MACM 442/MATH 742/MATH 800
Assignment 5, Fall 2008

Michael Monagan

This assignment is to be handed in on Thursday November 13th at the beginning of class.
Late penalty: 20% off for up to 24 hours late, zero after that.
Note, this assigment has a lot of calculations in finite fields.

Chapter 6.

Exercises 6.12 and 6.20.

For 6.20, implement Algorithm 6.6 and use it to answer the exercise.
You will have to "simulate" an oracle for computing $L_2(\beta)$.

Question 3: Suppose Bob wants to construct an ElGamal cryptosystem based on the finite field with 2^{128} elements, i.e. the group in which ElGamal is run will have $n = 2^{128} - 1$ elements. The security of the discrete logarithm problem depends on the largest prime dividing n. What is the largest prime dividing n? Using Maple, find an polynomial $f(x)$ of degree 128 in $\mathbb{Z}_2[x]$ that is irreducible over $\mathbb{Z}_2$. Then we have $F = \mathbb{Z}_2[x]/(f)$ is a finite field with 2^{128} elements. Using the factorization of $n = 2^{128} - 1$ determine the first primitive element in F, i.e., the first element in the sequence $0, 1, x, x+1, x^2, x^2+1, x^2+x+1, x^3, ...$ that has order n.

Question 4: Find an isomorphism between the group G = $(\mathbb{Z}_7^*, \times)$ and H = $(\mathbb{Z}_6, +)$.
Hint: Discrete Logarithms.

Question 5 (MATH 742 and 800 students only): On page 253 the text writes "Computation of inverses (in finite fields) can be done by using a straightforward adaption of the extended Euclidean algorithm." You are to explain how to do this as follows.

Let F be a field and $f, a \in F[x]$ with $a \neq 0$. Recall that the the Euclidean algorithm in $F[x]$ initializes $r_0 = f$ and $r_1 = a$ and computes polynomials $r_2, r_3, ..., r_n, r_{n+1} = 0$ by dividing r_{i-1} by r_i to get r_{i+1} satisfying

$$r_{i-1} = r_i q_{i+1} + r_{i+1} \quad \text{with} \quad r_{i+1} = 0 \quad \text{or} \quad \deg r_{i+1} < \deg r_i.$$

If $c = \mathrm{lc}(r_n)$, then $g = c^{-1} r_n$ is the monic gcd of f and a.

(i) Extend the Euclidean algorithm to compute also polynomials $s_0 = 1, s_1 = 0$ and $s_{i+1} = s_{i-1} - q_{i+1}s_i$ for $1 \leq i \leq n$ and polynomials $t_0 = 0, t_1 = 1$ and $t_{i+1} = t_{i-1} - q_{i+1}t_i$ for $1 \leq i \leq n$. Prove (by induction on i) that $s_i f + t_i a = r_i$ for $0 \leq i \leq n+1$. Hence prove that given $f(x), a(x) \in \mathbb{Z}_p[x]$, $a \neq 0$, there exist polynomials $s, t \in \mathbb{Z}_p[x]$ satisfying $sf + ta = g$ in $\mathbb{Z}_p[x]$ where $g = \gcd(f, a)$.

(ii) Now, letting $f(x) \in \mathbb{Z}_p[x]$ be irreducible over $\mathbb{Z}_p$ and $R = \mathbb{Z}_p[x]/f(x)$ be a finite field, explain how to compute the inverse of an element $a \in R$ using the extended Euclidean algorithm. Now illustrate your answer with the following example. For $f(x) = x^3+2x^2+1 \in \mathbb{Z}_3[x]$ and $a = x^2+x+2$ execute the extended Euclidean algorithm by hand showing the r_i, q_i, s_i, t_i polynomials and determine $a^{-1} \in \mathbb{Z}_3[x]/f(x)$.

Chapter 8

Exercises 8.5, 8.9.

Question 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 viewed 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 $s_1, s_2, ..., s_{16}$.

Question 9: Consider the example of the BBS Generator on page 337 of Chapter 8 with $n = 192649 = 383 \times 503$ and $s_0 = 101355^2 = 20749 \bmod n$. Implement the BBS generator and reproduce the 20 bit bit-string 11001110000100111010.

Now the BBS algorithm requires that $s_0 \in QR(n)$. 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 $n = 192649$ and display the data in a reasonable format. Hence determine (i) the period for $s_0 = 20749$ and (ii) the other possible periods for this BBS generator. The BBS algorithm also needs that s_0 not generate a small cycle!