Name:______________________

Introduction to Cryptography

The Repeated Squaring Algorithm and the Prime Exponentiation Cipher

(1) Using Python, write a function which
   - (a) accepts integers a, b, and n as arguments;
   - (b) implements the repeated squaring algorithm without using the pow(a,b,n) function;
   - (c) returns the value of $a^b \mod n$.

(2) Using Python, write a function which
   - (a) accepts an integer message, exponential key (integer), a modulus p (prime integer), and a boolean variable as arguments;
   - (b) checks that the key is valid;
   - (c) performs the exponential encryption to the integer message modulo p if the boolean variable is TRUE and performs the exponential decryption to the integer message modulo p if the boolean variable is FALSE;
   - (d) prints the encrypted/decrypted integer message.

(3) Suppose we publicly broadcast the modulus p and the encryption exponent e of the prime exponentiation cipher so that anyone can encrypt a message. This is called a **public key encryption** scheme. If the decryption exponent d is kept private (and not publicly broadcast), explain how an attacker would still be able to decrypt an intercepted ciphertext message.

(4) Suppose we intercept the following ciphertext message.

$$265447333455441482929244853322644679466$$

This message was encrypted using the public key encryption scheme outlined in the previous problem where the prime modulus is $p = 11111111122233333333444555667777777779999$ and the encryption exponent is $e = 19088892923$. Decrypt the message.