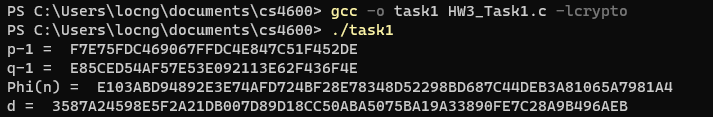
CS4600 Homework 3

1. Task 1 – First, I initialize the variables p, q, and e, assigning them value. Next, I make a variable to store the value 1 to subtract from p and q to find phi(n). Then, compute phi(n) and use modular inverse to find d, the decryption key.



1. Task 2 – First, I initialize the variable m, n, e, and d. Next, I use formula C = Me mod n to encrypt the plaintext. Then, I proceed to use the formula M = Cd mod n to decrypt the ciphertext. If everything goes smoothly, the plaintext should be the same as the original plaintext and are compared against each other by BN\_cmp().



Text

Description automatically generated

1. Task 3 - First, I initialize the variable m, n, e, d, and ciphertext. Next, I use the formula M = Cd mod n to decrypt the ciphertext. Then, I use the python command to convert the hex code to ASCII string to see what the plaintext is.



Text

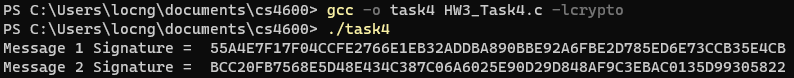
Description automatically generated with medium confidence

1. Task 4 – First, I use the python command to get the hex format for both messages. Next, I sign corresponding messages with formula Signature = m^d mod n by replace m correspondingly. The signatures are completely different because the original plaintext is different. This means that any change in the original data content can be detected when the digital signature changes.



Graphical user interface, text

Description automatically generated



1. Task 5 – First, I use the python command to get the hex format for the original message. Then, I verify the signature using the formula Plaintext = Signature^e mod n. This should yield a result that is the same as the original message. To be sure, I used BN\_cmp() to compare them and if it is, print “Plaintext Matched. Signature Verified!”. Otherwise, it would print "Plaintext Not Matched. Invalid Signature!".

Text

Description automatically generated

Text

Description automatically generated