BIJECTIONS, FLT, RSA, POLYNOMIALS, SECRET SHARING 3

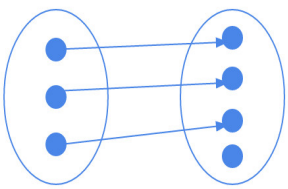
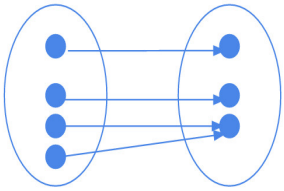
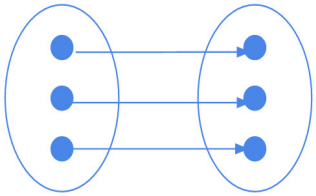
COMPUTER SCIENCE MENTORS 70

September 26 to September 30, 2016

1 Bijections

1.1 Introduction

1. Draw an example of each of the following situations

One to one AND NOT onto (injective but not surjective)	Onto AND NOT one to one (surjective but not injective)	One to one AND onto (bijection, i.e. injective AND surjective)
Solution: . 	**Solution:** . 	**Solution:** . 

2. Describe a function that is injective but not surjective and the set over which this applies. How about a function that is surjective but not injective?

Solution: ex: e^x: $R \to R$ is injective (one to one) but not surjective (onto) because while all real numbers map to something, nothing will map to 0 and negative

numbers. x2: x^2: $R \to R^+$ is surjective (onto) but not injective (one to one) because while all positive real numbers have something mapping to them, 4 has -2 and 2 mapping to it.

Note 1: Z_n denotes the integers mod n: $\{0, \ldots, n-1\}$
Note 2: in the following questions, the appropriate modulus is taken after applying the function

1.2 Questions

1. Are the following functions **bijections** from Z_{12} to Z_{12}?
   a. $f(x) = 7x$

   Solution: Yes: the mapping works, since 7 is coprime to 12, so there exists a multiplicative inverse to 7 in Z_{12} ($7x7 = 49 mod 12 = 1$, so $f^{-1}(x) = 7x$), which only occurs if the function is a bijection.

   b. $f(x) = 3x$

   Solution: No: $f(0) = f(4) = 0$.

   c. $f(x) = x - 6$

   Solution: Yes: can see its just $f(x) = x$, shifted by 6

2. Are the following functions **injections** from Z_{12} to Z_{24}?
   a. $f(x) = 2x$

   Solution: Yes: any two x_1 and x_2 will not equal each other as long as $x_1 \neq x_2$

   b. $f(x) = 6x$

   Solution: No: 0 and 4 both map to 0

   c. $f(x) = 2x + 4$

   Solution: Yes: same as 2x, except shifted

3. Are the following functions **surjections** from Z_{12} to Z_6? (Note: that $\lfloor x \rfloor$ is the floor operation on x)

a. $f(x) = \lfloor \frac{x}{2} \rfloor$

Solution: Yes: plug in every even number 0

b. $f(x) = x$

Solution: Yes: plug in 0 through 5

c. $f(x) = \lfloor \frac{x}{4} \rfloor$

Solution: No: the largest value we can get is f(12) which equals 3

4. Why can we not have a surjection from Z_{12} to Z_{24} or an injection from Z_{12} to Z_6?

Solution: Because there are more values in Z_{24} than Z_{12}, it is impossible to cover all the values in Z_{24} with mapping from Z_{12} . Similarly, because there are more values in Z_{12} than Z_6, there is not a unique element in Z_6 to assign to every Z_{12}.

2 Fermat's Little Theorem

2.1 Introduction

Fermat's Little Theorem: For any prime p and any $a \in \{1, 2, \ldots, p-1\}$, we have $a^{p-1} \equiv 1 \mod p$

1. Prove Fermat's Little Theorem.

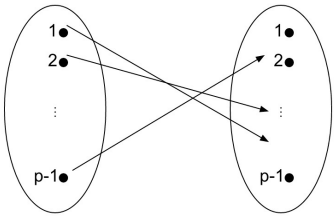
Solution: Proof from notes:
Claim: The function $a * x \mod p$ is a bijection where $x \in \{1, 2, \ldots, p-1\}$
The domain and range of the function are the same set, so it is enough to show that if $x \neq x'$ then $a * x \mod p \neq a * x' \mod p$.
Assume that $a * x \mod p \equiv a * x' \mod p$.
Since $gcd(a, p) = 1$, a must have an inverse: $a^{-1}(modp)$

$$ax \mod p \equiv ax \mod p$$

$$a^{-1} * a * x \mod p \equiv a^{-1} * a * x' \mod p$$

$$x \mod p \equiv x' \mod p$$

This contradicts our assumption that $x \neq x' \mod p$. Therefore f is a bijection. We want to use the above claim to show that $a^{p-1} \equiv 1 \mod p$. Note that now we have the following picture:



So if we multiply all elements in the domain together this should equal the product of all the elements in the image:

$$1 * 2 * \ldots * (p-1) \mod p \equiv (1a) * (2a) * \ldots * ((p-1)a) \mod p$$

$$(p-1)! \mod p \equiv a^{p-1} * (p-1)! \mod p$$

$$1 \equiv a^{p-1} \mod p$$

2.2 Questions

1. Find $3^{5000} \mod 11$

Solution:

$$(3^{10})^{500} \mod 11 = 1^{500} \mod 11 = 1$$

2. Find $2^{20} + 3^{30} + 4^{40} + 5^{50} + 6^{60} \mod 7$

Solution: By FLT:

$$2^6 \equiv 1 \mod 7$$
$$3^6 \equiv 1 \mod 7$$
$$4^6 \equiv 1 \mod 7$$
$$5^6 \equiv 1 \mod 7$$
$$6^6 \equiv 1 \mod 7$$

Apply the above facts to simplify each portion of the equation:

$$2^{20} = 2^2 * (2^6)^3 \to 2^{20} \mod 7 \equiv 2^2 \mod 7 \equiv 4 \mod 7$$
$$3^{30} = (3^6)^5 \to 3^{30} \mod 7 \equiv 1 \mod 7$$
$$4^{40} = 4^4 * (4^6)^6 \to 4^{40} \mod 7 \equiv 4^4 \mod 7 \equiv 4 \mod 7$$
$$5^{50} = 5^2 * (5^6)^8 \to 5^{50} \mod 7 \equiv 5^2 \mod 7 \equiv 4 \mod 7$$
$$6^{60} = (6^6)^{10} \to 6^{60} \mod 7 \equiv 1 \mod 7$$
$$2^{20} + 3^{30} + 4^{40} + 5^{50} + 6^{60} \mod 7 \equiv 4 + 1 + 4 + 4 + 1 \mod 7$$
$$\equiv 14 \mod 7 \equiv 0 \mod 7$$

3. Show that $n^7 - n$ is divisible by 42 for any integer n

Solution: $42 = 7 * 3 * 2 \leftarrow$these factors are prime so lets apply FLT!!

$$n^7 \equiv n \mod 7$$
$$n^3 \equiv n \mod 3$$
$$n^2 \equiv n \mod 2$$

Were interested in n^7 so lets modify the bottom two equations to write n^7 in mod 3 and mod 2

$$n^7 \equiv n^3 * n^3 * n \equiv n * n * n \equiv n^3 \equiv n \mod 3$$
$$n^7 \equiv n \mod 3$$

$$n^7 \equiv n^2 * n^2 * n^2 * n \equiv n * n * n * n \equiv n^2 * n^2 \equiv n * n \equiv n^2 \equiv n \mod 2$$
$$n^7 \equiv n \mod 2$$

$$n^7 \equiv n \mod 7$$
$$n^7 \equiv n \mod 3$$
$$n^7 \equiv n \mod 2$$

Wouldnt it be great if the above equations implied that $n7 \equiv n \mod 7 * 3 * 2$?
Lets try to prove that.
Claim: If

$$x \equiv y \mod a_1$$
$$x \equiv y \mod a_2$$
$$\dots$$
$$x \equiv y \mod a_n$$

are true and $a_1, \dots, a_n$ are coprime then $x \equiv y \mod a_1 a_2 \dots a_n)$
$x \equiv y \mod a_i \rightarrow x = y + c_i * a_i$ for some constant c_i

$$x = y + c_1 * a_1$$
$$x = y + c_2 * a_2$$
$$\dots$$
$$x = y + c_n * a_n$$

But this implies that $x = c * lcm(a_1, \dots, a_n) + y$
Since $a_1, \dots, a_n$ are coprime, $lcm(a_1, \dots, a_n) = a_1 * a_2 * \dots * a_n$
So we get $x = c * a_1 * a_2 * \dots * a_n + y$
Therefore $x \equiv y \mod a_1 * a_2 * \dots * a_n$
We can now say that $n^7 \equiv n \mod 7 * 3 * 2 \equiv n \mod 42$.

3 RSA

3.1 Questions

1. **How does RSA work?**

   a. Alice wants to send Bob a message $m = 5$ using his public key ($n = 26$, $e = 11$). What cipher text E(m) will Alice send?

   Solution:

$$\begin{aligned} 5^1 &= 5 \mod 26 \\ 5^2 &= 5 \mod 26 \\ &= -1 \mod 26 \\ 5^4 &= (-1)^2 \mod 26 \\ &= 1 \mod 26 \\ 5^8 &= 1 \mod 26 \\ 5^{11} &= 5^8 * 5^2 * 5^1 \mod 26 \\ &= 1 * -1 * 5 \mod 26 \\ &= -5 \mod 26 \\ &= 21 \mod 26 \end{aligned}$$

   b. What is the value of d (Bobs private key) in this scheme? Note that traditional RSA schemes use much larger prime numbers, so its harder to break n down into its prime factors than it is in this problem.

   Solution: $n = 26 \rightarrow$ because $26 = pq$ and $p \neq a * q$ for all a within integers, $p = 13$, $q = 2$

$$\begin{aligned} d &= e^{-1} \mod (13-1)(2-1) \\ d &= 11^{-1} \mod 12 \\ d &= 11 \end{aligned}$$

2. In RSA, if Alice wants to send a confidential message to Bob, she uses Bobs public key to encode it. Then Bob uses his private key to decode the message. Suppose that Bob chose $N = 77$. And then Bob chose $e = 3$ so his public key is (3, 77). And then Bob chose $d = 26$ so his private key is (26, 77).
   Will this work for encoding and decoding messages? If not, where did Bob first go wrong in the above sequence of steps and what is the consequence of that error? If it does work, then show that it works.

   Solution: e should be co-prime to $(p-1)(q-1)$.
   $e = 3$ is not co-prime to $(7-1)(11-1) = 60$, so this is incorrect, since therefore e does not have an inverse $\mod 60$.

3. **Coin tosses over text messages**
   You and one of your friends want to get your hands on the new gadget thats coming out. One of you has to wait in line overnight so that you have a chance to get the gadgets while they last. In order to decide who this person should be, you both agree to toss a coin. But you wont meet each other until the day of the actual sale and you have to settle this coin toss over text messages (using your old gadgets). Obviously neither of you trusts the other person to simply do the coin toss and report the results.

   How can you use RSA to help fix the problem?

   Solution: If there was a way for me to make my choice (i.e. toss the coin) without revealing to my friend what the result was before s/he makes her/his decision, then we would be in good shape. RSA enables us to do just that. One can commit to a choice without revealing what that choice really is. So here is how we proceed:

   1. First I select a public key (N,e) and a private key d. I toss a coin, but instead of sending the result to my friend, I first encrypt it using the public key (N,e). Then I send my friend the public key along with the encrypted message.
   2. My friend is supposedly (read the next part for why the word supposedly is used) unable to see what the result of the coin toss was and therefore cannot cheat. So s/he makes her/his choice (what HEADS and TAILS mean) and sends it to me.
   3. Once I have successfully received the result, I reveal the result of the coin toss by sending my friend the result in plain text (i.e. with no encryption). My friend can now verify that I have not cheated (i.e. I have not changed the result) by encrypting the result using the public key I have given her/him and making sure it was the same as the encrypted message I send her/him. Note that RSA encryption and decryption are both bijections, therefore if I

know the encrypted version of two messages are the same, then those two messages must be the same.

Note that I cannot cheat here, because I commit to the result of the coin toss before I know my friends choice. Commitment is a very useful primitive (used in many places in cryptography) that enables a party to convincingly commit to a choice without revealing it until they choose to reveal it. The party should not be able to change their mind after the commitment which is what the scheme guarantees.

4 Polynomials

4.1 Introduction

1. If polynomial P(x) has degree n - 1 then we can uniquely reconstruct it from any n distinct points.
2. If a polynomial P(x) has degree n - 1 then it can be uniquely described by its n coefficients

4.2 Questions

1. Define the sequence of polynomials by $P_0(x) = x+12$, $P_1(x) = x^2-5x+5$ and $P_n(x) = xP_{n-2}(x) - P_{n-1}(x)$. (For instance, $P_2(x) = 17x - 5$ and $P_3(x) = x^3 - 5x^2 - 12x + 5$.)

   (a) Show that $P_n(7) \equiv 0 \pmod{19}$ for every $n \in N$.

   Solution:

   (a) Prove using strong induction.

   Base Case There are two base cases because each polynomial is defined in terms of the two previous ones except for P_0 and P_1.

   $$P_0(7) \equiv 7 + 12 \equiv 19 \equiv 0 \pmod{19}$$
   $$P_1(7) \equiv 7^2 - 5 \cdot 7 + 5 \equiv 49 - 35 + 5 \equiv 19 \equiv 0 \pmod{19}$$

   Inductive Hypothesis Assume $P_n(7) \equiv 0 \pmod{19}$ for every $n \leq k$.

   Inductive Step Using the definition of P_{k+1}, we have that

   $$\begin{aligned} P_{k+1}(7) &\equiv xP_{k-1}(7) - P_k(7) \pmod{19} \\ &\equiv x \cdot 0 - 0 \pmod{19} \\ &\equiv 0 \pmod{19} \end{aligned}$$

   Therefore, $P_n(7) \equiv 0 \pmod{19}$ for all natural numbers n.

   (b) Show that, for every prime q, if $P_{2013}(x) \not\equiv 0 \pmod{q}$, then $P_{2013}(x)$ has at most 2013 roots modulo q.

Solution: This question asks to prove that, for all prime numbers q, if $P_{2013}(x)$ is a non-zero polynomial $\pmod q$, then $P_{2013}(x)$ has at most 2013 roots $\pmod q$.

The proof of Property 1 of polynomials (a polynomial of degree d can have at most d roots) still works in the finite field $GF(q)$. Therefore we need only show that P_{2013} has degree at most 2013. We prove that $\deg(P_n) \leq n$ for $n > 1$ by strong induction.

Base cases There are 4:

$$\begin{aligned}
\deg(P_0) &= \deg(x + 12) = 1 \\
\deg(P_1) &= \deg(x^2 - 5x + 5) = 2 \\
\deg(P_2) &= \deg(xP_0(x) - P_1(x)) \leq 2 \\
\deg(P_3) &= \deg(xP_1(x) - P_2(x)) \leq 3
\end{aligned}$$

Inductive Hypothesis Assume $\deg(P_n) \leq n$ for all $2 \leq n \leq k$.

Inductive Step Then

$$\begin{aligned}
&\deg(P_{k+1}(x)) \\
&\qquad \leq \max\{\deg(xP_{k-1}(x)), \deg(P_k(x))\} \\
&\qquad = \max\{1 + \deg(P_{k-1}(x)), \deg(P_k(x))\} \\
&\qquad \leq \max\{1 + k - 1, k\} \\
&\qquad \leq k \\
&\qquad \leq k + 1
\end{aligned}$$

5 Secret Sharing

5.1 Questions

1. Suppose the Oral Exam questions are created by 2 TAs and 3 Readers. The answers are all encrypted and we know that:
   (a) Both TAs should be able to access the answers
   (b) All 3 Readers can also access the answers
   (c) One TA and one Reader should also be able to do the same

   Design a secret sharing scheme to make this work.

Solution: Use a 2 degree polynomial which requires at least 3 shares to recover the polynomial. Generate a total of 7 shares, give each Reader a share, and each TA 2 shares. Then, all possible combinations will have at least 3 shares to recover the answer key. Basically the point of this problem is to assign different weights to different classes of people. If we give one share to everyone, then 2 Readers can also recover the secret and the scheme is broken.

2. An officer stored an important letter in her safe. In case she is killed in battle, she decides to share the password with her troops. Everyone knows there are 3 spies among the troops, but no one knows who they are except for the three spies themselves.The 3 spies can coordinate with each other and they will either lie and make people not able to open the safe, or will open the safe themselves if they can. Therefore, the officer would like a scheme to share the password that satisfies the following conditions:

   1. When M of them get together, they are guaranteed to be able to open the safe even if they have spies among them.
   2. The 3 spies must not be able to open the safe all by themselves.

   Please help the officer to design a scheme to share her password. What is the scheme? What is the smallest M? Show your work and argue why your scheme works and any smaller M couldnt work.

   Solution: The key insight is to realize that both polynomial-based secret-sharing and polynomial-based error correction work on the basis of evaluating an underlying polynomial at many points and then trying to recover that polynomial. Hence they can be easily combined. Suppose the password is s. The officer can construct a polynomial $P(x)$ such that $s = P(0)$ and share $(i, P(i))$ to the i-th person in her troops. Then the problem is: what should the degree of $P(x)$ be and what is the smallest M? First, the degree of polynomial d should not be less than 3. It is because when $d < 3$, the 3 spies can decide the polynomial $P(x)$ uniquely. Thus, n will be at least 4 symbols. Lets choose a polynomial $P(x)$ of degree 3 such that $s = P(0)$. We now view the 3 spies as 3 general errors. Then the smallest $M = 10$ since n is at least 4 symbols and we have $k = 3$ general errors, leading us to a codeword of $4 + 2 \cdot 3 = 10$ symbols (or people in our case). Even though the 3 spies are among the 10 people and try to lie on their numbers, the 10 people can still be able to correct the $k = 3$ general errors by the Berlekamp-Welch algorithm and find the correct $P(x)$.

   Alternative solution: Another valid approach is making $P(x)$ of degree $M1$ and adding 6 public points to deal with 3 general errors from the spies. In other words, in addition to their own point $(i, P(i))$, everyone also knows the values of 6 more points, $(t+1, P(t+1)), (t+2, P(t+2)), \ldots, (t+6, P(t+6))$, where t is the number of the troops. The spies have access to total of $3 + 6 = 9$ points so the degree $M - 1$ must be at least 9 to prevent the spies from opening the safe by themselves. Therefore, the minimum M is 10.