**Requirements**

Functional

Inputs:

1. User defined plaintext messages from keyboard
2. Ciphertext messages received from another user
3. (optional) Random numbers used for key establishment

Outputs:

1. Encrypted ciphertext made from user defined plaintext
2. Decrypted plaintext

Data to be stored:

1. Key
2. (optional) Output files

Computations to perform:

1. Key establishment (exponential modulation)
2. Scan user input from keyboard/clipboard/files
3. Using input, generate ciphertext
4. If ciphertext is entered in a separate text box, generate plaintext using same conditions

Timing and Synchronization:

1. Key establishment and initialization of scheme will come first and should take no longer than a few seconds depending on user decisions.
2. All computations will be in real time based off user input

Non-functional

1. Language to be used will be Python (for cross platform support)
2. Target user base is anyone to whom privacy and security are a concern
3. Software will function like google translate, providing instantaneous plain and ciphertext translations in real time depending on selected mode of operation. If one text box is altered, change will be shown in the other one as well.
4. Once user messages are encrypted, it would require a computing time complexity of about 2248 to crack without the key. As of 2015 there are no known attacks to crack it. ChaCha20 is a variant of Salsa20, one of the few alternatives to AES widely used.