Mathematisches Institut
der Universität München
Prof. Dr. O. Forster

WS 2015/16
2. Dez. 2015

Algorithmische Zahlentheorie und Kryptographie

Übungsblatt 7

Aufgabe 25

Man leite aus dem quadratischen Reziprozitäts-Gesetz für das Jacobi-Symbol und dem 1. Ergänzungssatz den 2. Ergänzungssatz ab.

Hinweis. Es gilt $\left(\frac{2}{a}\right) = \left(\frac{-1}{a}\right)\left(\frac{a-2}{a}\right)$ für ungerades $a \geqslant 3$.

Aufgabe 26

a) Sei $a \geqslant 2$ eine ganze Zahl und seien p, q ungerade Primzahlen mit $p \equiv q \bmod 4a$.
Man beweise mit Hilfe des quadratischen Reziprozitäts-Gesetzes, dass $\left(\frac{a}{p}\right) = \left(\frac{a}{q}\right)$.

b) Sei $p \geqslant 5$ eine Primzahl. Man zeige, dass das Legendre-Symbol $\left(\frac{3}{p}\right)$ nur von $p \bmod 12$ abhängt und gebe dafür eine Formel analog zum 2. Ergänzungssatz an.

Aufgabe 27

Sei p eine Primzahl mit $p \equiv 1 \bmod 2^2$, aber $p \not\equiv 1 \bmod 2^3$, d.h. $p \equiv 5 \bmod 8$.

a) Sei $j := 2^{(p-1)/4} \bmod p$. Man zeige $j^2 \equiv -1 \bmod p$.

b) Sei a eine ganze Zahl mit $\left(\frac{a}{p}\right) = 1$. Man zeige die Korrektheit des folgenden Algorithmus zur Berechnung der Quadratwurzel von a modulo p: Aus

$$x := a^{(p+3)/8} \bmod p \qquad \text{folgt} \qquad x^2 \equiv \pm a \bmod p.$$

Trifft das Pluszeichen zu, sind wir fertig.
Andernfalls genügt $z := jx$ der Kongruenz $z^2 \equiv a \bmod p$.

Aufgabe 28

Die 200-Bit-Zahl

$$N := 1\,07288\,94281\,14363\,63918\,26037\,39363\,39452\,81490\,03402\,27688\,58347\,94997,$$

hexadezimal

AA EBD0 494B 899F 884E E177 5ED4 DAFA 9E99 8432 CB87 AAD4 87F5

sei der Modul eines Blum-Blum-Shub Pseudo-Zufallsgenerators, d.h. $N = pq$ mit Primzahlen $p \equiv q \equiv 3 \bmod 4$. Ausgehend von einem quadratischen Rest $z_0 := a^2 \bmod N$, $\gcd(a, N) = 1$, sei die Folge $(z_i)_{i \geqslant 0}$, $1 \leqslant z_i < N$, rekursiv definiert durch $z_{i+1} := z_i^2 \bmod N$. Diese Folge definiert eine Pseudo-Zufallsfolge von Bits $b_i := z_i \bmod 2$.

---

b.w.

Der folgende Geheimtext der Länge von 138 Bytes

```
EAD3 99D0 4DF9 96AE 31D6 1D2B CC32 63A6 CD4F C77D 4D5C C3C2 6D57 353D
0B16 E50F 8821 A2E9 19A6 3AFF 0718 366E D3FF 95DE C9EE 42E5 24D3 3370
E37C AC2A 3305 B45B E691 5EAE 8ED7 046F 8A77 3AB3 CF18 A1E4 42C3 1FF3
C53D 30CB CE28 2684 24E3 2835 9C83 0C6B BC65 8FFA 7B3C 9130 D24B A8BC
0030 BD73 B0D0 9953 2BAD B86B 59B0 A97E 2727 CE2B AA0D 9A86 4947
```

entstand aus einem Klartext von 112 Bytes auf folgende Weise: Die Bits $(b_i)_{0 \leqslant i < 896}$ wurden zu einem One-Time-Pad der Länge 112 Bytes (= 896 Bits) zusammengefasst. (Jeweils 8 Bits $\beta_0, \beta_1, \ldots, \beta_7$ ergeben ein Byte $\xi = \sum_{i=0}^{7} \beta_i \cdot 2^i$.) Das One-Time-Pad wurde mit der Operation XOR auf den Klartext addiert, was die ersten 112 Bytes des Geheimtextes liefert. Anschließend folgt ein Nullbyte und dann 25 Bytes, welche die Zahl z_{896} in hexadezimaler (MSF, *most significant byte first*) Schreibweise darstellen.

Man bestimme z_0 und entschlüssele den Geheimtext.