Asynchronous Summer 2025 Math 673 (Sec. 699+700): Mathematical Foundation of Cryptography

HOMEWORK #1
Due 1PM, Saturday, May 31, 2025

0. Interview: Please schedule a time so we can meet and you can tell me a little about yourself: Your background and what you hope to gain from learning crypto. This should take just a few minutes, and will give you a chance to get familiar with our Zoom link. (We can do the interviews up until Monday, June 2, but please contact me ASAP to schedule.)

1. Reading:

- First 10 pages of Rogaway's essay on the moral character of cryptographic work
- Mira Bernstein's essay on mathematical proof
- Sections 1–6 of Ch. 1 of [HPS14] (our suggested textbook)
- Sections 1–3 of Ch. 2 of [HPS14]

The first two readings can be downloaded respectively from:
`https://people.tamu.edu/~rojas/bndemo.pdf`
`https://people.tamu.edu/~rojas/rogamo.pdf`

2. Written problems: Please solve Exercises 7 and 11 below, and then make a pdf of your write-up of the solutions and submit your pdf via Gradescope. (The Gradescope link will be available starting Friday evening, May 30.) *I highly recommend that you work out the other problems as well, because I will assume you know how to do them. If you have any trouble, you can of course ask me for hints.*

Exercises

1: Just as, over the rationals, $\frac{1}{2}$ is the unique number you multiply 2 by to get 1, fractions can be defined in $\mathbb{Z}/(N)$... sometimes. For instance, $\frac{1}{2} = 3$ mod 5 because

$$3 \cdot 2 = 6 = 1 \bmod 5$$

and no other integer in $\{0,1,2,3,4\}$ times 2 will get you 1. It looks weird, but it actually makes perfect sense mathematically!
(a) Please find $\frac{1}{7}$ mod 11. (This one you could almost just do in your head, or do a brute-force search.)
(b) Please find $\frac{1}{7}$ mod 101. (You could do a brute-force search but it is smarter to use the *Extended Euclidean Algorithm*, which is covered in the second video for Module #1.)
(c) Please prove that $\frac{1}{2}$ is *undefined* mod 6, i.e., there is *no* number in $\{0,1,2,3,4,5\}$ that will get you 1 if you multiply by 2.
(d) Please prove that $\frac{1}{2}$ is undefined mod N when N is even. **Hint:** If you just use the definition of congruence mod N, and basic properties of even numbers, you can prove this in under 5 lines.

2: It turns out that an integer (written as usual in base-10) is divisible by 3 if and only if its sum of digits is divisible by 3. (Also, you can use this recursively: For instance, 59388 is divisible by 3 because $5+9+3+8+8=33$ and $3+3=6$.) Please prove that this trick works. **Hint:** Note that $10=1$ mod 3 and observe that any integer written in base-10 is simply a sum of powers of 10. The proof is then short, and just a matter of setting notation correctly.

3: (a) Please write a program (in any language of your choice) to convert symbols into numbers as follows:

space	apostrophe	,	dash	.	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42

e.g., you program would take an input like `7,0a` and output the array `[14,4,7,17]`.
Your answer should *just be the source code*, but you must check yourself that the program works!
(b) Please write a program (in any language of your choice) to convert numbers into symbols as follows:

2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
space	apostrophe	,	dash	.	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

e.g., your program would take an input like `[14,4,7,17]` and output `7,0a` .
Your answer should *just be the source code*, but you must check yourself that the program works!.
Note. Let A denote the set whose elements are the symbols in the top row of the table in Part (a), and let B denote the set whose elements are the numbers in the bottom row of the table in Part (a). In more abstract notation, a sequence of n elements of A is just an element of A^n — the $n^{\underline{\text{th}}}$ *Cartesian power of A*. The collection of all finite (or empty) sequences of elements of A is usually called A^*. So, in fancier notation, Part (a) asks you to implement a function $f : A^* \longrightarrow B^*$ and Part (b) ask you to implement a function $g : B^* \longrightarrow A^*$.

4: Consider the following variant of the Caesar cipher: Take a plain-text message, convert it to numbers as in Exercise 3, and then multiply all the numbers by a key k mod 43. In particular, a poet sent the following encrypted message using the preceding scheme:
39 26 12 38 30 11 12 9 16 38 16 22 35 22 33 38 26 12 14 20 38 22 38 6 1 2 12 20 39 28 38 39 26 12 38 39 10 30 9 16 38 16 16 22 35 22 33 38 26 12 14 20 38 22 38 41 12 22 20 39 28 38 22 11 36 38 2 38 10 2 16 16 38 41 12 39 38 22 38 20 2 16 40 38 6 22 21 22 35 22 38 39 26 12 1 12 38 2 20 11 14 39 38 22 11 5 38 39 26 1 12 12 9 16 38 16 16 16 22 35 22 28
(a) What kind of message would you get if you tried to directly convert the preceding numbers to symbols using Exercise 3(b)? (You should get nonsense, since you are not decrypting yet...)
(b) Find the key k that was used to send the message. **Hint:** You could just use brute-force, since the key space is small. Also, brute-force search is very easy if you know enough to write the programs for Exercise 3.
(c) Who was the poet that wrote the encrypted poem?

5: Patrick is using an affine cipher to send messages to Spongebob in a boring crypto class. He ends each message with `hah` (before encrypting). If you intercept his cipher-text and see that it's $(17, 5, 0, 29, 33, 5, 0, 20, 16, 19, 19, 5, 5, 4, 35, 4)$, what was his affine map? What was his message? (You may assume he converted letters to numbers as in Exercise #3, before he applied his affine cipher.) **Hint:** If you assume his map was $f(x) = \alpha x + \beta$, the last 2 numbers give you an easy way to solve for α and β mod 43.

6: Please find a language, or computer algebra system, that you can code in *and* use to correctly do arithmetic on integers with hundreds of digits. (For instance, computing the full decimal expansion of 2^{51023} should be possible and easy for the system you ultimately use.) In particular, tell me what system you'll be using in your upcoming calculations.
Note: You should be aware that `Matlab` is an inadequate solution, unless you have access to their symbolic package. And even then, `Maple` is preferable since `Maple` does arbitrary

precision arithmetic from the get-go and is free to access via voal.tamu.edu .

7: Please clearly show how to compute 2^{144} mod 101 (without applying Fermat's Little Theorem or Euler's Theorem, if you already know them), by hand, using no more than 8 multiplications in $\mathbb{Z}/(101)$. **Hint:** It will turn out to be helpful to know the base-2 expansion of 144. Also, you might want to warm up with counting how many *squarings* you need to compute 2^2, 2^4, and 2^8...
REMINDER!: REDUCE MOD 101 ALONG THE WAY! Otherwise, you will get massive numbers that waste your time!

8: Consider the function $f : (\mathbb{Z}/(43)) \longrightarrow (\mathbb{Z}/(43))$ defined by $f(x) := 7x + 3$. (Such a function is an example of a method of encryption called an *affine cipher*, combining the simpler ideas of adding or multiplying mod n.) For the example at hand, what function g would yield $g(f(x)) = x$ for all $x \in \{0, \ldots, 42\}$? **Hint:** If you were over the rational numbers then this would be very easy. Perhaps you can do the same algebra over $\mathbb{Z}/(43)$, since you know fractions can sometimes be well-defined in $\mathbb{Z}/(N)$?

9: Solve the linear system

$$2x - 3y = 7$$

$$3x + 7y = 2$$

over $\mathbb{Z}/(43)$. **Hint:** If you are careful, you'll only need to use division mod 43 once.

10: Suppose m and n are positive integers, with m even. Please prove that $\gcd(mn - 1, mn + 1) = 1$.

11: Given $a \in \mathbb{Z}/(n)$ with $\gcd(a, n) = 1$, the *order of a in $\mathbb{Z}/(n)$* is the *least* positive k such that $a^k = 1$ mod n.
(a) Please find the order of 4 mod 11.
(b) Please find the order of 5 mod 12.
(c) Please find the order of 2 mod 101.
(d) Please describe how the powers of 2 eventually repeat mod 100. Is there any power of 2 equal to 1 mod 100?
Note: It's OK if you write a short loop to take care of the calculations for Part (c) and (d). But *you* should be the one writing the code!

12: In this problem, you'll consider the amount of work it takes to evaluate
$\alpha := \left(3^{(2^{2048}+1)} \text{ mod } 101\right)$.
(a) Please accurately estimate how many multiplications mod 101 you would have to do if you simply computed α naively.
(b) If you use the binary method (mentioned during the first two lectures) to compute α, please accurately estimate how many multiplications mod 101 you would have to do.
(c) Please estimate how many multiplications mod 101 you would have to do to compute α if you applied your solution to Exercise 11(d) above, along with the binary method.
(d) Compute α by hand. **Note:** I expect you to use the most efficient method you can, *without* a computer or calculator.