CS 70 Discrete Mathematics and Probability Theory
Summer 2017 Hongling Lu, Vrettos Moulos, and Allen Tang
HW 7

Sundry

Before you start your homework, write down your team. Who else did you work with on this homework? List names and email addresses. (In case of homework party, you can also just describe the group.) How did you work on this homework? Working in groups of 3-5 will earn credit for your "Sundry" grade.

Please copy the following statement and sign next to it:

I certify that all solutions are entirely in my words and that I have not looked at another student's solutions. I have credited all external sources in this write up.

1 Product of Two

Suppose that $p > 2$ is a prime number and S is a set of numbers between 1 and $p-1$ such that $|S| > p/2$. Prove that any number $1 \leq x \leq p-1$ can be written as the product of two (not necessarily distinct) numbers in S, mod p.

2 Solution for $ax \equiv b \pmod{m}$

In the lecture notes, we proved that when $\gcd(m,a) = 1$, a has a unique multiplicative inverse, or equivalently $ax \equiv 1 \pmod{m}$ has exactly one solution x (modulo m). The proof of the unique multiplicative inverse actually proved that when $\gcd(m,a) = 1$, the solution of $ax \equiv b \pmod{m}$ with unknown variable x is unique. Now let's consider the case where $\gcd(m,a) > 1$ and see why there is no unique solution in this case. Let's consider the general solution of $ax \equiv b \pmod{m}$ with $\gcd(m,a) > 1$.

(a) Let $\gcd(m,a) = d$. Prove that $ax \equiv b \pmod{m}$ has a solution (that is, there exists an x that satisfies this equation) if and only if $b \equiv 0 \pmod{d}$.

(b) Let $\gcd(m,a) = d$. Assume $b \equiv 0 \pmod{d}$. Prove that $ax \equiv b \pmod{m}$ has exactly d solutions (modulo m).

(c) Solve for x: $77x \equiv 35 \pmod{42}$.

3 Fermat's Little Theorem

Fermat's Little Theorem states that for any prime p and any $a \in \{1,2,\ldots,p-1\}$, we have $a^{p-1} \equiv 1 \pmod{p}$. Without using induction, prove that $\forall n \in \mathbb{N}$, $n^7 - n$ is divisible by 42.

4 How Fun Is RSA?

Your study group is trying to send each other encrypted messages during the homework party to find out how much fun everyone is having with RSA.

To send a message, you first make an array where each item in the array is the index of the corresponding letter in the alphabet. For instance the message "ABC" turns into the array $[1,2,3]$. You use the number 27 to denote a space. After this, you use RSA to encrypt your message.

(a) Let's consider the case where your friends want to send you messages to see how much fun you are having with RSA. Come up with a public and a private key, and show how your friends could encrypt the message "RSA FUN" and send it to you using the public key. After this, show how you could decrypt the encrypted message using your private key. You should pick a different key than the one in part (b).

- Pick two primes p,q.
- Pick an appropriate exponent e. If you are having trouble finding an appropriate e, it might be a better idea to decide on an easy e such as 3, and change the primes you have picked. If you decide to do this, think about what form your primes would need to be if $e = 3$.
- Convert the original message into an array and encrypt individual items using the public key to find the encrypted message your friends sent you. What is the encrypted message?
- Find the appropriate private key.
- Now use the private key to decrypt the encrypted message you received. Were you able to recover the original message back?

(b) Now we will consider the case where you want to ask your friends how much fun they are having with RSA. Let's assume that you picked $p = 17$, $q = 19$, $e = 23$.

- How would you encrypt the message "RSA FUN", so your friends can use your public key to decrypt it? Show both the encryption and decryption using the values you have picked.

- A TA catches one of your friends' responses. The response was

  $$[295, 59, 96, 107, 252, 13, 295].$$

  Say in addition to knowing your public key, they also knew that one of the primes you have picked is 17. Show how they could use this to decrypt your friend's response. What is your friend's response?

5 Squared RSA

(a) Prove the identity $a^{p(p-1)} \equiv 1 \pmod{p^2}$, where a is relatively prime to p and p is prime.

(b) Now consider the RSA scheme: the public key is $(N = p^2q^2, e)$ for primes p and q, with e relatively prime to $p(p-1)q(q-1)$. The private key is $d = e^{-1} \pmod{p(p-1)q(q-1)}$. Prove that the scheme is correct, i.e. $x^{ed} \equiv x \pmod{N}$. You may assume that x is relatively prime to both p and q.

(c) Continuing the previous part, prove that the scheme is unbreakable, i.e. your scheme is at least as difficult as ordinary RSA.

6 RSA with Three Primes

Show how you can modify the RSA encryption method to work with three primes instead of two primes (i.e. $N = pqr$ where p, q, r are all prime), and prove the scheme you come up with works in the sense that $D(E(x)) \equiv x \pmod{N}$.

7 Lagrange? More like Lamegrange.

In this problem, we walk you through an alternative to Lagrange interpolation.

(a) Let's say we wanted to interpolate a polynomial through a single point, (x_0, y_0). What would be the polynomial that we would get? (This is not a trick question.)

(b) Call the polynomial from the previous part $f_0(x)$. Now say we wanted to define the polynomial $f_1(x)$ that passes through the points (x_0, y_0) and (x_1, y_1). If we write $f_1(x) = f_0(x) + a_1(x - x_0)$, what value of a_1 causes $f_1(x)$ to pass through the desired points?

(c) Now say we want a polynomial $f_2(x)$ that passes through (x_0, y_0), (x_1, y_1), and (x_2, y_2). If we write $f_2(x) = f_1(x) + a_2(x - x_0)(x - x_1)$, what value of a_2 gives us the desired polynomial?

(d) Suppose we have a polynomial $f_i(x)$ that passes through the points (x_0, y_0), ..., (x_i, y_i) and we want to find a polynomial $f_{i+1}(x)$ that passes through all those points and also (x_{i+1}, y_{i+1}). If we define $f_{i+1}(x) = f_i(x) + a_{i+1} \prod_{j=0}^{i}(x - x_j)$, what value must a_{i+1} take on?

(e) If we use this method on a set of points $\{(x_0, y_0), ..., (x_n, y_n)\}$, do we get the same polynomial as with Lagrange interpolation? Why or why not?

8 Green Eggs and Hamming

In this problem, we consider Hamming distances. The Hamming distance between two length-n bit strings b_1 and b_2 is defined to be the minimum number of bits in b_1 you need to flip in order to get b_2. For example, the Hamming distance between 101 and 001 is 1 (since you can just flip the first bit), while the Hamming distance between 111 and 000 is 3 (since you need to flip all three bits).

(a) Sam-I-Am has given you a list of n situations, and wants to know in which of them you would like green eggs and ham. You are planning on sending him your responses encoded in a length n bit string (where a 1 in position i says you would like green eggs and ham in situation i, while a 0 says you would not), but the channel you're sending your answers over is noisy and sometimes corrupts a bit. Sam-I-Am proposes the following solution: you send a length $n+1$ bit string, where the $(n+1)$st bit is the XOR of all the previous n bits (this extra bit is called the parity bit). If you use this strategy, what is the minimum Hamming distance between any two valid bit strings you might send?

(b) If the channel you are sending over becomes more noisy and corrupts two of your bits, can Sam-I-Am still detect the error? Why or why not?

(c) If you know your channel might corrupt up to k bits, what Hamming distance do you need between valid bit strings in order to be sure that Sam-I-Am can detect when there has been a corruption? You should also prove that your answer is tight–that is, prove that if you used a smaller Hamming distance, Sam-I-Am might not be able to detect when there was an error.