code.html)

- Die nächsten fünf Stellen stehen für den Hersteller des Artikels (EAN 13)
- Die letzten fünf Stellen werden herstellerintern für die verschiedenen Produkte vergeben (EAN 13)
- Die letzte Stelle ist die Prüfziffer. Diese wird nach dem folgenden Schema berechnet: (Das z steht für die noch unbekannte Prüfziffer)

EAN - Code	1	2	3	4	5	6	7	8	9	0	1	2	z	
Multiplikator	1	3	1	3	1	3	1	3	1	3	1	3	1	
Ergebnis	1	6	3	12	5	18	7	24	9	0	1	6	z	$\sum 92 + z$

  Die Prüfziffer wird nun so ergänzt, dass die Summe aus der Prüfziffer und den 12 Ergebnissen der Muliplikationen der EAN Ziffern mit 1 bzw. 3 die nächste durch 10 teilbare Zahl ist. In diesem Falle ist die Prüfziffer also eine 8, da $92 + 8 = 100$ die nächste durch 10 teilbare Zahl ergibt.

(a) Eine in Bremen-Findorff erworbene Cola-Dose hat die EAN-Nummer

- In welchem Land wurde diese Cola-Dose abgefüllt?

- An der Scanner-Kasse des Findorffer Supermarktes wird die Cola-Dose zunächst nicht akzeptiert. Welcher Fehler kann beim Scannen aufgetreten sein?
- Nehmen wir an der Scanner erfasst die Nummer 5 494000 000996. Wird er die Dose akzeptieren?

(b) Die ersten 7 Stellen des EAN Codes einer 0,33l Flasche Becks lauten

- Berechne die Prüfziffer
- Gibt es in diesem Fall Vertauschungsfehler, welche beim Scannen unbemerkt bleiben?

2. ISBN-Nummer
Die ISBN-Nummer (z.B. 3-492-21165-8) besteht aus 10 Stellen und ist ähnlich aufgebaut wie der EAN-Code:

- Die erste Stelle steht für das Land in dem das Buch erschienen ist (z.B. 0: USA, 3: Deutschland)
- Die nächsten drei Stellen stehen für den Verlag. (z.B. 387 Springer USA, 540 Springer D, 528 Vieweg D, 423 dtv,...)
- Die nächsten 5 Stellen stehen für das Buch
- Die letzte Stelle ist wie beim EAN die Prüfziffer, welche auf eine ähnliche Art und Weise bestimmt wird:(Das z steht wieder für die noch unbekannte Prüfziffer)

ISBN-Nummer	3	4	9	2	2	1	1	6	5	z	
Multiplikator	10	9	8	7	6	5	4	3	2	1	
Ergebnis	30	36	72	14	12	5	4	18	10	z	$\sum 201 + z$

Beim ISBN Code soll die so entstehende Summe allerdings durch 11 und nicht wie beim EAN durch 10 teilbar sein. Wir teilen 201 mit Rest durch 11 und erhalten

$$201 = 18 \cdot 11 + 3.$$

Die nächstgrößere durch 11 teilbare ganze Zahl ist also 209 $(= 201 + (11 - 3))$ d.h. die Prüfziffer ist die $8 = 209 - 201$. Die vollständige ISBN-Nummer lautet also 3-492-21165-8.

(a) Häufig wird die ISBN-Nummer in Verbindung mit dem EAN-Code angewendet:

- Aus welchem Land und welchem Verlag ist dieses Buch?
- Teste ob die Prüfziffern vom ISBN bzw. EAN-Code korrekt sind!

(b) Die ersten 9 Stellen der ISBN Nummer des Langenscheidt Taschenwörterbuches Englisch lauten:

- Berechne die Prüfziffern des ISBN bzw. EAN Codes!
- Welcher Code ist sicherer gegenüber Vertauschungen und Lesefehlern?
- Ein Strichcode oder auch Barcode sind eigentlich zwei Codes in einem. Was sagst du zu dieser Aussage?

Hinweis UPC-A Code
Auf vielen Produkten (z.B. CD, Druckerpratronen, etc.) findet sich der UPC-A Code. Dieser besteht aus 12 Stellen und ist dem EAN Code sehr ähnlich.

- Hierbei bezeichnet die erste Stelle, was in dem Code verschlüsselt ist.

0	-	Normaler regulärer UPC Code	1	-	Reserviert (evtl. für spätere Nutzung)
2	-	Produkte, die nach Gewicht berechnet werden.	3	-	National Drug Code (NDC)
4	-	UPC Code, welcher ohne Format-Einschränkungen verwendet werden kann.	5	-	Coupon
6	-	Normaler regulärer UPC Code	7	-	Normaler regulärer UPC Codex
8	-	Reserviert für spätere Nutzung	9	-	Reserviert für spätere Nutzung

- Die zweite bis sechte Ziffer des UPC Codes kennzeichnen den Hersteller des Produktes (UPC ID Nummer). Diese Nummer wird von der Uniform Code Council (UUC), 7051 Corporate Way - Suite 201, Dayton, OH 45359-4292, USA vergeben.
- Die Ziffern der siebten bis zur elften Stelle des UPC Codes bilden die individuelle Artikelnummer und klassifizieren das Produkt des Herstellers.
- Die abschließende zwölfte Stelle des Codes ist die Prüfziffer. Diese wird durch dieselbe Rechnung wie beim EAN Code bestimmt.

Arbeitsblatt zum Thema
Formalisierung des Verschiebechiffre II

Kongruenzen und Restklassen
Beim Ver- und Entschlüsseln ist dir bestimmt aufgefallen, dass du mit den Zahlen, denen du beim Codieren einen Buchstaben zugewiesen hast nicht auskommst und dass sogar zu jedem Buchstaben mehrere Zahlen gehören.

Beispiel
Im folgenden wollen wir mit einer Codierung arbeiten, die $A \mapsto 0, \ldots, Z \mapsto 25$ abbildet. Wir verschlüssel den Satz „Schluss mit lustig“ mit dem Schlüssel „R“:

„SCHLUSS MIT LUSTIG“	$\overset{codieren}{\longrightarrow}$	18 02 07 11 20 18 18 12 08 19 11 20 18 19 08 06
↓ *verschlüsseln(+„R“)*		↓ *verschlüsseln(+17)*
„JTYCLJJ DZK CLJZKX“	$\overset{decodieren}{\longleftarrow}$	35 19 24 28 37 35 35 29 25 36 28 37 35 36 25 23

Damit der direkte Weg (+„R“) und der Weg über die Codierung gleich sind, müssen verschiedene Zahlen für dieselben Buchstaben stehen. Z.B. steht in der oberen Zeile die 11 für das „L“ , in der unteren (nach dem Verschlüsseln) steht aber die 37 für das „L“. Gibt es in diesem Text noch ähnliche Fälle?

L $\mapsto$ 11, _____ C $\mapsto$ 02, _____ J $\mapsto$ 09, _____
D $\mapsto$ _____ K $\mapsto$ _____

- Wieviele Zahlen gehören zu einem Buchstaben?
- Repräsentiert jede ganze Zahl einen Buchstaben?
- Wie kannst du alle Zahlen erfassen, die ausser 0 den Buchstaben „A“ repräsentieren?

Die ganzen Zahlen werden also in 26 *Klassen* eingeteilt. Jede dieser Klassen wird repräsentiert durch einen Buchstaben des Alphabets:
Welche Zahlen gehören den jeweiligen Klassen an?

Klasse „A“	Klasse „B“	...	Klasse „X“	Klasse „Y“	Klasse „Z“	$\mathbb{Z}/26\mathbb{Z}$
0	1		23	24	25	
26	27		49	50	51	
		⋮				$\mathbb{Z}$

Die Menge der Klassen bezeichnen wir auch mit $\mathbb{Z}/26\mathbb{Z} = \{\text{Klasse A}, \text{Klasse B}, \ldots, \text{Klasse Z}\}$

Arbeitsblatt zum Thema
Restklassen

Wir haben gesehen, dass durch Kongruenzrechnung modulo n die Menge der ganzen Zahlen in Klassen (die sogenannten Restklassen) eingeteilt wird. Für $n = 6$ und $k \in \mathbb{N}$ sieht diese Einteilung wie folgt aus:

0 6 12 $\vdots$ $\underbrace{6k}$	1 7 13 $\vdots$ $\underbrace{1+6k}$	2 8 14 $\vdots$ $\underbrace{2+6k}$	3 9 15 $\vdots$ $\underbrace{3+6k}$	4 10 16 $\vdots$ $\underbrace{4+6k}$	5 11 17 $\vdots$ $\underbrace{5+6k}$	$\mathbb{Z}_0^+$
$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
$[0]_6$	$[1]_6$	$[2]_6$	$[3]_6$	$[4]_6$	$[5]_6$	$\mathbb{Z}/6\mathbb{Z}$

- Kannst du die Aussage der Tabelle in Worte fassen?
- Kannst du eine ähnliche Tabelle für $n = 7$ und $n = 12$ anfertigen?
- Für $n = 26$ haben wir eine Anwendung in der Kryptographie gefunden (welche?). Für welche anderen Werte für n gibt es Anwendungen im täglichen Leben?
- Bisher haben wir lediglich die *positiven* ganzen Zahlen betrachtet. In welche Spalten der Tabelle gehören die *negativen* ganzen Zahlen $\mathbb{Z}^- = \{-1, -2, -3, ...\}$?

Hinweis
Am übersichtslichsten kann man die Addition und Multiplikation mit Hilfe von Additions- und Multiplikationstafeln (hier für $\mathbb{Z}/6\mathbb{Z}$) darstellen:

$+$	0	1	2	3	4	5
0						
1						
2						
3						
4						
5						

$\cdot$	0	1	2	3	4	5
0						
1						
2						
3						
4						
5						

Arbeitsblatt zum Thema
Rechnen mit Restklassen

1. Berechne die folgenden Aufgaben und versuche eine allgemeine Vermutung zur Lösung der Aufgabe

$$[a]_n + m \text{ bzw. } [a]_n \cdot m,$$

$a, n, m \in \mathbb{N}$ aufzustellen.

Bsp.	$\underline{[5]_8 + 6}$	$\underline{[4]_9 + 8}$	$\underline{[8]_{12} + 4}$
	$5 + 6 = 11 \in [3]_8$	$4 + 8$	$8 + 4$
	$13 + 6$	$13 + 8$	$32 + 4$
	$21 + 6$	$31 + 8$	$56 + 4$
	$(5 + 8k) + 6$	$(4 + 9k) + 8$	$(8 + 12k) + 4$
	$\underline{[3]_8 \cdot 4}$	$\underline{[7]_9 \cdot 3}$	$\underline{[2]_{14} \cdot 7}$
	$3 \cdot 4$	$7 \cdot 3$	$2 \cdot 7$
	$11 \cdot 4$	$16 \cdot 3$	$16 \cdot 7$
	$19 \cdot 4$	$43 \cdot 3$	$44 \cdot 7$
	$(3 + 8k) \cdot 4$	$(7 + 9k) \cdot 3$	$(2 + 14k) \cdot 2$

2. Versuche mit Hilfe der folgenden Aufgaben und deiner obigen Vermutung, herauszufinden, nach welchen Regeln Restklassen addiert bzw. multipliziert werden.

Bsp.	In $\mathbb{Z}/6\mathbb{Z}$:	In $\mathbb{Z}/7\mathbb{Z}$:	In $\mathbb{Z}/12\mathbb{Z}$:
	$\underline{[5]_6 + [2]_6 =?}$	$\underline{[5]_7 + [2]_7 =?}$	$\underline{[3]_{12} + [4]_{12} =?}$
	$5 + 2 = 7 \in [1]_6$	$5 + 2$	$15 + 16$
	$11 + 8$	$12 + 9$	$39 + 64$
	$17 + 14$	$19 + 16$	$3 + 4$
	$(5 + 6k) + (2 + 6k)$	$(5 + 7k) + (2 + 7k)$	$(3 + 12k) + (4 + 12k)$
	$\underline{[5]_6 \cdot [2]_6 =?}$	$\underline{[5]_7 \cdot [2]_7 =?}$	$\underline{[3]_{12} \cdot [4]_{12} =?}$
	$5 \cdot 2$	$5 \cdot 2$	$15 \cdot 16$
	$11 \cdot 8$	$12 \cdot 9$	$39 \cdot 64$
	$17 \cdot 14$	$19 \cdot 16$	$3 \cdot 4$
	$(5 + 6k)(2 + 6k)$	$(5 + 7k)(2 + 7k)$	$(3 + 12k)(4 + 12k)$

Arbeitsblatt zum Thema
Teilen in Restklassen und Inverse

1. Der folgende deutsche Text wurde mit einer Verschlüsselungsfunktion der Form

$$f(P) \equiv a \cdot P \mod 26$$

verschlüsselt (lineares Chiffre). Versuche den Schlüssel zu finden und den Text damit zu entschlüsseln.

„DPCJJCN GUPQCN KG VPCE“

- Ist eine Verschlüsselung der Form $f(P) \equiv a \cdot P \mod 26$ schwerer zu knacken, als das einfache Verschiebe-Chiffre $f(P) \equiv a + P \mod 26$?
- Verschlüssele den entschlüsselten Text mit der Verschlüsselungsfunktion

$$f(P) \equiv 4 \cdot P \mod 26$$

diese Funktion nicht sinnvoll ist?

2. Gib eine Restklasse $[x]_n$ an, so dass die Gleichungen erfüllt sind!

(a)
$$[4]_8 + [x]_8 = [0]_8$$
$$[4]_9 + [x]_9 = [0]_9$$
$$[4]_{10} + [x]_{10} = [0]_{10}$$
$$[4]_{11} + [x]_{11} = [0]_{11}$$

(b)
$$[4]_8 \cdot [x]_8 = [0]_8$$
$$[4]_9 \cdot [x]_9 = [0]_9$$
$$[4]_{10} \cdot [x]_{10} = [0]_{10}$$
$$[4]_{11} \cdot [x]_{11} = [0]_{11}$$

(c)
$$[4]_8 \cdot [x]_8 = [1]_8$$
$$[4]_9 \cdot [x]_9 = [1]_9$$
$$[4]_{10} \cdot [x]_{10} = [1]_{10}$$
$$[4]_{11} \cdot [x]_{11} = [1]_{11}$$

(d)
$$[2]_5 \cdot [x]_5 = [1]_5$$
$$[3]_5 \cdot [x]_5 = [1]_5$$
$$[4]_5 \cdot [x]_5 = [1]_5$$

(e)
$$[2]_6 \cdot [x]_6 = [1]_6$$
$$[3]_6 \cdot [x]_6 = [1]_6$$
$$[4]_6 \cdot [x]_6 = [1]_6$$
$$[5]_6 \cdot [x]_6 = [1]_6$$

(f)
$$[3]_8 \cdot [x]_8 = [1]_8$$
$$[4]_8 \cdot [x]_8 = [1]_8$$
$$[5]_8 \cdot [x]_8 = [1]_8$$
$$[6]_8 \cdot [x]_8 = [1]_8$$
$$[7]_8 \cdot [x]_8 = [1]_8$$

3. Trage in die folgende Tabelle ein, ob die Gleichung $[a]_n \cdot [x]_n = [1]_n$ lösbar ist oder nicht und bestimme den $\mathrm{ggT}(a, n)$!
Beispiel:

n=6

a	0	1	2	3	4	5
x	-	*1*	-	-	-	*5*
$\mathrm{ggT}(a,n)$	*6*	*1*	*2*	*3*	*2*	*1*

n=4

a	0	1	2	3
x				
$\mathrm{ggT}(a,n)$				

n=10

a	0	1	2	3	4	5	6	7	8	9
x										
$\mathrm{ggT}(a,n)$										

Hast du eine Vermutung, wann die Kongruenz $a \cdot x \equiv 1 \mod n$ lösbar ist? Wenn du dir nicht sicher bist, erstelle weiterer Tabellen für $n = 5, 9, 10$!

Arbeitsblatt zum Thema
Vignere- und Vernam-Chiffre

1. **Vignere-Chiffre**
   Wir stellen uns vor, wir wollen den Text

   „Treffen morgen um drei“

   in Blöcken von je zwei Buchstaben verschlüsseln. (ohne die Leerzeichen)
   Wir teilen den Text in Buchstabenpaare (xy),

   (TR)(EF)(FE)(NM)(OR)(GE)(NU)(MD)(RE)(IX)

   wobei wir eventuelle Leerstellen am Ende auffüllen, ohne den Inhalt zu gefärden (z.B. mit einem „X“).
   Wir verschlüsseln nun nicht Buchstabenweise, sondern Paarweise.
   Als Schlüsselwort benutzen wir z.B. das Buchstabenpaar (GN). Zum Verschlüsseln schreiben wir sich stetig wiederholend das Schlüsselwort unter den zu verschlüsselnden Text und veschlüsseln dann jeden Buchstabe des Klartextes mit dem unter ihm stehenden Schlüsselwortbuchstaben:

   (TR)(EF)(FE)(NM)(OR)(GE)(NU)(MD)(RE)(IX)
   (GN)(GN)(GN)(GN)(GN)(GN)(GN)(GN)(GN)(GN)
   ↓
   (ZE)(KS)(LR)(SZ)(UE)(MR)(TH)(SQ)(XK)(OM)

   „Treffen morgen um drei“ $\mapsto$ „Zekslre zuemrt hs qxkom“

   (a) Was fällt bei dieser Verschlüsselung auf?
   (b) Ist diese schwerer zu knacken als das Verschiebe-Chiffre?
   (c) Welche Information über deutsche Sprache ist nun für eine Häufigkeitsanalyse nötig?
   (d) Wie sieht die zugehörige Verschlüsselungsfunktion aus?
   (e) Verschlüssele den Text in 3-er Blöcken mit dem Schlüsselwort (UND)!

2. **Vernam-Chiffre**
   Der obige Text soll nun mit dem Schlüsselwort (EINSZWEIDREIVIERFUE) verschlüsselt werden.

   (a) Wie lautet nun der verschlüsselte Text?
   (b) Ist diese Verschlüsselung mit Hilfe einer statistischen Analyse zu knacken?
   (c) Wie könnte man dieses System nahezu perfekt sicher machen?
   (d) Wo liegen die Vor- und Nachteile dieses Systems? Warum hat es sich wohl in der Geschichte nicht durchsetzen können?

Arbeitsblatt zum Thema
Potenzfunktionen modulo n

1. **Potenzfunktionen modulo p**
   Wir betrachten Funktionen der Form

$$\begin{aligned} f : \mathbb{Z}/p\mathbb{Z} &\to \mathbb{Z}/p\mathbb{Z} \\ x &\mapsto x^a \mod p, \end{aligned}$$

   wobei p eine Primzahl ist.

   (a) Versuche mit Hilfe von Maple oder der Modulo-Funktion des TI-92 herauszufinden, welche Bedingung a erfüllen muss, damit f eineindeutig ist.
   (*Hinweise: Betrachte auch den $ggT(a, p-1)$!*
   Du kannst eine Tabelle der Form

p	$\varphi(p)$	e	$ggT(a,p-1)$	$x=0$	1	$\ldots$	$p-1$
				$x^a=$			

   benutzen.)

   (b) Untersuche, falls du es noch nicht getan hast, die Funktionen

$$\begin{aligned} x &\mapsto x^{p-1} \mod p \quad \text{und} \\ x &\mapsto x^p \mod p. \end{aligned}$$

2. **Potenzfunktionen modulo n**
   Wir betrachten nun Funktionen der Form

$$\begin{aligned} f : \mathbb{Z}/n\mathbb{Z} &\to \mathbb{Z}/n\mathbb{Z} \\ x &\mapsto x^a \mod n, \end{aligned}$$

   wobei $n = p \cdot q$ das Produkt aus zwei Primzahlen ist.

   (a) Stelle mit Hilfe der Ergebnisse der Aufgabe 1 eine Vermutung auf, unter welchen Bedingungen diese eindeutig sind!

   (b) Überprüfe deine Vermutung. Erweist sie sich in Beispielen als richtig, versuche eine Begründung für deine Vermutung zu finden!

Arbeitsblatt zum Thema
Blockcodierung

1. **Blockcodierung**
   Wir betrachten den Satz „Treffen morgen um drei“. Das Ziel ist es, ihn in Blöcken von je zwei Buchstaben zu codieren, d.h. je zwei Buchstaben wird bei der Codierung eine Zahl zugewiesen.
   Dazu benötigen wir zunächst eine einfache Buchstaben-Codierung. Außerdem wollen wir die Wörter aus den Buchstaben wie die Zahlen aus Ziffern im Stellenwertsystem betrachten.
   Beispiel
   Die Zahl 4567_{10} bedeutet nichts anderes als $4 \cdot 10^3 + 5 \cdot 10^2 + 6 \cdot 10^1 + 7 \cdot 10^0$ oder als Stellenwerttabelle:

...	10^3	10^2	10^1	10^0
0	4	5	6	7

   (a) Wie sieht die Stellenwert-Tafel im Achter-System aus? Welchen Wert (im Dezimalsystem) hat die Zahl 3425_8?

   (b) Wie sieht die Stellenwert-Tafel im Binär-System aus? Welchen Wert (im Dezimalsystem) hat die Zahl 11011_2?

   (c) Wie sieht die Stellenwert-Tafel im 26-System aus? Welchen Wert (im Dezimalsystem) hat die Zahl BAC_{26}?

   (d) Welchen Zahlwert im Dezimalsystem haben die Zahlen

TR_{26}	_____	OR_{26}	_____	MD_{26}	_____
RE_{26}	_____	GE_{26}	_____	RE_{26}	_____
FE_{26}	_____	NU_{26}	_____	IX_{26}	_____
NM_{26}	_____				

   (e) Ein Wort wurde als Dreierblock codiert und lautet nun „16346“. Wie lautet das Wort?

   (f) Beschreibe den Vorgang der Decodierung eines in Blöcke der Länge n codierten Wortes! Ist diese immer eindeutig?

   (g) Welchen Zahlwert im Binärsystem haben die Zahlen

TR_{26}	_____	OR_{26}	_____	MD_{26}	_____
RE_{26}	_____	GE_{26}	_____	RE_{26}	_____
FE_{26}	_____	NU_{26}	_____	IX_{26}	_____
NM_{26}	_____				

2. **Verschlüsseln**

   (a) Verschlüssele den oben in zweierblöcken codierten Satz „Treffen morgen um drei“mit dem Schlüssel 56!