MATHEMATISCHE GRUNDLAGEN DER KRYPTOLOGIE

KLASSENSTUFE 10

UNTERRICHTSVERLAUF UND HINTERGRUND

Markus Kammerer – E-Mail: markus.kammerer@zsl-rsfr.de – April 2020

Inhaltsverzeichnis

Einleitung

Kryptographie – ...

Ein spannendes, allgegenwärtiges Thema in unserer modernen Industriegesellschaft. Ob Geldautomat, Tankstelle, Internetbanking, Webshops, verschlüsselte Handys oder mehr... Eine Welt ohne verschlüsselte Kommunikation ist heute nicht mehr vorstellbar. Die Mathematik hinter dieser Thematik beginnt recht einfach, wird dann jedoch schnell komplex, abstrakt und reicht tief in das Gebiet der Zahlentheorie hinein. Dennoch kann es (bei entsprechender Reduzierung) gelingen, in der Schule einen wichtigen Einblick in die prinzipiellen Zusammenhänge und Strategien zu ermöglichen, der in ein tieferes Verständnis dieses uns alle umgebenden Themas ermöglicht.

Da im Bemühen um Reduktion auch stets die Gefahr der evtl. zu groben Vereinfachung liegt, bin ich sehr dankbar für kritsche Hinweise. Ebenso freuen würde ich mich über ergänzende Hinweise, didaktisch-methodischer oder fachlicher Art, die sich oft erst bei einer Bearbeitung im Unterricht ergeben. Hier erschließt sich ein weites Feld, einerseits in alternativen mathematischen Zugängen und Beweisen, von denen es viele gibt, als auch z.B. im Einsatz digitaler Werkzeuge: Angefangen von eigenen Programmierungen mit Tabellenkalkulationen oder CASen bis hin zum Einsatz fertiger Kryptotools, wobei der Übergang zur Informatik hier fließend ist. Eine erschöpfende Bearbeitung aller dieser Facetten in einem Unterrichtsgang erscheint wenig sinnvoll, wenn der rote Faden gewahrt werden soll. An mir besonders lohnenden Stellen sind mögliche Erweiterungen in Form von Hinweisen angeregt.

Der Unterrichtsgang ist wo immer möglich durch Arbeitsblätter umgesetzt, die ein selbständiges Arbeiten der SchülerInnen ermöglichen sollen. An manchen Stellen sehe ich jedoch die ordnende Hand der Lehrperson als unerlässlich an. Meine Hoffnung ist, dass diese Aufbereitung einen soliden Grundstein für einen methodisch abwechslungsreichen Unterricht bietet.

Ihnen und Ihren SchülerInnen wünsche ich viel Erfolg und Freude am weiteren Eintauchen in die Hintergründe der Kryptographie.

Markus Kammerer *markus.kammerer@zsl-rsfr.de*

Im April 2020

Der Unterricht

Einige grundsätzliche Bemerkungen

Zu den Arbeitsblättern:

In den Aufgaben der Arbeitsblätter ist der verpflichtende Inhalt des Bildungsplanes abgedeckt, manche Aufgaben oder -teile gehen darüber hinaus: diese sind je nach Schwierigkeitsgrad mit einem oder mehreren Sternen * in der Nummerierung gekennzeichnet.

Hin und wieder kann es zu Schwierigkeiten mit der Darstellung von Sonderzeichen kommen, z.B. im Thema „Kongruenzen“ beim Zeichen „≠“. Zur korrekten Darstellung muss der Zeichensatz *Arial Unicode MS* verfügbar sein (Zeichencode U+2262).

Zu den Hilfsmitteln:

Generell kann bei kleineren Zahlen der WTR als Hilfsmittel eingesetzt werden. Bei den Bemerkungen zu den entsprechenden Arbeitsblättern sind die benötigten Befehle ausgeführt. Bei größeren Zahlen kann eine Tabellenkalkulation hilfreich sein (z.B. bei Probierlösungen → Arbeitsblatt *03b.0_mgk_Verschlüsseln_durch_modulare_Mult*). Auch der Windows-Rechner bietet eine höhere Grenze der exakten Berechnung als der übliche WTR. Neben dem im Material zum Informatikunterricht enthaltenen Kryptotool lassen sich bei einfacher Suche im Internet sehr viele modulo-Rechner in unterschiedlichster Ausgestaltung (jedoch deshalb auch in unterschiedlicher Komplexität) finden. An passender Stelle und zur eigenverantwortlichen Kontrolle der Ergebnisse sind diese sehr gut einsetzbar.

Hinweis zur Verzahnung mit „Informationsgesellschaft und Datensicherheit“ (IuD):

Das Konzept der asymmetrischen Verschlüsselung ist vom Grunde her im Bereich der Informatik angesiedelt. Es ergeben sich jedoch auch im Mathematikunterricht *mathematische Grundlagen der Kryptologie* sinnvolle Gelegenheiten zur Besprechung. Aus diesem Grund sollte man sich absprechen:

Fall 1 : Im Mathematikunterricht wurde noch nicht mit *mathematischen Grundlagen der Kryptologie (mgK)* begonnen: Dann wird im Informatikunterricht *Informationsgesellschaft und Datensicherheit (IuD)* diese Einheit ‚normal‘ am Stück unterrichtet.

Fall 2: Der Mathematikunterricht beginnt mit *mathematischen Grundlagen der Kryptologie* bevor der Informatikunterricht mit *Informationsgesellschaft und Datensicherheit (IuD)* beginnt: Dann werden die beiden Kapitel

IuD - Wiederholung Kryptologie Klasse 8 und

IuD - Asymmetrische Verschlüsselung

im Mathematikunterricht als Einstieg unterrichtet und später im Informatikunterricht nicht mehr berücksichtigt. Der Informatikunterricht beginnt dann mit dem Kapitel *Man-in-the-middle*.

Tabellarische Stundenübersicht

Im Folgenden ist ein möglicher Stundenverlauf tabellarisch aufgelistet. Inhalte, die das Thema abrunden oder in einen größeren Kontext einbetten, jedoch über den verpflichtenden Inhalt des Bildungsplans hinausgehen, sind in der Spalte „Optional“ aufgeführt.

	Stundenthema	**Kerncurriculum**	**Optional**
1 – 3	CÄSAR-Verfahren, modulo-Operation, Kongruenzrelation	*01a.0_mgk_Caesar* oder *01a.1_mgk_Alternative_Caesar* *01b_mgk_mod_und_Kongruenz*	
4, 5	Modulares Addieren	*02a_mgk_Modulares_Addieren*	*Nr.5*
6, 7	Modulares Multiplizieren	*02b_mgk_Modulares_Multiplizieren*	*Nr.5*
8 – 10	Modulares Potenzieren	*02c_mgk_Mod. Potenzieren*	*Aufgabenteil b)*
11 – 15	Ver- und Entschlüsseln durch (modulare) Multiplikation	*03b.0_mgk_Verschlüsseln_durch_modulare_Mult*	*03a_mgk_Einstieg* *Seite 3 – Exkurs zu Einwegfunktion und Primzahlsuche*
+2h	Diophantische Gln.		*optionaler Exkurs: 03b.1_mgk_diophantische_Gln*
	Bestimmung des multiplikativen Inversen in mod	*03c_mgk_Erw_Euklid_Alg_S-zentriert*	*Nr.4*
+1h	Einweg- und Falltürfunktionen		*optionaler Exkurs: 03d_mgk_Einweg-und_Falltürfunktionen*
+2h	Neutrale und Inverse Elemente		*optionaler Exkurs: 03e_mgk_ Neutrale_und_inverse_Elemente*
16 - 18	RSA an einfachen Beispielen	*04_mgk_Das_RSA-Verfahren*	
	Vergleich aller Verfahren	*06_iud_ab_vergleich_verfahren*	

Die CÄSAR-Verschlüsselung

Zur Einleitung des Themas wird im Material eine sehr leichte und bekannte Wiederholung angeboten (*01a.0_mgk_Caesar*): ein gegebener Geheimtext soll entschlüsselt werden. Da die SuS in den vergangenen Jahren mehrere Verschlüsselungen kennengelernt haben, sollte dies kein Problem darstellen. Tipps in Form von Häufigkeitstabellen sollten dazu ausliegen. Der Fokus liegt bei dieser Einführungsaufgabe auf einer motivierenden Eröffnungsstunde und der Beschreibung des Verfahrens durch lediglich eine einzige Zahl, nämlich die Verschiebung von Klar- und Geheimalphabet gegeneinander. Mit der Methode des Think – Pair – Share soll der Schlüssel „so kompakt wie möglich“ dargestellt werden. Hierbei können sich naturgemäß auch andere als die intendierten Lösungen ergeben, spätestens in der Besprechung jedoch wird dann der Sachverhalt der „Schlüsselzahl“ und die Verbindung zur Modulo-Rechnung thematisiert, die zwingend bei Überschreitung von Z auftritt. Als Lösung wird die bekannte Formulierung mit der Verallgemeinerung mod 26 erarbeitet.

Daten und Diagramme zur Häufigkeitsverteilung der Buchstaben (als Tipp zur Auslage im Klassenraum) findet man z.B. bei

http://kryptografie.de/kryptografie/kryptoanalyse/haeufigkeitsverteilung.htm.

Hierbei kann auch die abweichende Verteilung in anderen Sprachen thematisiert werden.

Eine Alternative mit einem stärker geführten Vorgehen findet sich im Arbeitsblatt *01a_mgk_Alternative_Caesar.*

Wiederholung der Kenntnisse aus den vorangegangenen Klassen

Ein wichtiger Punkt, den es den Schülerinnen und Schülern klarzumachen gilt, ist der Unterschied zwischen Kongruenzen und Gleichheiten. In späteren Rechnungen und Beweisen wird immer wieder beides auftauchen, an manchen Stellen werden sogar aus Kongruenzen Gleichheiten erzeugt (Lösung linearer Kongruenzgleichungen im Zusammenhang mit der Bestimmung des multiplikativen Inversen einer Zahl a mod n). Je früher hier eine Sensibilisierung und Klärung erfolgt, umso einfacher wird in späteren Stunden das Verfolgen der entsprechenden Umformungen sein.

Das hierfür angelegte Arbeitsblatt ist *01b_mgk_mod_und_Kongruenz.*

Ein Rückgriff auf schon bekannte Aufgaben aus Klasse 9 kann hier ebenfalls sinnvoll sein, falls der Inhalt tiefer wiederholt werden muss.

Rechenregeln in mod: Addition, Multiplikation und Potenzierung

Zunächst lernen die SuS exemplarisch die grundlegenden Rechengesetze der modularen Addition kennen. Hierbei wird der Unterschied Kongruenz – Gleichheit explizit thematisiert, z.B.

$$a \bmod c + b \bmod c \equiv (a + b) \bmod c \quad \Leftrightarrow \quad (a \bmod c + b \bmod c) \bmod c = (a + b) \bmod c$$

Im Arbeitsblatt *02a_mgk_Modulares_Addieren* wird der Beweis zur Rechenregel in Form einer Aufgabe angesprochen. Beweise in diesem Teilthema gehen inhaltlich über den Bildungsplan hinaus; dieser fordert verbindlich lediglich die Anwendung und Verifizierung der Gesetze im Rahmen expliziter Zahlenbeispiele (BP-Item 3.3.2.1 (1)).

Der Beweis bietet jedoch wertvolle binnendifferenzierende und methodische Aspekte. Er kann z.B. mit der Methode der wachsenden Gruppe und/oder gestuften Hilfen bearbeitet werden. Ein Vorschlag für gestufte Hilfen befindet sich im Anschluss an das Arbeitsblatt.

Die modulare Multiplikation wird vom Ablauf her gleich wie die Addition behandelt (Arbeitsblatt *02b_mgk_Modulares_Multiplizieren*). Zum Beweis der Regel gilt das bei der Addition Gesagte. Für eine Bearbeitung spricht zudem, dass der Beweis prinzipiell dem Vorgehen des Beweises bei der Addition folgt, hier jedoch im Detail andere Umformungen mit Hilfe des Distributivgesetzes vorgenommen werden müssen. Nach Durchdringung des Additionsbeweises können die SuS hier ein Erfolgserlebnis erfahren.

Um die benötigten Rechenregeln zu komplettieren, wird im Arbeitsblatt *02c_mgk_Mod. Potenzieren_XX* zunächst diese Rechenregel nach bekannter Weise motiviert. Hierzu existieren zwei Kopiervorlagen; eine für die Verwendung des TI 30-X Pro Multiview, eine für den CASIO fx-87 DE X Classwiz.

Der (ebenfalls optionale) Beweis der Regel ist lohnend, weil die SuS hier leicht Erfolg haben können, obwohl sie sich von den bisherigen Beweisen lösen müssen: er weicht von den Vorgehensweisen, die bei Addition und Multiplikation zum Ziel führten, ab und stützt sich auf einfache Umformungen, bei denen lediglich auf die Definition der Potenz und die Multiplikationsregel zurückgegriffen wird. Zudem ist er sehr kurz:

z.z: $a^b \bmod c = (a \bmod c)^b \bmod c$

Bew: $a^b \bmod c = (a \cdot \cdots \cdot a) \bmod c = (a \bmod c \cdot \cdots \cdot a \bmod c) \bmod c = (a \bmod c)^b \bmod c$ ■

Daraufhin erfahren die SuS die Probleme beim Bestimmen großer Potenzen: Nach einer kurzen Sensibilisierungsphase, in der die SuS die schnell erreichten Grenzen des WTR erleben, wird ein effizientes Verfahren zur Berechnung von $a^b \bmod n$ bei großen a und b hergeleitet.

Bem: Genausogut wäre es möglich, das Thema $a^b \bmod n$ hier auszulassen und bei der Besprechung des RSA-Verfahrens die Problematik zu entdecken und zu lösen. Dann jedoch wäre nach der Einführung von RSA ein recht großer Block Theorie und Übung notwendig, bevor die SuS in der Lage wären, auch explizit zu verschlüsseln. Eine Platzierung dieses Themas an der hier vorgeschlagenen Stelle ermöglicht es, die Einheit insgesamt mit einer Chiffrier- und Dechiffrier-Phase abzuschließen. Eine erinnernde Bemerkung an das zeitlich etwas zurückliegende Thema befindet sich im Arbeitsblatt.

Verschlüsseln mittels modularer Multiplikation

Nachdem nun die grundlegenden mathematischen Kenntnisse gelegt sind, erfolgt der Übergang zur Anwendung in der Kryptographie:

Im Einstieg (*03a_mgk_Einstieg_-_Verschlüsseln_durch_Mult*) wird die Situation eines Spions beschrieben, der die Chance hat, das (stark vereinfachte) Verschlüsselungsverfahren eines Landes zu hacken. Ziel ist es hierbei, durch Kenntnis einiger Bedingungen das Verschlüsselungsprinzip *Dezimalcode des Zeichens → Verschlüsselung durch die Operation „Dezimalcode · 3 mod 55“* zu entdecken.

Bem.: Die ASCII-Tabelle ist fiktiv, da in der realen Tabelle viele für diesen Zweck nicht brauchbare Sonderzeichen enthalten sind.

Zur Erleichterung kann hierbei ein Exkurs in eine Tabellenkalkulation vorgenommen werden. Beispielsweise in OpenOffice als auch in EXCEL lautet der entsprechende Befehl =REST(Zahl;Divisor). Die Verwendung einer bedingten Formatierung erleichtert das Auffinden der gesuchten Zahlen. Dieses Blatt hat neben einem Einstieg, der von der Modulo-Problematik entlastet, vor allem auch motivatorischen Charakter. Sollte dieser Effekt nicht als notwendig erachtet werden, kann inhaltlich ohne Bruch oder Lücke auch gleich zu dem Arbeitsblatt *03b.0_mgk_Verschlüsseln_durch_modulare_Mult* übergegangen werden.

Im Arbeitsblatt *03b.0_mgk_Verschlüsseln_durch_modulare_Mult* wird die mod-Rechnung zunächst noch nicht bemüht. Durch einfach nachzuvollziehende Operationen wird das Prinzip verdeutlicht und herausgearbeitet, dass zum Hacken einer Nachricht die Faktorisierung das entscheidende Hilfsmittel ist.

Dadurch, dass auf den Seiten 1 und 2 von Hand multipliziert und faktorisiert wird, entsteht ein Gefühl für die auftretenden Schwierigkeiten beim Faktorisieren und bereitet intuitiv das Verständnis für Einweg- und Falltürfunktionen (*03d_mgk_Einweg-und_Falltürfunktionen*) vor. Der auf Seite 3 vorgenommene Hinweis nennt einige derzeit aktuelle Zahlen bei der Benutzung digitaler Hilfsmittel. Eine genauere Betrachtung der angewandten Algorithmen erfolgt in der Mathematik nicht.

Seite 3 des Arbeitsblattes beinhaltet einige Quellen, die die Suche nach Primzahlen zum Thema haben. Hier bietet sich eine Möglichkeit für historische Recherchen der SuS zur Entwicklung der Suche nach Primzahlen, der aktuell größten Primzahl usw.

Darauf aufbauend wird nun das Ver- und Entschlüsseln mittels modularer Multiplikation behandelt (Arbeitsblatt *03b.0_mgk_Verschlüsseln_durch_modulare_Mult,* Seiten 4 und 5). Hierbei stellt sich die Frage der Bestimmung des modularen Inversen, das bei der Entschlüsselung benötigt wird. Die Definition des Inversen wird mit Kenntnissen der Modulo-Operation so umgeschrieben, dass Probierlösungen möglich werden. Hierzu erfolgen einige Übungen. Das Teilthema endet mit einer Ver- und Entschlüsselungssequenz, bei der sich die SuS mittels modularer Multiplikation verschlüsselte Botschaften schicken und die erarbeiteten Zusammenhänge im Kontext anwenden.

Der Erweiterte Euklidische Algorithmus - Exkurs „Diophantische Gleichungen“

Zur Bestimmung des multiplikativen Inversen in mod ist im allgemeinen eine lineare Kongruenzgleichung zu lösen. Systematisch wird dies durch Anwendung des Erweiterten Euklidische Algorithmus erzielt. Der hierzu gehörende mathematische Hintergrund der diophantischen Gleichungen (s.u.) kann hierbei wertvolle mathematische Einsichten in umfangreichere Strukturen und v.a. auch in die Frage der Lösbarkeit linearer Kongruenzgleichungen ermöglichen und systematisches Denken schulen. Wird dies angestrebt, so kann als tiefergehende und allgemeinere Einbettung das Arbeitsblatt *03b.1_mgk_diophantische_Gln* bearbeitet werden. Hierbei wird das Auffinden von Lösungen, das Entdecken unendlich vieler Lösungen und ihrer Struktur schülerzentriert erarbeitet. Sollte dies nicht gewünscht werden, kann der Unterrichtsgang auch direkt mit der Erarbeitung des Erweiterten Euklidischen Algorithmus beginnen.

Zunächst jedoch einige Bemerkungen zum Exkurs:

Mathematischer Hintergrund: die diophantischen Gleichungen

Kurzer Exkurs: dieser Unterrichtsinhalt ist vom Bildungsplan nicht gefordert.

Bei der Bestimmung des multiplikativen Inversen, das als Entschlüsselungszahl d beim Verschlüsseln mittels Multiplikation benötigt wird, muss der Spezialfall einer linearen diophantischen Gleichung gelöst werden.

Def.: Eine lineare diophantische Gleichung ist eine Gleichung der Form

$a \cdot x + b \cdot y = c$ mit Variablen x, y und Koeffizienten a, b, $c \in \mathbb{Z}$.

Forderung: Die Lösungen x, y sollen ebenfalls ganzzahlig sein.

Anhand eines exemplarischen Beispiels kann man auch mit den SuS schnell erarbeiten, dass es unendlich viele Lösungen gibt und auch die Struktur dieser Lösungen ist relativ leicht durchschaubar, wenn man y in Abhängigkeit von x ausdrückt.

Geeignete Beispiele sind die folgenden Gleichungen mit ihren Lösungen:

Gleichung	Lösungen
$2 \cdot x + 1 \cdot y = 4$	(0 ; 4) , (1 ; 2) , (2 ; 0) ; (-1 ; 6) ; … ; (z ; 4 – 2z)
$3 \cdot x + 1 \cdot y = 2$	(0 ; 2) , (1 ; -1), (2 ; -4); (-1 ; 5) ; … ; (z ; 2 – 3z)
$3 \cdot x + 2 \cdot y = 4$	(0 ; 2) , (2 ; -1) , (-2 ; 5) , …

Die allgemeine Lösung ist hier etwas schwerer einzusehen:

Bem.: Ein Zurückbesinnen auf die Inhalte der Einheit lineare Gleichungen und Funktionen in Klasse 7 kann hilfreich sein: auch dort wurden schon lineare Gleichungen der Form $a \cdot x + b \cdot y = c$ in die Form $y = m \cdot x + d$ umgewandelt.

Umformung (mit der Umbennenung x = z zur Unterscheidung zwischen Lösung und Variable) ergibt $x = z\,;\, y = \frac{4 - 3z}{2}$. Hier ist y genau dann ganzzahlig, wenn 4 – 3z gerade ist, also wenn z gerade ist, d.h. wenn $z = 2k\,;\, k \in \mathbb{Z}$.

Es ergibt sich $y = \frac{4 - 3 \cdot 2k}{2}\,;\, k \in \mathbb{Z}$.

Damit lautet die allgemeine Lösung hier $(x\,;\, y) = (2k\,;\, 2 - 3k)\,;\, k \in \mathbb{Z}$

Jedoch sind nicht alle Gleichungen lösbar, z.B. $5 \cdot x + 10 \cdot y = 4$

Umformung ergibt $x = z\,;\, y = \frac{4 - 5z}{10}$ y ist also genau dann ganzzahlig, wenn gilt:

$4 - 5z = n \cdot 10$ mit $n \in \mathbb{N}$. Systematisches Ausprobieren mit z zeigt jedoch schnell, dass dies nicht erfüllbar ist. Damit besitzt diese diophantische Gleichung keine Lösungen.

Allgemein gilt der **Satz:**

Gegeben sei die diophantische Gleichung $a \cdot x + b \cdot y = c$; $a, b, c \in \mathbb{Z}$.

Diese besitzt genau dann Lösungen $x, y \in \mathbb{Z}$, wenn gilt: $ggT(a,b) \mid c$

Die Grundlage dieses Satzes ist als Lemma von Bézout bekannt. Dieser besagt, dass sich der ggT(a; b) als Linearkombination darstellen lässt.

Der Beweis ist nicht sehr schwer, jedoch lohnt hier der Zeitaufwand eher nicht

Zur Vollständigkeit sei er hier jedoch angebracht:

„⇒“: *sei (x_0, y_0) eine Lösung von $a \cdot x + b \cdot y = c$, also gilt $a \cdot x_0 + b \cdot y_0 = c$*

Klar ist: $ggT(a\,;\,b) \mid a \cdot x_0 + b \cdot y_0$, wie auch SuS leicht einsehen.

Und da $a \cdot x_0 + b \cdot y_0 = c$, folgt: $ggT(a\,;\,b) \mid c$.

„⇐“: *$ggT(a\,;\,b) \mid c \Rightarrow \exists\, q \in \mathbb{Z}$ mit $c = q \cdot ggT(a\,;\,b)$*

*Außerdem existiert nach dem Lemma von Bézout eine Linearkombination des ggT(a ; b). *Argumentation durch Konstruktion: Diese liefert der Erweiterte Euklidische Algorithmus (s.u.)*. Also existiert $s \cdot x_0 + r \cdot y_0 = ggT(a\,;\,b)$*

Multiplikation der Gleichung mit $h = \frac{c}{ggT(a\,;\,b)}$ liefert die Behauptung. ∎

Anmerkung: Die Plausibilisierung, dass die Lösbarkeit etwas mit den Teilern von a und b zu tun hat, ließe sich durch Besprechung einiger Beispiele wie im obigen Beispiel $5 \cdot x + 10 \cdot y = 4$ und evtl. dem anschließenden forschenden Auftrag an die SuS: „Konstruiert diophantische Gleichungen, die keine Lösung besitzen. Auf was kommt es dabei an?“ erreichen.

Dieser Exkurs ist für die Unterrichtseinheit inhaltlich nicht zwingend notwendig. Wird darauf verzichtet, so kann das Arbeitsblatt *04b.1_mgk_diophantische_Gln.* ausgelassen und gleich mit der Erarbeitung des Erweiterten Euklidischen Algorithmus' (wahlweise mit Arbeitsblatt *04c_mgk_Erw_Euklid_Alg_S-zentriert)* fortgefahren werden.

Bem.: es ergibt sich hiermit eine interessante Gelegenheit, auf die historischen Entwicklungen der Gleichungslehre zu sprechen zu kommen: Diophantos von Alexandria, nach dem die Gleichungen benannt sind, auf dessen Grabstein sich sogar eine Gleichung in verbalisierter Form befindet:

„Hier das Grabmal deckt Diophantos — ein Wunder zu schauen:

Durch des Entschlafenen Kunst lehrt dich sein Alter der Stein.
Knabe zu bleiben verlieh ein Sechstel des Lebens ein Gott ihm;
Fügend das Zwölftel hinzu, ließ er ihm sprossen die Wang;
Steckte ihm drauf auch an nach dem Siebtel die Fackel der Hochzeit,
Und fünf Jahre nachher teilt' er ein Söhnlein ihm zu.
Weh! unglückliches Kind, so geliebt! Halb hatt' es des Vaters
Alter erreicht, da nahms Hades, der schaurige, auf.
Noch vier Jahre den Schmerz durch Kunde der Zahlen besänft'gend
Langte am Ziele des Seins endlich er selber auch an.“

Algebraisierung: $x = \frac{x}{6} + \frac{x}{12} + \frac{x}{7} + 5 + \frac{x}{2} + 4$ *ergibt die Lösung* $x = 84$.

- Ende des Exkurses -

Zurück zu den BP-Inhalten „Lösung linearer Kongruenzgleichungen und Erweiterter Euklidischer Algorithmus“:

Im Fall der reinen Bestimmung des multiplikativen Inversen ohne den Hintergrund der Lösbarkeitsfrage, die im Exkurs „diophantische Gleichungen“ angesprichen wurde, reduziert sich die Schwierigkeit:

Die Bestimmungsgleichung des multiplikativen Inversen $e \cdot d \equiv 1 \bmod n$ lässt sich zur (diophantischen) Gleichung äquivalent umformen:

$$e \cdot d \equiv 1 \bmod n \quad \Leftrightarrow \quad e \cdot d = k \cdot n + 1 \quad \Leftrightarrow \quad e \cdot d - k \cdot n = 1$$

d und n sind dabei die Variablen der Gleichung.

Rückblickende Bem. Auf den Exkurs „diophantische Gleichungen“: Es gilt in unserem Kontext also stets $c = 1$. Hiermit erhält die Lösungsbedingung der linearen diophantischen Gleichung die Form $ggT(e,n) \mid 1$, also gilt $ggT(e,n) = 1$ und damit: e, n sind teilerfremd. Insbesondere ist dies erfüllt, wenn e und n Primzahlen sind.

Obwohl die Berechnung des ggT(e,n) in diesem Kontext aufgrund der Wahl als Primzahlen nicht mehr nötig ist, sollte sie im Beispiel doch durchgeführt werden, da der Erweiterte Euklidische Algorithmus durch Zurückrechnen für Schüler durchschaubar aus dem Euklidischen Algorithmus entsteht.

Hierzu ist eine Wiederholung des Euklidischen Algorithmus' angebracht. Entsprechende Aufgaben sind im Arbeitsblatt enthalten. Ist eine ausführlichere Wiederholung nötig, so kann auf die Materialien von Klasse 8 zurückgegriffen werden: *08_mgk_Euklid.*

Nach dieser wichtigen vorbereitenden Übung kann zur Erarbeitung des Erweiterten Euklidischen Algorithmus übergegangen werden.

Für die Herleitung eignet sich als Aufhängepunkt die kryptographische Kernaufgabe „gesucht wird das multiplikative Inverse d zu 4 (mod 7)“:

$$4 \cdot d \equiv 1 \pmod 7 \quad \Leftrightarrow \quad 4 \cdot d - k \cdot 7 = 1$$

Erster Schritt ist die Berechnung des ggT(7,4) mittels Euklidischem Algorithmus:

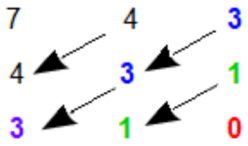
I $7 = 1 \cdot 4 + 3$

II $4 = 1 \cdot 3 + 1$

III $3 = 3 \cdot 1 + 0$

ggT Abbruchbedingung

optional in Matrixform :



Ansatz des Erweiterten Euklidischen Algorithmus ist nun der folgende Gedanke:

Die linke Seite der Gleichung $\mathbf{4} \cdot d - k \cdot \mathbf{7} = 1$ stellt eine Linearkombination der „1“ mittels der Zahlen 7 und 4 dar und es sollen nun die Koeffizienten d und k bestimmt werden.

Dies löst man durch Rückwärtsrechnen:

Zeile II, aufgelöst nach 1: $1 = 4 - 1 \cdot 3$ (*)

Bem.: die gesuchte „4“ taucht schon auf, „3“ muss ersetzt werden.

Nun eine Zeile höher: Zeile I, aufgelöst nach 3 ergibt $3 = 7 - 4$.

Dies wird nun in (*) eingesetzt: $1 = 4 - 1 \cdot 3$

$\Rightarrow \quad 1 = 4 - 1 \cdot (7 - 4) = 4 - 7 + 4$

Sortieren ergibt $1 = 2 \cdot 4 + (-1) \cdot 7 = 2 \cdot 4 - 1 \cdot 7$

Unsere lineare Kongruenzgleichung $4 \cdot d - k \cdot 7 = 1$ besitzt die Lösungen $d = 2$ und $k = 1$.

Wir benötigen davon lediglich $d = 2$.

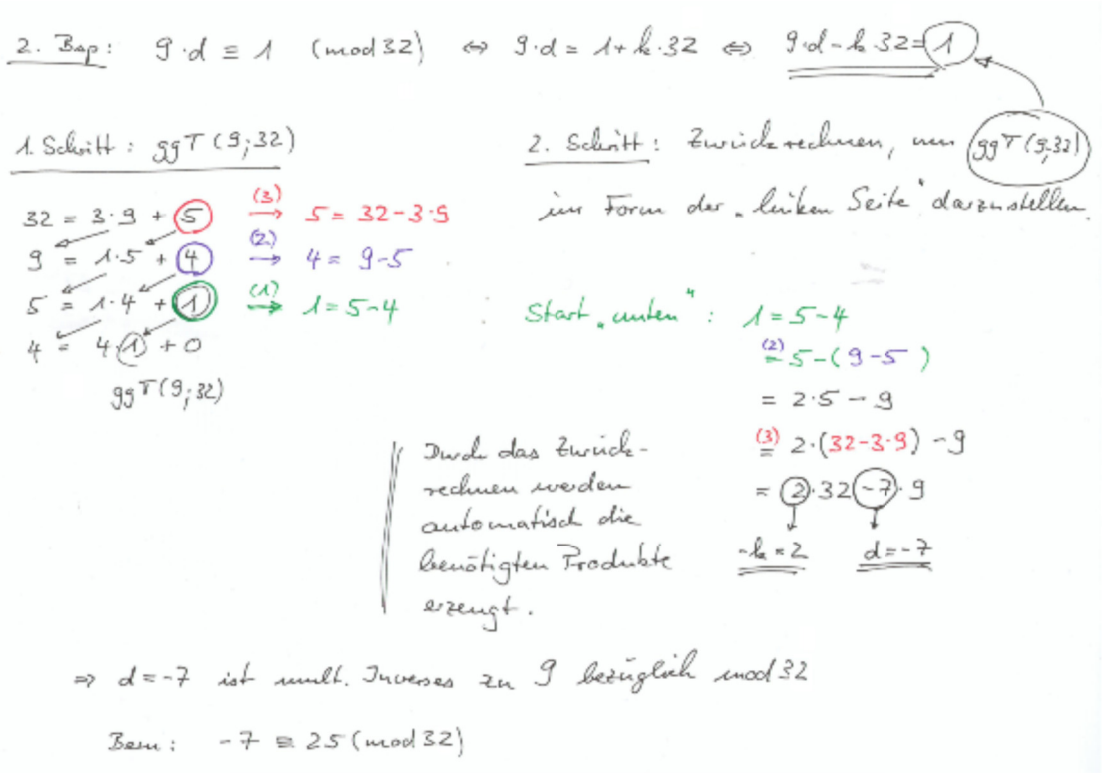
Bem.: auf dem AB 03c_mgk_Erw_Euklid_Alg_S-zentriert ist dieses Vorgehen für die SuS ausführlich und übersichtlich dargestellt.

Ergebnis: **Das multiplikative Inverse zu 4 bezüglich (mod 7) ist d = 2.**

Bemerkungen dazu:

- Eine Erarbeitung des Algorithmus im Unterrichtsgespräch ist angeraten, da gerade das Zurückrechnen und die dabei nötigen Gedanken und der notwendige Überblick, die letztendlich zum Erfolg bei der Erstellung der Linearkombination führen, eine Führung nötig erscheinen lassen. Gruppen, denen ein eigenständiges Erarbeiten des Themas zuzutrauen ist, können mit dem Arbeitsblatt *03c_mgk_Erw_Euklid_Alg_S-zentriert* an das Thema herangehen.
- Im Beispiel oben ist die Bestimmung des ggT und damit auch die Bestimmung des multiplikativen Inversen mit wenigen Schritten erledigt. Bei einer Berechnung mit größerer Schrittanzahl gilt es noch mehr, die Übersicht nicht zu verlieren. Eine ausgedehntere Übungsphase mit Beispielen aufsteigender Komplexität ist hier unerlässlich, damit SuS lernen, das Ziel nicht aus den Augen zu verlieren. Je nach Schülergruppe ist es angeraten, ein mehrschrittiges Beispiel noch einmal gemeinsam durchzurechnen. Hierzu eignet sich z.B. die auf dem Arbeitsblatt genannte Aufgabe $9 \cdot d \equiv 1 \pmod{33}$.

Ein möglicher Tafelaufschrieb:



- Die Matrixschreibweise ist als vereinfachte Schreibweise beim Euklidischen Algorithmus machbar. Beim Durchführen der Erweiterung jedoch erscheint hier die Gefahr des Verständnisverlustes zu hoch, die schrittweise Umformung der Gleichungen hilft den SuS, den Ablauf zu verinnerlichen und den Überblick zu behalten.
- Wurde der Exkurs „Diophantische Gleichungen" nicht vorgenommen, so genügt es für den Zusammenhang mit der Kryptographie, den Erweiterten Euklidischen Algorithmus wie

auf Arbeitsblatt *03c_mgk_Erw_Euklid_Alg_S-zentriert* einzuführen: *„Der EEA liefert für eine Gleichung der Form $a \cdot x + b \cdot y = ggT(a;b)$ mit $a, b \in \mathbb{N}$ (neben dem ggT(a;b) als Zwischenergebnis) die Lösungen $x, y \in \mathbb{Z}$.“* Mit dieser Feststellung wird klar, warum auf dem Euklidischen Algorithmus aufgebaut wird. Organisch folgt aus dem kryptographischen Kontext die Gleichung $e \cdot d - k \cdot n = 1$. Besonderes Augenmerk sollte dabei auch auf das Vorzeichen von k gelegt werden. Selbstverständlich kann die Umformung auch anders vorgenommen werden, damit die Identifikation der entstehenden Gleichung mit der Standardform der linearen Kongruenzgleichung $a \cdot x + b \cdot y = ggT(a;b)$ leichter ist. Hier sollte nach eigenem Ermessen, jedoch im weiteren Unterrichtsverlauf konsequent verfahren werden.

- Die Erstellung weiterer Übungsaufgaben (die auch noch schwerer zu erraten sind → Sinnhaftigkeit des Algorithmus) ist denkbar einfach:

  Ansatzpunkt ist die Forderung ggT(a;b) = 1 (a, b teilerfremd; am leichtesten zu realisieren durch Aufstellen einer entsprechenden Primzahlzerlegung). Dann lautet die Aufgabe „finde d so, dass $a \cdot d \equiv 1 \pmod{b}$“. Die Bearbeitung dieser Umkehraufgabe kann eine gute binnendifferenzierende Aufgabe sein (Übung Nr.3 auf dem Arbeitsblatt).

Exkurs: Einweg- und Falltürfunktionen

Diese (optionale) Stunde baut den systematisch grundlegenden Begriff der Einweg- und Falltürfunktionen in der Kryptographie anhand von außermathematischen Beispielen weiter aus. Hierbei sollte deutlich gemacht werden, dass diese Funktionen rar und entsprechend schwer zu finden sind. Das Konstruieren solcher Zusammenhänge stellt eine beachtliche Leistung dar und ist das Herz der Kryptographie. In diesem Zusammenhang bietet sich ein historischer Exkurs an, bei dem z.B. die Bemühungen und Erfolge der Mathematiker Diffie und Hellman in den 70er-Jahren des 20. Jahrhunderts genannt werden können, die auch den Grundstein zu den heutigen Public-Key-Verfahren legten.

Im Material finden Sie hierzu das Arbeitsblatt *03d_mgk_Einweg-und_Falltürfunktionen.*

Bem.: Eine interessante Anregung zu einer weiteren Einwegfunktion unter Verwendung von Graphen findet sich bei *https://classic.csunplugged.org/wp-content/uploads/2014/12/unplugged-18-public_key_encryption_0.pdf* (abgerufen 10.5.2020)

Exkurs: Neutrale und inverse Elemente

Zuerst bei der Verschlüsselung mittels Multiplikation aufgetaucht, wird uns das Problem der Bestimmung des Inversen auch im Zusammenhang mit dem RSA-Verfahren gegen Ende der Einheit wieder beschäftigen. Aus diesem Grund erscheint es sinnvoll, den an entscheidender Stelle immer wieder auftretenden Begriff der „Inversen“ etwas weiter auszuleuchten und die mathematischen Zusammenhänge systematisch darzustellen. In diesem Zusammenhang ist dann natürlich auch die Erwähnung des „neutralen Elements“ zwingend. Die Inhalte dieses Blocks sind optional und gehen über den Bildungsplan hinaus.

Das Arbeitsblatt *03e_mgk_ Neutrale_und_inverse_Elemente* stellt zunächst die Begriffe des inversen und des neutralen Elements vor und definiert diese. Es wird thematisiert, dass diese abhängig von der jeweiligen der Verknüpfung sind und eine sinnvolle Bearbeitung nur mit dieser im Zusammenhang vorgenommen werden kann. Als Beispiel dient die Menge ganzen Zahlen mit der Verknüpfungen „Multiplikation". Die Tatsache, dass ein Inverses überhaupt nicht existieren muss bzw. dass es möglicherweise ein inverses Element gibt, das jedoch nicht in der betrachteten Zahlenmenge liegt, werden angesprochen. Der Begriff der Abgeschlossenheit könnte hier ebenfalls aufgegriffen werden, wird in diesem Unterrichtsgang aber nicht weiter vertieft.

Die Bearbeitung der Verknüpfung „Addition" wird als Übungsaufgabe vorgenommen.

Einige Detailprobleme stellen sich hier in der Formulierung der Zusammenhänge, die im Unterricht unbedingt geklärt und abgegrenzt werden müssen:

- Das neutrale Element, wird standardmäßig mit „e" bezeichnet. Dies doppelt sich hier mit einem anderen in der Kryptographie zentralen Begriff, nämlich dem der Verschlüsselungszahl (e für „encryption"). Aus diesem Grund wird hier die Bezeichnung „a" für das neutrale Element gewählt.
- Das zu einer Zahl a inverse Element a^{-1} erzeugt bei den SuS die vertraute Vorstellung des Kehrwerts. Dass in dieser Formulierung viel mehr steckt, weil der Begriff des Kehrwerts evtl. in manchen Zusammenhängen gar nicht sinnvoll ist, sollte klar werden. Aus diesem Grund wird hier die Bezeichnung „b" für das inverse Element gewählt.

Als Beispiel, dass sich die Begriffe inverses und neutrales Element nicht nur in Zahlenmengen manifestieren, wird neutrales und inverses Element bei Potenzfunktionen unter der Verknüpfung „Verkettung" bestimmt. Obwohl der Unterrichtsinhalt „Verkettung" erst in der Kursstufe besprochen wird, ergibt sich hier ein für die SuS gut zu verstehendes, jedoch anspruchsvolles Betätigungsfeld, das durchaus binnendifferenzierend eingesetzt werden kann.

Der hier vorgenommene Einblick kann aus Zeitgründen nur ein begrenzter Exkurs sein. Wie oben schon angesprochen, wird auf die Frage der Abgeschlossenheit nicht weiter eingegangen. Jedoch auch auf weitere, durchaus interessante Problematiken und lohnende Betätigungsfelder wie das Problem der links- und rechtsseitigen Verknüpfungen wird ebenfalls nicht eingegangen.

An dieser Stelle bieten sich weitere mögliche Felder zu mathematischem Tun: sei es in Form weiterer Exkurse im Unterricht oder auch z.B. als lohnende GFS-Themen. Ein Beispiel hierfür wäre der Themenkreis „Umkehrfunktionen", der sich organisch aus Aufgabe 4 ergeben kann.

Das RSA-Verfahren

Abschließend wird als real benutztes Verfahren das RSA-Verfahren im Rahmen des Arbeitsblattes *07_mgk_Das_RSA-Verfahren* vorgestellt.

Bemerkung: Im Zusammenhang mit dem RSA-Verfahren kann auch der Schlüsselaustausch nach Diffie-Hellman als ein weiteres Beispiel für ein asymmetrisches Verfahren behandelt werden. Dies ist auch aus historischer Sicht interessant. Wenn dies im Rahmen der RSA-Einheit thematisiert werden soll (was durchaus sinnvoll ist), so kann hier das Arbeitsblatt 02_iud_ab_asym_RSA aus der Einheit „Informationsgesellschaft und Datensicherheit (iud)" verwendet werden.

Eine Liste verfügbarer Primzahlen, bei denen noch mit geringen Hilfmitteln gerechnet werden kann, findet man im Internet, z.B. bei Walter Fendt:

https://www.walter-fendt.de/html5/mde/primenumbers_de.htm,

ebenso ASCII-Tabellen der Buchstaben und Zahlen, z.B. unter

https://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange#ASCII-Tabelle

Um die Komplexität zu reduzieren, wird hier in diesem Unterrichtsgang lediglich der Kommunikationsweg A → B betrachtet, nicht die beiderseitige Kommunikation. Nachdem der Unterrichtsgang bis hier verfolgt wurde, kann die Erweiterung auf eine beiderseitige Kommunikation aufgesetzt werden.

Tiefergehende Hintergrundinformationen zu grundlegenden Fragen:

Wie sicher ist RSA?

Um den Algorithmus anwenden zu können, muss der Benutzer das Produkt $(p-1) \cdot (q-1)$ kennen. Der öffentliche Schlüssel informiert aber lediglich über die Zahl N (bestimmt als $N = p \cdot q$) und nicht über die beiden erzeugenden Primfaktoren p und q selbst. Diese müssten eben über Faktorisierung herausgefunden werden, Stichwort „Einwegfunktion“.

In der Praxis setzt man also N aus so großen Primfaktoren zusammen, dass die Zerlegung selbst mit leistungsfähigen Computern Jahre dauern würde (siehe Info in den Arbeitsblatts).

Da das Problem der Faktorisierung prinzipiell gelöst werden kann bedeutet das, dass RSA theoretisch gesehen nicht sicher ist,. Es dauert eben nach derzeitigem Stand der Technik und Mathematik so lange, dass die Information bis zum Zeitpunkt der Lösung nicht mehr relevant ist (diesen generellen Aspekt von „Sicherheit“ lohnt es sich durchaus, im Unterricht zu thematisieren: Auch z.B. Tresore werden mit einer Garantie verkauft, wie lange sie unbefugten Öffnungsversuchen widerstehen können)

In dieser Hinsicht ist also RSA solange sicher, bis ein entsprechend schneller Algorithmus für die Primfaktorzerlegung gefunden ist.

Bemerkung zu Diskrepanzen im Vergleich zu realem RSA:

Beim zur Verfügung gestellten Arbeitsblatt werden jeweils nur einzelne Zahlen (nämlich die Dezimalcodes der Buchstaben nach der ASCII-Tabelle) mit RSA verschlüsselt.

- *In der Regel werden keine Dezimal-, sondern Hexadezimalzahlen verwendet. Für das RSA-Prinzip ist dieser Unterschied jedoch nicht von Bedeutung*
- *In der Realität macht eine RSA-Verschlüsselung der einzelnen Zeichen keinen Sinn: würde so verschlüsselt werden, so ergäbe sich eine eindeutige Zuordnung der Klar- und Geheimbuchstaben wie im monoalphabetischen Fall. Dies würde die Errungenschaften von RSA zunichte machen, da man nun mit einer einfachen Häufigkeitsanalyse angreifen könnte. Das bedeutet, dass in der Realität die ASCII-Codes maskiert werden müssen. Hierzu gibt es verschiedenste Möglichkeiten, wobei sich eine häufig gebrauchte oder gar einheitliche Methode nicht angeben lässt. Eine mögliche sei hier beschrieben (aufbauend auf Joachim Mohr, https://kilchb.de/bsp_rsa.php, 06.07.2020):*

- *Die Buchstaben werden z.B. in 2er-Gruppen zusammengefasst. Die ASCII-Codes werden z.B. maskiert, indem man die Zahl des zweiten Buchstabens mit einer Zahl multipliziert, die größer ist als die größte zu verschlüsselnde Zahl (ASCII besitzt 255 Zeichen, also Multiplikation mit einer Zahl z > 255). Danach wird das Ergebnis zur ersten Zahl (des ersten Buchstabens) addiert.*

  Beispiel: Verschlüsselt werden soll ZPG IMP.

  2er-Gruppen: *ZP* *GI* *MP.*

  Nach der ASCII-Tabelle ergibt sich *136 126* *117 119* *123 126.*

  Nun wird maskiert mit z.B.256: $136 + 126 \cdot 256 = \mathbf{32392}$

  $117 + 119 \cdot 256 = \mathbf{30581}$

  $123 + 136 \cdot 256 = \mathbf{34939}$

- *Diese Zahlen werden nun mit RSA ver- und entschlüsselt. Um die ursprüngliche Kombination wieder zu erhalten, bestimmt man (hier am ersten Beispiel durchgeführt) 32392 mod 256 = 136, bestimmt daraus den Faktor 126 und hat so beide ursprünglichen ASCII-Codes zurückgewonnen. Ein ausführliches Beispiel ist unter der o.g. Quelle einsehbar.*

Hier sind weitere Möglichkeiten denkbar, z.B. die Zusammenfassung größerer Buchstabengruppen mit der Maskierung durch Multiplikation mit Potenzen der gewählten Zahl. Bsp.: ZPG IMP mit 3er-Gruppen und gewählter Zahl 301 ergibt die mit RSA zu verschlüsselnden Zahlen:

$136 + 126 \cdot 301 + 117 \cdot 301^2 = \mathbf{10.638.379}$ *und*

$119 + 123 \cdot 301 + 126 \cdot 301^2 = \mathbf{11.452.868}$

Demaskierung würde hier erfolgen (am ersten Beispiel) durch 10.638.379 mod 301^2 und anschließend den erhaltenen Rest mod 301 inklusive der Bestimmung der Faktoren 117 im ersten Schritt und 126 im zweiten Schritt.

Das RSA-Verfahren durchzuführen ist mit den besprochenen Inhalten nun problemlos möglich. Schwieriger wird es jedoch, wenn man die Frage nach dem grundlegenden mathematischen Fundament des RSA-Verfahrens stellt: Warum funktioniert RSA? Welche Zusammenhänge müssen dazu bewiesen werden? Hierbei stößt man sehr schnell an die Grenzen dessen, was in der Schule machbar ist.

Hier als Anregung einige grundlegenden Sätze und der Beweis zur Korrektheit von RSA:

Grundlegende Sätze:

1) Die Euler'sche φ-Funktion

Ist n eine natürliche Zahl, so gibt $\varphi(n)$ die Anzahl der zu n teilerfremden Zahlen an, die nicht größer als n sind. Damit ist insbesondere klar: Ist p eine Primzahl, so ist p teilerfremd zu den Zahlen 1 bis p-1.

Daher gilt für Primzahlen p: $\varphi(p) = p - 1$.

Für teilerfremde Zahlen m und n gilt: $\varphi(n) \cdot \varphi(m) = \varphi(n \cdot m)$

Daher gilt für Primzahlen p und q: $\varphi(p \cdot q) = \varphi(p) \cdot \varphi(q) = (p - 1) \cdot (q - 1)$

Beweise zur Euler'schen φ-Funktion sind sehr langwierig und in diesem Zusammenhang nicht weiterführend. Deshalb sei hier für Interessierte auf die gängige Fachliteratur verwiesen.

2) Der Satz von Fermat-Euler und der kleine Satz des Fermat

Satz von Fermat-Euler: Für $a, m \in \mathbb{N}$ gilt: $ggT(a, m) = 1 \Rightarrow a^{\varphi(m)} \equiv 1 \bmod m$.

Einen Spezialfall davon stellt der *kleine fermat'sche Satz* dar:

Ist p eine Primzahl und $a \in \mathbb{N}$, gilt: $ggT(a, p) = 1 \Rightarrow a^{p-1} \equiv 1 \bmod p$.

Bem.: Satz von Fermat-Euler für m = p, da $\varphi(q) = (p - 1)$

Der Beweis des Satzes von Fermat-Euler geht über das in Klasse 10 Mögliche hinaus. Eine mögliche kombinatorische Variante (als „Kettenbasteln") kann evtl. für sehr interessierte SuS thematisiert werden. Diese und einige weitere Beweismöglichkeiten findet man unter https://de.wikibooks.org/wiki/Beweisarchiv:_Zahlentheorie:_Elementare_Zahlentheorie:_Kleiner_Satz_von_Fermat

Der Beweis: Warum funktioniert RSA?

Die Beweise zur theoretischen Begründung der Korrektheit des RSA-Verfahrens sind ebenfalls oft aufwändig und mathematisch in der Schule nicht durchführbar. Hier jedoch eine mögliche Skizze, die man (unter Auslassung der Beweise zur Euler'schen Funktion) schrittweise nachvollziehen kann:

Die Botschaft B wird zum Geheimtext S verschlüsselt durch von $B^e \bmod N$ (= S)

Die Entschlüsselung wird vorgenommen durch $S^d \bmod N$ (= B)

Also gilt $S^d \bmod N = (B^e \bmod N)^d \bmod N = B^{ed} \bmod N$.

Dabei war d das (modulare) multiplikative Inverse zu e bezüglich mod $(p - 1) \cdot (q - 1)$,

also gilt: $(e \cdot d) \bmod \varphi(N) = 1$ ($\Leftrightarrow (e \cdot d) \bmod \varphi(p \cdot q) = (e \cdot d) \bmod (p-1) \cdot (q-1) = 1$)

Damit gilt mit $N = p \cdot q$ und $\varphi(p \cdot q) = (p-1) \cdot (q-1)$

die Beziehung

$$e \cdot d = k \cdot \varphi(N) + 1 = k \cdot \varphi(p \cdot q) + 1 = k \cdot (p-1) \cdot (q-1) + 1$$

und damit

$$e \cdot d \bmod \varphi(N) = (k \cdot (p-1) \cdot (q-1) + 1) \bmod \varphi(N)$$

Wir erinnern uns: Der entschlüsselte Geheimtext war $S^d \bmod N = (B^e)^d = B^{ed} \bmod N$.

Wenn RSA funktionieren soll, muss also gelten $B^{ed} \bmod N = B$

Mit der obigen Beziehung $e \cdot d = k \cdot (p-1) \cdot (q-1) + 1$ muss also gelten: $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod N = B$

Zusammenfassend die zu beweisende zentrale Aussage (Korrektheit von RSA):

> Es seien p, q verschiedene Primzahlen und $B \in \mathbb{N}$ mit $B \leq p \cdot q$.
>
> Dann gilt für jede Zahl $k \in \mathbb{N}$: $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod pq = B$

Beweis[1]**:** Zunächst getrennt nach den Primzahlen p und q (1), (2), dann Zusammenführung (3):

(1) $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod p = B$

(2) $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod q = B$

(3) $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod pq = B$

(1) $B^{k \cdot (p-1) \cdot (q-1) + 1}$ *elementare Potenzregel* $a^{n+m} = a^n \cdot a^m$

$= B \cdot B^{k \cdot (p-1) \cdot (q-1)}$ *elementare Potenzregel* $a^n \cdot a^m = (a^n)^m$

$= B \cdot (B^{p-1})^{k \cdot (q-1)}$

Damit: $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod p$

$= B \cdot (B^{p-1})^{k \cdot (q-1)} \bmod p$ *Regel modulares Potenzieren*

$= B \cdot (B^{p-1} \bmod p)^{k \cdot (q-1)} \bmod p$ *kl. Fermat: ggT(p, B) = 1* $\Rightarrow$ $B^{p-1} \equiv 1 \bmod p$.

ggT(p, B) = 1, da p Primzahl und B < p.

$= B \cdot (1 \bmod p)^{k \cdot (q-1)} \bmod p$

$= B \cdot 1^{k \cdot (q-1)} \bmod p$ *Potenzrechnung:* $1^n = 1$

$= B \bmod p$

Zusammenfassend: $B^{k \cdot (p-1) \cdot (q-1) + 1} \bmod p = B \bmod p$

1 Angelehnt an *http://www.mathematik-netz.de/pdf/RSA.pdf*

(2) Für q analog, lediglich mit anderer Sortierung im Exponenten.

Zusammenfassend: $B^{k\cdot(p-1)\cdot(q-1)+1} \bmod q = B \bmod q$

(3) N = pq: Hierzu führen wir der Übersichtlichkeit halber ein $h := k\cdot(p-1)\cdot(q-1)+1$.

In den beiden in (1) und (2) erhaltenen Gleichungen

$$B^h \bmod p = B \bmod p \quad \text{und} \quad B^h \bmod q = B \bmod q$$

bringen wir jeweils B auf die andere Seite und erhalten

$$B^h \bmod p - B \bmod p = 0 \quad \text{und} \quad B^h \bmod q - B \bmod q = 0.$$

Mit Anwendung der Regel über modulare Addition folgt:

$$(B^h - B) \bmod p = 0 \quad \text{und} \quad (B^h - B) \bmod q = 0$$

und damit:

$$p \text{ teilt } B^h - B \quad \text{und} \quad q \text{ teilt } B^h - B.$$

p und q sind jedoch verschiedene Primzahlen, also muss $(p \cdot q)$ die Zahl $B^h - B$ teilen.

Damit gilt: $(B^h - B) \bmod (p\cdot q) = 0$ *Rechenregel modulares Addieren*

$\Leftrightarrow$ $B^h \bmod (p\cdot q) - B \bmod (p\cdot q) = 0$ $B < p \cdot q$

$\Leftrightarrow$ $B^h \bmod (p\cdot q) - B = 0$

$\Leftrightarrow$ $B^h \bmod (p\cdot q) = B$ ■

Die einzelnen Schritte sind so mit den im Verlauf des Unterrichts bewiesenen Gesetzen begründet nachvollziehbar. Allerdings bleibt der Beweis aufwändig.