Fall 2020 CIS 3362 Homework #6: Public Key Encryption
Check WebCourses for the due date

1) One of the primitive roots (also called generators) mod 43 is 19. There are 11 other primitive roots mod 43. One way to list these is $19^{a1} \bmod 41$, $19^{a2} \bmod 41$, … $19^{a12} \bmod 41$, where $0 < a1 < a2 < \ldots < a12$. (Note: it's fairly easy to see that $a1 = 1$, since 19 is a primitive root.) Find the values of a10, a11 and a12 and the corresponding remainders when 19^{a10}, 19^{a11} and 19^{a12} are divided by 43. Use a program or calculator to quickly simplify the modular exponentiations that arise, but show what each calculation is.

2) In the Diffie-Hellman Key Exchange, let the public keys be $p = 43$, $g = 26$, and the secret keys be $a = 13$ and $b = 22$, where a is Alice's secret key and b is Bob's secret key. What value does Alice send Bob? What value does Bob send Alice? What is the secret key they share? Use a program or calculator to quickly simplify the modular exponentiations that arise, but show what each calculation is.

3) In an RSA scheme, $p = 37$, $q = 19$ and $e = 77$. What is d?

4) In Elliptic Curve Arithmetic what is the sum of the points (22, 17) and (8, 28) on the curve $E_{37}(15, 4)$?

5) In Elliptic Curve Arithmetic calculate 4 x (22, 17) on the curve $E_{37}(15, 4)$? (Note: This will require you to multiply by two twice.)

6) Consider an El Gamal cryptosystem with the prime $q = 37$ and the primitive root $a = 15$. Alice picks $X_A = 22$ for her secret key. What is the public key Y_A that Alice posts? Now, consider sending the message $M = 31$ to Alice. Give two different ordered pairs that you could send to Alice using her public keys to encrypt M. For each, write down which value of k you picked, the corresponding value of K, as well as the cipher text, the ordered pair (C_1, C_2). Use a program or calculator to quickly simplify the modular exponentiations that arise, but show what each calculation is.

7) For this question, you are going to implement a RSA protocol to send the TAs and me (Arup) a message. For our RSA system, the public keys are as follows:

n = 5959543795627426174320202010482251983
e = 2362345234523453452345234523452345243447

Your message must be in Radix-64. Please google this format. It allows for 64 characters, encoding each with 6 bits. The characters are: all lowercase letters, all uppercase letters, all digits, the plus sign(+) and a forward slash (/).

First, type your message in a textfile only using those 64 characters. Type 20 characters per line. To encrypt, you will encrypt each line, one by one. Please pad the last line with '+' characters as needed. Convert each line of 20 Raxix-64 characters to a 120 bit integer. This will be your plaintext for RSA. Use the public keys given above and calculate the ciphertext, which will be a number from 1 to n-1. Output this number to a textfile. Do this for each line of the message. Here is what you need to turn in for this question:

1. Your code. **(Please use either Java or Python so you have support for Big Integers, naturally.)**

2. A text file with your ciphertext. This should have one number per line, for each block of 20 Radix-64 characters.

If you did everything to specification, the TAs and I should be able to read your message. **Please keep it clean =)** You may address any one of the four of us in your message, or all four of us, if you'd like!