CS 70 Discrete Mathematics and Probability Theory
Fall 2011 Rao HW 5

Due Friday, September 30, 5:00pm

You *must* write up the solution set entirely on your own. You must never look at any other students' solutions (not even a draft), nor share your own solutions (not even a draft).

Please put your answer to each problem on its own sheet of paper, and paper-clip (don't staple!) the sheets of paper together. Label each sheet of paper with your name, your discussion section number (101–108), your login, and "CS70–Fall 2011". Turn in your homework and problem x into the box labeled "CS70 – Fall 2011, Problem x" whereon the 2nd floor of Soda Hall. Failure to follow these instructions will likely cause you to receive no credit at all.

1. (20 pts.) Polynomial interpolation

Consider the set of four points $\{(0,1),(1,-2),(3,4),(4,0)\}$.

(a) Construct the unique degree-3 polynomial (over the reals) that passes through these four points by writing down and solving a system of linear equations.

(b) Repeat part (a) but using the method of Lagrange interpolation. Show your working clearly.

2. (25 pts.) Representing polynomials

Let f be a polynomial of degree at most d. The *coefficient representation* of f is the sequence $(a_0, a_1, \ldots, a_d)$ of coefficients of f. A *point-value representation* of f is a collection $\{(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_t, f(x_t))\}$ of values of f at any t points $x_1, x_2, \ldots, x_t$, where $t \geq d+1$. (Recall that a polynomial of degree d is completely determined by its values at any $d+1$ points. Note that t may be greater than $d+1$, so more points than necessary may be given.)

In the following questions, let f and g be any two real polynomials of degree at most d.

(a) What is the maximum degree of the product polynomial fg?

(b) Given coefficient representations of f and g, explain how to compute the coefficient representation of fg using $O(d^2)$ arithmetic operations (additions/subtractions/multiplications/divisions) over real numbers.

(c) Now suppose that f and g are specified by point-value representations at t points for some $t \geq d+1$, i.e., f is specified as $(x_1, f(x_1)), (x_2, f(x_2)), \ldots, (x_t, f(x_t))$, and g as $(x_1, g(x_1)), (x_2, g(x_2)), \ldots, (x_t, g(x_t))$. With a suitable value of t (which you should specify), show how to compute a point-value representations of fg using only $O(d)$ arithmetic operations.

(d) Suppose that polynomial g divides polynomial f, and that f, g are given in point-value representation as in part (c) with $t = d+1$. Show how to compute a point-value representation for the quotient f/g using $O(d)$ arithmetic operations, and justify your algorithm carefully.

(e) Suppose you are given f in coefficient representation, and you want to compute a point-value representation for f at $t = d+1$ points. Show how to do this using $O(d^2)$ arithmetic operations. [Hint: Show how to evaluate f at one point using $O(d)$ operations; to do this, consider writing f in the form $f(x) = a_0 + xh(x)$, where h is polynomial of degree at most $d-1$, and iterating.]

3. (12 pts.) Polynomials over GF(p)

(a) Deduce from Fermat's Little Theorem (see Lecture Note 10) that, if p is prime, then any polynomial over GF(p) is equivalent to a polynomial of degree (at most) $p-1$. In this problem, we consider a polynomial f as equivalent to another g if $f(x) = g(x)$ for all $x \in \text{GF}(p)$.
[Hint: Be careful! It is not true that $x^{p-1} \equiv 1 \pmod{p}$ for all $x \in \{0, 1, \ldots, p-1\}$ (why not?).]

(b) Using part (a), deduce that the number of distinct (i.e. inequivalent) polynomials over GF(p) is exactly p^p.

(c) Using part (b), deduce that every function on GF(p) is equivalent to a polynomial over GF(p). [Hint: A function f on GF(p) assigns to each $x \in \text{GF}(p)$ a value $f(x) \in \text{GF}(p)$. How many functions are there on GF(p)?]

4. (12 pts.) Hierarchical secret sharing

Consider the following variant of the secret sharing problem. We wish to share a secret among fifteen people, divided into three groups of five, so that the following condition is satisfied. A subset of the fifteen people can recover the secret if and only if it contains majorities (at least three out of five) of at least two of the groups. How would you modify the standard secret sharing scheme to achieve this condition? Briefly justify your answer. [Hint: The title of this problem may help you! Try a two-level scheme, one level for groups, the other for people within the group.]

5. (10 pts.) Error-correcting codes

(a) Recall from class the error-correcting code for erasure errors, which protects against up to k lost packets by sending a total of $n+k$ packets (where n is the number of packets in the original message). Often the number of packets lost is not some fixed number k, but rather a *fraction* of the number of packets sent. Suppose we wish to protect against a fraction α of lost packets (where $0 < \alpha < 1$). How many packets do we need to send (as a function of n and α)?

(b) Repeat part (a) for the case of general errors.

6. (21 pts.) Berlekamp–Welsh algorithm

In this question we will go through an example of error-correcting codes with general errors. We will send a message (m_0, m_1, m_2) of length $n = 3$. We will use an error-correcting code for $k = 1$ general error, doing arithmetic modulo 5.

(a) Suppose $(m_0, m_1, m_2) = (4, 3, 2)$. Use Lagrange interpolation to construct a polynomial $P(x)$ of degree 2 (remember all arithmetic is $\bmod 5$) so that $(P(0), P(1), P(2)) = (m_0, m_1, m_2)$. Then extend the message to lengeth $n+2k$ by appending $P(3), P(4)$. What is the polynomial $P(x)$ and what is the message $(c_0, c_1, c_2, c_3, c_4) = (P(0), P(1), P(2), P(3), P(4))$ that is sent?

(b) Suppose the message is corrupted by changing c_0 to 0. We will locate the error using the Berlekamp–Welsh method. Let $E(x) = x + b_0$ be the error-locator polynomial, and $Q(x) = P(x)E(x) = a_3x^3 + a_2x^2 + a_1x + a_0$ be a polynomial with unknown coefficients. Write down the system of linear equations (involving unknowns a_0, a_1, a_2, a_3, b_0) in the Berlekamp–Welsh method. You need not solve the equations.

(c) The solution to the equations in part (b) is $b_0 = 0, a_0 = 0, a_1 = 4, a_2 = 4, a_3 = 0$. Show how the recipient can recover the original message (m_0, m_1, m_2).