Maximilian **Kolbe** Schule
Menschen | Bildung | Zukunft

Seminararbeit 2015/16

Thema:

**Erweiterter Euklidischer Algorithmus in $\mathbb{N}$ -
eine Untersuchung seiner Geschichte, Funktionsweise und dessen Anwendung am Beispiel des RSA-Algorithmus**

Name der betreuenden Lehrkraft: Ghiroga, Ionut

Name: Matthias Uschold **Klasse:** 13 BT 1

Schule, an der die 13. Klasse besucht wird:

Name: Maximilian-Kolbe-Schule

Straße: Kerschensteinerstraße 7

Ort: 92318 Neumarkt i. d. OPf.

1 Einleitung

Diese Facharbeit beschäftigt sich mit dem Erweiterten Euklidischen Algorithmus. Die einfache Form davon, der Euklidische Algorithmus, ermöglicht die schnelle Berechnung des größten gemeinsamen Teilers zweier Zahlen.
Als ich im Mai 2015 in einem Lexikon auf diesen stieß, konnte ich mir nicht erklären, warum er funktioniert. Ich war aber fasziniert von der Einfachheit und zugleich Richtigkeit des Algorithmus. Im Rahmen des Seminarfachs entschloss ich mich daher, diesen Algorithmus näher zu untersuchen.
Das Ziel meiner Arbeit ist es, den Gedankengang von der Herleitung der Grundlagen bis zum Beweis des Euklidischen Algorithmus anschaulich darzustellen. Weiterhin werde ich die Erweiterung behandeln, welche neue Berechnungsmöglichkeiten eröffnet. Im letzten Kapitel werde ich darstellen, welche konkreten Anwendungsmöglichkeiten das gewonnene Wissen in der Kryptografie bietet. Dafür habe ich mir den RSA-Algorithmus ausgesucht.
Da der Euklidische Algorithmus bereits vor über 2000 Jahren entstand, werde ich mich zunächst seiner Geschichte zuwenden.

Zur besseren Veranschaulichung stelle ich neben den hier gegebenen Erklärungen Computerapplikationen auf der beiliegenden CD im Ordner „Applikationen“ bereit. In diesem Fall befindet sich ein solcher Hinweis am Beginn des Kapitels: ►*beispiel.html*

2 Geschichte

2.1 Euklid

Der Mathematiker Euklid von Alexandria, auch Eukleides genannt (Herrmann, 2014, 101), nicht zu verwechseln mit Euklid von Megara (Herrmann, 2014, 103), wurde ca. 360 v. Chr. geboren. (Herrmann, 2014, 101) Er wirkte um 300 v. Chr. in Alexandria. Über seine Lebensdaten ist nichts Genaueres bekannt. (Froese, 2015, 3)
Da sein Verständnis stark von der platonischen Lehre geprägt ist, vermutet man, er hätte einige Jahre an der platonischen Akademie in Athen verbracht, bevor er sich nach Alexandria begab, um dort die Alexandrinische Schule der Mathematik zu gründen. (Froese, 2015, 3)

Euklids bekanntestes Werk sind „Die Elemente“, welches aus 13 Büchern besteht. Es gilt als das einflussreichste Werk der gesamten Mathematik, (Duden, 2011, 194) das noch bis ins 19. Jahrhundert zur Einführung in die akademische Mathematik verwendet wurde. (Froese, 2015, 3) Die Besonderheit an diesem Buch ist der für die weitere Mathematik bedeutsame Versuch, die Geometrie axiomatisch aufzubauen und somit erste mathematische Prinzipien aufzustellen. (Herrmann, 2014, 104) Die Vorgehensweise folgt dabei stets dem strengen Aufbau *Definition, Satz, Beweis*. (Froese, 2015, 11)

Das Buch VII beginnt mit der Vorstellung des Euklidischen Algorithmus zur Bestimmung des größten gemeinsamen Teilers (Froese, 2015, 21), welcher später in dieser Arbeit vorgestellt wird.
Weiterhin sind dort viele mathematische Erkenntnisse enthalten, die uns heute selbstverständlich erscheinen.
So beweist Euklid, dass es unendlich viele Primzahlen gibt. Erwähnenswert ist, dass Euklids Beweis hierzu zu einem der 10 schönsten Beweise der Mathematik gewählt wurde. (Herrmann, 2014, 129f) Auch, dass die Innenwinkelsumme eines Dreiecks 180° beträgt, (Herrmann, 2014, 131) sowie der Satz des Pythagoras werden in Euklids Werk bewiesen. (Froese, 2015, 10)

2.2 RSA-Verfahren

Schon seit Jahrtausenden beschäftigt sich die Menschheit mit der Verschlüsselung von Nachrichten. Eine Schwierigkeit bestand darin, dass der Schlüssel zur Verschlüsselung gleichzeitig den Schlüssel zur Entschlüsselung darstellte. Man musste also die Schlüssel geheim austauschen, um zu verhindern, dass die Verschlüsselung geknackt wird. (Calderbank, 2007, 1)

Dies änderte sich 1975, als Whitfield Diffie und Martin Hellmann das Konzept der asymmetrischen Verschlüsselung entwickelten, bei der sich die Schlüssel zur Verschlüsselung und Entschlüsselung unterscheiden. Dadurch kann der öffentliche Schlüssel (zur Verschlüsselung) gefahrlos unverschlüsselt an die andere Partei übertragen werden, ohne die Sicherheit der Verschlüsselung zu gefährden. Diffie und Hellmann beschrieben Anforderungen an einen solchen Algorithmus, konnten aber selbst keine Einwegfunktion finden, die diesen genügte. (Calderbank, 2007, 2)

Am Massachusetts Institute of Technology forschten die Wissenschaftler Rivest, Shamir und Adleman an diesem Problem. (Busse, u. a., 1999) Ronald Rivest und Adi Shamir (Busse, u. a., 1999) sind Informatiker, die immer wieder neue Ideen für eine solche Einwegfunktion entwickelten. (Calderbank, 2007, 2) Leonard Adleman (Busse, u. a., 1999) ist Mathematiker und half den beiden Informatikern dabei, auszusortieren, welche Lösungsvorschläge Erfolg versprachen und welche mathematische Fehler beinhalteten. Im April 1977 kam Rivest auf eine Idee, die schließlich zum Erfolg führte. Ohne die Hilfe der anderen beiden Wissenschaftler wäre dies jedoch nicht gelungen.
Zu Ehren der Erfinder wurde der Algorithmus später **R**ivest-**S**hamir-**A**dleman, kurz RSA, genannt. (Calderbank, 2007, 2)

3 Normen für diese Seminararbeit

Um Missverständnissen aus dem Weg zu gehen, möchte ich an dieser Stelle einige Normen definieren:

So ist unter Mathematikern umstritten, ob die Null zu der Menge der natürlichen Zahlen gehört (Mathe-Lexikon.at, kein Datum). Für diese Seminararbeit lege ich daher Folgendes fest:
$\mathbb{N} := \{1; 2; 3; 4; \dots\}$
$\mathbb{N}_0 := \{0;\ 1; 2; 3; 4; \dots\}$

Weiterhin werden öfter Namen für Aussagen festgelegt. So bedeutet „$(A00)$ ….“, dass die Aussage, die dort steht, den Namen „$A00$“ erhält. Die Namen beginnen mit A und es folgt eine Zahl. Die Zahlen werden je nur einmal und in aufsteigender Reihenfolge vergeben. Ohne Klammern wird auf die Aussage verwiesen.

Ebenso werden innerhalb von Beweisen Gleichungen oder Definitionen mit römischen Zahlen bezeichnet. Diese Bezeichnungen gelten nur innerhalb des Beweises. Im darauffolgenden Beweis können diese neu vergeben werden. Ebenso benutze ich hier den Buchstaben C für Aufspaltungen einer Aussage in mehrere Bedingungen (engl. conditions) und Y für den zweiten Teil einer Zeile. Ein Beweis ist immer mit dem Zeichen ■ beendet.
Allerdings wird nicht jede Aussage ausführlich bewiesen, da dies den Rahmen dieser Arbeit sprengen würde.

4 Grundlagen

4.1 Teilbarkeit

Der erste Begriff ist die Teilbarkeit, die für das Verständnis der Seminararbeit erforderlich ist.

Definition:
(A01) Man bezeichnet a als Teiler von b $(a, b \in \mathbb{N}_0)$, wenn es ein $n \in \mathbb{N}_0$ gibt, sodass $a * n = b$ gilt.
Mathematisch wird dies als $a \mid b$ (sprich: „a teilt b“) geschrieben (Kramer/von Pippich, 2013, 18).
b ist dann ein Vielfaches von a (Duden, 2005, 20).

Beispiele: $2 \mid 8$, da $2 * 4 = 8$
$7 \mid 14$, da $7 * 2 = 14$
$4 \nmid 7$ (4 teilt 7 nicht) da $n \notin \mathbb{N}_0$ für $4 * n = 7$

Folgende Eigenschaften gelten bei der Teilbarkeit ($a, b, c \in \mathbb{N}_0$):

(A02) $a \mid a \ \forall a \in \mathbb{N}_0$ (Jede Zahl teilt sich selbst), da $a * 1 = a$ mit $1 \in \mathbb{N}_0$

(A03) $1 \mid a \ \forall a \in \mathbb{N}_0$, da $1 * a = a$

(A04) $a \mid 0 \ \forall a \in \mathbb{N}_0$, da $a * 0 = 0$

(A05) $a \mid b \Rightarrow a \mid b * c$

Beweis:

$a * n = b$ (aus Angabe)

$a * n * c = b * c$ (Multiplikation mit c)

$a * n_2 = b * c$ mit $n_2 = n * c \in \mathbb{N}_0$

$a \mid b * c$ ■

(A06) $a \mid b$ und $a, b \neq 0 \Rightarrow a \leq b$

Beweis:

$a * n = b$ (aus A01)

$(I) n = \frac{b}{a}$

Fall 1: $n = 0$ nur für $b = 0$ (ist bereits ausgeschlossen)

Fall 2: $b \neq 0 \Rightarrow n \neq 0$

Mit $n \in \mathbb{N}_0$ folgt, dass $n \geq 1$

Mit I folgt $b \geq a$ ■

(A07) Transitivität: $a \mid b$ und $b \mid c \Rightarrow a \mid c$ (Forster, 2004)

Beweis:

$(I)\ a * n_1 = b$	(Voraussetzung mit A01)
$(II) b * n_2 = c \quad (n_1, n_2 \in \mathbb{N}_0)$	(Voraussetzung mit A01)
$a * n_1 * n_2 = c$	(I in II einsetzen)
$a * n = c \ mit \ n_1 * n_2 = n \in \mathbb{N}_0$	(Zusammenfassen)

$\Rightarrow a \mid c$ ■

(A08) $a = b \pm c$; $d \mid b$ und $d \mid c \Rightarrow d \mid a$ (Duden, 2005, 21)

Beweis:

$(I)\ d * n_1 = b$	(Voraussetzung mit A01)
$(II) d * n_2 = c \quad (n_1, n_2 \in \mathbb{N}_0)$	(Voraussetzung mit A01)
$a = d * n_1 \pm d * n_2$	(Einsetzen von I und II in Angabe)

$a = d * (n_1 \pm n_2) \quad mit \ (n_1 \pm n_2) \in \mathbb{N}_0, da \ a, d \in \mathbb{N}_0$

$\Rightarrow d \mid a$ ■

(A09) $b = c + a$; $d \mid b$ und $d \mid c \Rightarrow d \mid a$ (Umstellung von A08)

4.2 Teilermenge

Definition:

(A10) Jede Zahl $a \in \mathbb{N}$ besitzt eine Teilermenge $T(a)$, welche alle ihre Teiler beinhaltet, mathematisch geschrieben als $T(a) := \{d \in \mathbb{N} \mid d\ teilt\ a\}$
(Bescherer, 2006, 1)

Beispiel: $T(12) = \{1; 2; 3; 4; 6; 12\}$

Nachfolgend gilt $a \in \mathbb{N}$:

(A11) $\{1; a\} \subseteq T(a)\ \forall a \in \mathbb{N}$ (aus A02 und A03)

(A12) $T(1) = \{1\}$

(A13) $T(a) = \{1; a\}$ und $a \neq 1 \iff$ a ist eine Primzahl
Begründung: Eine Primzahl ist eine natürliche Zahl, die nur durch 1 und sich selbst teilbar ist. (Kramer/von Pippich, 2013, 21)

(A14) Das größte Element der Teilermenge $T(a)$ ist a selbst, da sich jede Zahl selbst teilt (aus A02) und ein Teiler nicht größer als die Zahl selbst sein kann (aus A06).

4.3 Division mit Rest

Die Division mit Rest kennen viele Menschen bereits aus der Grundschule.
Beispiel: $7 : 3 = 2\ Rest\ 1$
Da $3 \nmid 7$ lässt sich $7 : 3$ nicht als natürliche Zahl darstellen. Hier behilft man sich mit oben gezeigter Schreibweise: Die 3 „passt“ zweimal in die 7, dann bleibt noch ein Rest von 1. An der Schreibweise lässt sich der Zusammenhang leicht erkennen, mathematisch korrekt ist sie jedoch nicht (Universität Erfurt, kein Datum, 1).

Besser ist folgende Schreibweise, die den gleichen Sachverhalt ausdrückt: $7 = 2 * 3 + 1$
Allgemein gilt:

(A15) Für gegebene $a, b \in \mathbb{N}_0;\ b \neq 0$ gibt es eindeutige $q,\ r \in \mathbb{N}_0$
für die gilt $a = q * b + r$ mit $0 \leq r < b$ (Ziegenbalg, 2015, 25)

Beweis zu A15:
Der Begriff „eindeutig“ lässt sich in zwei Teile aufspalten:
(C1) Es gibt mindestens eine Kombination von q und r, für die die Gleichung erfüllt ist.
(C2) Es gibt nicht mehr als eine solche Kombination.

Zu C1: Beweis durch vollständige Induktion
b sei beliebig festgelegt, aber innerhalb der vorgegebenen Menge.
a soll sich also als $(I) a = q * b + r$ mit $0 \leq r < b$ darstellen lassen.

Induktionsanfang: Für $a = 0$ gelten $q = 0$ und $r = 0$. Es ist also möglich, a so wie angegeben darzustellen.

Induktionsschritt: Gibt es für ein beliebiges a eine Möglichkeit, so gibt es auch eine solche, $a + 1$ als $a + 1 = q' * b + r'$ mit $0 \leq r' < b$ darzustellen. (wobei q' und r' neue Variablen sind)

Aus I ergibt sich durch Addition: (II)$a + 1 = q * b + r + 1$
Fall 1: $r = b - 1$

$r + 1 = b$	(Umstellen)
$a + 1 = q * b + b$	(In II einsetzen)
$a + 1 = (q + 1) * b + 0$	(Zusammenfassen)
$\Rightarrow q' = q + 1;\ r' = 0$	

Fall 2: $r < b - 1$

$r + 1 < b$	(Umstellen)
$\Rightarrow\ q' = q;\ r' = r + 1$	

(Ziegenbalg, 2015, 27)

Zu C2: Beweis durch Widerspruch
Annahme: Es gibt mindestens verschiedene zwei Arten, a darzustellen.
Zwei davon sind:
(I) $a = q_1 * b + r_1$ mit (III) $0 \leq r_1 < b$
(II) $a = q_2 * b + r_2$ mit (IV) $0 \leq r_2 < b$
wobei (V) $q_1 \neq q_2$ und $r_1 \neq r_2$, da es sich um verschiedene Arten handeln soll
sowie $q_1, q_2, r_1, r_2 \in \mathbb{N}_0$
Da die beiden Zeilen vertauschbar sind, sei (VI)$q_1 > q_2$ (vgl. V).
(I-II) $0 = (q_1 - q_2) * b + r_1 - r_2$
$r_2 = (q_1 - q_2) * b + r_1$
Mit $(q_1 - q_2) \geq 1$ (aus VI) folgt, dass
$r_2 \geq b$
⇒Widerspruch zu IV ■

4.4 Modulo-Rechnung

Manchmal möchte man eine Division mit Rest durchführen, es ist jedoch nur der Rest r als Ergebnis erwünscht. Hier führt man eine Modulo-Rechnung durch.

(A16) Gilt A15, so lässt sich auch (ohne Veränderung der Buchstaben) schreiben:
$a\ mod\ b := r$ (Sprich: „a modulo b gleich r“) (Ganter, 2007/2008, 5)

4.5 Kongruenzen modulo m

(A17) Haben zwei Zahlen $a, b \in \mathbb{N}_0$ bei der Division mit Rest durch $m \in \mathbb{N}$ den gleichen Rest, so schreibt man $a \equiv b \ (mod\ m)$ (Arens, u. a., 2013, 57)
Sprich: a ist kongruent zu b modulo m (Ganter, 2007/2008, 7)

(A18) $a \equiv b \ (mod\ m) \Leftrightarrow a \ mod\ m = b \ mod\ m$ (folgt aus Definition in A17)

(A19) $a \equiv b \ (mod\ m) \Leftrightarrow m \mid |a - b| \Leftrightarrow a = q * m + b$ mit $q \in \mathbb{Z}$
Beweis (von links nach rechts)
Mit A18 folgt, dass $a \ mod\ m = b \ mod\ m$
Mit A16 und A15 erhält man $a - q_1 * m = b - q_2 * m$
Durch Umstellung folgt: $a - b = (q_1 - q_2) * m$
Beziehungsweise $a - b = q * m$ mit $q \in \mathbb{Z}$
Da $m \in \mathbb{N}$, $|a - b| = |q| * m$
$\Rightarrow m \mid |a - b|$ ■

5 größter gemeinsamer Teiler (ggT)

5.1 Definition

In einem Lehrbuch wird der Begriff des größten gemeinsamen Teilers wie folgt definiert:

> *„Der größte gemeinsame Teiler zweier ganzer Zahlen a und b ist die größte natürliche Zahl, die a und b teilt“* (Arens, u. a., 2013, 1049)

Im Folgenden möchte ich mich jedoch für a und b auf natürliche Zahlen einschließlich Null beziehen. Diese Einschränkung ist ohne Änderung der Definition möglich, da $\mathbb{N}_0 \subset \mathbb{Z}$.

Der Begriff „größter gemeinsamer Teiler“ kann durch ggT abgekürzt werden (Duden, 2015a). Mathematisch schreibt man $ggT(a; b)$ (Arens, u. a., 2013, 1050)

5.2 Beweis der Existenz

Im Folgenden soll bewiesen werden, dass es zu jedem $a, b \in \mathbb{N}$ einen eindeutigen ggT gibt.
Wir setzen $ggT(a; b) =: d$
Aus der Definition folgt, dass $d \in T(a)$ sowie $d \in T(b)$.
Somit ist $d \in T(a) \cap T(b)$

Aus A11 wissen wir, dass $1 \in T(a)$ und $1 \in T(b)$. Daraus folgt, dass $1 \in T(a) \cap T(b)$. Die Schnittmenge beider Teilermengen ist also nicht leer. Die beiden Teilermengen

enthalten nur natürliche Zahlen und sind nach oben begrenzt (vgl. A14). Somit enthält auch $T(a) \cap T(b)$ nur eine begrenzte Anzahl an Elementen.

Laut Definition des ggT ist d das größte Element dieser Schnittmenge. Da jede endliche, nichtleere Menge ein größtes Element besitzt (Wikipedia, 2015), besitzt auch $T(a) \cap T(b)$ ein eindeutig bestimmbares, größtes Element, welches wir als $ggT(a; b)$ bezeichnen. ■

5.3 Eigenschaften

für $a, b, c \in \mathbb{N}_0$ gilt:

(A20) $ggT(a, b) = ggT(b, a) \ \forall a, b \in \mathbb{N}_0$ (Ziegenbalg, 2015, 40)

(A21) $ggT(a, 0) = a \ \forall a \in \mathbb{N}_0$ (Ziegenbalg, 2015, 40)
da alle natürlichen Zahlen Null teilen (vgl. A04)

(A22) Ergänzend definiert man $ggT(0, 0) = 0$ (Arens, u. a., 2013, 1050)

(A23) $ggT(a, b, c) = ggT\big(a,\ ggT(b, c)\big) \ \forall a, b, c \in \mathbb{N}_0$ (Schulze-Pillot, 2015, 47)

(A24) $ggT(a, b) = 1 \Leftrightarrow$ a und b nennt man zueinander teilerfremd
(Ziegenbalg, 2015, 39)

(A25) Zwei unterschiedliche Primzahlen sind zueinander immer teilerfremd.
Beweis:
Die Primzahlen haben die Teilermengen $T(a) = \{1; a\}$ sowie $T(b) = \{1; b\}$ (aus A13), wobei $a \neq b$. Somit ist $T(a) \cap T(b) = \{1\}$. Das größte (und auch einzige) Element in dieser Schnittmenge ist 1. Mit A24 folgt, dass die beiden Zahlen a und b teilerfremd sind. ■

(A26) $ggT(a, b) \geq 1$
Beweis durch Widerspruch:
Es wird angenommen, dass für bestimmte Werte von a und b $(I) ggT(a, b) =: k < 1$ gilt. Da aber $1 \mid a$ und $1 \mid b$ (vgl. A03) sowie $1 > k$, kann k nicht der größte gemeinsame Teiler sein. Dies ist ein Widerspruch zu I. ■

5.4 Primitive Bestimmung

► *greatest_common_divisor.html*

Bis jetzt habe ich den Begriff des ggT beschrieben, jedoch keine Methode, diesen zu finden. Die primitivste Methode besteht darin, einfach die Teilermengen miteinander zu vergleichen.

Beispiel: Der ggT(24; 44) soll bestimmt werden.

$T(24) = \{1; 2; 3; \mathbf{4}; 6; 8; 12; 24\}$
$T(44) = \{1; 2; \mathbf{4}; 11; 22; 44\}$

Die größte Zahl, die in beiden Mengen vorkommt, ist die 4. Somit ist $ggT(24;44) = 4$ Offensichtlich ist hier jedoch, dass diese Vorgehensweise bei größeren Zahlen schnell sehr aufwendig wird.

5.5 Bestimmung mithilfe der Primfaktorzerlegung

►*prime_decomposition.html*

Eine andere Methode, die viele Schüler auch in der Unterstufe lernen, ist die Bestimmung mithilfe der Primfaktorzerlegung. Diese beruht darauf, dass sich jede natürliche Zahl als Produkt von Primzahlen eindeutig darstellen lässt (Knapp, 2006, 5). Die Primfaktorzerlegung der oben verwendeten Zahlen sieht wie folgt aus:

$24 = 2 * 2 * 2 * 3 = 2^3 * 3^1$
$44 = 2 * 2 * 11 = 2^2 * 11^1$

Der ggT ist nun das Produkt der Primfaktoren mit ihrem jeweils niedrigsten Exponenten, der in einer der beiden Zerlegungen vorkommt (Bauerhenne, 2010).

$ggT(24, 44) = 2^2 * 3^0 * 11^0 = 4$

Bei der Primzahl 2 ist der Exponent 2 in der zweiten Zeile der Niedrigere. Für die Primzahl 3 könnte man die zweite Zeile um den Faktor 3^0 ergänzen, da $3^0 = 1$. Für die Primzahl 3 ist dadurch 0 der niedrigste Exponent. Dasselbe gilt für 11.

6 Euklidischer Algorithmus (EA)

Eine weitere Methode zur Bestimmung des ggT ist der Euklidische Algorithmus, der auch den Hauptteil dieser Seminararbeit darstellt.

6.1 Funktionsweise

►*euclidean_algorithm.html*

Beim EA wird eine Serie von Divisionen mit Rest durchgeführt. In der ersten dieser Divisionen sind die beiden Zahlen, von denen der ggT berechnet werden soll, jeweils der Dividend bzw. Divisor. Nach der Division mit Rest wird die nächste Zeile gebildet: Der Divisor wird zum neuen Dividenden und der Rest wird zum neuen Divisor. Nach diesem Schema geht man solange vor, bis bei einer Division der Rest 0 bleibt. Der Divisor in dieser Zeile ist der ggT der beiden Ausgangszahlen (Oswald/Steuding, 2015, 67). Ein Beispiel mit den bereits verwendeten Zahlen 24 und 44 soll dies verdeutlichen:

$$44 = 1 * 24 + 20$$
$$24 = 1 * 20 + 4$$
$$20 = 5 * 4 + 0$$

Da in der letzten Zeile der Rest 0 bleibt, ist 4 der größte gemeinsame Teiler von 24 und 44. (Die Richtigkeit wird an späterer Stelle auch noch bewiesen)

Allgemeiner kann man also formulieren:
Will man $ggT(a,b)$ berechnen, so setzt man $r_0 := a$ und $r_1 := b$ und führt dann folgende Serie an Divisionen mit Rest durch:

(Zeile 0)	r_0	$=$	$q_1 * r_1$	$+$	r_2
(Zeile 1)	r_1	$=$	$q_2 * r_2$	$+$	r_3
…..	….	….	….	….	….
(Zeile i)	r_i	$=$	$q_{i+1} * r_{i+1}$	$+$	r_{i+2}
…..	….	….	….	….	….
(Zeile n-1)	r_{n-1}	$=$	$q_n * r_n$	$+$	$\boldsymbol{r_{n+1}}$
(Zeile n)	r_n	$=$	$q_{n+1} * \boldsymbol{r_{n+1}}$	$+$	0

(Oswald/Steuding, 2015, 67)

(Anmerkung: Zeile i stellt eine beliebige Zeile dar)
Wie bei jeder Division mit Rest gilt zusätzlich $0 < r_{i+2} < r_{i+1}$ für $0 \leq i < n$. (Oswald/Steuding, 2015, 67) Da die Reste abnehmen, wird nach einer endlichen Anzahl von Schritten der Rest 0 erreicht. Dies ist in der Zeile n der Fall. Nach der Formulierung von oben gilt $r_{n+1} = ggT(a,b)$.
Um später einfacher argumentieren zu können, setze ich $d := r_{n+1}$.

6.2 Beweis der Richtigkeit

Das Wichtigste fehlt noch: Es muss bewiesen werden, dass der Euklidische Algorithmus wirklich den ggT der beiden Ausgangszahlen herausgibt.
Die Aussage lässt sich in zwei Teilaussagen aufteilen
(C1) Die gefundene Zahl d teilt beide Ausgangszahlen.
(C2) Es gibt keine größere Zahl als d, die beide Ausgangszahlen teilt.

zu C1: Beweis durch vollständige Induktion
Induktionsanfang:
Betrachten wir die Zeile n. Da $d = r_{n+1}$ gilt logischerweise $d \mid r_{n+1}$ (vgl. A02).
Da r_n ein Vielfaches von r_{n+1} ist, gilt auch $d \mid r_n$ (vgl. A05)

Induktionsschritt:
Betrachten wir die Zeile i für $0 \leq i < n$. Unter der Voraussetzung, dass $d \mid r_{i+2}$ und $d \mid r_{i+1}$, folgt mit A05, dass $d \mid q_{i+1} * r_{i+1}$ und sodann mit A08, dass $d \mid r_i$.

Durch den Induktionsanfang und Induktionsschritte kann man darauf schließen, dass $d \mid r_1 = b$ und $d \mid r_0 = a$.

(Arens, u. a., 2013, 1050)

zu C2: Beweis durch vollständige Induktion und Widerspruch
Wir nehmen an, es gäbe ein d', das Teiler beider Zahlen a und b ist und für das (I) $d' > d$ gilt.

Induktionsanfang:
Betrachten wir die Zeile 0. Durch die Definition von d' ergibt sich, dass $d' \mid r_0 = a$ und $d' \mid r_1 = b$.

Induktionsschritt:
Betrachten wir die Zeile i für $0 \leq i < n$. Unter der Voraussetzung, dass $d' \mid r_i$ und $d' \mid r_{i+1}$, folgt mit A05, dass $d' \mid q_{i+1} * r_{i+1}$ und sodann mit A09, dass $d' \mid r_{i+2}$.

Mithilfe des Induktionsanfangs und der Induktionsschritte lässt sich in Zeile n-1 folgern, dass $d' \mid r_{n+1} = d$. Mit A06 folgt, dass $d' \leq d$. Dies ist jedoch ein Widerspruch zu I. ■

6.3 Vor- und Nachteile

Nun habe ich drei Methoden vorgestellt, mit denen man den ggT zweier Zahlen bestimmen kann. Selbstverständlich liefern alle drei das gleiche Ergebnis. Im Folgenden möchte ich darauf eingehen, warum man den Euklidischen Algorithmus vorziehen sollte und auch, welche Nachteile dieser bietet.

So kann es ein Nachteil sein, dass bei kleineren Zahlen der größte gemeinsame Teiler fast schon offensichtlich ist. Zwei Primzahlen sind z.B. immer zueinander teilerfremd (vgl. A25). Der EA kann jedoch mehrere Schritte benötigen, bis dieses Ergebnis gefunden wird. In Abbildung 1 im Anhang wird dies am Beispiel der Primzahlen 11 und 19 verdeutlicht. Somit kann eine der anderen beiden Methode schneller zum Ziel führen als der EA.

Ein weiterer Nachteil ist, dass die Richtigkeit des EA dem Betrachter nicht sofort ersichtlich ist. Es bedarf einiger Vorüberlegungen sowie eines Beweises, der auch etwas Platz in Anspruch nimmt. Bei der primitiven Methode, die in Kapitel 5.4 beschrieben wird, ist die Richtigkeit offensichtlich und deshalb ist sie leichter zu verstehen.

Allerdings bietet der Euklidische Algorithmus auch Vorteile:
Bei der Computerberechnung benötigt dieser besonders bei großen Zahlen weniger Zeit zur Berechnung. So zeigt ein Beispiel der Queen's University mit zwei knapp 20-stelligen Zahlen, dass mithilfe des EA gut 99,9 % der Rechenzeit verglichen zu der Zeit der Primfaktor-Methode eingespart werden kann (Kani, 1999). Dies liegt daran, dass eine Primfaktorzerlegung für Computer sehr aufwendig ist. Bei geschickt gewählten, sehr

großen Zahlen kann die Zerlegung sogar praktisch unmöglich sein, wenn sie z.B. mehrere Jahre dauern würde (Arens, u. a., 2013, 1049).

Weiterhin werden aufwendige Divisionen vermieden. So heißt es zwar „Division mit Rest“, diese kann jedoch ohne eigentliche Divisionen, sondern nur mit Subtraktionen und Additionen verwirklicht werden. So setzt auch das der Seminararbeit beiliegende Programm einen solchen Algorithmus ein, der im Anhang beschrieben wird (siehe Verfahren 1). Dadurch kann weitere Rechenzeit eingespart werden.

Außerdem ist der Euklidische Algorithmus ein Verfahren, bei dem eindeutig ist, welcher Schritt als nächstes auszuführen ist. Durch jede Zeile kommt man (außer evtl. durch die erste Zeile, wenn die zweite Zahl die Größere ist) dem Ergebnis einen Schritt näher. Im Gegensatz dazu ist bei der Primfaktormethode die Vorgehensweise eine andere: Entweder findet man durch Erraten die Primfaktoren oder durch Ausprobieren aller Möglichkeiten bis zu einer bestimmten Zahl. In beiden Fällen führt man „nutzlose“ Rechnungen durch. So kommt man bei der Primfaktorzerlegung keinen Schritt weiter, wenn man versucht, 21 durch 5 zu teilen, außer natürlich, dass man dann die 5 ausschließen kann.

Wie erläutert, bietet der Euklidische Algorithmus einige Vorteile gegenüber den anderen beiden Methoden. Dadurch empfiehlt sich vor allem in Computeralgorithmen der Einsatz, wo sich auch die Nachteile des Algorithmus verschmerzen lassen.

7 Erweiterter Euklidischer Algorithmus (ErwEA)

7.1 Ziel

Lemma von Bezout: Der größte gemeinsame Teiler von $a, b \in \mathbb{N}$ lässt sich als Linearkombination wie folgt darstellen: $ggT(a, b) = s * a + t * b$ mit $s, t \in \mathbb{Z}$ (Schwan, 2002, 11).
Die Zahlen s und t sind dabei aber nicht eindeutig, sondern es gibt unendlich viele Zahlenpaare (Arens, u. a., 2013, 1051).
Der Erweiterte Euklidische Algorithmus ist ein Verfahren, um diese Zahlen zu finden. (Lehrstuhl für Netz- und Datensicherheit Bochum, 2014, 1)

Wie der Name schon andeutet, handelt es sich um eine Erweiterung des Euklidischen Algorithmus. Deshalb beziehe ich mich im Folgenden auch auf Kapitel 6.

7.2 Beweis der Möglichkeit

Die Zeilen des Euklidischen Algorithmus wie in Kapitel 6.1 werden jetzt rückwärts, d.h. von unten nach oben betrachtet.

Induktionsanfang:
Es gilt $d := ggT(a,b) = r_{n+1}$.
Aus Zeile n-1 folgt mit Umstellung und Einsetzen $(I) ggT(a,b) = 1 * r_{n-1} - q_n * r_n$. Der ggT lässt sich also als Linearkombination von r_{n-1} und r_n darstellen.

Induktionsschritt:
Es gilt $0 \leq i < n$.
Voraussetzung: $ggT(a,b)$ lässt sich als Linearkombination von r_{i+1} und r_{i+2} darstellen, d.h. $ggT(a,b) = s_{i+1} * r_{i+1} + t_{i+1} * r_{i+2}$.
Zeile i wird umgestellt: $r_{i+2} = r_i - q_{i+1} * r_{i+1}$ und eingesetzt:
$ggT(a,b) = s_{i+1} * r_{i+1} + t_{i+1} * (r_i - q_{i+1} * r_{i+1})$
Zusammenfassen ergibt:
$(II) ggT(a,b) = t_{i+1} * r_i + (s_{i+1} - t_{i+1} * q_{i+1}) * r_{i+1}$
Somit lässt sich der ggT auch als Linearkombination von r_i und r_{i+1} darstellen.

Durch den Induktionsanfang und Induktionsschritte folgt, dass sich $ggT(a,b)$ als Linearkombination von $r_0 = a$ und $r_1 = b$ darstellen lässt. ■

(Arens, u. a., 2013, 1051)

7.3 Beispiel

► *extended_euclidean_algorithm.html*

Ich greife auf mein Beispiel aus Kapitel 6.1 zurück. Zuerst wird der EA wie beschrieben durchgeführt:

$$44 = 1 * 24 + 20$$
$$24 = 1 * 20 + 4$$
$$20 = 5 * 4 + 0$$

Aus der zweiten Zeile ergibt sich
$(I)\ 4 = 1 * 24 - 1 * 20$
Hier muss aber noch die 20 eliminiert werden. Aus Zeile 1 folgt:
$(II) 20 = 1 * 44 - 1 * 24$
II wird nun in I eingesetzt.
$4 = 1 * 24 - 1 * (1 * 44 - 1 * 24)$
Zusammengefasst gilt also:
$4 = ggT(44,\ 24) = -1 * 44 + 2 * 24$

Wie bereits erwähnt, ist dies nicht die einzige Möglichkeit der Darstellung. So ist auch $ggT(44,24) = 5 * 44 + (-9) * 24$ eine gültige Lösung.

7.4 Auffinden weiterer Darstellungen

Durch den ErwEA lässt sich eine mögliche Linearkombination herausfinden. Doch wie kommt man auf weitere Darstellungen?
Es gibt bereits die folgende Darstellung:
$ggT(a,b) = s * a + t * b$
Um auf die „benachbarte“ Darstellung zu kommen, muss zu einem Summanden ein bestimmtes k addiert, vom anderen dieses subtrahiert werden (oder umgekehrt). Das bedeutet (Es wird im Folgenden nur eine Richtung gezeigt):
$ggT(a,b) = s * a + k + t * b - k$ mit $k \in \mathbb{N}; k\ m\ddot{o}glichst\ klein$
$ggT(a,b) = (s + \frac{k}{a}) * a + (t - \frac{k}{b}) * b$
Damit $(s + \frac{k}{a}) \in \mathbb{Z}$, muss $a \mid k$ und dass $(t - \frac{k}{b}) \in \mathbb{Z}$, muss $b \mid k$ gelten.
k ist somit ein Vielfaches von a und b und da k möglichst klein sein soll, handelt es sich um das sog. kleinste gemeinsame Vielfache, abgekürzt kgV. (Duden, 2015b)
Für das kgV gilt: $kgV(a,b) = \frac{a*b}{ggT(a,b)} = k$ (Arens, u. a., 2013, 1057)
Eingesetzt in die vorherige Gleichung erhält man:
$ggT(a,b) = \left(s + \frac{b}{ggT(a,b)}\right) * a + \left(t - \frac{a}{ggT(a,b)}\right) * b$
Dies ist die nächstmögliche Darstellung bei Erhöhung von s und Verminderung von t. Für die entgegengesetzte Richtung müssen Addition und Subtraktion entsprechend umgekehrt werden.

7.5 Verallgemeinerung

Aktuell ist der ErwEA verständlich und durchführbar, allerdings gibt es noch eine Verallgemeinerung, die ich hier durchführe. Diese soll es erleichtern, einen Computeralgorithmus zu schreiben, der eine dieser Linearkombinationen berechnen kann. Dazu erweitere ich die Vorgehensweise aus Kapitel 6.1 um je eine Gleichung pro Zeile:

(Zeile 0)	r_0	=	$q_1 * r_1$	+	r_2	(Y 0)	$d =$	$s_0 * r_0$	+	$t_0 * r_1$
…..	….	…	….	…	….	….	….	….	.	….
(Zeile i)	r_i	=	$q_{i+1} * r_{i+1}$	+	r_{i+2}	(Y i)	$d =$	$s_i * r_i$	+	$t_i * r_{i+1}$
(Zeile i+1)	r_{i+1}	=	$q_{i+2} * r_{i+2}$	+	r_{i+3}	(Y i+1)	$d =$	$s_{i+1} * r_{i+1}$	+	$t_{i+1} * r_{i+2}$
…..	….	…	….	…	….	….	….	…	.	….
(Zeile n-1)	r_{n-1}	=	$q_n * r_n$	+	$\boldsymbol{r_{n+1}}$	(Y n-1)	$d =$	$s_{n-1} * r_{n-1}$	+	$t_{n-1} * r_n$
(Zeile n)	r_n	=	$q_{n+1} * \boldsymbol{r_{n+1}}$	+	0					

Somit wird $d \coloneqq ggT(r_0, r_1) = r_{n+1}$in jeder Zeile als Linearkombination des jeweiligen Dividenden und Divisors angeben. Jetzt soll eine Möglichkeit gefunden werden, wie die jeweiligen s, t gefunden werden, wenn die der darunter liegenden Zeile sowie der vordere Teil der Zeile bekannt ist.

Aus Kapitel 7.2 Aussage II lässt sich entnehmen, dass aus Y i+1 mit Zeile i für $0 \leq i < n$ folgt:
$d = t_{i+1} * r_i + (s_{i+1} - t_{i+1} * q_{i+1}) * r_{i+1}$
Y i besagt, dass $d = s_i * r_i + t_i * r_{i+1}$
Werden beide Aussagen verglichen, so folgt:
$s_i = t_{i+1}$; $t_i = s_{i+1} - t_{i+1} * q_{i+1}$

Für den Beginn des ErwEA in der Zeile n-1 lässt sich aus Kapitel 7.2 Aussage I folgern, dass $s_{n-1} = 1$ sowie $t_{n-1} = -q_n$

7.6 Berechnung modularer Inversen

Definition:
(A27) Sind $a, b \in \mathbb{N}$ und beide Zahlen zueinander teilerfremd, so gibt es für $ax \equiv 1\ (mod\ b)$ mögliche $x \in \mathbb{Z}$. (Chamberlain, u. a., 2014, 24)
x wird dann auch als $x = a^{-1}\ mod\ b$ dargestellt und man sagt: „x gleich dem Inversen von a modulo b“. (Lehrstuhl für Netz- und Datensicherheit Bochum, 2014, 2)
(Anmerkung: Mit dem Exponenten -1 ist hier nicht der Kehrbruch gemeint.)

Beispiel: $7 = 5^{-1}\ mod\ 17$, da $7 * 5 = 35 \equiv 1\ (mod\ 17)$

Mithilfe des ErwEA lassen sich modulare Inverse berechnen (Lehrstuhl für Netz- und Datensicherheit Bochum, 2014, 2).
Da a und b teilerfremd sein müssen, gibt es eine Darstellung der Form $ggT(a, b) = 1 = s * a + t * b$
Mit A19 folgt, dass $1 \equiv s * a\ (mod\ b)$. Über A27 kommt man dann darauf, dass $s = a^{-1}\ mod\ b$. Auf gleiche Weise lässt sich auch das Inverse modulo a von b finden.

8 Anwendung am RSA-Algorithmus

8.1 Ziel

Der RSA-Algorithmus ist eine Verschlüsselungsmethode, die es ermöglicht, verschlüsselte Nachrichten ohne vorherigen Schlüsseltausch zu versenden. (Chamberlain, u. a., 2014, 2f) .
Unter Schlüsseltausch versteht man den Vorgang, dass zwei Parteien, die miteinander verschlüsselt kommunizieren möchten, sich vorher im Geheimen auf einen Schlüssel einigen müssen, mit dem die Nachrichten ver- und entschlüsselt werden können. Dies ist bei symmetrischen Verschlüsselungsverfahren notwendig. (Meisel & Mileski, 2006, 7 u. 11)

Im Gegensatz dazu werden bei public-key-Verfahren wie RSA zwei Schlüssel generiert, nämlich der öffentliche Schlüssel zum Verschlüsseln und der private Schlüssel zum Entschlüsseln (Meisel & Mileski, 2006, 8). Der öffentliche Schlüssel kann dabei, wie der Name schon verrät, veröffentlicht werden, denn der private Schlüssel lässt sich daraus nicht berechnen (Meisel & Mileski, 2006, 11f).

Die Parteien können diese Schlüssel auch durch unsichere Leitungen austauschen, da ein Mithörer durch Kenntnis dieser nicht in der Lage ist, die darauffolgende Kommunikation abzuhören. Solche Verfahren ermöglichen es also Parteien, die vorher nie miteinander in Kontakt getreten sind, verschlüsselte Verbindungen aufzubauen.

8.2 Notwendigkeit großer Primzahlen

Für das Generieren eines solchen Schlüsselpaares werden zwei große Primzahlen benötigt, empfohlen wird zurzeit die Größenordnung 2^{1024} (Arens, u. a., 2013, 1068).
Das Produkt beider Primzahlen stellt einen Teil des öffentlichen Schlüssels dar. Mit den beiden Primzahlen wird über eine andere Verfahrensweise der private Schlüssel bestimmt (Beutelspacher, u. a., 2005, 117). Das heißt, wer die beiden Primzahlen kennt, kann den privaten Schlüssel berechnen.
Um dies zu verhindern, hält man diese Zahlen geheim. Das Produkt muss allerdings veröffentlicht werden. Kennt man das Produkt zweier Primzahlen, so kann man aber diese beiden Primzahlen durch Primfaktorzerlegung bestimmen, zumindest theoretisch.

Man vermutet nämlich, dass es sich bei der Multiplikation zweier großer Primzahlen um eine Einwegfunktion handelt (Beutelspacher, u. a., 2005, 117). Darunter versteht man eine Funktion, die einfach zu berechnen ist, aber deren Umkehrung praktisch unmöglich ist. (Meisel & Mileski, 2006, 26)
So ist es relativ einfach zwei Primzahlen miteinander zu multiplizieren, die Faktorisierung jedoch nach heutigem Erkenntnisstand ungleichmäßig komplizierter. (Beutelspacher, u. a., 2005, 117). Bei sehr großen Primzahlen aus dem oben genanntem Bereich beträgt die Berechnungsdauer mit heutiger Rechenleistung mehrere Tausend Jahre (Arens, u. a., 2013, 1068).

Durch große Primzahlen ist es also praktisch unmöglich, die Faktorisierung durchzuführen und die Primzahlen bleiben geheim.

8.3 Funktionsweise

►*rsa_alice.html*
►*rsa_bob.html*

Wie auch in der Literatur teilweise üblich, werde ich die beiden Parteien im Folgenden Alice und Bob nennen.
Bob möchte Alice eine verschlüsselte Nachricht schicken. Zuerst muss Alice ein eigenes Schlüsselpaar generieren.

Dazu wählt sie zwei zufällige, große Primzahlen $p, q \in \mathbb{P};\ p \neq q$. Solche findet man, in dem man Zufallszahlen im gewünschten Bereich mit Primzahlentests prüft. (Arens, u. a., 2013, 1068)
Anmerkung: Das RSA-Verfahren funktioniert grundsätzlich auch mit kleineren Primzahlen. Allerdings handelt es sich bei der Multiplikation dann nicht mehr um eine Einwegfunktion (siehe vorheriges Kapitel) und die Verschlüsselung kann geknackt werden.

Alice berechnet dann $n = p * q$ und $\varphi(n) = (p-1) * (q-1)$.
Anschließend wählt sie ein $e \in \mathbb{N}$ mit $1 < e < \varphi(n)$ und $ggT\big(e, \varphi(n)\big) = 1$. Das Tupel (n, e) stellt nun den öffentlichen Schlüssel von Alice dar. (Arens, u. a., 2013, 1068)

Ihren privaten Schlüssel $d \in \mathbb{N}$ kann sie über die Beziehung $d * e \equiv 1\ (mod\ \varphi(n))$ berechnen. (Arens, u. a., 2013, 1068)

Bob besorgt sich jetzt den öffentlichen Schlüssel von Alice.
Er wandelt sodann seine geheime Nachricht in die Zahl $m \in \mathbb{N}$ um (Chamberlain, u. a., 2014, 29), wobei $m < n$ gelten muss (Arens, u. a., 2013, 1068). Dafür muss die Nachricht notfalls in mehrere Blöcke aufgeteilt werden (Chamberlain, u. a., 2014, 29).

Anschließend berechnet Bob $y \coloneqq\ m^e\ mod\ n$ und überträgt dieses Ergebnis an Alice. Diese kann die ursprüngliche Nachricht mit $m =\ y^d\ mod\ n$ bestimmen. (Chamberlain, u. a., 2014, 29)

Möchte auch Alice an Bob eine verschlüsselte Nachricht schicken, muss der gesamte Prozess andersherum stattfinden, d.h. auch Bob muss ein Schlüsselpaar generieren.

8.4 Einsatz des EA und ErwEA

Beim RSA-Algorithmus können an zwei Stellen EA und ErwEA angewendet werden:

Erstens muss e teilerfremd zu $\varphi(n)$ gewählt werden.
Dazu wählt man eine Zahl e, sodass $1 < e < \varphi(n)$ und prüft mit dem Euklidischen Algorithmus, ob $ggT\big(e,\ \varphi(n)\big) = 1$ gilt.

Der Euklidische Algorithmus kann dabei schneller als die Primfaktormethode angewandt werden, wie bereits in Kapitel 6.3 festgestellt wurde.
Weiterhin birgt die Primfaktormethode aber auch ein Sicherheitsrisiko: $ggT(e, \varphi(n))$ soll mithilfe dieser bestimmt werden. Folglich müssen e und $\varphi(n)$ faktorisiert werden. Das ist praktisch nicht möglich, es sei denn, man wählt die beiden Primzahlen p und q so klein, dass $\varphi(n)$ in kurzer Zeit faktorisierbar wäre. Da $\varphi(n)$ und n jedoch in etwa die gleiche Größenordnung haben, ist es nicht unwahrscheinlich, dass ein Angreifer mit gleicher Rechenleistung wie Alice in der Lage ist, n in seine Primfaktoren zu zerlegen. Genau dieser Fall soll jedoch verhindert werden.
Aus Gründen der Sicherheit und Schnelligkeit sollte man also dazu den Euklidischen Algorithmus anwenden.

Einen Schritt später muss Alice ihren privaten Schlüssel d berechnen. Dazu benutzt sie – wie oben dargestellt – die Formel $d * e \equiv 1\ (mod\ \varphi(n))$. e ist bereits bekannt. Umstellen nach d ergibt: $d = e^{-1}\ mod\ \varphi(n)$
Hierzu wird der Erweiterte Euklidische Algorithmus eingesetzt (Beutelspacher, u. a., 2005, 122). Ein weiterer Vorteil neben der Geschwindigkeit ist, dass ein Teil der Berechnung bereits vorliegt: Für die Berechnung von d wird zunächst der EA auf e und $\varphi(n)$ angewendet. Dieser Vorgang wurde jedoch schon im vorherigen Schritt durchgeführt. Durch Zwischenspeichern der Teilergebnisse kann man sich Rechenarbeit und damit Zeit bei der Ausführung sparen.

8.5 Anwendungen

Bei RSA handelt es sich um das meisteingesetzte Kryptografie-Verfahren (Meisel & Mileski, 2006, 46). Es hat für die moderne Kommunikation eine wichtige Bedeutung. Nachfolgend nenne ich einige Beispiele:

So kann das Verfahren bei der SSL-Verschlüsselung zum Einsatz kommen. (Meisel & Mileski, 2006, 45) SSL wird häufig eingesetzt, wenn ein Benutzer eine verschlüsselte Verbindung zu einem Server aufbauen möchte und persönliche Daten wie z.B. Passwörter oder Kreditkartennummern übertragen will. (DigiCert, 2015) Hier macht man sich die Tatsache zunutze, dass der öffentliche Schlüssel des Servers auch unverschlüsselt übertragen werden kann, ohne die Sicherheit des Verfahrens zu beeinträchtigen.
Eine Stichprobe von mir hat ergeben, dass Unternehmen wie z.B. Google, Microsoft und Amazon SSL mithilfe des RSA-Verfahrens für ihre verschlüsselten Verbindungen einsetzen.

Weiterhin ist es möglich, mithilfe von PGP verschlüsselte E-Mails zu versenden (Meisel & Mileski, 2006, 45). Dies steht für „Pretty Good Privacy“ und kann u.a. auch RSA einsetzen (Rouse, 2014). Besonders bekannt wurde das Verfahren dadurch, dass Edward Snowden im Jahr 2013 PGP zur verschlüsselten Informationsweitergabe an Journalisten benutzte (Calderone, 2013).

Schließlich kann mit RSA ein Dokument signiert werden.
Mit dem öffentlichen Schlüssel kann theoretisch jeder eine Nachricht an den Empfänger schreiben. Dieser kann sich somit nicht sicher sein, wer der Absender ist.
Hier schafft das RSA-Verfahren Abhilfe, indem das Dokument mithilfe des privaten Schlüssels des Senders signiert wird. Mit dem öffentlichen Schlüssel des Senders kann der Empfänger verifizieren, dass die Nachricht tatsächlich vom richtigen Absender stammt. (Meisel & Mileski, 2006, 43f)

9 Schlussbemerkung

Der Euklidische Algorithmus wurde vor sehr langer Zeit entwickelt. Trotzdem stellt er ein effizientes Verfahren dar, um den größten gemeinsamen Teiler zweier Zahlen zu berechnen. Mit der Erweiterung des Algorithmus eröffnet sich zusätzlich die Möglichkeit, modulare Inverse zu erhalten.
In der Kryptografie hat dies enorme Auswirkungen: Das vielseitig einsetzbare RSA-Verfahren basiert genau auf der schnellen Ausführung dieser beiden Operationen bei großen Zahlen. Da die anderen bekannten Verfahren, die ich zur Berechnung des ggT vorgestellt habe, hier sehr viel leistungsschwächer sind, wäre der RSA-Algorithmus kaum ohne den EA einsetzbar.
Der RSA-Algorithmus wird häufig für den Aufbau von verschlüsselter elektronischer Kommunikation verwendet. Somit kommt auch dem Erweiterten Euklidischen Algorithmus eine elementare Bedeutung für die Privatsphäre im Internet zu.

Anhänge

Abbildung 1: ggT-Berechnung von 11 und 19, selbst erstelltes Werk (vgl. die der Seminararbeit beiliegende Applikation)

Calculating the gcd of 11 and 19 using the Euclidean Algorithm....

1: 11 = 0 * 19 + 11
2: 19 = 1 * 11 + 8
3: 11 = 1 * 8 + 3
4: 8 = 2 * 3 + 2
5: 3 = 1 * 2 + 1
6: 2 = 2 * 1 + 0

Result: gcd(11, 19) = 1
These numbers are coprime.

Verfahren 1: Computer-Algorithmus zur Division mit Rest

Mit folgendem Verfahren soll die Darstellung der Division mit Rest gefunden werden, ohne eine „echte“ Division durchzuführen.

Verfahrensbeschreibung

$a \in \mathbb{N}_0$ soll mit Rest durch $b \in \mathbb{N}$ dividiert werden. Man setzte $q = 0$ und $r = a$. Dann wird q jeweils um 1 erhöht und r um b reduziert, bis die Bedingung $0 \leq r < b$ erfüllt ist. $a = q * b + r$ stellt sodann die Division mit Rest dar.

Beweis

Die eindeutige (vgl. A15) Darstellung $(I) a = q^* * b + r^*;\ \ q^*, r^* \in \mathbb{N}_0$ wird gesucht, die die Bedingung $0 \leq r^* < b$ erfüllt.

Induktionsanfang: Es gilt $a = q_0 * b + r_0$ mit $q_0 = 0;\ r_0 = a$, da $a = 0 * b + a$

Für $0 \leq i \leq n$ gilt $a = q_i * b + r_i$
Induktionsschritt: Es gilt also auch $a = q_i * b + r_i + b - b$ (Addition und Subtraktion von b). Umstellen ergibt $a = (q_i + 1) * b + (r_i - b)$.
Anders ausgedrückt: $a = q_{i+1} * b + r_{i+1}$ mit $q_{i+1} = q_i + 1$ und $r_{i+1} = r_i - b$ für $0 \leq i < n$

Nach endlicher Anzahl von Induktionsschritten wird die Darstellung $(II)\ a = q_n * b + r_n$ erreicht, bei der $(III) q_n = q^*$ gilt. Durch Vergleich von I, II und III stellt man fest, dass auch $r_n = r^*$ gilt. Die Darstellung der Division mit Rest ist somit gefunden. ■

Literaturverzeichnis

Bücher

Arens, u. a., 2013	Arens, Tilo/ Busam, Rolf/ Hettlich, Frank/ Karpfinger, Christian/ Stachel, Hellmuth: Grundwissen Mathematikstudium, Berlin 2013
Beutelspacher, u. a., 2005	Beutelspacher, Albrecht/ Neumann, Heike/ Schwarzpaul, Thomas: Kryptografie in Theorie und Praxis, Wiesbaden 2005
Duden, 2005	Duden: Formelsammlung Mathematik, Mannheim 2005
Duden, 2011	Duden: Was jeder wissen muss, Mannheim 2011
Herrmann, 2014	Herrmann, Dietmar: Die antike Mathematik, Heidelberg 2014
Knapp, 2006	Knapp, Anthony W: Basic Algebra, New York 2006
Kramer/ von Pippich, 2013	Kramer, Jürg/ von Pippich, Anna-Maria: Von den natürlichen Zahlen zu den Quaternionen, Wiesbaden 2013
Oswald/ Steuding, 2015	Oswald, Nicola/ Steuding, Jörn: Elementare Zahlentheorie, Berlin 2015
Schulze-Pillot, 2015	Schulze-Pillot, Rainer: Einführung in Algebra und Zahlentheorie, Berlin 2015
Ziegenbalg, 2015	Ziegenbalg, Jochen: Elementare Zahlentheorie, Wiesbaden 2015

Internetquellen

Bauerhenne, 2010	Bauerhenne, Bernd: ggT und kgV, 2010, Internetpublikation unter http://www.mathematik.de/ger/fragenantworten/erstehilfe/ggtundkgv/ggtundkgv.html?print=1 [Zugriff am 25.08.2015]
Bescherer, 2006	Bescherer, C.: Didaktik der Zahlenbereiche (4), 2006, Internetpublikation unter http://www.math.uni-augsburg.de/de/prof/dida/studium/lehre/ws0607/didazahlensystem/Folien/Folien05.pdf [Zugriff am 01.08.2015]
Busse, u. a., 1999	Busse, Michael/ Schmitt, Matthias/ Steeg, Jörg: Der RSA-Algorithmus, 1999, Internetpublikation unter http://www.zum.de/Faecher/Inf/RP/infschul/kr_rsa.html [Zugriff am 31.10.2015]

Calderbank, 2007 — Calderbank, Michael: The RSA Cryptosystem: History, Algorithms, Primes, 2007, Internetpublikation unter http://www.math.uchicago.edu/~may/VIGRE/VIGRE2007/REUPapers/FINALAPP/Calderbank.pdf [Zugriff am 22.11.2015]

Calderone, 2013 — Calderone, Michael: How Glenn Greenwald Began Communicating With NSA Whistleblower Edward Snowden, 2013, Internetpublikation unter http://www.huffingtonpost.com/2013/06/10/edward-snowden-glenn-greenwald_n_3416978.html [Zugriff am 31.10.2015]

Chamberlain, u.a., 2014 — Chamberlain, E./ Guevara Vasquez, F./ Hohenegger, C./ Korevaar, N.: ACCESS: Cryptography, 2014, Internetpublikation unter http://www.math.utah.edu/~fguevara/ACCESS2014/lecturenotes.pdf [Zugriff am 27.09.2015]

DigiCert, 2015 — DigiCert: What is SSL and what are SSL Certificates?, 2015, Internetpublikation unter https://www.digicert.com/ssl.htm [Zugriff am 14.11.2015]

Duden, 2015a — Duden: ggT, 2015, Internetpublikation unter http://www.duden.de/node/819874/revisions/1104985/view [Zugriff am 25.08.2015]

Duden, 2015b — Duden: kgV, 2015, Internetpublikation unter http://www.duden.de/node/819883/revisions/1132323/view [Zugriff am 02.09.2015]

Forster, 2004 — Forster, O.: Einführung in die Zahlentheorie, 2004, Internetpublikation unter http://www.mathematik.uni-muenchen.de/~forster/v/zth/inzth_02.pdf [Zugriff am 29.07.2015]

Froese, 2015 — Froese, Norbert: Euklid und die Elemente, 2015, Internetpublikation unter http://www.antike-griechische.de/Euklid.pdf [Zugriff am 22.11.2015]

Ganter, 2007/2008 — Ganter, Bernhard: Rechnen modulo m, 2007/2008, Internetpublikation unter http://www.math.tu-dresden.de/~ganter/inf0708/folien/inf07-Modulo.pdf [Zugriff am 05.08.2015]

Kani, 1999 — Kani, Ernst: The GCD-Formula vs. the Euclidean Algorithm, 1999, Internetpublikation unter http://www.mast.queensu.ca/~math211/M211OH/M211OH20.pdf [Zugriff am 31.08.2015]

Lehrstuhl für Netz- und Datensicherheit Bochum, 2014	Lehrstuhl für Netz- und Datensicherheit Bochum: Euklidischer Algorithmus und Inversenberechnung, 2014, Internetpublikation unter https://www.nds.rub.de/media/attachments/files/2014/04/Euklidischer_Algorithmus_und_Inversenberechnung.pdf [Zugriff am 01.09.2015]
Mathe-Lexikon.at, kein Datum	Mathe-Lexikon.at: Natürliche Zahlen, kein Datum, Internetpublikation unter http://www.mathe-lexikon.at/mengenlehre/zahlenmengen/natuerliche-zahlen.html [Zugriff am 31.07.2015]
Meisel/ Mileski, 2006	Meisel, Andreas/ Mileski, Robert: Public-Key-Kryptografie mit dem RSA-Schema, 2006, Internetpublikation unter http://www.cs.uni-potsdam.de/ti/lehre/06-Kryptographie/slides/slides-06.pdf [Zugriff am 01.11.2015]
Rouse, 2014	Rouse, Margaret: What is Pretty Good Privacy (PGP)?, 2014, Internetpublikation unter http://searchsecurity.techtarget.com/definition/Pretty-Good-Privacy [Zugriff am 31.10.2015]
Schwan, 2002	Schwan, Matthias: RSA-Verschlüsselung, 2002, Internetpublikation unter http://www.math.uni-hamburg.de/home/werner/Kryptographie.pdf [Zugriff am 01.11.2015]
Universität Erfurt, kein Datum	Universität Erfurt: Division mit Rest, kein Datum, Internetpublikation unter http://www.uni-erfurt.de/fileadmin/user-docs/Mathematik/BA_Arithmetik_Vorlesung/veranstaltung13.pdf [Zugriff am 30.07.2015]
Wikipedia, 2015	Wikipedia, die freie Enzyklopädie: Größtes und kleinstes Element, 2015, Internetpublikation unter https://de.wikipedia.org/w/index.php?title=Gr%C3%B6%C3%9Ftes_und_kleinstes_Element&oldid=141626144 [Zugriff am 25.08.2015]

Selbständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Seminararbeit selbständig und nur unter Zuhilfenahme der angegebenen Hilfsmittel angefertigt habe.

Neumarkt, den _____

_____

(Unterschrift des Schülers)