Assignment Discussion

For assignment 2, I completed the task of implementing a crypto machine similar to the Enigma using C++. To use my implementation, you need to compile the program using the command in the Readme file. Once the data is compiled, run the program, and it will display two options to the user. The first option lets the user encrypt a string, and the second option enables the user to decrypt a string. If the user selects option 1, then they will be asked to enter a 10-character key. The user must enter ten characters from 0 to 9 in any order making sure not to use a character more than once. This key will determine how the string the user enters will be transposed. After the 10-character key is entered, the 3-character key most be entered. This key determines what character each rotor will start on; the first character corresponds to the left rotor, the second to the middle rotor, and the third to the right rotor. Lastly, the user must enter a string to be encrypted. The second option will first ask the user for a 10-character string used to encrypt the message. Then it will ask for the 3-character key that was used. After the keys have been entered, type in the encrypted message and press enter. The program will then output the decrypted message.

My testing plan for this assignment consisted of testing if my program could encrypt and decrypt a simple 10-character string. Next, I tested to see if my program could encrypt and decrypt greater than 10-characters. Then I test to see if my program could encrypt and decrypt less than 10-charact. After I made sure my program was working probably, I then went on to see if I could decrypt messages from other students. When I tried to decrypt messages from other students, I would get different outputs. This was because my program did not account for the positional offset between the current wheel and previous. Below are images of me testing my code and the results.

In conclusion, in comparison to the original Enigma machine, this implementation would be harder to crack, especially using the bombe-approach. The first being that the original Enigma machine dealt only with 26 characters, and this one deals with 36. Another key difference is the 10-character key that transposes the string. Assuming that a string has ten characters, then there are 3,628,800 permutations, and there are another 42,840 permutations when deciding the 3-character key.

Test Cases:

1.

```
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 1
Enter 10-character string key: 0123456789
Enter 3-character string key (left to right): 000
Enter a message to encrypt (no special characters): cortney

Encrypted message: 7lzvy4r
ccr56@tux3:~/CS-475/Assignment2$ ./enigma_encryption
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 2
Enter 10-character string key: 0123456789
Enter 3-character string key (left to right): 000
Enter a message to decrypt: 7lzvy4r

Decrypted message: cortney
```

2.

```
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 1
Enter 10-character string key: 3145926870
Enter 3-character string key (left to right): 203
Enter a message to encrypt (no special characters): helloworld

Encrypted message: u4ekmo7ltu
```

```
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 2
Enter 10-character string key: 3145926870
Enter 3-character string key (left to right): 203
Enter a message to decrypt: u4ekmo7ltu

Decrypted message: helloworld
```

3.

```
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 1
Enter 10-character string key: 0123456789
Enter 3-character string key (left to right): 000
Enter a message to encrypt (no special characters): abcdefghijabcdefghijabcdefghij

Encrypted message: rrrrreefffddddskkkkk42222hhhaa
ccr56@tux3:~/CS-475/Assignment2$ ./enigma_encryption
Welcome to the crypto machine similar to the Enigma
        1. Encrypt msg.
        2. Decryot msg.
Select an option: 2
Enter 10-character string key: 0123456789
Enter 3-character string key (left to right): 000
Enter a message to decrypt: rrrrreefffddddskkkkk42222hhhaa

Decrypted message: abcdefghijabcdefghijabcdefghij
```