Mauro Gabriel Elias Vazquez

Kasey Nguyen

24FAL-CIS-7-28754

December / Fall 2024

| VIGENERE CIPHER DECRYPTION | SUMMARY |

| FINAL COURSE PROJECT |

| PART 2.1 |

TEAM MEMBERS

The only member of this team is myself: Mauro Gabriel Elias Vazquez

PROJECT INFORMATION AND DETAILS

1. WHAT PROBLEMS ARE YOU SOLVING IN THIS PROJECT?

   The problem to be solved with respect to what this program does is to be able to secure and protect messages that can only be revealed or converted by means of the logic of the vigenere system. In other words, the problem I am solving is a "security" problem that guarantees accurate and proper encryption and decryption.

In “technical” matters, the problems I am solving to improve the performance and efficiency of this program are related to its instructions or algorithms to make it convert and reveal vigenere encryption. While the idea does not seem complicated to me, it became a challenge to have to be reviewing the program and verifying that it uses the idea of encryption correctly and that the password also contributes to the message to be decrypted properly.

2. WHAT SOLUTIONS ARE YOU IMPLEMENTING IN THE PROJECT?

   What I can say about the solutions I have implemented for my code is related to ensuring that encryption and decryption work accurately. Among the things related to this solution is to make the key match the words, that the input and output text contains valid and appropriate characters (that what the program displays makes sense) and to make the logic or structure of the program simple. Another “solution” is to add “toupper” so that the letters are converted to uppercase and so that the caculations are made easier, recognizable and consistent for the program.

3. PROVIDE EXPLANATION OF CALCULATIONS AND ALGORITHM IMPLEMENTATION

   The calculations that this program does includes:

   The creation of a password: The program will have to adjust the password to the text that the user enters. At the end each character of the text will have to match the characters of the key. If the number of characters exceeds the number of characters in the key, then the key will be repeated to match that number.

Ex: “Hello” ---> “Keyke” (If, **KEY** = *Key*)

The encryption process: Here the program will have to move the characters along the guidelines of the current cipher, making sure that letters from A to Z are used. The formula used here is as follows: Ei = (Pi + Ki) mod 26. What the formula does is to take characters (in the format of letters) and convert them to numbers, then add them together keeping the result within a range of 26 letters and at the end convert the number to a letter or letter equivalent.

The decryption process: This process works the same as the previous one except that here the letters are converted to numbers and subtracted from the number of the encrypted letter. The process is reversed, so that in this section you get back to the original text.

4. WHAT IS THE PROGRAM OBJECTIVES? EXPLAIN HOW YOUR PROGRAM IS INTERACTING WITH THE USER AND ITS PURPOSE

   The main purpose of my program is to encrypt and decrypt text in “Vigenère Cipher”. Regarding the use that this program in C++ can have in real life, I could say that it is a useful tool to (looking at its structure) understand how concepts like cryptography and discrete structures are applied. An advanced version of this program could be used to encrypt messages or the text of a document in particular so that certain information cannot be read easily. An even more advanced version could perhaps be used to hide passwords or access codes of personal accounts practically and simply so that only I can decrypt them. The way this code interacts with people is through text input and output. The program will presenst itself in a

friendly way and explain to the user what it can do. The user will then be required to choose whether to encrypt or decrypt a message. Later the user will enter a word and a password, or a word to decrypt. At the end, the program will do its job and terminate automatically.

5. HOW ARE DISCRETE STRUCTURES IMPLEMENTED IN THE C++ PROGRAM?

   In this program, discrete structures are implemented through the use of arithmetic, "logical" decision making and sequences.

   The program is able to process different kinds of text (normal text and key).

   The program uses operations or modular arithmetic by using formulas such as Ei = (Pi + Ki) mod 26 - Di = (Ei - Ki) mod 26.

   The program also makes important decisions, responsible for the correct flow of the program. Among these important and logical decisions is the choice to encrypt, decrypt, or exit the program.

6. WHAT ARE THE LIMITATIONS OF THE PROGRAM?

   The limitations of the program lie in the spacing of the words and I had trouble getting the program to encrypt phrases such as: "Hello _ how _ are _ you" or "Good _ night".

7. PROVIDE RECOMMENDATIONS ON IMPROVING THE LIMITATIONS OF THE PROGRAM.

   To take my program to the next level in the future, I would implement advanced techniques to make it able to encrypt complete sentences. This would be useful to

develop the version I introduced earlier, where it would be able to encrypt large texts in files or chats.

| VIGENERE CIPHER DECRYPTION | PSEUDOCODE |

| FINAL COURSE PROJECT |

| PART 2.2 |

THE PROGRAM

My project is a program that implements the vigenere encryption. This is an encryption method that works by substituting and using a key. The key is aligned to the text until its characters equal the length of the text. When encrypting and decrypting, each character is shifted according to the “alphabetic” value of each letter of the alphabet, adding or subtracting the values. My program is able to encrypt and decrypt text as long as the user provides text or words with appropriate characters.

VARIABLES / FUNCTIONS

Generate_The_Key : generates the password / key

Encrypt_The_Text : encrypts the text

text : original text

Key : password / key

Encrypt_The_Text : encrypt the text

Decrypt_The_Text : decrypt the text

Ciphertext: cipher text

Choice : choice

User_Text : user's choice

The_Key : the main key

Tot_Key : totak key

Res_Encrypted_Text : result: encrypted text

Res_Decrypted_Key : result: decrypted text

HOW IT WILL LOOK (PSEUDOCODE)

"MAIN MENU"

- This first part is a menu
- It will greet the user and ask what they want to do.
- The user will be asked to enter a number from 1 to 3.
- The program will reject with a friendly message the user's request if a character other than 1, 2 or 3 is used.

"OPTION 1"

- If option 1 is selected, the text will be encrypted
    - The user will be required to type a text to be encrypted
        + The text will be stored
        + The text will have to be typed all together and without spaces

    - The user will be required to enter a key
        + The key will be stored
        + The key has to be a word or phrase that will be used to encrypt the text
- generate_the_key will be called
    - The key will be adjusted to the length of the text
- encrypy_the_text will be called to encrypt the text
- The result will be displayed
- The end of option 1

"OPTION 2"

- If option 2 is selected, the text will be decrypted
    - The user will be required to type a text to be decrypted
        + The text will be stored
        + The text will have to be typed all together and withous spaces
    - The user will be required to enter a key (The same used to encrypt the original text)
        + The key will be stored
- generate_the_key will be called
    - The key will be adjusted to the length of the text
- decrypt_the_text will be called to decrypt the text
- The result will be displayed
- The end of option 2

"OPTION 3"

- If option 3 is selected, the program will end / exit
- A kind goodbye message will be displayed
- The program will stop
- The end of option 3

"INVALID OPTION"

- If an invalid command is entered, a message will be displayed to the user
    - The user will have to change his response
- The end of invalid option.