What RSAying?

Last class, we examined Caesar ciphers and more general mono-alphabetic ciphers. Caesar ciphers proved to be relatively insecure: a simple iteration through the possible shifts was enough to crack them. General mono-alphabetic ciphers proved to be a lot more resilient: we had to use frequency analysis and some knowledge about English letters to crack them. Today, we'll create a particular kind of mono-alphabetic cipher using some neat properties of modular arithmetic!

Abstract Algebra Refresher

Problem 1. *What is the multiplicative inverse of a number?*

Problem 2. *Find the multiplicative inverses of:*

(i) *2*

(ii) *100*

(iii) *1/7*

Problem 3. *What happens when you multiply a number by its multiplicative inverse? Does every number have a unique multiplicative inverse?*

Hopefully your answers were accompanied by a big 'IF'. Recall that abstract ideas of addition and multiplication can apply to systems beyond just the real numbers! Some axioms hold for certain number systems, while others do not. For example, we previously defined modular arithmetic as:

Definition 1. *For any integer x, we let $(x \mod n)$ equal the unique integer in $0, 1, 2, \ldots, n-1$ that is congruent to x modulo n. In other words, if you divide x by n, then we define $(x \mod n)$ to be the positive integer remainder (if the remainder is negative, simply add n to it).*

It holds true for any two integers x, y that

$$(x + y \mod n) = \big((x \mod n) + (y \mod n) \mod n\big)$$

and that

$$(x \cdot y \mod n) = \big((x \mod n) \cdot (y \mod n) \mod n\big).$$

Problem 4. *Fill out the following table for multiplication in $\mathbb{Z}_{10}$. For each cell, you compute the row label times the column label.*

$\cdot_{10}$	0	1	2	3	4	5	6	7	8	9
0										
1										
2										
3										
4										
5										
6										
7										
8										
9										

Problem 5. *Using the table, find the multiplicative inverses of the following numbers in $\mathbb{Z}_{10}$, or explain why no inverse exists.*

(i) *1*

(i) *3*

(i) *5*

(i) *9*

Problem 6. *Does every number in $\mathbb{Z}_{10}$ have a multiplicative inverse? If so, write down the inverse of every number. If not, describe a pattern for numbers that have inverses.*

In a Caesar cipher, the alphabet is shifted a certain number of places to encode a message. Numbers that get shifted past Z wrap around to A.

Problem 7. *Explain how you might use modular arithmetic to describe a Caesar cipher. (Hint: If we're encrypting the p^{th} letter of the alphabet with a shift of k, where along the alphabet will the shifted letter end up?)*

Improving the Caesar Cipher

Last class, we showed that encrypting messages with Caesar ciphers isn't particularly safe. Let's see if we can improve on the work of the Romans. Instead of shifting/adding k to each letter of the alphabet, what happens if we multiply by a number p?

Problem 8. *Let's first examine how a multiplicative scheme would apply to digits. Fill in the chart below by multiplying each digit by 7 (mod 10).*

d: Unencrypted Number	e: Encrypted Number (Multiplying by 7 in Modulo 10)
0	$0 \times 7 = 0 \pmod{10}$
1	
2	
3	
4	
5	
6	
7	
8	
9	

Problem 9. *For the sake of encryption, do you think that 7 is a good choice for p? Why or why not?*

Problem 10. *Let's see what happens when we encrypt digits with a different choice of p. Fill in the chart below by multiplying each digit by 5 (mod 10).*

d: Unencrypted Number	e: Encrypted Number (Multiplying by 5 in Modulo 10)
0	$0 \times 5 = 0 \pmod{10}$
1	
2	
3	
4	
5	
6	
7	
8	
9	

Problem 11. *Do you think that 5 is a good choice for p? Why or why not?*

Problem 12. *Do you notice a pattern for which numbers make good choices for p? How does this relate to our review of multiplicative inverses?*

As you can see, not all numbers make good choices for p. Let's call the numbers that encode unencrypted numbers as *unique* encrypted numbers **multiplicative encoders**.

Problem 13. *Is $p = 1$ a multiplicative encoder? If so, would you use it to encrypt a message? Why or why not?*

Problem 14. *List all the multiplicative encoders in $\mathbb{Z}_{10}$. For each encoder p, find gcd(p, 10), i.e., the greatest common divisor of p and 10.*

Problem 15. *Let's extend the range of our encryption scheme to the alphabet, in $\mathbb{Z}_{26}$. List all the multiplicative encoders that we could use to encrypt messages containing just alphabetical letters.*

Definition 2. *Two numbers p and m are called* ***co-prime*** *if the greatest common divisor of p and m is 1.*

Proposition 1. *If p and m are co-prime, then there exists $q < m$ such that $p \times q = 1 \mod m$.*

In other words, if we choose our multiplicative encoder, p, such that p is co-prime to our modulus m, we are guaranteed to have an inverse for p. Let's denote this inverse q, and see how it relates to decryption.

Problem 16. *Let's revisit encrypting digits with $p = 7 \pmod{10}$.*

(i) *Find q, the multiplicative inverse of p.*

(ii) *Encrypt the digits in the table by multiplying by p. Afterward, decrypt your encrypted digits by multiplying by q.*

Unencrypted Number	Encrypted Number (Multiplying the Unencrypted Number by p in Modulo 10)	Decrypted Number (Multiplying the Encrypted Number by q in Modulo 10)
0	0xp=0 (mod 10)	0xq=0 (mod 10)
1		
2		
3		
4		
5		
6		
7		
8		
9		

(iii) *Were you able to successfully encrypt and decrypt all your digits? Explain why or why not.*

Simplified RSA

When we first studied abstract algebra, you may have wondered about practical applications for studying abstract number systems. One incredibly important example is the **RSA algorithm**. RSA is an encryption scheme based on the idea of using multiplicative inverses to safely transmit information.

Since RSA in its full implementation is already complicated enough for us instructors, we've simplified some of its ideas into a more approachable cipher. Let's call this simplifed RSA, and explore how it works!

Problem 17. *Suppose our entire class agrees to use $p = 11$ as our multiplicative encoder, with modulus $m = 50$.*

(i) *Come up with a message that you want to securely send to a partner. Convert the message into a numerical form, by replacing letters with their positions in the alphabet.*

(ii) *Encrypt your numerical message by multiplying each number by $p = 11$ (mod $m = 50$).*

(iii) *Send your partner your encrypted message. Since there are fewer than 50 letters in the alphabet, you can simply send the encrypted message in numerical form.*

(iv) *Decrypt your partner's message. (Hint: the multiplicative inverse of 11 (mod 50) is 41.)*

As you've seen, when we work with large numbers for p and m, even if we know what the multiplicative encoder is, finding the right decoder can be difficult. But what if someone was determined enough to try a bunch of different multiplicative decoders?

Problem 18. *Suppose a nosy instructor knew that you and your partner were communicating in modulo 50, but not what you chose as your multiplicative encoder. How many different decoders would they have to try before they are guaranteed to decode your message?*

Problem 19. *Suppose instead that you and your partner agreed on your modulus, m, in private. Without knowing your communication modulus, how could an instructor go about cracking your cipher?*

Problem 20. 🌶️ *A paranoid student might point out that to communicate with their partner, they have to exchange both m and p. Anyone listening in on their communication can encrypt and decrypt their messages with the same degree of difficulty. Can you devise a way to exchange m and p such that a nosy instructor can't intercept your messages?*