1. Modulare Arithmetik

Dreizehn Jahre lang hatten die Briten und Franzosen geglaubt, die Enigma-Verschlüsselung sei nicht zu knacken, doch nun schöpften sie Hoffnung. Die polnischen Erfolge hatten bewiesen, daß die Enigma angreifbar war, und dies stärkte die Moral der alliierten Kryptoanalytiker.

[aus Simon Singh: "The Codebook"]

Wozu braucht man modulare Arithmetik? – Nicht erst mit dem Internet wächst der Bedarf, sensible Informationen verschlüsselt zu übertragen. Aber dazu müssen Sender und Empfänger erstmal einen Schlüssel übertragen. Was, wenn der im Klartext übertragene Schlüssel belauscht wird?

Lange Jahrhunderte war das die große Schwäche der Kryptographie, denn fast alle früheren Verschlüsselungsverfahren (z.B. Cäsar's Alphabet-Verschieber) waren so, dass das Wissen zum Verschlüsseln einer Nachricht auch zum Entschlüsseln gut war.

Die modulare Arithmetik verbringt ein erstaunliches Kunststück: Mit dem **RSA-Verfahren** gibt es ein sog. **asymmetrisches Verschlüsselungsverfahren**. Den Schlüssel zum Kodieren einer Nachricht darf jeder lesen (**public key**), aber niemand kann die Nachricht entschlüsseln, ausser dem Empfänger, der einen speziellen Schlüssel (**private key**) besitzt.

Analogie Vorhängeschloß (oder auch Tresor): Jeder kann ein offenes Vorhängeschloß schließen, aber nur der Schlüsselbesitzer bekommt es wieder auf.

Nur wollen und können wir ja keine materiellen Dinge wie Vorhängeschlösser (oder gar tonnenschwere Tresore) über das Internet verschicken und das Erstaunliche ist, dass diese Form der Undurchdringbarkeit eben auch mit – immateriellen – Zahlen geht.

Wir werden in diesem Kapitel die mathematischen Grundlagen entwickeln, um RSA zu verstehen. Auf diesem Wege werden wir auch folgende Fragen beantworten:

- Wieviele Primzahlen gibt es?

- Woran erkennt man, ob eine Zahl durch 9 teilbar ist?
- Wie funktioniert modulare Arithmetik?
- Wie berechnet man für große ganze Zahlen (1000 Stellen oder mehr) effizient den ggT (größten gemeinsamen Teiler)? – Antwort: Algorithmus von Euklid.
- Wie rechnet man überhaupt mit sehr großen Zahlen? – Antwort: Chinesischer Restsatz.

1.1. Primzahlen

Def D1-1: Für a, b $\in \mathbf{Z}$ gilt:

- **Teilbarkeit**: Man sagt, b **teilt** a, wenn es eine ganze Zahl x $\in \mathbf{Z}$ gibt mit a=xb.
- Eine Zahl p $\in \mathbf{N}$ mit p≥2, heißt **Primzahl**, wenn es keine Zahl s $\in$ {2,3,...,p-1} gibt, die p teilt.
- Der größte gemeinsame Teiler, **ggT(a,b)**, ist die größte Zahl g, für die gilt:
  g teilt a UND g teilt b.
- Das kleinste gemeinsame Vielfache, **kgV(a,b)**, ist die kleinste Zahl K $\in \mathbf{N}$, für die gilt:
  a teilt K UND b teilt K.
- Zwei Zahlen a und b heißen **teilerfremd**, wenn ggT(a,b)=1.

Englische Bezeichner (>> Maple):

ggT ≡ **gcd** (**g**reatest **c**ommon **d**ivisor)

kgV ≡ **lcm** (**l**east **c**ommon **m**ultiple)

Beispiele: ggT(32,80)=16, ggT(7,8)=1, also sind 7 und 8 teilerfremd.

ggT(-6,-20)=2, d.h. ggT und kgV sind auch für negative Zahlen definiert.

kgV(6,20)= 60.

Für "mittelgroße" Zahlen findet man ggT und kgV recht schnell über die Primzahlzerlegung:

45	=	3·3·5
60	=	2·2·3 ·5
ggT(45,60)	=	3 ·5 = 15

kgV(45,60) = 2·2·3·3·5 = 180

Für sehr große Zahlen N ist aber die Primzahlzerlegung ein (höchstwahrscheinlich) nicht polynomial lösbares Problem und daher sehr schwer. Für den ggT gibt es zum Glück den Algorithmus von Euklid, der nur die Laufzeit O(log N) hat.

Trotz der Einfachheit der Definitionen ergeben sich, besonders aus dem Begriff der Primzahl, eine Vielzahl von Fragen: Kann man einer Zahl "ansehen", ob sie eine Primzahl ist? Wieviele Primzahlen gibt es? Gibt es einen effizienten Algorithmus zur Bestimmung von Primzahlen?

Da diese Fragen schon die alten Griechen beschäftigten, gibt es zumindest einen der frühesten überlieferten Algorithmen, das **Sieb des Eratosthenes**:

> *Um die in 2,...,N enthaltenen Primzahlen zu finden, schreibe alle Zahlen 2,...,N auf, streiche die echten Vielfachen von 2, die echten Vielfachen der nächstgrößeren verbleibenden Zahl usw. bis man die Vielfachen von $\sqrt{N}$ erreicht. Die Zahlen, die übrig bleiben, sind Primzahlen.*

Beispiel für N=33: (**rot**: die Zahlen, deren Vielfache gerade gestrichen werden. Wegen 6^2=36 braucht man die 6 nicht mehr zu untersuchen)

2	3	~~4~~	5	~~6~~	7	~~8~~	9	~~10~~	11	~~12~~	13	~~14~~	15	~~16~~	17	~~18~~	19	~~20~~	21	~~22~~	23	~~24~~	25	~~26~~	27	~~28~~	29	~~30~~	31	~~32~~	33
2	**3**		5		7		~~9~~		11		13		~~15~~		17		19		~~21~~		23		25		~~27~~		29		31		~~33~~
2	3		**5**		7				11		13				17		19				23		~~25~~				29		31		
2	3		5		7				11		13				17		19				23						29		31		

Wieso reicht es, bis $\sqrt{N}$ zu gehen? – Sei P die erste ungestrichene Zahl > $\sqrt{N}$. Für 1<k<P kann das Vielfache k·P nicht in den ungestrichenen Zahlen sein, denn es wäre schon gestrichen worden, als k an der Reihe war. Ein Vielfaches k·P mit k≥P liegt aber sicher oberhalb von N, ist also uninteressant. ϒ

Satz S1-1 (Fundamentalsatz der Arithmetik): Jede natürliche Zahl n≥2 läßt sich eindeutig als Produkt von Primzahlen p_i schreiben:

$$n = p_1^{e_1} \cdot p_2^{e_2} \cdot \ldots \cdot p_k^{e_k} \quad \text{mit} \quad e_i \in \{1,2,...\}$$

Beweis evtl. später als Übungsaufgabe!

Beispiel: Um eine Zahl n möglichst schnell in kleinere Zahlen zu zerlegen, startet man zweckmäßigerweise bei der kleinsten Primzahl und probiert sukzessive die nächstgrößeren erst dann, wenn durch die kleineren nicht mehr teilbar:

300	=	2·150
	=	2·2·75
	=	2·2·3·25
	=	2·2·3·5·5
300	=	$2^2 \cdot 3 \cdot 5^2$

Allerdings wird für große n (100 Stellen und mehr) die Primzahlzerlegung sehr aufwendig, genauer gesagt, es ist bis heute kein Algorithmus bekannt, der es in polynomialer Zeit schaffen würde. Gerade weil sich dies nicht "knacken" läßt, beruht auf eben diesem Prinzip die Stärke vieler kryptographischer Verfahren, s.u.

Satz S1-2: Es gibt unendlich viele Primzahlen.

Der Beweis, der auf den großen griechischen Mathematiker Euklid von Alexandria (300 v. Chr.) zurückgeht, wird in der Vorlesung gebracht.

1.2. Modulare Arithmetik

Def D1-2: Zwei Zahlen a,b $\in \mathbf{Z}$ heißen **kongruent modulo m**, geschrieben

a = b (mod m)

genau dann, wenn a – b ein Vielfaches von m ist. D.h. es gibt ein t $\in \mathbf{Z}$: a – b = tm.

Sei $\mathbf{Z}_m$ = {0,1,...,m-1}. Der ganzzahlige **Rest r bei Division** von a durch m

r = a mod m

ist diejenige Zahl r$\in \mathbf{Z}_m$, für die a – r ein Vielfaches von m ist.

Die Zahl m heißt **Modul**.

Anmerkungen: (1) Diese beiden Definitionen schauen recht ähnlich aus, bezeichnen aber verschiedene Dinge. Die Beziehung a = b (mod m) definiert eine **Äquivalenzklasse**, die ∞ viele Zahlen a,b enthält. Dagegen definiert r = a mod m genau eine Zahl r, den Rest. Es gilt:

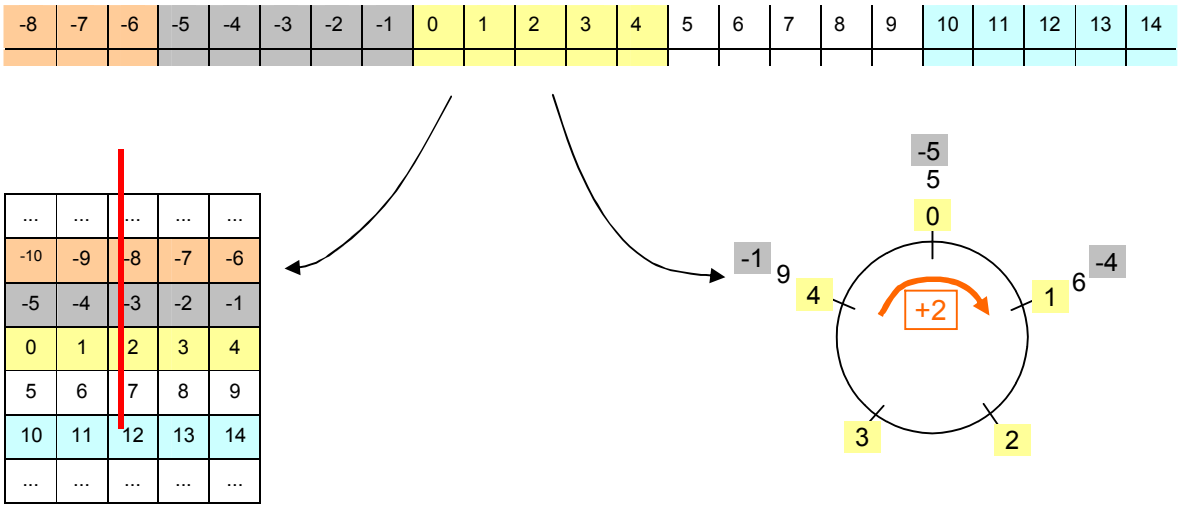
$r = a \bmod m \quad \Rightarrow \quad a = r \pmod m$,

aber die Umkehrung gilt i.A. nicht. Die Umkehrung gilt genau dann, wenn $r \in \mathbf{Z}_m$ ist.

(2) Aus $a \equiv b \pmod m$ folgt $b \equiv a \pmod m$. (klar aus Visualisierung, s.u.)

Visualisierung: Durch die mod-Operation wird die Zahlengerade in Stücke der Länge m zerschnitten. Stapeln wir diese Stücke übereinander so liegen alle Zahlen einer Äquivalenzklasse vertikal übereinander.

Abbildung 1: Zahlengerade, Beispiel für m=5:



Links: Alle Zahlen der Äquivalenzklasse 2 (mod 5) liegen übereinander (rote Linie). Offensichtlich gilt deshalb die Äquivalenz $2 \equiv 7 \pmod 5 \Leftrightarrow 7 \equiv 2 \pmod 5$.
Rechts: Ordnen wir die Zahlen auf einem Ring an, so sehen wir, dass die Addition 2 + 4 (mod 5) auf die Äquivalenzklasse 1 (mod 5) führt.

Satz S1-3 (Rechenregeln für modulare Arithmetik):

Seien $a,b,c,d \in \mathbf{Z}$ und $m \in \mathbf{N}$. Dann gilt:

Wenn $a = b \pmod m$ und $c = d \pmod m$

dann folgt:

$$a + c = b + d \pmod m$$
$$ac = bd \pmod m$$

Beweis: Es gibt $t,s \in \mathbf{Z}$: a-b=tm und c-d=sm $\Rightarrow$ a+c-(b+d) = (t+s)m $\Rightarrow$ a+c $\equiv$ b+d (mod m). Für die Multiplikation gehen wir von der Gleichung für c aus und multiplizieren mit a=b+tm durch:

c = d + sm | ·a

a·c = a·d + a·sm

a·c = (b+tm)·d + a·sm $\Rightarrow$

a·c = b·d (mod m). Υ

In der letzten Zeile haben wir davon Gebrauch gemacht, dass in der (mod m)-Welt alle Vielfachen von m, also tm und a·sm, gestrichen werden können.

Beispiele:

- Was ist 48 + 25 (mod 11)? – Wegen 48 = 44 + 4 = 4 (mod 11) und 25 = 22 + 3 = 3 (mod 11) ist 48 + 25 = 4+3 (mod 11) = 7 (mod 11).
- Es gilt 27 = 2 (mod 5) und 11 = 1 (mod 5). Also ist 27·11 = 2·1 (mod 5).
- Man beachte: "Kürzen" ist i.a. nicht erlaubt. D.h. aus a·c $\equiv$ b·c (mod m) folgt **NICHT** a $\equiv$ b (mod m). Denn es ist z.B. 2·4 $\equiv$ 5·4 (mod 6), denn 8 und 20 haben den gleichen Rest 2 (mod 6), aber sicherlich gilt NICHT 2 $\equiv$ 5 (mod 6).

1.2.1. Teilbarkeitsregeln

Woran erkennt man, ob eine Zahl M durch 3 teilbar ist? Ist 11736 durch 3 teilbar? – Den aus der Schule bekannten Satz: "Eine Zahl ist durch 3 teilbar, wenn ihre Quersumme durch 3 teilbar ist", können wir mit modularer Arithmetik wie folgt beweisen:

- Dezimaldarstellung: $M = \sum_{i=0}^{n} a_i 10^i = a_0 10^0 + ... + a_n 10^n$

	Rest bei Division durch 3	M =	M (mod m)

10^0	1	$a_0 \cdot 10^0 +$	$a_0 \cdot 1 +$
10^1	1	$a_1 \cdot 10^1 +$	$a_1 \cdot 1 +$
10^2	1	$a_2 \cdot 10^2 +$	$a_2 \cdot 1 +$
…	…	…	…
10^n	1	$a_n \cdot 10^n$	$a_n \cdot 1$
			= Quersumme

- In der letzten Spalte haben wir 10^0, 10^1, 10^2,… jeweils durch einen einfacheren Vertreter aus der (mod 3)-Welt ersetzt, in diesem Fall immer durch 1.
- Die Zahl M hat also den gleichen Rest bei Division durch 3 wie ihre Quersumme.

Beispiel:

- 11736 hat die Quersumme 1+1+7+3+6=18, ist also durch 3 teilbar. Es gilt 11736 = 3·3912.
- Allgemeiner: Wir können aus der Quersumme direkt den Rest ablesen, den eine Zahl bei Division durch 3 hat: 11740 hat Quersumme 1+1+7+4 = 13 = 1 (mod 3), also ist 11740 mod 3 = 1.

Übung: Finden und beweisen Sie nach ähnlichem Muster eine 9er-Teilbarkeitsregel. Welchen Rest hat M=1234567 bei Division durch 9?

Übung(+): Finden und beweisen Sie nach ähnlichem Muster eine 7er-Teilbarkeitsregel. Welchen Rest hat M=1234567 bei Division durch 7?

Der Satz S1-3 hat eine weitere interessante Folgerung:

$$a^{x-1} \equiv 1 \pmod{m} \Rightarrow a^x \equiv a \pmod{m}$$

(einfach beide Seiten mit a durchmultiplizieren). In Worten: *Wenn* es einen Exponenten (x-1) gibt, der auf a angewendet die Äquivalenzklasse 1 (mod m) liefert, dann reproduziert der Exponent x wieder a selbst. *Wenn* x = k·g zusätzlich noch ein Produkt ist, dann kann ein Sender eine Zahl (Nachricht) a durch $b=a^k$ verschlüsseln und später ein Empfänger aus b durch $a=b^g$ wieder a zurückgewinnen. Genau dies ist die Kernidee für RSA, welches ein Rezept liefert, wie man ein solches x = k·g bekommt.

Eine weitere Folgerung ergibt sich für das Rechnen mit Rest

Folgerung S1-4 (Rechnen mit Rest):

Seien a,b $\in \mathbf{Z}$ und k,m $\in \mathbf{N}$. Dann gilt: Beim (mod m)-Rechnen mit den Zahlen a,b als Summand, Faktor oder Basis einer Potenz ist ein zusätzliches "mod m" für beliebige Zwischenergebnisse eine Äquivalenzumformung:

$$(a + b) = [(a \bmod m) + (b \bmod m)] \pmod m$$

$$(a \cdot b) = [(a \bmod m) \cdot (b \bmod m)] \pmod m$$

$$a^k = (a \bmod m)^k \pmod m$$

Beachte: Es handelt sich nicht um das bloße Vertauschen von mod mit dem anderen Operator, denn am Ende muss immer noch ein "mod m" stehenbleiben. Ein "mod m" zuviel ist nie schädlich, da "mod m" auf $\mathbf{Z}_m$ eine Identitätsoperation ist.

Beweis: Wir zeigen stellvertretend die Identität für das Produkt, die Summengleichung ergibt sich analog. Sei r_a=a mod m und r_b=b mod m. Es gibt dann Zahlen s,t $\in \mathbf{Z}$, so dass gilt:

$$(a \cdot b) \bmod m = (r_a+sm)(r_b+tm) \bmod m = [r_a r_b+(s+t+stm)m] \bmod m = r_a r_b \bmod m.$$

Die Potenzregel wird auf die k-fache Produktregel zurückgeführt, $a^k = a \cdot a \cdot \ldots \cdot a$. ϒ

Beispiel: $223^5 \bmod 3 = 551473077343 \bmod 3 = 1$, wobei das 12-stellige Zwischenergebnis oft in (Taschen-) Rechnern nicht mehr exakt darstellbar sein wird. Viel einfacher geht es mit Folgerung S1-4, denn $223^5 \bmod 3 = (223 \bmod 3)^5 \bmod 3 = 1^5 \bmod 3 = 1$.

1.2.2. Der Chinesische Restsatz (CRT)

Aktivierung Kartenspieltrick:

> Der „Magier“ bittet einen Zuschauer, sich 1 aus 20 Karten zu merken. Er teil die 20 Karten in 5 Stapeln vor sich aus, der Zuschauer sagt, welcher Stapel „seine“ Karte enthält. Dies wiederholt sich mit 4 Stapeln. Nachdem der Zuschauer wiederum „seinen“ Stapel benannt hat, sagt der Magier dem Zuschauer sofort auf den Kopf zu, welche Karte es war.

Wie kann der Magier das so schnell wissen / rechnen?

Dies ist ein Beispiel für simultane Kongruenzen, die mit dem Chinesischen Restsatz gelöst werden können. Zwei Zahlen heißen nach Def D1-1 dann kongruent, wenn sie den gleichen Rest modulo m haben. Für viele Anwendungen muss man **simultane Kongruenzen** (oder

Systeme von Kongruenzen) bestimmen, d.h. eine Zahl x soll zu verschiedenen Modulen gleichzeitig bestimmte Kongruenzen erfüllen. Neben dem Kartenspieltrick braucht man dies für

- Astronomie: Wann stehen mehrere Planeten in Konjunktion?
- Multiplikation großer Zahlen: Aufwand von $O(n^2)$ auf $O(n)$ reduzierbar (!, s. Kap. 3.1)

Weil die erste Erwähnung des Probleme und erste Speziallösungen auf die Chinesen zurückgehen:

> Im 1. Jahrhundert v. Chr. stellte der chinesische Mathematiker Sun-Tsu folgendes Rätsel: „Ich kenne eine Zahl. Wenn man sie durch 3 dividiert, bleibt der Rest 2, wenn man sie durch 5 dividiert, bleibt der Rest 3, wenn man sie durch 7 dividiert, bleibt der Rest 2. Wie lautet diese Zahl?“

nennt man den zugehörigen Satz den *Chinesischen* Restsatz:

Satz S1-5 Chinesischer Restsatz (CRT)

Sind $m_1, \ldots, m_n$ paarweise *teilerfremde* Module, dann haben die simultanen Kongruenzen (hat das System von Kongruenzen)

$$x = a_1 \pmod{m_1}$$
$$\vdots$$
$$x = a_n \pmod{m_n}$$

eine eindeutige Lösung $x \in \mathbf{Z}_m$, wobei $m = m_1 \cdot \ldots \cdot m_n$ das Produkt der einzelnen Module ist.

Anmerkungen:

- Der Satz gilt nur, wenn die Module teilerfremd sind. Sind sie nicht teilerfremd, so kann das System keine oder mehrere Lösungen in Z_m haben.
- **CRT** = **C**hinese **R**emainder **T**heorem

Man kann sogar leicht eine Lösung x explizit angeben, wenn man den Begriff des multiplikativen Inversen einführt:

Def D1-3 Multiplikatives Inverses

Wenn es zu $e \in \mathbf{Z}_m$ eine Zahl $d \in \mathbf{Z}_m$ gibt mit

$$e \cdot d = 1 \pmod{m}$$

so nennt man d den **Kehrwert** oder das **multiplikative Inverse** zu e modulo m.

Satz S1-6 Existenz eines multipliaktiven Inversen

Für $e \neq 0$ in $\mathbf{Z}_m$ gilt: Es gibt ein multiplikatives Inverses zu e modulo m genau dann, wenn e und m teilerfremd sind.

Für kleine Module m kann man die multiplikativen Inversen schnell aus der Multiplikationstabelle ablesen. Betrachten wir als Beispiel die Fälle m=5 und m=6, also $\mathbb{Z}_m$ ={0,1,2,3,4} bzw. $\mathbb{Z}_m$ ={0,1,2,3,4,5}:

m=5:

•	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

m=6:

•	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	4	0	2	4
3	0	3	0	3	0	3
4	0	4	2	0	4	2
5	0	5	4	3	2	1

Im Falle m=5 (Primzahl) hat jedes e≠0 ein Inverses, z.B. hat e=2 die Zahl d=3 als Inverses.

Im Falle m=6 sind nur die Zahlen 1 und 5 zum Modul 6 teilerfremd, mithin besitzen nur 1 und 5 multiplikative Inverse (in diesem Fall sich selbst).

Für große Module m ist die Aufstellung einer Verknüpfungstabelle (bzw. einer Zeile aus ihr) zu aufwendig, hier gibt es zum Glück mit dem Algorithmus Erweiterter Euklid (Satz S1-8) eine effizientere Methode.

Übung: Ermitteln Sie die Multiplikationstabelle für m=9! Überlegen Sie **vorher**: Welche Zahlen werden kein multiplikatives Inverses (mod 9) besitzen?

Nun aber zurück zum CRT:

Satz S1-7 Explizite Lösung für CRT (Simultane Kongruenzen)

Für das in Satz S1-5 definierte CRT-Problem kann man Lösung x wie folgt konstruieren:

1. Berechne $M_k=m/m_k$, das ist das Produkt aller Module außer m_k.
2. Berchne für jedes M_k das multiplikative Inverse $N_k \in Z_{m_k}$ für das $M_k N_k=1 (\text{mod } m_k)$.
3. Dann ist

$$x = (a_1 \cdot M_1 \cdot N_1 + \ldots + a_n \cdot M_n \cdot N_n) \text{ mod } m$$

die Lösung der simultanen Kongruenzen.

Beweis in Vorlesung.

Beispiel: Löse mit Satz S1-7 das Problem von Sun-Tsu, d.h. bestimme die kleinste natürliche Zahl x mit

$x = 2 \text{ (mod 3)}$

$x = 3 \text{ (mod 5)}$

$x = 2 \text{ (mod 7)}$

Lösung:

i	1	2	3
m_i	3	5	7
M_i	$5 \cdot 7 = 35$	$3 \cdot 7 = 21$	$3 \cdot 5 = 15$
N_i	2	1	1

Wie bestimmt man N_1? – Gesucht ist eine Zahl N_1 mit

$$35 \cdot N_1 = 1 \text{ (mod 3)} \quad \Leftrightarrow \quad (33+2) \cdot N_1 = 1 \text{ (mod 3)} \quad \Leftrightarrow \quad 2 \cdot N_1 = 1 \text{ (mod 3)},$$

was offensichtlich durch $N_1=2$ erfüllt wird. Analog für N_2 und N_3. Damit ist

$$\begin{aligned} x &= (a_1 \cdot M_1 \cdot N_1 + a_2 \cdot M_2 \cdot N_2 + a_3 \cdot M_3 \cdot N_3) \text{ mod } 105 \\ &= (2 \cdot 35 \cdot 2 + 3 \cdot 21 \cdot 1 + 2 \cdot 15 \cdot 1) \text{ mod } 105 \\ &= 233 \text{ mod } 105 \\ &= 23 \end{aligned}$$

Die gesuchte Zahl ist also x=23.

Übung: Welche Zahl x erfüllt die simultanen Kongruenzen

$$x = 1 \pmod 4$$

$$x = 5 \pmod 7$$

?

Übung: Welche Gleichung muss man im Kopf lösen, um den Kartenspielertrick aus der Aktivierung zu beherrschen? Bestimmen Sie die Lösung, wenn der Zuschauer beim 1. Auslegen auf Stapel 4, beim 2. Auslegen auf Stapel 2 zeigt (die Stapel seien 0,1,2,3,4 bzw. 0,1,2,3 nummeriert).

1.2.3. Der Algorithmus Erweiterter Euklid

Mit dem Algorithmus Erweiterter Euklid bestimmt man

(a) den größten gemeinsamen Teiler ggT(m,n) zweier Zahlen m, n

(b) das multiplikative Inverse (Def D1-3) und

(c) die Lösung linearer diophantischer Gleichungen (s. Kap. 2)

Satz S1-8: Der **Algorithmus ErweiterterEuklid** berechnet für alle $m,n \in \mathbf{N}$ mit $m \le n$ ganze Zahlen $x,y \in \mathbf{Z}$ mit

$$ggT(m,n) = mx + ny.$$

Dabei funktioniert der Algorithmus rekursiv wie folgt:

```
(x,y) = func ErweiterterEuklid(n, m) {
        if (m teilt n) return (1,0);
        else {
                (x',y')=ErweiterterEuklid(m, n mod m);
                (x,y) = (y'-x' floor(n/m),  x');
                return (x,y);
        }
}
```

Beispiel: Was ist ggT(24948, 8712)?

n	m	floor(n/m)	x	y
24948	8712	2	-20	7
8712	7524	1	7	-6
7524	1188	6	-6	1
1188	396		1	0

Die Pfeile zeigen an, dass zuerst (n,m)-Werte berechnet werden, bis man bei 396 landet, das 1188 teilt, also (x,y)=(1,0). Man nimmt also immer die kleinere der beiden Zahlen (m), teilt sie durch die größere (n) und erhält einen Rest (n mod m, was sicher kleiner als m ist). Das bisherige m wird das neue n (die größere Zahl) und der Rest wird das neue m.

Dann berechnet man die (x,y)-Werte von unten nach oben. Der y-Wert ergibt sich jeweils aus dem x-Wert eine Zeile darunter. Der x-Wert ist „y eine Zeile darunter“ minus das Produkt der beiden Zahlen, die an die leere x-Zelle links und unten angrenzen.

Das Ergebnis ggT(24948, 8712)=396 läßt sich entweder aus der untersten m-Zahl ablesen oder aus der ersten Zeile berechnen: $8712 \cdot (-20) + 24948 \cdot 7 = 396$.

Hinweis: Wie berechnet man 24948 / 8712 = 2 Rest 7524 mit dem Taschenrechner?

24948 / 8712 = 2.863636…

Notiere 2 = floor(24948 / 8712), ziehe diese Zahl ab und multipliziere den echten Bruch wieder mit 8712:

0.83636… * 8712 = 7524 (der Rest)

Folgerung S1-9: Wenn $m,n \in \mathbb{N}$ mit $m \le n$ zwei **teilerfremde** Zahlen sind, dann läßt sich für die Gleichung

$$m \cdot x = 1 \pmod{n}$$

die Lösung x aus dem x des Algorithmus ErweiterterEuklid(n, m) gewinnen.

Beweis in Vorlesung.

Übung: Führe den Algorithmus Erweiterter Euklid für n = 2121 und m = 567 durch!

Übung: Besitzt 26 ein multiplikatives Inverses (mod 27)? Wenn ja, ermitteln Sie es!