Dr. Andreas Borchert
Christian Ehrhardt

Allgemeine Informatik III (WS 2006/2007)

Abgabetermin 14.Nov 2006

ACHTUNG!

Leider hatte sich in die veröffentlichten verschlüsselten Texte ein Fehler eingschlichen. Wer diese bereits vor Donnerstag 12h heruntergeladen hat, hat die falschen bekommen. Es tut mir leid, wenn das zu Verwirrung geführt hat und bedanke mich für den freundlichen Hinweis.

Geheimsachen oder Blum-Blum-Shub (10 Punkte)

Das Prinzip

Um einen Text zu verschlüsseln, haben Sie sich überlegt, daß es vielleicht eine gute Idee wäre, wenn einfach die Buchstaben des Alphabets rotiert würden. Wird zum Beispiel um $k = 3$ rotiert, dann wird aus einem `A` ein `D`, aus einem `B` ein `E` und aus einem `X` wieder ein `A`. Analoges gilt natürlich auch für Kleinbuchstaben. Alle anderen Zeichen werden überhaupt nicht verschlüsselt.
Weil es Ihnen zu einfach erscheint, wenn jeder Buchstabe um die gleiche Anzahl k von Buchstaben verschoben wird, beginnen Sie mit einem Startwert k_1, der für den ersten Buchstaben gilt. Für den zweiten Buchstaben wird k_2 berechnet als $k_2 = k_1 * k_1 \bmod M$, wobei M zunächst frei gewählt werden kann, es sollte nur nicht zu klein sein. Auf diese Weise wird für jeden Buchstaben, der zu verschlüsseln ist, ein neuer Wert für k berechnet.
Um ein Gefühl für das System zu bekommen, ist es am sinnvollsten, das Verfahren einmal mit kleinen Werten von Hand auszuprobieren. Ist etwa $k_1 = 17$ und $M = 2021$, dann erhält man für k_1 bis k_5 nacheinander die Werte 17, 289, 660, 1085 und 1003. Das bedeutet, daß aus `Hallo` der Text `Ydved` wird.

Ein erster Versuch (4 Punkte)

Implementieren Sie ein Programm, das von der Standardeingabe einen Text einliest und wie oben beschrieben verschlüsselt. Die Werte für k_1 und M sollen im Programm mit Hilfe von Konstanten festgelegt werden. Wenn das funktioniert, wird natürlich auch ein Programm zum Entschlüsseln benötigt, das man durch einige kleine Veränderungen am Verschlüsselungsprogramm erhalten kann.

Gute Werte für M und k_1

Nicht jede Wahl von M und k_1 liefert gleich gute Ergebnisse, da die erzeugte Folge der k_i unter Umständen eine sehr kurze Periodenlänge hat. Die Erfinder des Verfahrens (Lenore Blum, Manuel Blum und Michael Shub – daher auch der Name Blum-Blum-Shub) haben zwei Bedingungen angegeben, unter denen die Periodenlänge groß wird:

- M ist des Produkt von zwei (größeren) Primzahlen P und Q, die beide einen Rest von 3 bei der Division durch 4 lassen.
- Der größte gemeinsame Teiler von $\phi(P-1)$ und $\phi(Q-1)$ ist klein. Was hierbei $\phi(X)$ bedeutet wird weiter unten erklärt.

Gute Primzahlen (3 Punkte)

Ändern Sie Ihre Programme so ab, daß ab sofort nicht mehr M direkt sondern zwei Primzahlen P und Q angegeben werden. M ergibt sich dann als das Produkt dieser beiden Primzahlen.
Schreiben Sie weiterhin eine Funktion, die überprüft, ob eine gegebene Zahl überhaupt eine Primzahl ist und ob die Zahl bei Division durch 4 den Rest 3 läßt. Mit dieser Funktion werden beim Programmstart die Werte für P und Q überprüft. Sollte die Überprüfung für einen der beiden Werte fehlschlagen, dann verweigert das Programm mit einer Fehlermeldung die Arbeit.

Die ϕ-Funktion

Die ϕ-Funktion für eine Zahl x kann wie folgt berechnet werden.

- Das vorläufige Ergebnis wird mit 1 initialisiert.
- Man beginnt von 2 an aufwärts nach Teilern von x zu suchen. Hat man einen Teiler t gefunden, wird x durch t dividiert und das vorläufige Ergebnis mit $t-1$ multipliziert.
- Sollte nach diesem Schritt x immernoch durch t teilbar sein, dann wird x solange durch t dividiert, bis t kein Teiler von x mehr ist. Dabei wird das vorläufige Ergebnis jedes Mal mit t (nicht mehr mit $t-1$) multipliziert.
- Anschließend wird t um 1 erhöht.
- Sobald x den Wert 1 erreicht wird das vorläufige Endergebnis zum endgültigen Ergebnis.

Anmerkungen: Die ϕ-Funktion spielt in der Zahlentheorie eine große Rolle. Unter anderem gibt $\phi(x)$ an, wieviele Zahlen, zwischen 1 und x teilerfremd zu x sind. Mathematisch läßt sich die ϕ-Funktion auch wie folgt beschreiben:

$$\phi(x) = x \cdot \prod_{p|x,\ p \text{ prim}} \frac{p-1}{p}$$

Implementierung der ϕ-Funktion (3 Punkte)

Implementieren Sie die im letzten Abschnitt beschriebene *phi*-Funktion, sowie eine Funktion, die den größten gemeinsamen Teiler zweier Zahlen berechnet. Überprüfen Sie damit die zweite oben angegebene Bedingung an P und Q beim Programmstart. Dabei soll *klein* bedeuten, daß der berechnete größte gemeinsame Teiler kleiner als 10 ist.

Hinweise

- Der hier beschriebene *Blum-Blum-Shub* Zufallszahlengenerator gilt zwar als kryptographisch sicher, die Art, wie er hier angewendet wird, ist es aber nicht.
- Wenn $M = P \cdot Q$ größer wird, dann kann es sein, daß das Quadrat von k_i nicht mehr in einem `int` Platz hat. Es empfiehlt sich daher ein `long long int` zur Berechnung dieses Zwischenergebnisses.
- Auf der Homepage der Vorlesung gibt es einige Texte, die mit unterschiedlichen Werten für P, Q und k_1 verschlüsselt wurden. Damit kann das eigene Entschlüsselungsprogramm getestet werden.
- Die zum Verschlüsseln verwendeten Werte für P, Q und den Startwert $S := k_1$ stehen in der Datei. Da Zahlen nicht mitverschlüsselt werden, sind diese Werte im Klartext zu lesen.
- Nicht alle verwendeten Werte erfüllen die Bedingungen an P und Q. Eine vollständige Lösung sollte das natürlich feststellen.
- Bei einem Beispiel sind die Zwischenergebnisse unter Umständen sogar so groß, daß sie auch in einem `long long` keinen Platz finden. Wer diese Datei auch entschlüsseln möchte, muß sich noch etwas zusätzliches einfallen lassen. Dieses Problem zu lösen ist aber nicht Bestandteil dieses Blatts.