Dylan's Team

Members: Dylan Kral

The problems that I will be facing or solving would simply be the implementation of vigenere cipher itself. To be more precise, this involves taking a word or phrase from the user and converting that word or phrase to an encrypted message. The program will then ask the user for a desired key and map that key along with the word that they wish to encrypt or decrypt. Expressed mathematically, the encryption of the message at letter *i*, is equal to the alphabetic value of *i* in the plaintext plus the alphabetic value of the corresponding *i* in the key. The encryption process uses the equation $E_k(M_i) = (M_i + K_i) \bmod 26$. The decryption is the same process reversed, subtracting the key instead of adding to arrive back at the original plaintext value. This uses the equation $D_K(C_i) = (C_i - K_i) \bmod 26$. Discrete structures is implemented by applying the algorithms and mathematics behind the scenes. Afterall, we use discrete structures to write precise mathematical statements that captures what we want in each application, and learn to prove things about these statements. In this case, the application was for the encryption and decryption of words or phrases. A limitation that I have encountered in the program is that it makes you close the program after each encryption and decryption. This would be something that I would like to improve within the program. Another small limitation within the program would be that the encryption and decryption is only returned in capitals rather than the same way it came back. This is not a major issue but it is nice to have.

Pseudocode

The following program lets you encrypt and decrypt using the key provided

```
1. function messageAndKey()

{

1.prompt user to input the message in message and

  then convert the input message in uppercase and

  then store it in message array letter by letter.

2.prompt user to input the key in key variable and

  then convert the input key in uppercase and

  then store it in key array letter by letter.

3.next step is to map key to message

  Loop until one loop variable is less than message.length()

            inside loop

            if letter of message array at index i equal to 32

                        then add 32 to keymap variable.

            else
```

if second loop variable j is less than key.length()

add key to keymap

else

make second variable j equal to 0 and add key to key map

increment j

increment i

end loop

4. Store message and mapped key i some variable

end

}

2.Create a 2-d array

3.function visibleTable()

{

loop till variable i less than 26

inner loop run till j less than 26

if i+65+j is greater than 90

store i+65+j-26 in a temporary variable

else

store i+65+j in temporary variable

end if

store the value of temporary variable in the 2-d array at i,j position.

end for

end for

print the 2-D array in form of matrix

loop till variable i less than 26

inner loop run till j less than 26

print 2-D array value at i,j

end for

end for

end

}

4.Create a 2-d array

5. function createVigenereTable(){

  loop till variable i less than 26

        inner loop run till j less than 26

            if i+65+j is greater than 90

                store i+65+j-26 in a temporary variable

            else

                store i+65+j in temporary variable

            end if

            store the value of temporary variable in the new 2-d array at i,j position.

        end for

  end for

end

}

6. function cipherEncryption(argument one , argument two)

{

  call function createVigenereTable()

  loop till i is less than message.length()

        if value of argument one at i is equal to 32 and value of argument two at i is 32

            add " " to encrypted text variable.

        else

            store value of argument one in variable x after subtracting 65

            store value of argument two in variable y after subtracting 65

            add value of second 2-D array at x,y to encrypted text.

        end if

  end for

  print out the encrypted text.

end

}

6. function itrCount(argument one,argument two) to count number of iterations.

{

  loop till i is less than 26

    if argument one + i is greater than 90

      add argument one + i -26 to result variable

    else

      add argument one + i to result variable.

    end if

  end for

  loop till i is less than result.length()

    if value of result at i is equal to argument two

      then break

    else

      increment counter variable

    end if

  end for

  return counter variable

end

}

7.function cipherDecryption(argument one argument two){

  loop till i is less than argument one length

    if value of argument one at i is equal to 32 and value of argument two at i is equal to 32

      add " " to decryptedText variable

    else

      call and store function itrCount(for argument two at i, argument one at i) value in temp variable

      add 65 + temp variable to decrypted text.

    end if

end for

print the decrypted text

end

}

8.{In the main function

print out the menu where user can choose from various options i.e. 1. for encrypt

2. for decrypt 3. to see table and instructions to use.

take user choice

if user chooses 1 to encrypt

then call function messageAndKey()

call function cipherEncryption() with appropriate arguments

else

if user chooses option 2 to decrypt

then call messageAndKey() function

then call cipherDecryption() function with appropriate arguments

else

if user chooses option 3

then prompt user to confirm

then if user confirms call visibleTable() function

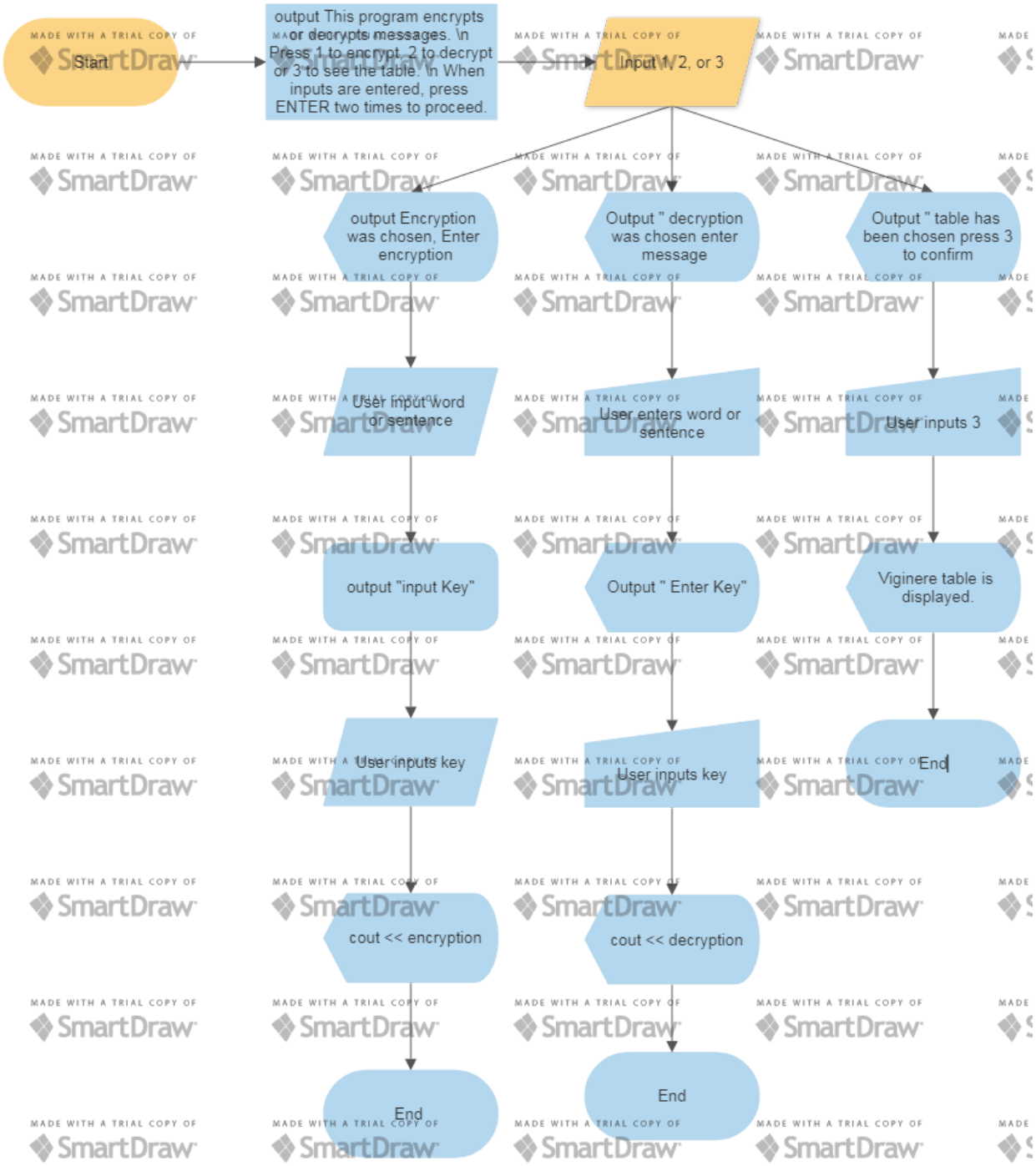
else

ask user to enter correct choice

end if

end

}


Start
output This program encrypts or decrypts messages. \n Press 1 to encrypt, 2 to decrypt or 3 to see the table. \n When inputs are entered, press ENTER two times to proceed.
Input 1, 2, or 3
output Encryption was chosen, Enter encryption
Output " decryption was chosen enter message
Output " table has been chosen press 3 to confirm
User input word or sentence
User enters word or sentence
User inputs 3
output "input Key"
Output " Enter Key"
Viginere table is displayed.
User inputs key
User inputs key
End
cout << encryption
cout << decryption
End
End