Project Documentation

1. Team Name: IDK NVM.

Members: Miguel Ibarra, Issac Martinez, Jered Napalan, Lissette Serrato

1. Project Information and Details:

**What problems are you solving in this project?**

In this program, we are trying to solve various problems associated with the Vigenere Encryption process; this include being able to encrypt, decrypt, write and decode a letter in a message utilizing the Vigenere Cipher. In order for our program to work as desired, some problems we will have to solve will deal with how we will ensure the correct number of shifts will be used during the encryption/decryption process as well as how to store the Vigenere table in either an array or vector and how to properly display the table in the correct format.

**What solutions are you implementing in the project?**

In order to achieve our goal, we will be utilizing a menu system, allowing for the program’s user to select from a variety of options pertaining to the cipher progress. The options will include a way to view the Vigenere Table and either encrypt or decrypt a string of text based on a key provided by the user. Each menu option will correlate with a function in our code that will implement the correct algorithm required to properly complete the task assigned to that function.

**Provide explanation of calculations and algorithm implementation.**

Within each function, we utilize various algorithms in order to effectively encrypt and decrypt the string as necessary. When encrypting/decrypting will we pass in the desired parameters (in our case, the string and shift number) and will use a loop to iterate through the string; we then utilize the shift number and the ASCII numbers associated with alphabet to properly alter the letter to the desired new letter. In regard to the Vigenere table, we will utilize a matrix array to store the actual table. Using several loops, we then will fill in the array with the letters of the alphabet while shifting the letter over to the right by one as we traverse down the columns of the table.

**What are the program objectives? Explain how your program is interacting      with the user and its purpose.**

The overall objective of the program is to provide a seamless experience for the user, allowing them to choose from a variety of different options regarding the Vigenere Cipher experience. If the user chooses to do so, our program would allow them to enter in any message they would prefer and allow them to encrypt the message using the Vigenere cipher. The program would also allow for the user to decrypt the message, assuming that they have the shift value on hand as well.

**How are discrete structures implemented in the C++ program?**

Since we are dealing with various forms of arithmetic in this program, data structures are implements throughout the entirety of our cipher program. For example, we are dealing with a set of all the letters in the alphabet which could then be sorted in an array of vector. We would then traverse through this set/vector in order to match that letter/character with that of a user inputted shift value that would match the letter to its new value.

**What are the limitations of the program?**

There are several limitations found within our program, such as the limitation of the number of shifts that we are trying to implement which is set by the programmer. For example, if we are trying to implement a shift of seven spaces to the left for each letter in the alphabet, then the shift will persist throughout the program.

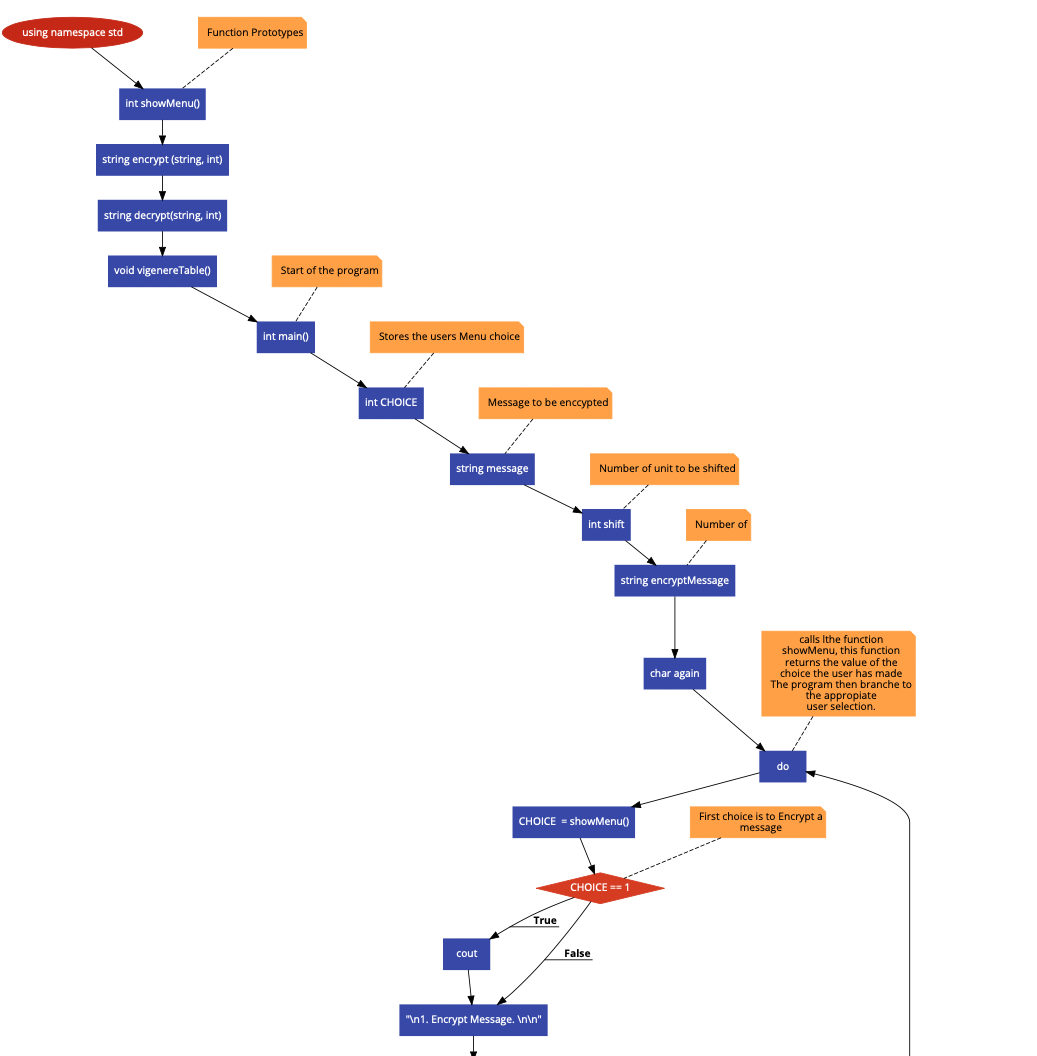
**Provide recommendation on improving the limitations of the program.**

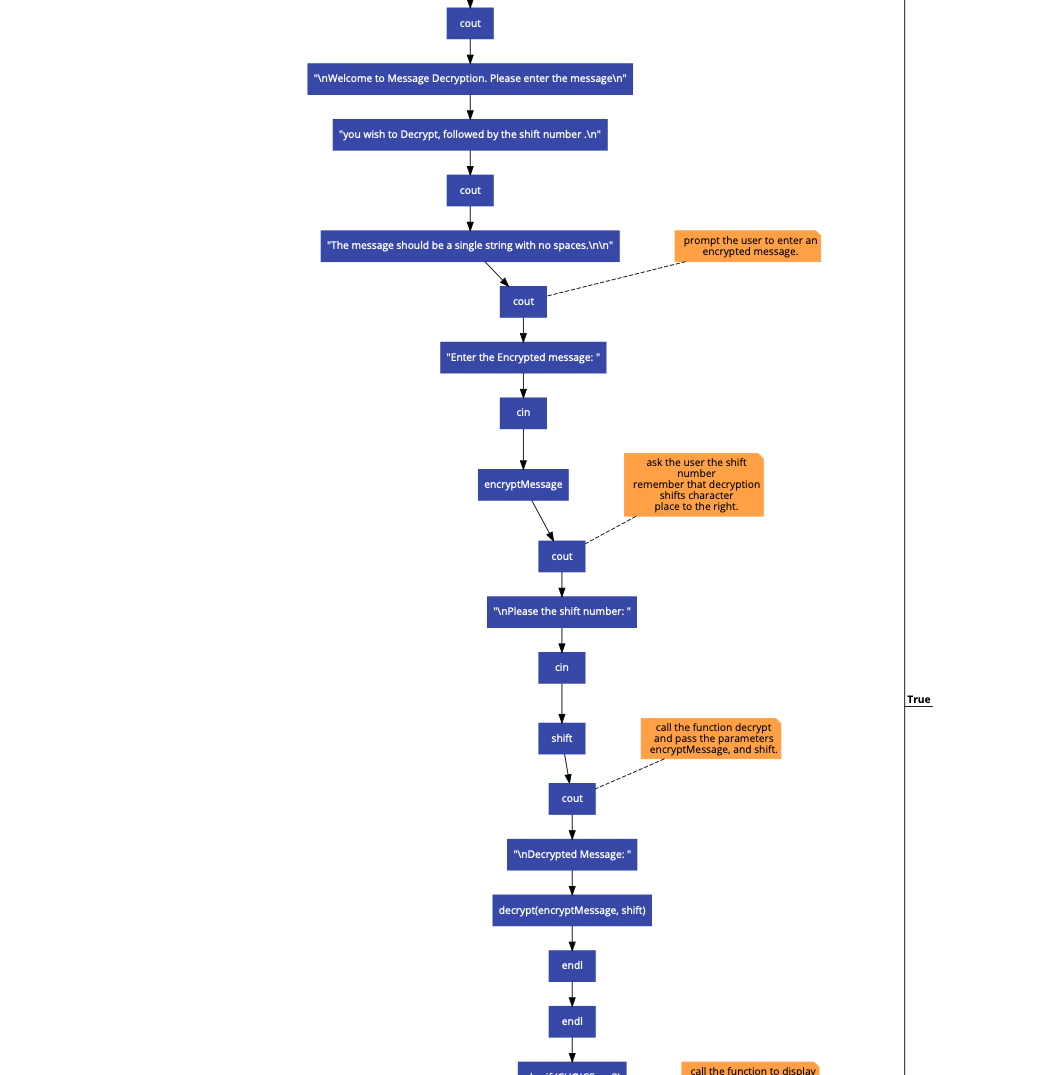
One recommendation for overcoming the limitation would be the idea of incorporating a function or system that would allow for multiple shifts to be used when encrypting or decrypting multiple messages. One possibility would be to expand our messaging system to ask the user if they would like to encrypt or decrypt a new message, which would then ask for a new shift value as well.

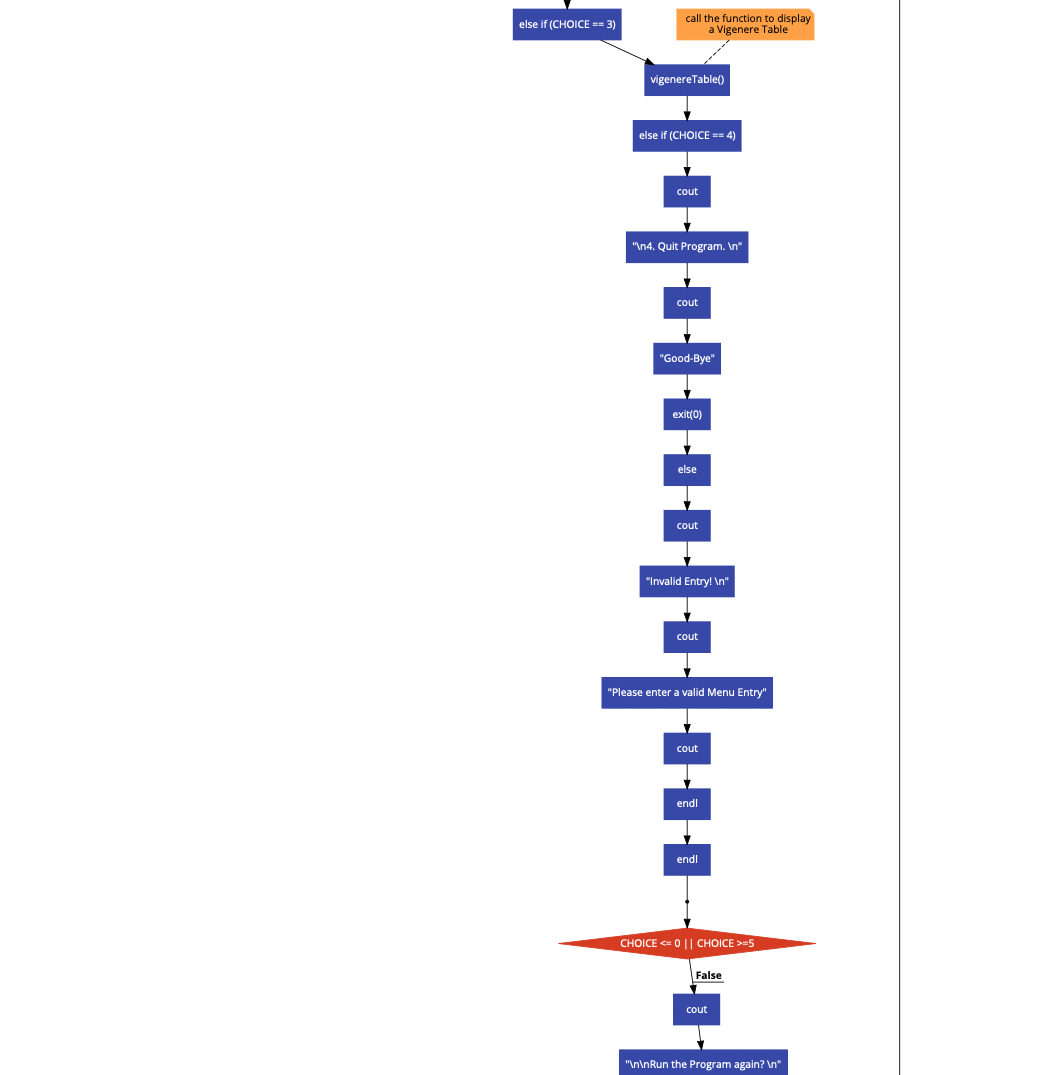
1. Flowchart & Psuedo Code.

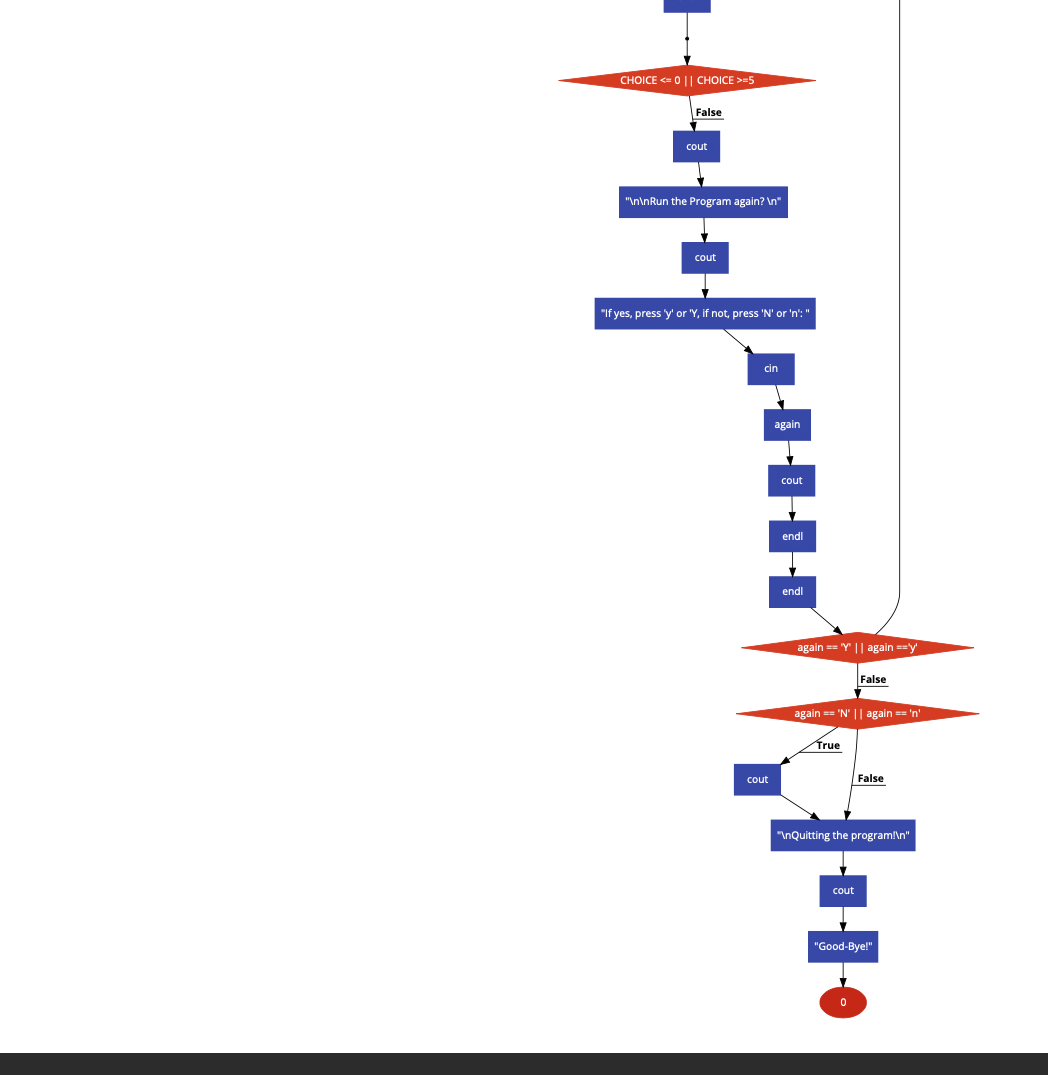
**Flowchart**

\*See attached PDF for a clearer photo.

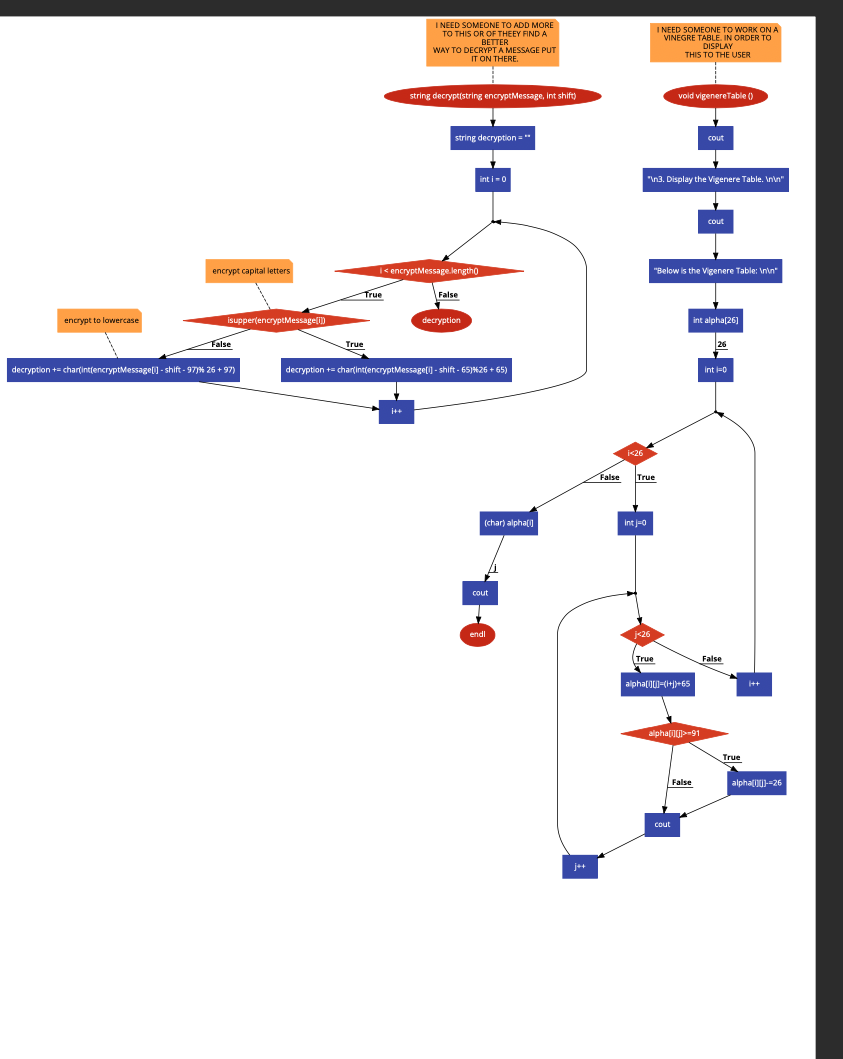












**Psuedo Code**

This program is meant to do one of three things depending on the user’s selection. It will Encrypt/decrypt a message based on the Vigenere Cipher, or it will print a display of the Vigenere table itself.

In the main function:

Initialize variables

* CHOICE - int, used to store the user’s choice in the menu
* Message - string, stores the message that will be encrypted
* Shift - int, shifts used for the encryption
* encryptMessage - string, stores the final encrypted message

We use a do while loop to loop back to the menu if an invalid option is selected.

Options for the menu include:

1. Encrypt - Calls the encrypt message function.

Displays welcome message.

Asks for user input for the desired message for encryption

Asks for the shift number.

Passes this info into the encrypt message function.

2. Decrypt - Calls the decrypt message function.

Similarly to the prev option:

Displays welcome message.

Asks for user input for the desired message for decryption

Asks for the shift number.

Passes this info into the decrypt message function.

3. Display Vigenere Table - Calls Vigenere Table function.

Calls Vigenere Table function.

4. Quit Program - Ends the program

Displays good bye message.

Ends program.

Anything else results in the invlaid selection message and returns to the menu.

Show Menu Function :

Prints message displaying entire menu.

We then ask for the user to enter their selection.

This gets returned to the main function where the proper follow up function is then called with the proper parameters.

String Encrypt Function :

* Parameters passed through is the message and the shift number.
* Empty string is initialized to later store our message.
* For loop is used to iterate through the entire string.
  + if -> the letter at the index is uppercase, use uppercase encryption method.
    - using the character at the index, add the shift to that character and subtract 65 so that the character is converted into a number. We then mod that number using 26 in order to encrypt it. Finally we add 65 back to the number in order to convert the number into our new encrypted ASCII coded character.
  + else -> use lowercase encryption method.
    - Lowercase letters are treated in the same manner as before, except this time we subtract and add 95 in order to get the proper lowercase ASCII.
* Once our string is completely encrypted, we return the string to main where it is printed for the user to see.

String Decrypt Function :

* Parameters passed through is once again the message and intended shift number.
* Empty string is initialized to store our message.
* For loop is used to literature through the entire string.
  + if -> the letter at the index is uppercase, use uppercase decryption method.
    - Similar to our encryption method, but instead of adding the shift we subtract the shift.
  + else -> use lowercase encryption method.
    - Similar to our encryption method, but instead of adding the shift we subtract the shift.
* Once our string is completely decrypted, we return the string to main where it is printed for the user to see.

Void Vigenere Table Function :

* Displays welcome messages for the function.
* Alpha is initialized to serve as the array storing our letters for the table.
* We have a nested for loop used to fill in the table with each row shifting the letter one to the right.