 Lebanese American University

Department of Computer Science and Mathematics

Text

Description automatically generated

**CSC 435 – Computer Security**

**Lab 4**

**Instructor: Doctor Ayman Tajjedien**

**Team #1**

Fatima Alzahraa Maarouf - 201900571

Rabih Yamani **-** 201706550

Yehya Alameh - 201905950

Mohamad Chebli **-** 201608008

1. Consider RSA with p = 5 and q = 11.
   1. What are n and z?
   2. Let e be 3. Why is this an acceptable choice for e?
   3. Find d such that de = 1 (mod z) and d < 160.
   4. Encrypt the message m = 8 using the key (n, e). Let c denote the corresponding ciphertext. Show all work. Hint: To simplify the calculations, use the fact: [(a mod n) . (b mod n)] mod n = (a . b) mod n

Answers

1. N = pq = 11 x 5 = 55

Z = (p-1) (q-1) = 4 x 10 = 40

1. e should be less than n which is 55 and it should not have common factors with z making e and z relatively prime
2. de= 1 % z d < 160 ed – 1 % z = 0 where z=40 and e=3

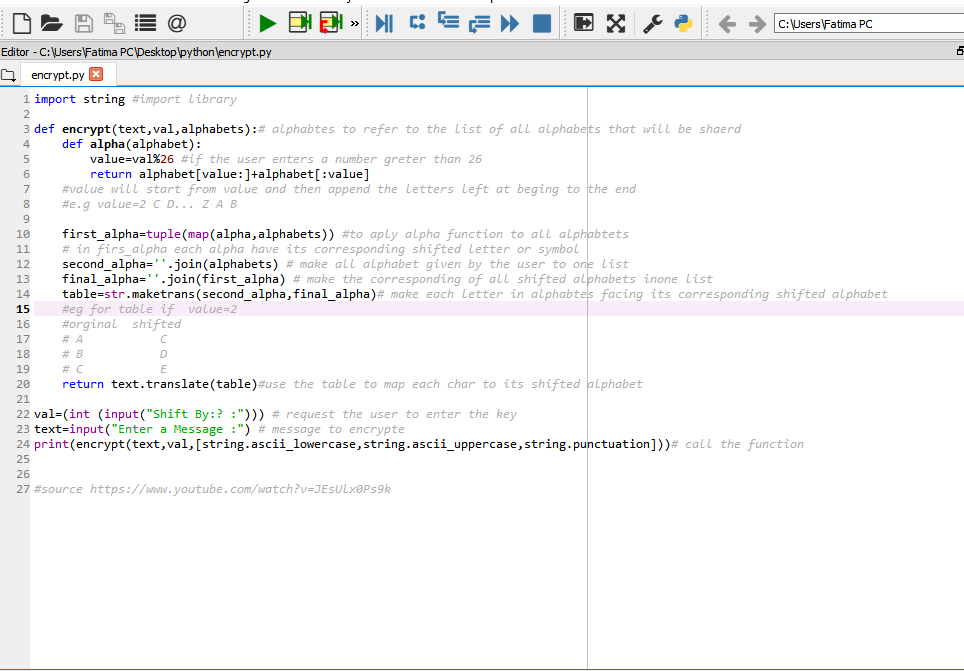
strategy is have to get ed-1 >= 40 if d=27 then ed=81 and ed-1 =80 which is divisible by 40

Thus d=27

1. to encrypt me =83=512

c= me % n 512%55 = 17

1. Encryption



Text Code

import string #import library

def encrypt(text,val,alphabets):# alphabtes to refer to the list of all alphabets that will be shaerd

def alpha(alphabet):

value=val%26 #if the user enters a number greter than 26

return alphabet[value:]+alphabet[:value]

#value will start from value and then append the letters left at beging to the end

#e.g value=2 C D... Z A B

first\_alpha=tuple(map(alpha,alphabets)) #to aply alpha function to all alphabtets

# in firs\_alpha each alpha have its corresponding shifted letter or symbol

second\_alpha=''.join(alphabets) # make all alphabet given by the user to one list

final\_alpha=''.join(first\_alpha) # make the corresponding of all shifted alphabets inone list

table=str.maketrans(second\_alpha,final\_alpha)# make each letter in alphabtes facing its corresponding shifted alphabet

#eg for table if value=2

#orginal shifted

# A C

# B D

# C E

return text.translate(table)#use the table to map each char to its shifted alphabet

val=(int (input("Shift By:? :"))) # request the user to enter the key

text=input("Enter a Message :") # message to encrypte

print(encrypt(text,val,[string.ascii\_lowercase,string.ascii\_uppercase,string.punctuation]))# call the function

#source <https://www.youtube.com/watch?v=JEsUlx0Ps9k>

1. Brute Force

Text

Description automatically generated

Text Code:

import string

print('Please write the message you want to decode')

inputString = input() # request the user to enter the text he want to decode

inputString = inputString.lower() # make it lower case

# dictionary used to decode Containing shifted values

transAlphabet = {}

def createDict(shift):# creating function to check the shifted possible amount

for i in range(0,26): # looping

letter = alphabet[i]

transAlphabet[letter] = alphabet [(i+shift)%26]# we checking possible shifted letters, %26 incase more than 26 loop back to begining

def decodeMessage(message): # creating fuction to decode the msg

cypherText='' # empty string

for letter in message: # looping for each letter in the input

if letter in transAlphabet: # if the letter is an alphabet

letter = transAlphabet [letter]

cypherText = cypherText + letter # keep adding possible shifted letter into the string

# incase space it will add a space

else:

cypherText = cypherText + ' '

print(cypherText) # print the full string with all possible shifted letters

for i in range(0,26): # loop on all possible shift combinations

createDict(i)

decodeMessage(inputString)

# source https://www.youtube.com/watch?v=0wdRh7QVzbw&t=396s