Foundations of Computer Science Lab 7

We found out that Loyola has been sniffing our network and stealing our secret messages. We want to put an end to this, so we decide to write a program that implements three very common encryption schemes:

1. Shift Ciphers
2. Vigenere Ciphers
3. RSA Public Key Encryption

The mathematical formulation to each encryption scheme is as follows:

Shift Cipher

Assume you have a message. We know from the past cycle that each character on our keyboard has a corresponding numeric value from the ASCII table. The shift cipher algorithm is as follows:

1. Convert our message into a list of ASCII Values
2. Add our "key" to each individual character of our ASCII Value List
3. Convert our message back into its corresponding ASCII Characters

The Shift Cipher is an example of a **Symmetric Encryption Scheme**, in the sense that encrypting and decrypting messages follows the same pattern. For example, The algorithm to decrypt a shift cipher is as follows:

1. Convert our Ciphertext into a list of ASCII Values
2. Subtract our "key" to each individual character of our ASCII Value List
3. Convert our message back into its corresponding ASCII Characters

EXAMPLE RUN

```
Message: Hello calvert hall
Cipher: Jgnnq"ecnxgtv"jcnn
unencrypted: Hello calvert hall
```

Vigenere Cipher

It turns out that our Shift Ciphers are very easily broken…. Is there a better way to perform a similar cipher? The answer is YES! We can perform a Vigenere cipher which can be thought of as a character by character shift cipher. The algorithm is as follows

1. First we need to check our key to make sure it is the correct length. The key and message MUST be the same length.
    a. Example
        i. Message: Hello World
        ii. Original Key: “KEY”
        iii. Updated Key: “KEYKEYKEYKE”
2. Convert both the Message and Key to their corresponding ASCII Values
3. Character by Character, **ADD** the ASCII Values together and perform the MOD operation by **MOD 255**. Save the result into a third variable called encrypted
4. Convert the encrypted message back into its corresponding ASCII Characters

The Vigenere Cipher is also an example of a **Symmetric Encryption Scheme.** The inverse algorithm is as follows:

1. First we need to check our key to make sure it is the correct length. The key and message MUST be the same length.
    a. Example
        i. Message: Hello World
        ii. Original Key: “KEY”
        iii. Updated Key: “KEYKEYKEYKE”
2. Convert both the Message and Key to their corresponding ASCII Values
3. Character by Character, **SUBTRACT** the ASCII Values together and perform the MOD operation by **MOD 255**. Save the result into a third variable called encrypted
4. Convert the encrypted message back into its corresponding ASCII Characters

EXAMPLE RUN

KEY = “Let’s Go Hall”

```
Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall
!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hal
l!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets go hall!Lets
Nkxk2&u{&igt&k~vruxk&u{x&vgyyouty&zu&xkgin&u{x&vuzktzogr&gtj&xkgrok&u{x&v{xvuyk2&ut&u{x&igsv{y&gtj&ot&znk&}uxrj&hkutj4&G&Igr|kxz&
Ngrr&kj{igzout&oy&g&zxgtyluxsgzo|k&k~vkxoktik&zngz&iutzot{ky&rutm&glzkx&mxgj{gzout4&U{x&yz{jktzy&hkiusk&Skt&ul&Otzkrrkiz2&Skt&ul&
Lgozn2&gtj&Skt&ul&Otzkmxoz2&gtj&znk&xkrgzoutynovy&znk&jk|kruv&u|kx&lu{x&kgxy&gz&Znk&Ngrr&rgyz&luxk|kx4
Here, you can explore your passions to reach your potential and realize your purpose, on our campus and in the world beyond. A Ca
lvert Hall education is a transformative experience that continues long after graduation. Our students become Men of Intellect, M
en of Faith, and Men of Integrity, and the relationships they develop over four years at The Hall last forever.
```

RSA Public Key Encryption

It turns out that some people cannot be trusted with keeping their keys a secret. Therefore we can utilize a modern and advanced form of cryptography called **Public Key Encryption.** RSA Encryption is one of the most frequently used forms of Encryption and is effectively impossible to break without the use of Quantum Computers. Therefore, we'd like to use RSA to encrypt our messages. There are two stages to RSA

1. Key Generation
2. Encryption and Decryption

The Key Generation is as follows:

1. Choose two very large prime numbers (big prime number generator) and multiply them together, call that N
2. Compute the Euler Totient Function of p and q, call the result ON
    a. (p-1)*(q-1)
3. Choose a number that is coprime with ON, call it E.
4. Compute the modular inverse of E and ON. Call the result D
5. You now have two keys, (E,N) and (D,N).

To encrypt a message:

1. Convert your message into ASCII Values
2. Character by Character, **perform modular exponentiation by E MOD N**
3. Convert your list back into its ASCII Characters

To Decrypt a message:

1. Convert your message into ASCII Values
2. Character by Character, **perform modular exponentiation by D MOD N**
3. Convert your list back into its ASCII Characters

EXAMPLE RUN

Our message was "CHC" which converted to a subset of 26 characters is 383

```
result of p times q: 17094377
Result of euler's totient function: 17086020
modular inverse of e mod o(n) -1
encrypted message: 5719925
Decrypted message: 383
```

This is the result of performing RSA Encryption, both encrypted and decrypted.

Wrapping it all together:

You should write a simple command line interface that lets the user choose between which of the three encryption schemes they want:

```
Welcome to the encryption calculator!
1. Shift Cipher
2. Vigenere Cipher
3. RSA Encryption
4. Quit
Choose which Encryption Scheme you would like to use:
```

You should also ask for the message and the Key (if applicable, for RSA it should ask for two primes):

```
enter the message you would like to encrypt: Hello CHC
Enter the amount you would like to shift by 4
```

```
enter the message you would like to encrypt: Hello chc
Enter prime 1: 5
Enter prime 2: 7
```

Then, you should ask the user if they would like to encrypt/decrypt and perform the necessary algorithms. Make sure to print out these three pieces of information:

1. The original Message
2. The (modified) Key(s)
3. The result of your operation

WARNING: I KNOW THAT MANY OF THIS CAN BE FOUND ONLINE – IF I CATCH YOU USING ANY OF THE GEEKSFORGEEKS OR OTHER ONLINE SOLUTIONS, YOU WILL GET A 0 ON THIS LAB