CS 70 Discrete Mathematics and Probability Theory
Summer 2012 Filiba HW 3

Due Thursday, July 5, 4:59pm

1. (15 pts.) Stable Marriage True-or-False?
For each of the following claims, state whether the claim is true or false. If it is true, give a *short* proof; if it is false, give a *simple* counterexample.

(a) In a stable marriage instance, if man M and woman W each put each other at the top of their respective preference lists, then M must be paired with W in every stable pairing.

(b) In a stable marriage instance with at least two men and two women, if man M and woman W each put each other at the bottom of their respective preference lists, then M cannot be paired with W in any stable pairing.

(c) For every $n > 1$, there is a stable marriage instance with n men and n women which has an unstable pairing in which every unmatched man-woman pair is a rogue couple.

2. (12 pts.) Modular Arithmetic

(a) Give the addition and multiplication tables for modular-5 arithmetic. Write down the inverse for each of the elements which have one, and identify the ones which have no inverse.

(b) Solve the following equations for x and y or show that no solution exists. Show your work (in particular, what division must you carry out to solve each case).

(i) $5x+23 \equiv 6 \pmod{47}$

(ii) $9x+80 \equiv 2 \pmod{81}$

(iii) The system of simultaneous equations
$30x+3y \equiv 0 \pmod{37}$ and $y \equiv 4+13x \pmod{37}$

(c) Compute $\gcd(5688, 2010)$ and show your steps.

(d) Use Extended Euclid's algorithm to find some pair of integers j,k such that $52j+15k = 3$. Show your work.

3. (10 pts.) GCD
In class we saw that, if $\gcd(m,x) = 1$ then there are m distinct elements in the set $\{\mathrm{mod}(ax,m) : a \in \{0,\ldots,m-1\}\}$. If $gcd(m,x) > 1$, how many distinct elements are there? Prove your answer.

4. (10 pts.) Poker mathematics
A *pseudorandom number generator* is a way of generating a large quantity of random-looking numbers, if all we have is a little bit of randomness (known as the *seed*). One simple scheme is the *linear congruential generator*, where we pick some modulus m, some constants a,b, and a seed x_0, and then generate the sequence of outputs $x_0, x_1, x_2, x_3, \ldots$ according to the following equation:

$$x_{t+1} = \mathrm{mod}\,(ax_t + b, m)$$

(Notice that $0 \le x_t < m$ holds for every t.)

You've discovered that a popular web site uses a linear congruential generator to generate poker hands for its players. For instance, it uses x_0 to pseudo-randomly pick the first card to go into your hand, x_1 to pseudo-randomly pick the second card to go into your hand, and so on. For extra security, the poker site has kept the parameters a and b secret, but you do know that the modulus is $m = 2^{31} - 1$ (which is prime).

Suppose that you can observe the values x_0, x_1, x_2, x_3, and x_4 from the information available to you, and that the values $x_5, \ldots, x_9$ will be used to pseudo-randomly pick the cards for the next person's hand. Describe how to efficiently predict the values $x_5, \ldots, x_9$, given the values known to you.

5. (12 pts.) RSA

In this problem you play the role of Amazon, who wants to use RSA to be able to receive messages securely.

(a) Amazon first generates two large primes p and q. He picks $p = 13$ and $q = 19$ (in reality these should be 512-bit numbers). He then computes $N = pq$. Amazon chooses e from $e = 37, 38, 39$. Only one of those values is legitimate, which one? (N, e) is then the public key.

(b) Amazon generates his private key d. He keeps d as a secret. Find d. Explain your calculation.

(c) Bob wants to send Amazon the message $x = 102$. How does he encrypt his message using the public key, and what is the result?

(d) Amazon receives an encrypted message $y = 141$ from Charlie. What is the unencrypted message that Charlie sent him?

6. (10 pts.) Easy RSA

In class, we said that RSA uses as its modulus a product of two primes. Let's look at a variation that uses a single prime number as the modulus. In other words, Bob would pick a 1024-bit prime p and a public exponent e satisfying $2 \le e < p-1$ and $\gcd(e, p-1) = 1$, calculate his private exponent d as the inverse of e modulo $p-1$, publish (e, p) as his public key, and keep d secret. Then Alice could encrypt via the equation $E(x) = \bmod(x^e, p)$ and Bob could decrypt via $D(y) = \bmod(y^d, p)$.

Explain why this variation is insecure. In particular, describe a procedure that Eve could use to recover the message x from the encrypted value y that she observes and the parameters (e, p) that are known to her. Analyze the running time of this procedure, and make sure to justify why Eve could feasibly carry out this procedure without requiring extravagant computation resources.

7. (10 pts.) Practice with Lagrange interpolation

This problem will have you practice with Lagrange interpolation. Here, we are looking for a polynomial $p(x)$ of degree at most 2 that passes through the points $(1,2)$, $(2,3)$, and $(3,5)$, working in $GF(7)$. In other words, we want $p(x)$ to satisfy $p(1) \equiv 2 \pmod 7$, $p(2) \equiv 3 \pmod 7$, and $p(3) \equiv 5 \pmod 7$.

(a) Find the three polynomials $\Delta_1(x), \Delta_2(x), \Delta_3(x)$. Simplify them to the form $ax^2 + bx + c \pmod 7$ where a, b, c are integers satisfying $0 \le a, b, c < 7$. Circle or box your final answer.

(b) Using your answer to part 1 and Lagrange interpolation, find the polynomial $p(x)$. Simplify it to the form $ax^2 + bx + c \pmod 7$ where a, b, c are integers satisfying $0 \le a, b, c < 7$. Circle or box your final answer.

8. (6 pts.) I'm not a spy, but I play one in CS70

You are sent an encoded message $(c_1, c_2, c_3, c_4, c_5, c_6)$ where $c_i = \sum_{j=0}^{3} m_j \cdot i^j \bmod 7$, and the m_j are integers mod 7. You actually receive $(5, X, 2, 5, X, 6)$, where X means "missing". Reconstruct the original message (m_0, m_1, m_2, m_3). Justify your answer.

9. (12 pts.) ISBN checksums

An ISBN is a 10-digit number that serves as a serial number for books. The last digit is a checksum, which can be helpful for detecting data entry errors when typing in an ISBN. If the first nine digits are given by $x_1, \ldots, x_9$ (where $0 \le x_i \le 9$), the checksum digit x_{10} is given by

$$x_{10} = \bmod(x_1 + 2x_2 + \cdots + 8x_8 + 9x_9, 11).$$

(The checksum digit is in the range $0 \le x_{10} \le 10$. If the checksum digit is "10", the letter X is substituted when writing out an ISBN.) An equivalent way to describe the ISBN algorithm is like this: the checksum digit x_{10} is chosen so that the following equation is true:

$$10x_1 + 9x_2 + \cdots + 3x_8 + 2x_9 + x_{10} \equiv 0 \pmod{11}.$$

For instance, a sample ISBN is 0201896834; this has a valid checksum, since

$$10 \cdot 0 + 9 \cdot 2 + 8 \cdot 0 + 7 \cdot 1 + 6 \cdot 8 + 5 \cdot 9 + 4 \cdot 6 + 3 \cdot 8 + 2 \cdot 3 + 1 \cdot 4 = 176 \equiv 0 \pmod{11}.$$

For each of the following claims about this checksum algorithm, say whether the claim is true or false. Justify your answer.

(a) The ISBN checksum detects all single-digit errors (i.e., all errors where a single digit is entered incorrectly).

(b) The ISBN checksum detects all two-digit errors (i.e., all errors where a pair of digits, not necessarily adjacent, are entered incorrectly).

(c) The ISBN checksum detects all errors where a pair of adjacent digits are transposed (e.g., where we enter 0021896834 instead of 0201896834).

(d) The ISBN checksum detects all errors where any pair of digits (not necessarily adjacent) are transposed (e.g., where we enter 3201896804 instead of 0201896834).