MACM 498/CMPT 881/MATH 800
Assignment 5, Fall 2004

Michael Monagan

This assignment is to be handed in on Tuesday November 23rd at the beginning of class. Late penalty: 10% off for each day late.

Chapter 6.

1: Let $f(z) \in \mathbb{Z}_p[z]$ have degree greater than 0. Consider the finite ring

$$R = \mathbb{Z}_p[z]/(f) = \{u \in \mathbb{Z}_p[z] \mid \deg(u) < \deg(f)\}.$$

Let $u \in R$. Show that u is invertible in R if and only if $\gcd(u, f)$=1.

2: In the ElGamal crytosystem in $\mathbb{Z}_p^*$, to encrypt $x \in \mathbb{Z}_p^*$, Alice chooses a random integer $0 < r < p$ then sends the pair (y_1, y_2) to Bob where $y_1 = \alpha^r \bmod p$ and $y_2 = x\beta^r \bmod p$. Bob decrypts by computing $x = y_2 y_1^{-a} \bmod p$.

In class we proposed that if Alice wants to send Bob several messages $x_1, x_1, ..., x_m$, she might use the same r to speed up encryption as follows: she computes $c = \alpha^r \bmod p$ and $d = \beta^r \bmod p$ once. Now to encrypt $x_1, x_2, ..., x_m$ she sends $(c, dx_i \bmod p)$ for $i = 1..m$ which requires m multiplications in $\mathbb{Z}_p$.

Explain how Bob can similarly speed up decryption.
Explain why this might not be a good idea.

3: Find an isomorphism between the group G = $(\mathbb{Z}_7^*, \times)$ and H = $(\mathbb{Z}_6, +)$.
Hint: Discrete Logarithms.

4: For CMPT 881 and MATH 800 students: Implement Algorithm 6.6 and use it to answer exercise 6.2. You will have to "simulate" an oracle for computing $L_2(\beta)$.

Chapter 2

5: Exercise 2.2

6: For the One-Time-Pad, to encrypt one bit, let $K \in 0, 1$ be the key. Show that if the $\Pr(K = 0) \neq 1/2$ then the One-Time-Pad does NOT have perfect secrecy.

Chapter 12

7: Exercise 12.3.

8: Consider the linear congruential generator based on the finite field GF(2^k) with 2^k elements. Let α be a primitive element from $GF(2^k)$ and let $s_0 \in GF(2^k)^*$ be the seed. Compute

$$s_i = \alpha s_{i-1} \quad \text{for} \quad i = 1, 2, ..., m$$

and convert each s_i to a k bit bit-string: If $s_i = a_0 + a_1 y + ... + a_{k-1} y^{k-1}$ then the bit-string is $a_0 a_1 ... a_{k-1}$. This will produce a bit string of length km and thus it can be viewd as a (k, l)-Pseudo Random Bit Generator with seed s_0.

Implement this generator for $GF(2^{16})$. To construct the field you need to find an irreducible polynomial $f(y)$ of degree 16 in $\mathbb{Z}_2[y]$. Use the `Nextprime` command in Maple to find one. Now choose a random primitive element $\alpha \in \text{GF}(2^{16}) = \mathbb{Z}_2[y]/f(y)$. Now compute $s_1, ..., s_{16}$ and convert each s_i to a bit-string. This will produce a bit string of length 256.

Now explain why (k, l)-PRBGs constructed in this way are not secure for cryptographic purposes. Demonstrate this by showing how to compute f, α, s_0 from $s_1, s_2, ..., s_{16}$.

9: Consider the example of the BBS Generator on page 372 of Chapter 12 with $n = 192649 = 383 \times 503$ and $s_0 = 20749$. Reproduce the 20 bit bit-string 11001110000100111010. The BBS algorithm requires that $s_0 \in QR(n)$. Thus one possibility for s_0 is $s_0 = 1$ which is not a good choice! I claim that the map $x \to x^2 \bmod n$ partitions QR(n) into a set of cycles $C_1, C_2, ...,$. Compute these cycles and their cardinality for this n and display the data in a reasonable format. Hence determine (i) the period for $s_0 = 20749$ and (ii) the possible periods for this n.