MATH 4440/5440 Assessment, Module 2 (Discrete Log Problem, More Modular Arithmetic)

Katherine Stange, CU Boulder, Fall 2020

Honor Code Rules

Assessments are open book, but are to be completed on your own without collaboration. To be specific, you may use your course notes, textbook, course website resources, course videos. You may not use the internet beyond the course websites. You may not ask anyone else for help (except your professor), including other humans, or posting/entering your question or any part of it into the internet. You may not share the questions or answers with anyone else. You may not use calculators (even from the course websites) unless explicitly permitted in the question.

Have you read, understood, and followed the honor code rules above?

$$YES \quad / \quad NO$$

Some instructions on formatting.

You may use the accompanying LaTeX source document to produce LaTeX'ed answers. You may typeset answers separately. You may print the pages and solve the questions on them by hand. You may handwrite answers on separate sheets. You may upload PDF or image files (JPG or PNG). No matter what you do, just make sure it is clearly and easily legible before you upload it to canvas.

1 Question 1

(10 points) Compute the following expression by hand. Please show your work neatly and **show all steps**.

$$63002^{72048} \pmod{63}.$$

Solution. We know we can reduce the base modulo 63 and the exponent modulo $\phi(63)$.

$$\phi(63) = \phi(9 \cdot 7) = 6 \cdot 6 = 36.$$

Therefore,

$$63002^{72048} \pmod{63}. = 2^{12} \pmod{63}.$$

This is now a manageble problem via successive squaring.

$$2^1 \equiv 2, 2^2 \equiv 4, 2^4 \equiv 16, 2^8 \equiv 256 \equiv 4,$$

The result is therefore

$$2^{12} \equiv 2^8 \cdot 2^4 \equiv 4 \cdot 16 \equiv 64 \equiv 1.$$

2 Question 2

(10 points) Compute the inverse of 30 modulo 97 by hand. Note that 97 is prime. Show your work neatly please, and **show all steps**.

Solution.

We use Gauss' method.

First, divide 97 by 30:

$$97 = 3 \cdot 30 + 7$$

We learn that

$$30 \cdot (-3) \equiv 7 \pmod{97}.$$

Then, divide 97 by 7:

$$97 = 13 \cdot 7 + 6$$

Adding a copy of 7, we obtain

$$97 = 14 \cdot 7 - 1$$

We learn that

$$7 \cdot 14 \equiv 1 \pmod{97}.$$

Taking our two "learned facts" together gives

$$30 \cdot (-3) \cdot 14 \equiv 1 \pmod{97}.$$

Therefore the inverse of 30 is

$$-3 \cdot 14 \equiv -42 \equiv 55 \pmod{97}.$$

3 Question 3

(10 points) You wish to solve a discrete logarithm problem $h \equiv g^x \pmod{137}$. Note that 137 is prime.

You know the following facts:

$$\begin{aligned} g^{125} &\equiv 12 \pmod{137}, \\ g^{59} &\equiv 6 \pmod{137}, \\ g^{78} &\equiv 30 \pmod{137}, \\ h^2 g^3 &\equiv 60 h^3 g \pmod{137}. \end{aligned}$$

What is $x = L_g(h)$? Please compute by hand, be neat and **show all steps**.

Solution.

Write $X_2 = L_g(2)$, $X_3 = L_g(3)$, $X_5 = L_g(5)$. Then we can write the facts above as

$$\begin{aligned} 125 &\equiv 2X_2 + X_3 \pmod{136} \\ 59 &\equiv X_2 + X_3 \pmod{136} \\ 78 &\equiv X_2 + X_3 + X_5 \pmod{136} \\ 2x + 3 &\equiv 2X_2 + X_3 + X_5 + 3x + 1 \pmod{136} \end{aligned}$$

We need to solve this system of equations. From the first two equations, $X_2 \equiv 125 - 59 \equiv 66$. From the second and third, $X_5 \equiv 78 - 59 \equiv 19$. From the second, using $X_2 \equiv 66$, $X_3 \equiv 59 - 66 \equiv -7$.

Plugging all that into the last, we have

$$2x + 3 \equiv -4 - 7 + 19 + 3x + 1$$

which simplifies to

$$x \equiv -6 \equiv 130 \pmod{136}.$$

4 Question 4

Miscellaneous Brief Questions (20 points total).

4.1 Short Answer Question

(2 points)

1. Suppose Bob has an El Gamal private key a and public key h. The modulus p and primitive root g are common knowledge. Consider a ciphertext (r, t) that encrypts plaintext message m, encrypted to Bob. Suppose an attacker sees (r, t), but does not know m, or the private key a of Bob. Can this attacker generate, in polynomial time, a valid ciphertext (r, t') that correctly encrypts the plaintext $2m$ to Bob? In fact, he can. Give a formula or algorithm for t' that this attacker can use to compute it.

4.2 True/False Questions

(14 points) Please indicate, for each statement, whether it is true or false.

1. The Computational Diffie-Hellman Problem reduces to the Discrete Logarithm Problem.
2. Solving the Discrete Logarithm Problem suffices to break the El Gamal cryptosystem.
3. The key obtained in Diffie-Hellman key exchange can be pre-determined by one party before the exchange.
4. The method of successive squaring for modular exponentiation has exponential runtime in the bitlength of the exponent.
5. The Birthday Attack on the Discrete Logarithm Problem has polynomial runtime in the bitlength of the modulus.
6. The runtime for Index Calculus is superior to that of Baby-Step-Giant-Step.
7. For $g, a, b \in (\mathbb{Z}/p\mathbb{Z})^*$, $L_g(ab) = L_g(a)L_g(b)$.

4.3 Fill in the blank

(4 points) Only the answer matters.

1. The number of cycles that will appear in the additive dynamics of +15 modulo 1000 is _____.
2. Let g be a primitive root modulo p. Ignoring 0, the number of cycles that will appear in the multiplicative dynamics of $\times g^2$ modulo p is _____.

Solutions.

1. $t' = 2t$.

1. True
2. True
3. False
4. False
5. False
6. True
7. False

1. 5
2. 2

5 Question 5

(10 points) Allow me to remind you of the definition of Big-Oh notation.

Definition 1. . *Let $f : (0, \infty) \to \mathbb{R}$ and let $g : (0, \infty) \to \mathbb{R}^{\geq 0}$. Then we say that $f = O(g)$ if there exist real constants $c > 0$ and $x_0 > 0$ such that, for $x > x_0$, we have $|f(x)| \leq cg(x)$.*

Prove that, for any non-negative integer k, $x^k = O(e^x)$. (Hint: induction on k combined with calculus.)
Solution 1.
This solution uses L'Hôpital's Rule. We will show that

$$\lim_{x \to \infty} \frac{x^k}{e^x} = 0.$$

If that holds, then clearly $x^k < e^x$ for sufficiently large x, so $x^k = O(e^x)$.

Let us induct on k. The base case is $k = 0$. In that case we have

$$\lim_{x \to \infty} \frac{x^0}{e^x} = \lim_{x \to \infty} \frac{1}{e^x} = 0,$$

since e^x tends to ∞.

For the inductive step, we assume

$$\lim_{x\to\infty} \frac{x^k}{e^x} = 0.$$

Then by L'Hôpital's Rule, since the limit above exists,

$$\lim_{x\to\infty} \frac{x^{k+1}}{e^x} = \lim_{x\to\infty} \frac{(k+1)x^k}{e^x} = (k+1) \lim_{x\to\infty} \frac{x^k}{e^x} = 0.$$

Note on the above solution: The limit formulation implies the big-Oh formulation, but the converse is not in general true.

Solution 2.

We wish to show that for all k, there exists a C and an x_0 so that for all $x > x_0$,

$$x^k \leq ce^x.$$

The equation above is equivalent to the equation

$$k\log(x) - \log(c) \leq x,$$

at least for $k \geq 0$ and positive c and x.

For $k = 0$, this equation holds for $x > x_0$ if we take $x_0 > 0$ and $c > 0$.

We can write

$$\lim_{x\to\infty} \frac{k\log(x) - \log(c)}{x} = k \lim_{x\to\infty} \frac{\log(x)}{x}.$$

This reduces us to showing

$$\lim_{x\to\infty} \frac{\log(x)}{x} = 0.$$

That can be accomplished by, for example, L'Hôpital's rule.

Notes on this solution: It avoids induction because we can easily factor the k out of the limit.

Solution 3.

This can be accomplished by use of the series expansion for e^x:

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + x^2/2 + x^3/3! + \cdots$$

Let $x_0 > 0$. If $x > x_0 > 0$, then all the contributing terms are positive. Hence e^x exceeds any one of the terms:

$$e^x \geq x^k/k!.$$

Hence, taking $c = k!$, we obtain the result.

Solution 4.

Let us induct on k. The base case is $k = 0$. But $1 = O(e^x)$ since $e^x > 1$ for all $x > 0$.

Suppose it holds for some $k \geq 0$, that $x^k = O(e^x)$. Then, for sufficiently large x and some c, $ce^x - x^k > 0$. Since $e^x > 1$, by doubling c, we can guarantee $ce^x - x^k > 1$. Consider the quantity $f(x) = (k+1)ce^x - x^{k+1}$. Its derivative is $f'(x) = (k+1)(ce^x - x^k)$, which by the inductive hypothesis, satisfies $f'(x) \geq k+1 \geq 1$. Hence $f(x)$ tends to infinity and will therefore eventually be positive, which is to say, $x^{k+1} = O(e^x)$.

6 Question 6

(10 points) Let p be an odd prime, and g a primitive root modulo p. Prove that a non-zero residue $a \in \mathbb{Z}/p\mathbb{Z}$ has a square root (i.e. is a square of something in $\mathbb{Z}/p\mathbb{Z}$) if and only if $L_g(a)$ is even.

Solution.

Proof. Every non-zero residue a can be written as $a \equiv g^x \pmod{p}$, where $x = L_g(a)$.

If x is even, say $x = 2k$, then $(g^k)^2 \equiv g^x \equiv a \pmod{p}$, so a has a square root, namely g^k.

Conversely, if a has a square root, say b, then $b^2 \equiv a \pmod{p}$. Since b is nonzero (otherwise a would be zero), we can write $b = g^k$ for some k. Then $g^x \equiv a \equiv b^2 \equiv g^{2k} \pmod{p}$. Hence $x \equiv 2k \pmod{p-1}$.

In other words, $x = 2k + \ell(p-1)$. But since $p-1$ is even, x is therefore even (and, in fact, 'even' and 'odd' are well-defined notions modulo $p-1$, meaning that any other residue congruent to x will have the same parity as x). □

Notes: The last paragraph is not strictly required in your writeup, but it's a good thing to ponder a little here: if $p-1$ were odd, the idea of 'even' wouldn't even make sense for a discrete log.