BIJECTIONS, FLT, CRT, RSA 3

COMPUTER SCIENCE MENTORS 70

February 12 to February 17, 2017

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)

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?

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. $f(x) = 7x$

   b. $f(x) = 3x$

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

2. Are the following functions **injections** from Z_{12} to Z_{24}?

   a. $f(x) = 2x$

   b. $f(x) = 6x$

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

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

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

   b. $f(x) = x$

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

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

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.

2.2 Questions

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

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

3 Chinese Remainder Theorem

3.1 Introduction

Chinese Remainder Theorem: The Chinese Remainder theorem says that a sequence of remainders with pairwise coprime divisors defines a unique remainder modulo the product of those divisors. Formally, if x can be expressed as

$$x \equiv a_1 (\mod m_1)$$
$$x \equiv a_2 (\mod m_2)$$

where m_1 and m_2 are relatively prime to each other, CRT tells us that there is an unique number mod $m_1 m_2$ that satisfies this equation.
In simple cases, we can often use extended Euclid's algorithm in simple cases to find x. However, a failsafe equation is given by:
$x = \sum_{i=1}^{k} a_i b_i \mod N$, where b_i are defined as $\left(\frac{N}{n_i}\right)\left(\frac{N}{n_i}\right)^{-1}_{\mod n_i}$ and $N = n_1 \cdot n_2 ... \cdot n_k$.

3.2 Questions

1. Find an integer x such that x is congruent to $3 \mod 4$ and $5 \mod 9$.

2. The supermarket has a lot of eggs, but the manager is not sure exactly how many he has. When he splits the eggs into groups of 5, there are exactly 3 left. When he splits the eggs into groups of 11, there are 6 left. What is the minimum number of eggs at the supermarket?

4 RSA

4.1 Introduction

RSA: Given two large primes, p and q, and a message x,

$$\begin{aligned} N &= pq \\ E(x) &= x^e \text{mod} N \\ D(x) &= x^d \text{mod} N \end{aligned}$$

where e is relatively prime to $(p-1)(q-1)$, and $ed = 1$. The pair (N, e) is the recipient's **public key**, and d is the recipient's **private key**. The sender sends $E(x)$ to the recipient, and the recipient uses $D(x)$ to recover the original message.

4.2 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?

   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.

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.

5 Extra RSA Fun! :-)

1. **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?