CIS-7 Project Documentation

Team Names:

* Demajay Francis
* Eric Solorzano
* Bryan Cruz

Project Information:

* What problems are you solving in this project?
  + One major problem we had to solve in our code was getting the code to work with both uppercase and lowercase letters. When the code was first written, it worked perfectly with uppercase letters, but it would not encrypt or decrypt correctly when lowercase letters were attempted.
* What solutions are you implementing in the project?
  + To resolve this issue, we used the transformation function to convert the string to all uppercase letters and then proceeded to the encoding and decoding process.
* Provide explanation of calculations and algorithm implementation.
  + In this program, the function is encrypted in the encryption function located in the encryption header file. After the key is extended to match the size of the plain text entered by the user, the for loop iterates through each element in the word variable. Every element of the word is added to every element of the key with respect to the current i value. Then the modulo 26 operator limits the operation values to the range of the uppercase letters. Finally, A is added to convert the result to its corresponding uppercase letter ASCII value. The decryption is very similar to the encryption method, essentially doing the reverse. After the key is extended to match the length of the encrypted message, the for loop in the decryption function located in the decryption header iterates through each element of the message and the key. It subtracts the element of the key from the element of the word with respect to the current i value. Then modulo 26 limits the values to the range of the uppercase letter ASCII values. Finally, A is added to convert the result back into the corresponding uppercase letter ASCII value.
* What are the program Objectives? Explain how your program is interacting with the user and its purpose.
  + The program’s primary objective is to encrypt and decrypt messages using the Vigenère cipher. The program uses a switch statement to interact with the user. A menu is displayed listing 3 choices for the user, to encrypt, to decrypt, and to quit the program, then prompts the user to enter their choice. Once the user makes their choice, it is saved in the choice variable, which is then used in a switch statement to go to the portion of the program that the user wishes to utilize. If the user enters 1, the switch statement will go to case 1, where the program encrypts messages. The user is then prompted to enter their plain text message stored in the word variable. Then a key of their choice is stored in the key variable, and their encrypted message is generated and displayed. The decryption process is very similar, prompting the user to enter their encrypted message, then the key to decrypt the message then their decrypted message will be displayed.
* How are discrete structures implemented in the C++ program.
  + One way that discrete structures are implemented in this program is through modular arithmetic. During the encrypting and decrypting process, modular arithmetic is used to lock the element's value in between the range of the capital letters on the ASCII table and then add the value of the element of the key.
* What are the limitations of the program
  + There are a few limitations to this program. The first limitation is that the program cannot encrypt, decrypt, or use a key with lowercase letters. The second limitation is that the program cannot encrypt spaces and cannot encrypt multiple words written in a standard sentence format. The third limitation is that this program cannot encrypt special characters, so there cannot be any function in the words encrypted.
* Provide recommendation on improving the limitations of the program.
  + One recommendation to improve the program would be to use getline() to allow sentences with spaces. Another thing that could be done to improve the program is to use an if else statement to separate upper-case and lower-case letters adding either ‘a’ or ‘A’ depending on what is needed.

Pseudo Code:

function main()

choice = 0

condition = true

print "Welcome to our Vigenere Cypher Encryption & Decryption Program"

while condition

word

key

print "1.Encrypt"

print "2.Decrypt"

print "3.Quit"

user input choice

switch choice

case 1:

print "Type a word or phrase you wish to encrypt. (No Spaces, Numbers or Symbols)"

user input word

Word = word to uppercase

print "Type the key needed for encryption and decryption. (No Spaces, Numbers or Symbols)"

user input key

key = key to uppercase

Call function ExtendKey(key, word.length)

Call function Encrypt(word, key, word.length)

break

case 2:

print "Type a word or phrase you wish to decrypt. (No Spaces, Numbers or Symbols)"

user input word

word = word to uppercase

print "Type the key needed for encryption and decryption. (No Spaces, Numbers or Symbols)"

user input key

key = key to uppercase

Call fucntion ExtendKey(key, word.length)

Call function Decrypt(word, key, word.length)

break

case 3:

print "You chose to quit."

condition = false

break

default:

print "Invalid input, try again."

Break

function ExtendKey(key: string, wordSize: integer)

if length of key == length of word then

return

else if length of key > length of word then

temp = key.size() - wordSize

t = key.substr(0, key.size() - temp)

key = t

else

temp = legnth of key

i = 0

j = 0

for i and j = 0; i less than (wordSize - temp); i++

key = key + key[j]

j = j + 1

if j == length of key then

j = 0

end for

end if

end function

function Decrypt(word, key, size)

for i from 0 to size-1

if word[i] is not a space

decryptedCharIndex = ((word[i] - key[i] mod 26) mod 26) + 'A'

word[i] = decryptedCharIndex

print "Decrypted word: " + word

function Encrypt(word, key, size)

for i from 0 to size-1

if word[i] is not a space

encryptedCharIndex = ((word[i] + key[i] % 26) % 26) + 'A'

word[i] = encryptedCharIndex

print "Encrypted word: " + word