4. Übungsblatt
Diskrete Strukturen (Winter 2018/19)

Bauhaus-Universität Weimar, Professur für Mediensicherheit
Dr. Stefan Lucks, Nathalie Dittrich
URL: `http://www.uni-weimar.de/de/medien/professuren/mediensicherheit/teaching`

Abgabe: Bis zum 11. Dezember 2018 vor Beginn der Übung oder per E-Mail an `eik.list(at)uni-weimar.de`. Lösungen sind bevorzugt in LaTeX zu verfassen. Ein Template finden Sie auf der Übungsseite der Veranstaltung.

Aufgabe 1 – Modulare Arithmetik (4 Punkte)
Alice zieht um und muss ihre Büchersammlung auf Umzugskisten verteilen. Sie teilt sie zunächst einfach wie folgt auf:

- Wenn sie je 7 Bücher in jede Kiste legt, bleibt am Ende ein Buch übrig.
- Wenn sie je 10 Bücher in jede Kiste legt, bleiben zwei übrig.
- Wenn sie je 11 Bücher in jede Kiste legt, bleiben auch sechs übrig.

Finden Sie mit Hilfe des naheliegenden Satzes der Vorlesung die kleinstmögliche positive Anzahl der Bücher in Alice' Büchersammlung heraus. Erläutern Sie Ihren Rechenweg.

Aufgabe 2 – Gruppeneigenschaften (8 Punkte)
Lösen Sie die folgenden Aufgaben zum Thema Gruppeneigenschaften.

a) Untersuchen Sie die folgende potentielle Gruppe $(\{0,1\}^n, \oplus)$ auf Ihre Gruppeneigenschaften. $\{0,1\}^n$ meint dabei n-bit Werte und $\oplus$ XOR.

b) Zeigen Sie, dass $(\mathbb{Z}_n \setminus \{0\}, \cdot)$ genau dann eine Gruppe ist, wenn n eine Primzahl ist. Die Operation '$\cdot$' bezeichne dabei die Multiplikation mod n.

c) Untersuchen Sie die folgende potentielle Gruppe auf ihre Gruppeneigenschaften:
$(\mathbb{Z}_{13}, \circ)$ mit $a, b \in \mathbb{Z}_{13}$ und $a \circ b := (2 \cdot a + 2 \cdot b) \bmod 13$.

d) Sei $\mathrm{Dlog}_{2,11}(a)$ der diskrete Logarithmus zur Basis 2 in $\mathbb{Z}_{11}$. Es gilt für alle $a \in \mathbb{Z}_{11}$: $\mathrm{Dlog}_{2,11}(2^a \bmod 11) = a$. Zum Beispiel ist $\mathrm{Dlog}_{2,11}(5) = 4$, denn $2^4 \bmod 11 \equiv 5$. Untersuchen Sie die folgende potentielle Gruppe auf ihre Gruppeneigenschaften:
$(\mathbb{Z}_{11}, \circ)$ mit $a, b \in \mathbb{Z}_{11}$ und $a \circ b := \mathrm{Dlog}_2(2^a \cdot 2^b \bmod 11)$.

Aufgabe 3 – Eindeutigkeit (4 Punkte)
Beweisen Sie Satz 52 und Satz 53 aus der Vorlesung.

> **Satz 52 (Eindeutigkeit des neutralen Elements)**
> In einer Halbgruppe gibt es höchstens ein neutrales Element.

> **Satz 53 (Eindeutigkeit des Inversen)**
> In einem Monoid gibt es zu jedem Element höchstens ein Inverses. Hat x ein Inverses x^{-1}, dann ist x selbst das Inverse von x^{-1}.

Aufgabe 4 – Fermat-Test + RSA (Programmieraufgabe) (5 Punkte)
Implementieren Sie den Fermat-Test und den RSA-Algorithmus aus der Vorlesung (Folien 134 und 123) in Python. Das Programm soll dabei vier Werte p ,q, x und k als Kommandozeilenparameter entgegennehmen und folgende Funktionalität bieten:

- Behandlung fehlerhafter Eingaben, z. B.: falsche Anzahl oder falscher Typ von Kommandozeilenparametern.
- Testen mittels des Fermat-Tests, ob p und q prim sind. Die Anzahl an verwendeten Zufallszahlen soll dabei k entsprechen. k soll optional sein. Wenn p und/oder q keine Primzahlen sind, soll ihr Programm `p = <xx> is not prime` bzw. `q = <xx> is not prime` ausgeben.
- Berechnen des RSA Modulus $n = p \cdot q$ und der φ-Funktion $\varphi(n) = (p-1) \cdot (q-1)$.
- Auswahl des kleinstmöglichen öffentlichen Exponenten $e \in \mathbb{N}^*_{\varphi(n)}$ mit $\text{ggT}(\varphi(n), e) = 1$.
- Berechnen des geheimen Exponenten d mit Hilfe des Erweiterten Euklidischen Algorithmus' (s. Folien 106 ff.).
- Verschlüsseln von Nachrichten M mit (e, n) mit $C = M^e \bmod n$.
- Entschlüsseln von Chiffretexten C mit (d, n) mit $M' = C^d \bmod n$.

Beispielaufruf:

```
python3 rsa_<xxxxxx>.py --mode=encrypt 73 67 507
3648
python3 rsa_<xxxxxx>.py --mode=encrypt 73 67 507 -k 15
3648
python3 rsa_<xxxxxx>.py --mode=decrypt 73 67 3648
507
python3 rsa_<xxxxxx>.py --mode=encrypt 72 67 3648
p = 72 is not prime
python3 rsa_<xxxxxx>.py --mode=encrypt 73 66 3648
q = 66 is not prime
```

Schicken Sie Ihre Lösung als Anhang einer E-Mail als Pythondatei

rsa_<IhreMatrikelnummer>.py

an **eik.list(at)uni-weimar.de** mit dem Betreff **[DS WS2018/19 Beleg 4]**. Es reicht für **IhreMatrikelnummer** eine Matrikelnummer Ihrer Gruppe. Die vollständigen Namen und Matrikelnummern sollen als Kommentar in der Pythonabgabe stehen.