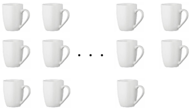
Inhalt

Einführungs-Rätsel

- Bei Umzug: Kiste mit Tassen geht kaputt.
- Frage: Wie viele Tassen waren in der Kiste?

Einführungs-Rätsel – bekannte Infos

Vorletzter Umzug:
Dreierkartons benutzt,
2 übrigbleibende Tassen



Letzter Umzug:
Fünferkartons benutzt,
1 übrigbleibende Tasse

Dieser Umzug:
Siebnerkartons benutzt,
5 übrigbleibende Tassen

Einführungs-Rätsel – mathematische Formulierung

Gesucht ist die Lösung des Gleichungssystems

$$\begin{aligned} x &= 2 \pmod 3 \\ x &= 1 \pmod 5 \\ x &= 5 \pmod 7 \end{aligned}$$

- Lösung durch Pröbeln: $x = ???$
- Weitere Lösungen (in der Theorie)?

Verallgemeinerung

- Informell: Kennt man die Reste von x bezüglich der Faktoren von m, so ist $x \pmod{m}$ eindeutig bestimmt.
  (Im obigen Beispiel ist $m = 105$ und $x = 26$.)
- Formal:

Chinesischer Restsatz

Für paarweise teilerfremde Zahlen $m_1, m_2, \ldots, m_n$ hat jedes Gleichungssystem der Form

$$\begin{aligned} x &= a_1 \pmod{m_1} \\ x &= a_2 \pmod{m_2} \\ &\vdots \\ x &= a_n \pmod{m_n} \end{aligned}$$

genau eine Lösung mod $m := m_1 \cdot m_2 \cdots m_n$

Chinesischer Restsatz – Nutzen

- Rechnen modulo einer grossen Zahl ist effizienter durchführbar!
- Illustration für Multiplikation (mit Zahlen aus Anfangs-Rätsel):

$$
\begin{tikzcd}
\mathbb{Z}_{105} \arrow[r] \arrow[d, "\odot"'] & \mathbb{Z}_3 \times \mathbb{Z}_5 \times \mathbb{Z}_7 \arrow[d, "\odot"] \\
\mathbb{Z}_{105} & \mathbb{Z}_3 \times \mathbb{Z}_5 \times \mathbb{Z}_7 \arrow[l]
\end{tikzcd}
$$

Gesucht: $26 \cdot 37 \pmod{105}$

- Repräsentation modulo 3,5,7: $26 \rightarrow (2, 1, 5)$, $37 \rightarrow (1, 2, 2)$
- elementweise Mulitplikation: $(2, 1, 5) \cdot (1, 2, 2) = (2, 2, 3)$
- Repräsentation modulo 105: $(2, 2, 3) \rightarrow 17$
- direkte Berechnung: $26 \cdot 37 = 17 \pmod{105}$

- Vorteil des "blauen Weges": Alle Rechnungen erfolgen modulo relativ kleiner Zahlen; dies ist viel effizienter! (v.a. bei sehr grossen Modul-Werten von Bedeutung)

Chinesischer Restsatz – Hilfs-Satz für Begründung

Hilfs-Satz

Für jedes Element u einer Gruppe $\mathbb{Z}_m^*$ lässt sich $u^{-1} \pmod{m}$ effizient bestimmen.

Begründung:

- Mithilfe des erweiterten Euklid-Algorithmus lassen sich effizient x, y finden mit

$$xu + ym = 1$$

- Somit: $xu = 1 \pmod{m} \;\Rightarrow x = u^{-1} \pmod{m}$

Bem: $u^{-1} \pmod{m}$ heisst das modulare Inverse

Befehle

- PARI/GP: $\mathrm{Mod}(u, m)^{-1}$
- Java: $u.\mathrm{modInverse}(m)$

Zunächst: Betrachtung des **Spezialfalls** $n = 2$

Chinesischer Restsatz (Spezialfall $n = 2$)

Für teilerfremde Zahlen m_1, m_2 hat jedes Gleichungssystem der Form

$$\begin{aligned} x &= a_1 \pmod{m_1} \\ x &= a_2 \pmod{m_2} \end{aligned}$$

genau eine Lösung mod $m := m_1 \cdot m_2$

Begründung:

- $u_1 := m_2^{-1} \pmod{m_1} \Rightarrow u_1 m_2 = 1 \pmod{m_1}$
- $u_2 := m_1^{-1} \pmod{m_2} \Rightarrow u_2 m_1 = 1 \pmod{m_2}$
- $x := a_1(u_1 m_2) + a_2(u_2 m_1) \pmod{m}$
  - Modulo m_1: $x := a_1\underbrace{(u_1 m_2)}_{1} + a_2\underbrace{(u_2 m_1)}_{0} = a_1$
  - Modulo m_2: $x := a_1\underbrace{(u_1 m_2)}_{0} + a_2\underbrace{(u_2 m_1)}_{1} = a_2 \Rightarrow x$ erfüllt Bed. $\square$

Gleichungssystem (Recap):

$$\begin{aligned} x &= a_1 \pmod{m_1} \\ x &= a_2 \pmod{m_2} \\ &\vdots \\ x &= a_n \pmod{m_n} \end{aligned}$$

Bestimmung von x: $m := m_1 \cdot m_2 \cdots m_n$

$$\begin{array}{ll} M_1 := \frac{m}{m_1} = m_2 \cdots m_n & u_1 := M_1^{-1} \pmod{m_1} \Rightarrow u_1 M_1 = 1 \pmod{m_1} \\ M_2 := \frac{m}{m_2} = m_1 \cdot m_3 \cdots m_n & u_2 := M_2^{-1} \pmod{m_2} \Rightarrow u_2 M_2 = 1 \pmod{m_2} \\ \vdots & \vdots \\ M_n := \frac{m}{m_n} = m_1 \cdots m_{n-1} & u_n := M_n^{-1} \pmod{m_n} \Rightarrow u_n M_n = 1 \pmod{m_n} \end{array}$$

- $x := \sum_{i=1}^{n} a_i \cdot (u_i M_i) \pmod{m}$
  - Modulo m_1: $x = a_1 \underbrace{(u_1 M_1)}_{1} + a_2 \underbrace{(u_2 M_2)}_{0} + \ldots + a_n \underbrace{(u_n M_n)}_{0} = a_1$
  - Die Verifikation der übrigen Gleichungen geht analog.

Chinesicher Restsatz – Begründung allgemeiner Fall

Zusatzbemerkungen:

- Die vorhergehende Begründung zeigt eine stärkere Version des chinesischen Restsatzes: Die Lösung des Gleichungs-Systems kann effizient bestimmt werden. (Wichtig für Anwendungen!)
- Streng genommen müsste man sich noch vergewissern, dass es modulo m niemals 2 (oder mehrere) Lösungen geben kann: Gäbe es 2 Lösungen x, x' fürs Gleichungssystem, so wäre $d := x - x' = 0$ bezüglich aller Moduln $\Rightarrow$ $d = 0 \pmod{m} \Rightarrow x = x' \pmod{m}$

Beispiele

Aufgabe: Bestimmen Sie ein x, so dass

$$\begin{aligned} x &= 2 \pmod 3 \\ x &= 5 \pmod 7 \end{aligned}$$

Beispiele

Aufgabe: Bestimmen Sie ein x, so dass

$$\begin{aligned} x &= 4 \pmod{7} \\ x &= 2 \pmod{11} \\ x &= 8 \pmod{13} \end{aligned}$$

Effizienzgewinn beim RSA-Verfahren

- Entschlüsselung bei RSA: c^d (c: Chiffrat, d: privater Schlüssel)
- Kennt man die Faktorisierung $m = p \cdot q$, so kann man die Entschlüsselung folgendermassen berechnen:
  1. Berechnung von $c_1 = c \pmod{p}$, $\quad d_1 = d \pmod{p-1}$, $a_1 := c_1^{d_1} \pmod{p}$
  2. Berechnung von $c_2 = c \pmod{q}$, $\quad d_2 = d \pmod{q-1}$, $a_2 := c_2^{d_2} \pmod{q}$
  3. Bestimmung der Zahl mit den obigen Resten a_1 und a_2 (via chinesischer Restsatz)

Bem: Die Durchführung der obigen Schritte ist effizienter als die herkömmliche Berechnung in $\mathbb{Z}_m$.

Beispiel

Aufgabe: Wir setzen

- $p = 23$, $q = 31$ und $m = p \cdot q = 23 \cdot 31 = 713$
- $e = 19$ (öffentlicher Schlüssel) und $d = 139$ (privater Schlüssel).

Führen Sie die auf der vorhergehenden Folie beschriebenen Schritte durch, um den Klartext für das Chiffrat $c = 100$ zu bestimmen.

Chinesischer Restsatz – Anwendung 2

Verschlüsselung von Datenbanken

- Grund-Idee: Erstellung von read-keys für Benutzer einer Datenbank.
- Ziel: Bei jedem File hat nur ein bestimmter Teil der Benutzer Zugriffsrechte (z.B. die involvierten Mitarbeiter)
- Formal: Datenbank besteht aus Files $F_1, F_2, \ldots, F_n$. Jedes File F_i wird als ganze Zahl codiert.
- Umsetzung der Zugriffsberechtigung (via "trusted authority"):
  1. Bestimmung von Primzahlen $p_1, \ldots, p_n$ (alle verschieden) mit $p_i > F_i$ für alle i
  2. Bestimmung des Chiffrates C, so dass
  $$\begin{aligned} C &= F_1 \pmod{p_1} \\ C &= F_2 \pmod{p_2} \\ &\vdots \\ C &= F_n \pmod{p_n} \end{aligned}$$
  3. Für jedes File F_i: Berechtigte erhalten zugehöriges p_i ("read-key').

Bem: Mithilfe von C und seinen read-keys kann jeder Benutzer auf seine Files zugreifen.