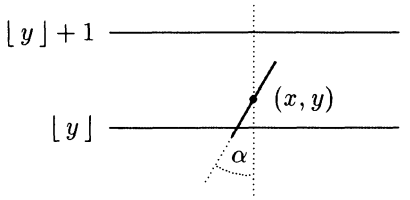
wo $\alpha \in [-\pi/2, \pi/2[$ der Winkel ist, den die Richtung der Nadel mit der Richtung der senkrechten Koordinatenachse einschließt; dabei trifft die Nadel die erste unter (x, y) liegende Parallele, genau wenn $d \leq (a/2) \cdot \cos\alpha$ ist, und die erste über (x, y) liegende Parallele, genau wenn $1 - d \leq (a/2) \cdot \cos\alpha$ ist. Die Würfe lassen sich also durch Paare (d, α) repräsentieren, wobei d in $[0, 1[$ und α in $[-\pi/2, \pi/2[$ unabhängig voneinander zufällig gewählt sind. Wie man mittels einer einfachen geometrischen Überlegung zeigt, trifft die Nadel bei einem Wurf eine der Parallelen mit der Wahrscheinlichkeit $2a/\pi$.

(a) Man simuliere mittels Zufallszahlen das Buffonsche Nadelwerfen und gewinne so Schätzwerte für die Zahl π.

(b) Man lese in dem Buch [78] von J. Pfanzagl den Abschnitt 5.1, in dem das Buffonsche Nadelproblem ausführlich behandelt wird.



Aufgabe 5: Man informiere sich in Knuth [55], Abschnitt 3.3.4, über den Spektraltest und programmiere ihn in MuPAD. Man wende diesen Test auf die in (8.12), in (8.13) und in Aufgabe 2 angegebenen L-Folgen an.

9 Ein wenig Kryptologie

(9.1) Der Wunsch, eine Information so zu verschlüsseln, daß sie nur von denen gelesen und verstanden werden kann, die dazu berechtigt sind, ist wohl uralt. Wie der römische Schriftsteller Sueton schreibt, verschlüsselte schon Gaius Julius Caesar Briefe, indem er für einen jeden Buchstaben einen anderen schrieb, etwa A statt D, B statt E und so fort (vgl. [107], Divus Julius 56). Von diesen und anderen in der Antike verwendeten Methoden, militärische und politische Informationen vor den Augen Unbefugter zu verbergen, erzählt auch Aulus Gellius in [41], XVII, 9. Man sieht daran, daß die Kryptologie, also die Lehre vom Verschlüsseln und Entschlüsseln von Informationen, schon in der Antike in militärischen und in politischen Belangen wichtig war. Heute ist die Kryptologie auch im alltäglichen Leben von großer Bedeutung; kein Mensch möchte, daß seine e-mail oder die vielen Informationen, die in allerlei Datenbanken über sein Leben, seine Finanzverhältnisse, seine Krankheiten stehen, von Leuten gelesen werden, die dazu nicht berechtigt sind.

Von den vielen Methoden der Kryptologie soll hier nicht die Rede sein; vielleicht darf an Stelle von Beispielen auf einige berühmte Kriminalgeschichten hingewiesen werden: Arthur Conan Doyle in [27] und [28] und Dorothy Sayers

in [98] beschreiben jeweils eine klassische kryptographische Methode. In den folgenden Abschnitten werden nur einige neuere Verfahren behandelt, die sich mit Hilfe der Zahlentheorie beschreiben und diskutieren lassen.

(9.2) Die klassischen Verfahren der Kryptologie sind, wie man sagt, symmetrische Verfahren: Sender und Empfänger einer Nachricht haben einen Schlüssel verabredet, der sowohl bei der Chiffrierung des Klartexts wie auch bei der Dechiffrierung des Geheimtexts verwendet wird. Dieser Schlüssel muß geheimgehalten werden, denn die Kenntnis des Schlüssels ermöglicht die Entschlüsselung. Auch der Data Encryption Standard (DES), der 1977 in den USA für "unclassified computer data" eingeführt wurde, ist ein symmetrisches Verfahren. Die Schwierigkeit beim Einsatz solcher Verfahren war immer, daß zwei Partner vor dem ersten Austausch einer verschlüsselten Nachricht einen geheimen Schlüssel verabreden müssen, was nicht möglich ist, wenn sie nur über ein öffentliches Kommunikationssystem, wie eine Telephonleitung oder ein Computernetz, miteinander Verbindung haben. Einen Ausweg aus einer solchen Situation bietet ein Verfahren, das W. Diffie und M. E. Hellman 1976 in der auch heute noch lesenswerten Arbeit [26] beschrieben haben. Dieses Verfahren macht es möglich, daß zwei Partner, die ein symmetrisches kryptologisches Verfahren, wie zum Beispiel DES, einsetzen wollen, gewissermaßen in aller Öffentlichkeit einen gemeinsamen Schlüssel für dieses Verfahren verabreden.

(9.3) Schlüssel-Austausch nach Diffie und Hellman: Zwei Benutzer A und B eines öffentlichen Kommunikationssystems möchten über dieses System einen Schlüssel für ein symmetrisches kryptographisches Verfahren verabreden. Sie einigen sich auf eine große Primzahl p und eine Primitivwurzel $g \in \mathbb{Z}$ modulo p. A wählt eine Zufallszahl $k_A \in \{0, 1, \ldots, p-2\}$, die er geheimhält, und teilt B die Zahl $m_A := g^{k_A} \bmod p$ mit; B wählt eine Zufallszahl $k_B \in \{0, 1, \ldots, p-2\}$, die er geheimhält, und teilt A die Zahl $m_B := g^{k_B} \bmod p$ mit. A berechnet $s := m_B^{k_A} \bmod p = g^{k_A k_B} \bmod p$, und B berechnet $m_A^{k_B} \bmod p = g^{k_A k_B} \bmod p = s$. Diese Zahl s wird nun von A und von B als gemeinsamer Schlüssel für das verabredete symmetrische Verfahren verwendet.

Die Sicherheit des Systems ist gewährleistet, solange nicht ein Dritter aus der Kenntnis von p, g und m_A und m_B eine der Zahlen k_A und k_B berechnen kann. Könnte man für jedes $a \in \{1, 2, \ldots, p-1\}$ schnell den Index $\mathrm{ind}(a)$ von a zur Primzahl p und zur Primitivwurzel g berechnen (vgl. (5.7)), so wäre das Verfahren unbrauchbar: Es ist $k_A = \mathrm{ind}(m_A)$ und $k_B = \mathrm{ind}(m_B)$.

(9.4) Neben die klassischen, symmetrischen Verfahren der Kryptologie sind in den letzten zwanzig Jahren verschiedene Verfahren getreten, bei denen die Methode und der Schlüssel zur Chiffrierung von Nachrichten an einen Empfänger A öffentlich bekannt sind, nicht aber der Schlüssel, den A zur Dechiffrierung

verwendet, und bei denen zwischen Sender und Empfänger einer Nachricht keine geheime Absprache erforderlich ist. Diese Verfahren werden als asymmetrisch bezeichnet oder, weil bei ihnen die (zur Chiffrierung verwendeten) Schlüssel öffentlich bekannt sind, als public-key-Verfahren. Diese Verfahren sind für die Kommunikation in Rechnernetzen besonders geeignet und wurden daher ausführlich untersucht.

(9.5) Hilfssatz: *Es seien p und q verschiedene Primzahlen; es sei $m := pq$, und es sei r eine natürliche Zahl mit $r \equiv 1 \pmod{\varphi(m)}$. Für jede ganze Zahl a ist $a^r \equiv a \pmod{m}$.*

Beweis: Es sei a eine ganze Zahl.
(a) Es gibt ein $k \in \mathbb{N}_0$ mit $r = 1 + k\varphi(m) = 1 + k\varphi(pq) = 1 + k\varphi(p)\varphi(q) = 1 + k(p-1)(q-1)$. Gilt $p \nmid a$, so gilt $a^{p-1} \equiv 1 \pmod{p}$ (vgl. (4.21)), und es folgt $a^r = a \cdot (a^{p-1})^{k(q-1)} \equiv a \pmod{p}$. Gilt $p \mid a$, so gilt $a^r \equiv 0 \equiv a \pmod{p}$.
(b) Nach (a) gilt $a^r \equiv a \pmod{p}$, und ebenso ergibt sich $a^r \equiv a \pmod{q}$. Da p und q teilerfremd sind, folgt $a^r \equiv a \pmod{m}$.

(9.6) Das public-key-Verfahren von Rivest, Shamir und Adleman: Die Benutzer eines öffentlichen Kommunikationssystems wollen über dieses System verschlüsselte Botschaften austauschen. Für den Klartext wird ein Alphabet mit N Zeichen (Buchstaben, Ziffern und Sonderzeichen) verwendet. Diese Zeichen werden numeriert: $b_0, b_1, \ldots, b_{N-1}$, und diese Reihenfolge wird stets beibehalten. Es werden natürliche Zahlen k und l mit $k < l$ gewählt, für die N^k und N^l 300 bis 400 Dezimalstellen besitzen. Das Alphabet, die Reihenfolge der Zeichen und die Zahlen k und l werden (im "Telephonbuch" des Systems) veröffentlicht.
(1) Jeder Benutzer A des Kommunikationssystems wählt zwei verschiedene Primzahlen p_A und q_A, jede mit 150 bis 200 Dezimalstellen und so, daß für $m_A := p_A q_A$ gilt: Es ist $N^k < m_A < N^l$. Er berechnet $\varphi(m_A) = \varphi(p_A)\varphi(q_A) = (p_A - 1)(q_A - 1)$, wählt ein $d_A \in \{1, 2, \ldots, \varphi(m_A) - 1\}$ mit $\mathrm{ggT}(d_A, \varphi(m_A)) = 1$ und berechnet die Zahl $e_A \in \{1, 2, \ldots, \varphi(m_A) - 1\}$ mit $d_A e_A \equiv 1 \pmod{\varphi(m)}$ (vgl. (4.6)). Die Zahlen m_A und e_A werden im "Telephonbuch" veröffentlicht, die Zahlen p_A, q_A und d_A werden von A geheimgehalten. Bei der Wahl von p_A, q_A und d_A ist ein Zufallszahlengenerator zu verwenden: Um eine geeignete Primzahl p_A zu finden, wähle man eine hinreichend große Zufallszahl x und suche dann – mit Hilfe eines Primzahltests – die kleinste Primzahl $p \geq x$. (Wenn man dabei einen stochastischen Primzahltest wie den von Rabin verwendet, so muß man darauf gefaßt sein, daß die gelieferte Zahl p unter Umständen keine Primzahl ist).
(2) Ein Benutzer B des Kommunikationssystems möchte an A eine Botschaft schicken. Er teilt den Klartext in Blöcke aus je k Zeichen ein, wobei er am Ende

der Nachricht eventuell noch einige ergänzende Zeichen anfügt, und ersetzt jedes Zeichen durch sein numerisches Äquivalent, d.h. er ersetzt für jedes $i \in \{0,1,\ldots,N-1\}$ das Zeichen b_i durch seinen Index i. So entstehen k-tupel aus Zahlen in $\{0,1,\ldots,N-1\}$; jedes solche k-tupel wird nun für sich verschlüsselt.
(a) Die Verschlüsselung: Es sei $(\alpha_0,\alpha_1,\ldots,\alpha_{k-1})$ ein k-tupel aus Zahlen in $\{0,1,\ldots,N-1\}$. B entnimmt dem "Telephonbuch" die unter dem Eintrag von A stehenden Zahlen m_A und e_A und berechnet

$$\begin{aligned} x &:= \sum_{j=0}^{k-1} \alpha_j N^j \quad \in \{0,1,\ldots,N^k-1\} \subset \{0,1,\ldots,m_A-1\}, \\ x^* &:= x^{e_A} \bmod m_A \in \{0,1,\ldots,m_A-1\} \subset \{0,1,\ldots,N^l-1\} \end{aligned}$$

und schließlich die Zahlen $\beta_0, \beta_1,\ldots,\beta_{l-1} \in \{0,1,\ldots,N-1\}$ mit

$$x^* = \sum_{j=0}^{l-1} \beta_j N^j.$$

Dann schickt B das l-tupel $(b_{\beta_0}, b_{\beta_1},\ldots,b_{\beta_{l-1}})$ über das öffentliche Kommunikationssytem an A.
(b) Die Entschlüsselung: A gewinnt aus $(b_{\beta_0}, b_{\beta_1},\ldots,b_{\beta_{l-1}})$ zuerst die Zahlen $\beta_0, \beta_1,\ldots,\beta_{l-1}$ und berechnet wieder

$$x^* = \sum_{j=0}^{l-1} \beta_j N^j$$

und daraus

$$x^{**} := (x^*)^{d_A} \bmod m_A.$$

Nach (9.5) gilt $x^{**} \equiv (x^*)^{d_A} \equiv x^{d_A e_A} \equiv x \pmod{m_A}$, und daher ist $x^{**} = x$. A berechnet die Zahlen $\alpha_0, \alpha_1,\ldots, \alpha_{k-1} \in \{0,1,\ldots,N-1\}$ mit

$$x = x^{**} = \sum_{j=0}^{k-1} \alpha_j N^j$$

und hat damit das k-tupel $(\alpha_0,\alpha_1,\ldots,\alpha_{k-1})$ gewonnen, aus dem er den zugehörigen Block $b_{\alpha_0}, b_{\alpha_1},\ldots, b_{\alpha_{k-1}}$ des Klartexts herstellen kann.
(c) B kann mit seiner Nachricht an A eine "Unterschrift" mitschicken, die beweist, daß die Nachricht von ihm kommt. Er codiert seinen Namen, eine Angabe über den Zeitpunkt des Abschickens und eventuell andere Angaben, die sich aus dem Klartext seiner Nachricht berechnen lassen, durch ein k-tupel

$(\alpha_0, \alpha_1, \ldots, \alpha_{k-1}) \in \{0, 1, \ldots, N-1\}^k$ (oder eventuell durch mehrere solche k-tupel) auf dieselbe Weise, wie er das für den Klartext selbst getan hat, und berechnet damit zuerst

$$y := \sum_{j=0}^{k-1} \alpha_j N^j \in \{0, 1, \ldots, N^k - 1\}$$

und dann mit den von ihm gewählten Zahlen m_B und d_B

$$y^* := \begin{cases} \left(y^{d_B} \bmod m_B\right)^{e_A} \bmod m_A, & \text{falls } m_B < m_A \text{ ist,} \\ \left(y^{e_A} \bmod m_A\right)^{d_B} \bmod m_B, & \text{falls } m_A < m_B \text{ ist,} \end{cases}$$

codiert y^* wie auch bei der Verschlüsselung des Klartextes durch ein l-tupel $(\beta_0, \beta_1, \ldots, \beta_{l-1}) \in \{0, 1, \ldots, N-1\}^l$ und schickt die so gewonnene Zeichenkette $(b_{\beta_0}, b_{\beta_1}, \ldots, b_{\beta_{l-1}})$ mit seiner Nachricht an A. A entnimmt dem "Telephonbuch" die unter dem Eintrag von B stehenden Zahlen m_B und e_B und berechnet zuerst

$$y^* = \sum_{j=0}^{l-1} \beta_j N^j.$$

Ist $m_B < m_A$, so berechnet er dann

$$y^{**} := (y^*)^{d_A} \bmod m_A \quad \text{und} \quad y^{***} := (y^{**})^{e_B} \bmod m_B,$$

und wie in (b) folgt mit Hilfe von (9.5)

$$y^{**} = y^{d_B} \bmod m_B \quad \text{und} \quad y^{***} = y.$$

Ist aber $m_A < m_B$, so berechnet er

$$y^{**} := (y^*)^{e_B} \bmod m_B \quad \text{und} \quad y^{***} := (y^{**})^{d_A} \bmod m_A,$$

und wie in (b) folgt mit Hilfe von (9.5)

$$y^{**} = y^{e_A} \bmod m_A \quad \text{und} \quad y^{***} = y.$$

Aus y gewinnt A das k-tupel $(\alpha_0, \alpha_1, \ldots, \alpha_{k-1})$ und daraus schließlich die "Unterschrift" von B.

(3) Das Verfahren kann nicht mehr verwendet werden, wenn ein Benutzer C einen schnellen Faktorisierungsalgorithmus für große natürliche Zahlen kennt. Wenn C aus m_A die Primzahlen p_A und q_A berechnen kann, so kann er $\varphi(m_A) = (p_A - 1)(q_A - 1)$ berechnen und damit aus der öffentlich bekannten Zahl e_A die Zahl d_A, mit deren Hilfe er jede an A gerichtete Botschaft entschlüsseln kann.

Es ist nicht bekannt, ob es einen effizienten Algorithmus zur Entschlüsselung gibt, der ohne die Kenntnis der Primfaktoren von m_A auskommt.
(4) Die Primzahlen p_A und q_A, sowie die Zahl d_A muß A mit Sorgfalt wählen. Einer genaueren Diskussion des Verfahrens (vgl. etwa Bauer [11]) entnimmt man, daß $|p_A - q_A|$ nicht zu klein sein sollte und daß für p_A und q_A "sichere" Primzahlen zu wählen sind; eine Primzahl p heißt sicher, wenn auch $(p-1)/2$ eine Primzahl ist. Auch d_A und e_A dürfen nicht zu klein sein.

(9.7) Bemerkung: Das in (9.6) beschriebene Verschlüsselungsverfahren haben R. L. Rivest, A. Shamir und L. M. Adleman im Jahr 1978 in [91] angegeben; man findet es in der Literatur unter dem Namen "RSA-Verfahren". Das RSA-Verfahren ist für die Verschlüsselung großer Datenmengen zu langsam. Es ist aber gut geeignet, Schlüssel für ein symmetrisches Verfahren wie DES zwischen den Benutzern eines öffentlichen Kommunikationsnetzes zu tauschen, und wird bereits in verschiedenen Programmen, etwa in PGP (Pretty Good Privacy) oder in ssh (Secure Shell), die zur Verschlüsselung von e-mail und anderen über ein Computernetz verschickten Daten und zur Authentifikation in einem Computernetz dienen, dazu verwendet.

Im nächsten Abschnitt wird ein public-key-Verfahren beschrieben, das 1985 von T. ElGamal in [31] angegeben wurde.

(9.8) Das public-key-Verfahren von ElGamal: Die Benutzer eines öffentlichen Kommunikationssystems wollen über dieses System verschlüsselte Botschaften austauschen. Sie einigen sich auf eine große Primzahl p und auf eine Primitivwurzel g modulo p. Als Klartext werden Zahlen $x \in \{1, 2, \ldots, p-1\}$ verwendet. Jeder Benutzer A wählt eine Zufallszahl $d_A \in \{0, 1, \ldots, p-2\}$, die er geheimhält, berechnet $e_A := g^{d_A} \bmod p$ und veröffentlicht e_A im "Telephonbuch" des Systems.
(1) (a) Die Verschlüsselung: Ein Benutzer B will an einen Benutzer A eine Nachricht $x \in \{1, 2, \ldots, p-1\}$ schicken. Er entnimmt dem "Telephonbuch" des Systems die unter dem Eintrag von A stehende Zahl e_A, wählt eine Zufallszahl $k \in \{0, 1, \ldots, p-2\}$, berechnet $y := g^k \bmod p$ und $z := (x \cdot e_A^k) \bmod p$ und schickt das Paar (y, z) an A.
(b) Die Entschlüsselung: Wegen $p \nmid g$ und $y \equiv g^k \pmod p$ gilt $p \nmid y$, also kann A mit Hilfe des erweiterten Euklidischen Algorithmus aus (1.18) ein $v \in \mathbb{Z}$ mit $yv \equiv 1 \pmod p$ berechnen. Es gilt

$$zv^{d_A} \equiv xe_A^k v^{d_A} \equiv xg^{d_A k} v^{d_A} = x(g^k v)^{d_A} \equiv x(yv)^{d_A} \equiv x \pmod p,$$

und daher ist $x = (zv^{d_A}) \bmod p$.
(c) B kann einer Nachricht $x \in \{1, 2, \ldots, p-1\}$ an A eine Unterschrift mitgeben, die nur von jemandem stammen kann, der seinen geheimen Schlüssel d_B

kennt. Er wählt eine Zufallszahl $k \in \{0, 1, \ldots, p-2\}$ mit $\mathrm{ggT}(k, p-1) = 1$ und berechnet damit $r := g^k \bmod p$ und die eindeutig bestimmte Zahl $s \in \{0, 1, \ldots, p-2\}$ mit

$$ks \equiv x - rd_B \pmod{(p-1)}$$

(vgl. (4.9)(2)). Hierfür gilt

$$r^s e_B^r \equiv g^{ks} g^{rd_B} = g^{ks+rd_B} \equiv g^x \pmod{p}.$$

B schickt das Paar (r, s) zusammen mit der Verschlüsselung von x an A. A berechnet zuerst den Klartext x, entnimmt dem "Telephonbuch" die unter dem Eintrag von B stehende Zahl e_B und überprüft dann, ob

$$g^x \bmod p = r^s e_B^r \bmod p$$

gilt.

(3) Ein Benutzer C, der einen schnellen Algorithmus zur Berechnung von Indizes kennt und eine an A gerichtete chiffrierte Nachricht abgefangen hat, berechnet aus der öffentlich bekannten Zahl e_A den Index $\mathrm{ind}(e_A) = \mathrm{ind}(g^{d_A}) = d_A$ von e_A zur Primzahl p und zur Primitivwurzel g und kann die Nachricht an A ebenso dechiffrieren wie A selbst. Offensichtlich kann C auch Unterschriften fälschen. Das public-key-Verfahren von ElGamal kann also nicht mehr verwendet werden, sobald ein schneller Algorithmus zur Berechnung von Indizes gefunden ist.

(9.9) Bemerkung: Die Literatur zur Kryptologie ist recht umfangreich. In Koblitz [57] findet man weitere Verschlüsselungsverfahren, die man mit Hilfe der Zahlentheorie beschreiben und diskutieren kann, eine allgemeinere Einführung in die Kryptologie ist das Buch [11] von F. L. Bauer, und eine umfassende Darstellung der heute wichtigen kryptologischen Methoden und der dazu nötigen Hilfsmittel aus Algebra und Zahlentheorie ist das Handbuch [69] von A. J. Menezes, P. C. van Oorschot und S. A. Vanstone. Eine sehr ausführliche Darstellung der Geschichte der Kryptologie bietet das Buch [51] von D. Kahn; darin werden viele klassische kryptologische Verfahren beschrieben.

(9.10) Aufgaben:

Aufgabe 1: Man schreibe MuPAD-Funktionen zum Verschlüsseln und Entschlüsseln nach dem RSA-Verfahren. Wie man dabei einen Klartext in eine Folge von Zahlen verwandelt, ist nicht weiter wichtig. Benötigt man nur Kleinbuchstaben und als Sonderzeichen nur einen Worttrenner, so kann man etwa diesen durch 0, a durch 1, b durch 2 und schließlich z durch 26 ersetzen. Man kann aber auch jedes Zeichen durch seinen ASCII-Code ersetzen, also

durch eine ganze Zahl zwischen 0 und 127. (Dazu kann man sich die Definition der MuPAD-Funktionen `numlib::toAscii` und `numlib::fromAscii` ansehen, indem man entweder die Textfiles ansieht, in denen diese Funktionen erklärt sind, oder indem man innerhalb einer MuPAD-Sitzung die Anweisungen `numlib::fromAscii;` und `numlib::toAscii;` eingibt).

Aufgabe 2: Man schreibe eine MuPAD-Funktion, die zu einer natürlichen Zahl a die kleinste sichere Primzahl $\geq a$ berechnet (vgl. (9.7)(4)).

Aufgabe 3: Man schreibe MuPAD-Funktionen zum Verschlüsseln und Entschlüsseln nach dem Verfahren von ElGamal.

Aufgabe 4: Zum public-key-Verfahren von ElGamal: Ein Benutzer schickt an den Benutzer A zwei Nachrichten x_1, $x_2 \in \{1, 2, \ldots, p-1\}$ und verschlüsselt beide wie in (9.8)(1) beschrieben, wobei er bei beiden dieselbe Zufallszahl $k \in \{0, 1, \ldots, p-2\}$ verwendet. Man überlege sich, daß dann ein unbefugter Dritter, der die beiden verschlüsselten Nachrichten abgefangen hat, x_2 ohne Schwierigkeit berechnen kann, falls er x_1 kennt.

Aufgabe 5: Das folgende kryptographische Verfahren wurde im Jahr 1985 von J. L. Massey und J. K. Omura angegeben:
Die Benutzer eines öffentlichen Kommunikationssystems verabreden eine große Primzahl p. Jeder Teilnehmer A wählt Zahlen d_A, $e_A \in \{1, 2, \ldots, p-1\}$ mit $d_A e_A \equiv 1 \pmod{p}$, die er beide geheimhält. Als Klartext werden Zahlen aus $\{1, 2, \ldots, p-1\}$ verwendet.
Ein Benutzer B, der an den Benutzer A eine Nachricht $x \in \{1, 2, \ldots, p-1\}$ schicken will, berechnet die Zahl $x^* := x^{d_B} \bmod p$ und sendet sie an A, A berechnet $y := (x^*)^{d_A} \bmod p$ und schickt y an B, B berechnet die Zahl $y^* := (y^{e_B}) \bmod p$ und schickt sie an A. Dann berechnet A die Zahl $(y^*)^{e_A} \bmod p$ und hat damit den Klartext x gewonnen.

(a) Worauf beruht die Sicherheit dieses Verfahrens? Warum erhält A am Ende den Klartext x?
(b) Dieses Verfahren erfordert eine Authentifikation. Wie kann ein dritter Benutzer C, der sich in die Kommunikation zwischen A und B einschalten kann, den Klartext einer von B an A geschickten Nachricht gewinnen?
(c) Man überlege sich, daß dieses Verfahren nicht mehr verwendet werden darf, wenn ein Benutzer einen schnellen Algorithmus zur Berechnung von Indizes kennt.