CASER CIPHER

ENCODER AND DECODER

A PROJECT REPORT

Submitted by

P INDIVAR[RA2011042010107]
GARAGAPATI PRUDHVIJA[RA2011042010126]
KOSURU SAI[RA2011042010127]
EEDARA REVANTH [RA2011042010133]
M SRAVAN KUMAR[RA2011042010135]

Under the guidance of

Dr. J JEBA SONIA

(Associate Professor, Department of Data Science and Business Systems)

in partial fulfillment for the award of the

degreeof

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND BUSINESS SYSTEMS

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

S.R.M. Nagar, Kattankulathur, Kancheepuram District

APRIL 2023

BONAFIDE CERTIFICATE

Certified that this project report titled “**CAESER CIPHER ENCODER AND DECODER**” is the bonafide work of “**P INDIVAR, GARAGAPATI PRUDHVIJA, KOSURU SAI, EEDARA REVANTH, M SRAVAN KUMAR**” who carried out the project work under my supervision as a batch. Certified further, that to the best of my knowledge the work reported herein does not form any other project report on the basisof which a degree or award was conferred on an earlier occasion for this or any other candidate.

M. Lakshmi

Date : Project Supervisor Head of the Department

Submitted for University Examination held on ----------------------------in the Department of Data Science and Business Systems, SRM Institute of Science and Technology, Kattankulathur.

Date: Internal Examiner External Examiner

ABSTRACT

Communication through internet brings teams together. Three main parts of communication are sender, medium and receiver. Now days, transmission of data over internet is not safe without any encryption method. All corporate sectors, banking sectors, government sectors and many other sectors are sharing their data through internet. Hackers always try to attack on the transmitted data and try to recover the data. Various techniques are developed for providing the data security. Cryptography is used for safe transmission of data. In cryptography, Encryption is done at sender side and decryption is done at receiver side. In the encryption technique, Caesar cipher is one of the best example. The analysis of Basic Caesar cipher, Delta formation Caesar cipher and XOR Caesar cipher is done on the basis of many parameters like Avalanche Effect, Frequency Test and Brute force attack.

ACKNOWLEDGEMENTS

We express our humble gratitude to **Dr. C. Muthamizhchelvan**, Vice Chancellor (I/C), SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to **Dr. Revathi Venkataraman**, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for his invaluable support.

We wish to thank **Dr. M. Lakshmi**, Professor & Head, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor **Dr. E. Sasikala**, Assistant Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology,for her great support at all the stages of project work.

Our inexpressible respect and thanks to my guide, **Dr. J. Jeba Sonia,** Assistant Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology,for providing me an opportunity to pursue my project under her mentorship. She provided me the freedom and support to explore the research topics of my interest. Her passion for solving real problems and making a difference in the world has always been inspiring.

We sincerely thank staff and students of the Data Science and Business Systems Department, SRM Institute of Science and Technology, for their help during my research. Finally, we wouldlike to thank my parents, our family members and our friends for their unconditional love, constant support and encouragement.

P Indivar [RA2011042010107]

G Prudhvija [RA2011042010126]

Kosuru sai [RA2011042010127]

Eedara Revanth[RA2011042010133]

M Sravan Kumar[RA2011042010135]

TABLE OF CONTENTS

KEYWORDS

Cryptography
Caesar Cipher
Encryption
Decryption
Security
Brute Force attack.

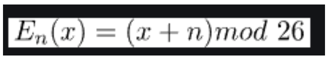
INTRODUCTION

The Caesar cipher is a classic example of ancient cryptography and issaid to have been used by Julius Caesar. The Caesar cipher is based on transposition and involves shifting each letter of the plaintext message by a certain number of letters, historically three. The ciphertext can be decrypted by applying the same number of shifts in the opposite direction. This type of encryption is known as a substitution cipher, due to the substitution of one letter for another in a consistent fashion.

The Caesar Cipher technique is one of the earliest and simplest methods of encryption technique. It’s simply a type of substitution cipher, i.e., each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet. For example with a shift of 1, A would be replaced by B, B would become C, and so on. The method is apparently named after Julius Caesar, who apparently used it to communicate with his officials.

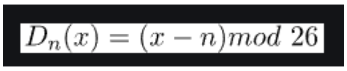
Thus to cipher a given text we need an integer value, known as a shift which indicates the number of positions each letter of the text has been moved down.

The encryption can be represented using modular arithmetic by first transforming the letters into numbers, according to the scheme, A = 0, B = 1,…, Z = 25. Encryption of a letter by a shift n can be described mathematically as.



$$E_n(x) = (x + n) mod\ 26$$

(Encryption Phase with shift n)



$$D_n(x) = (x - n) mod\ 26$$

(Decryption Phase with shift n)

Literature Survey

A.RAJAN (2014) et al. In paper entitled “ADVANCEMENT IN CAESAR CIPHER BY RANDOMIZATION AND DELTA FORMATION” discusses the Caesar cipher by involving the Delta formation method.
By adding this algorithm, it’s not easy for attacker to crack the cipher because the character replaced
is randomly generated. Brute force attacker will also not be able to crack it because the characters are replaced by the other character according to delta formation. This method makes the transmission of data more secure. These delta formation Caesar ciphers divide in three portions.

A. Alphabetic order.

B. Character can be made by the combination of characters.

C. Delta formation.

Computational complexity in this cipher is less then hill cipher and play fair cipher.

ATM cards data are encrypted by using this cryptography.

P. Verma (2016) et al. In paper entitled “EXTENDED CAESAR CIPHER FOR LOW POWERED DEVICES” demonstrate the Caesar cipher by adding new function in basic Caesar cipher which strengthen its impact to withstand against severe attacks. This extended Caesar cipher has also three parts:

A. Key generation Process

B. Encryption Process

C. Decryption Process

Basic Caesar cipher has also similar parts but in this paper, the authors have added more operations in all the process. Like in key generation, factorial function is added and then key value is taken in binary form. In Encryption process, the XOR of key and plaintext is done. By adding these functions, this technique has higher avalanche effect and more equalization in frequency test.

P. Garg et al. In paper entitled “A Review Paper on Cryptography and Significance of Key Length” states the importance of the key length. In this paper, author explains number of algorithm. Public Key cryptography, secrete key cryptography, and Hash Function are these algorithms. A single key is used in secrete key cryptography for both encryption and decryption. Sender uses the key for encrypt the data and then send the key to receiver for decrypt the data. \For encryption and decryption of a message

METHODOLOGY

Modern encryption algorithms are very complicated and (ideally) difficult to break. However, encryption has been around for thousands of years—long before computers existed. Leaders throughout history have used various types of encryption to send messages to allied countries and military leaders during wartime. One famous example is the **Caesar cipher**, used by Julius Caesar in ancient Rome. The Caesar cipher is an example of a **substitution cipher**, where each letter of the alphabet (in English, 26 letters) is replaced by another letter of the alphabet. This is done by "shifting" the entire alphabet by a certain number of spaces. This number is called the **key**. For example, here is a shift of 3 (note how the alphabet "wraps around" from the end):

Original alphabet:	ABCDEFGHIJKLMNOPQRSTUVWXYZ
Shifted alphabet:	DEFGHIJKLMNOPQRSTUVWXYZABC

To encode a message, each letter in the original message (called the **plaintext**) is replaced with the letter directly below it in the shifted alphabet (A becomes D, B becomes E, and so on). The result is called the **ciphertext**. Here is a plaintext message encrypted using a shift of 3:

Plaintext:	THIS IS A SECRET MESSAGE
Ciphertext:	WKLV LV D VHFUHW PHVVDJH

In order to share secret messages, you and your friend need to agree on a key in advance. Then, you can use the key to encrypt messages, and your friend can use the same key (shifting the alphabet in the opposite direction) to decrypt them. Anyone who intercepts the messages will be unable to read them if they do not know the key.

But, what if a very determined person wants to crack your code? How could they do it? One major weakness of the Caesar cipher is that it is vulnerable to a **brute-force attack**, an attack that tries all possible keys to decrypt a message. Since there are only 25 possible keys in English (using a key of 26 gets you back to the original alphabet), for very short encrypted messages it would not take you long to manually try all the keys. For example, here is a short encrypted message (note that this simple version of the Caesar cipher only changes letters; punctuation remains unchanged).

RPC NDJ RGPRZ IWT RDST?

What happens if we try to decrypt this message using a shift of 1? That would mean that during encryption, A became B, B became C, and so on. To decrypt the message, we work backwards (B becomes, A, C becomes B, and so on). If we try this on the entire message, we get this result:

QOB MCI QFOQY HVS QCRS?

The message is still gibberish, so we know that 1 is not the key (assuming the original message was actually in English!). Can you try to decrypt the message using the other 24 possible keys? Keep trying different keys until you get a sentence that makes sense in English. How long does it take you to do it by hand?

Another method that can be used to crack a Caesar cipher (or any other type of substitution cipher) is **frequency analysis**. Frequency analysis is based on the fact that certain letters appear with different frequencies in English writing—for example, E usually occurs the most often, followed by T and A; whereas Q and Z appear the least often (Figure 1).

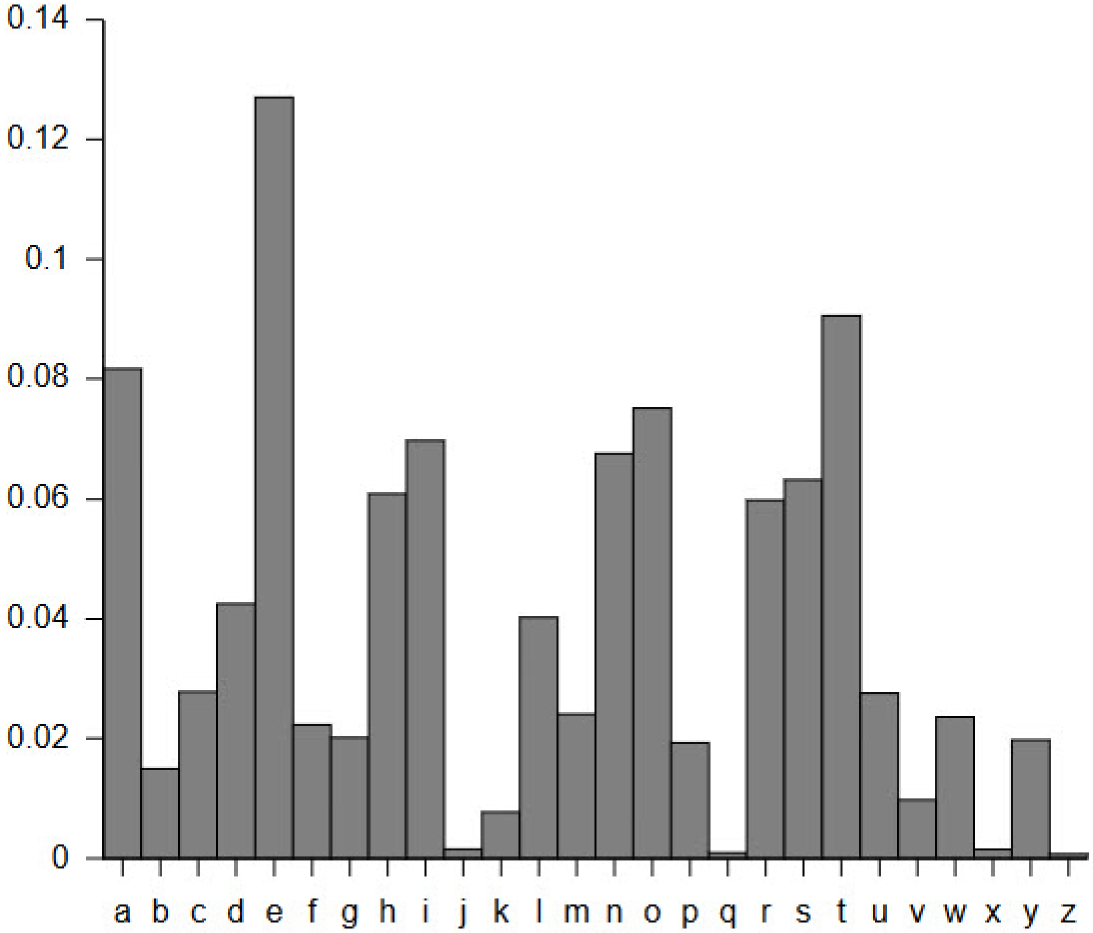


Figure 1. Letter frequencies in the English language.

Plaintext: It is a simple message written by the user.

Ciphertext: It is an encrypted message after applying some technique.

The formula of encryption is:

$E_n (x) = (x + n) \bmod 26$

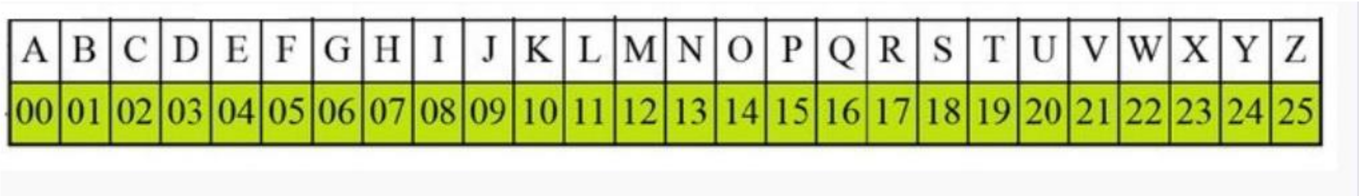
The formula of decryption is:

$D_n (x) = (xi - n) \bmod 26$

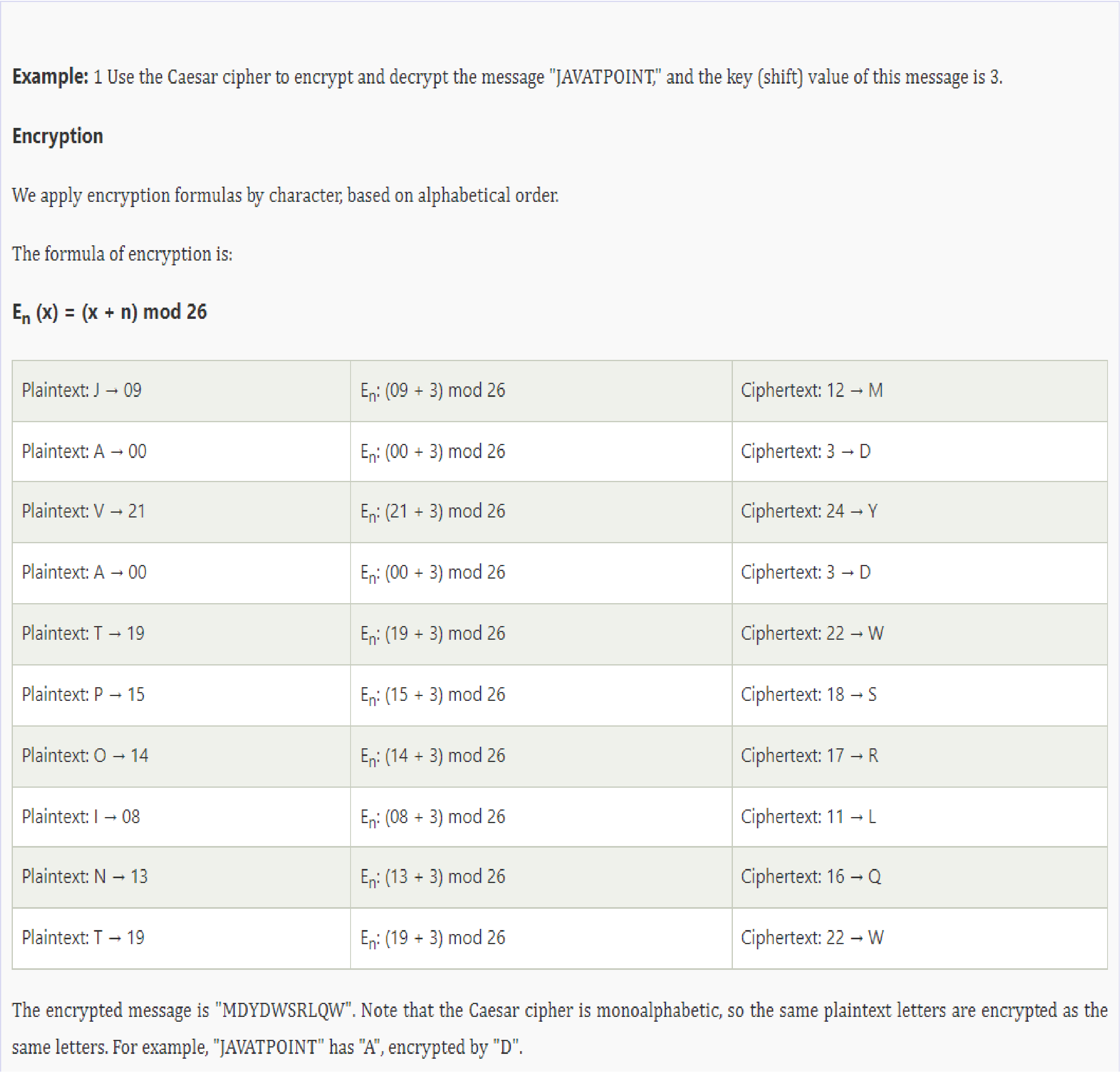
If any case (Dn) value becomes negative (-ve), in this case, we will add 26 in the negative value.

Where,

E denotes the encryption
D denotes the decryption
x denotes the letters value
n denotes the key value (shift value)



A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25



Example: 1 Use the Caesar cipher to encrypt and decrypt the message "JAVATPOINT," and the key (shift) value of this message is 3.

Encryption

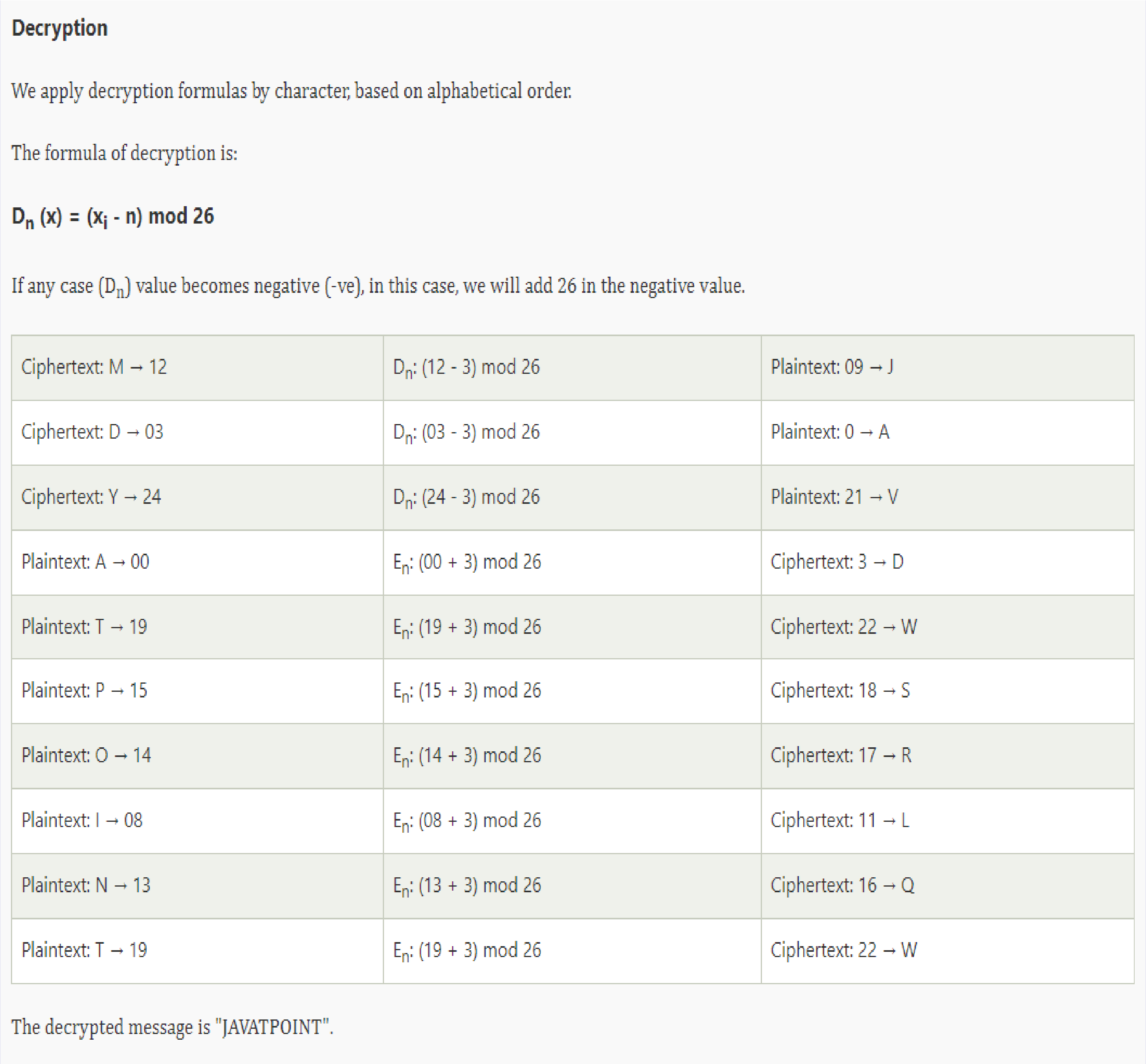
We apply encryption formulas by character, based on alphabetical order.

The formula of encryption is:

E_n (x) = (x + n) mod 26

Plaintext: J → 09	E_n: (09 + 3) mod 26	Ciphertext: 12 → M
Plaintext: A → 00	E_n: (00 + 3) mod 26	Ciphertext: 3 → D
Plaintext: V → 21	E_n: (21 + 3) mod 26	Ciphertext: 24 → Y
Plaintext: A → 00	E_n: (00 + 3) mod 26	Ciphertext: 3 → D
Plaintext: T → 19	E_n: (19 + 3) mod 26	Ciphertext: 22 → W
Plaintext: P → 15	E_n: (15 + 3) mod 26	Ciphertext: 18 → S
Plaintext: O → 14	E_n: (14 + 3) mod 26	Ciphertext: 17 → R
Plaintext: I → 08	E_n: (08 + 3) mod 26	Ciphertext: 11 → L
Plaintext: N → 13	E_n: (13 + 3) mod 26	Ciphertext: 16 → Q
Plaintext: T → 19	E_n: (19 + 3) mod 26	Ciphertext: 22 → W

The encrypted message is "MDYDWSRLQW". Note that the Caesar cipher is monoalphabetic, so the same plaintext letters are encrypted as the same letters. For example, "JAVATPOINT" has "A", encrypted by "D".



Decryption

We apply decryption formulas by character, based on alphabetical order.

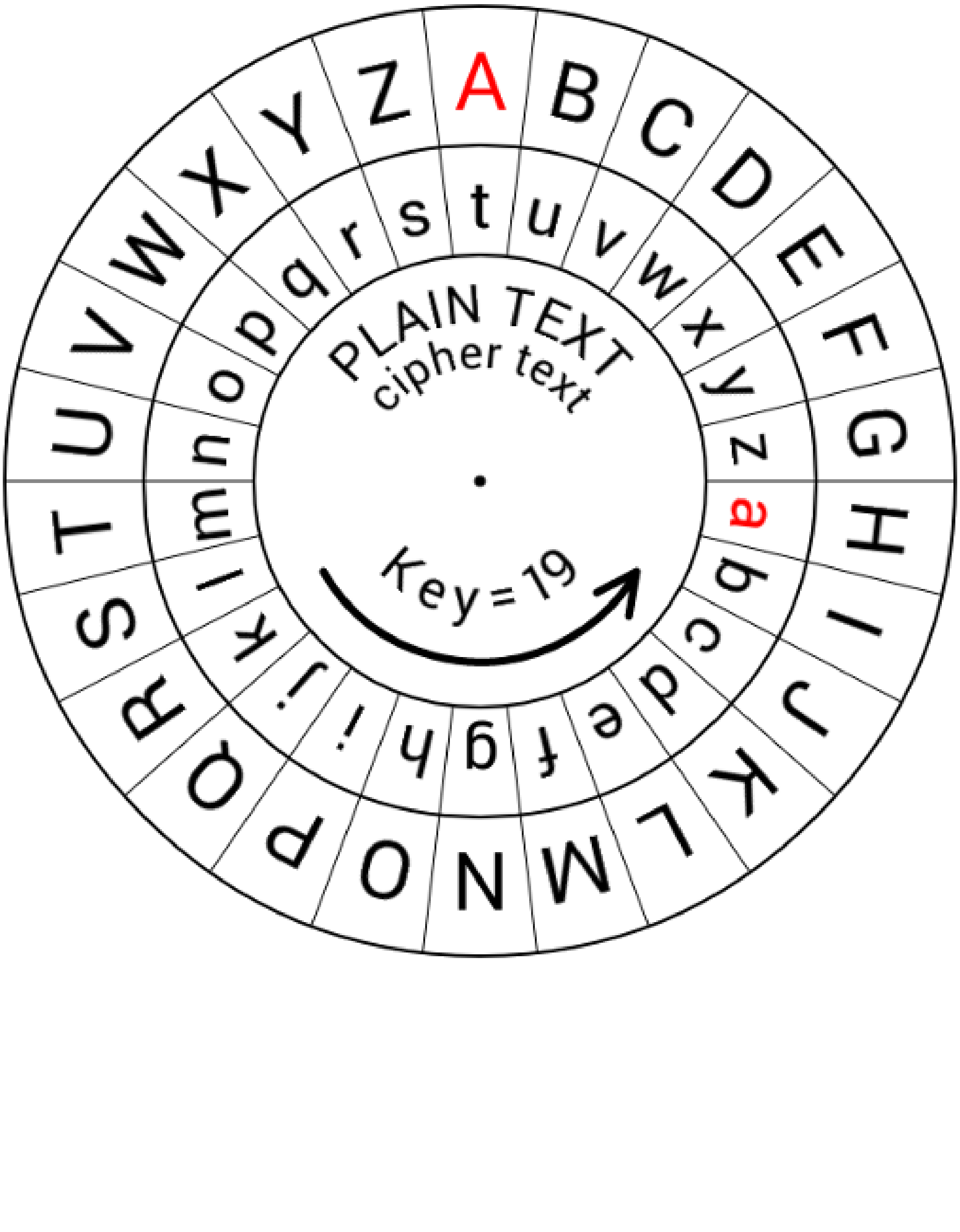
The formula of decryption is:

$$D_n(x) = (x_i - n) \bmod 26$$

If any case (D_n) value becomes negative (-ve), in this case, we will add 26 in the negative value.

Ciphertext: M → 12	D_n: (12 - 3) mod 26	Plaintext: 09 → J
Ciphertext: D → 03	D_n: (03 - 3) mod 26	Plaintext: 0 → A
Ciphertext: Y → 24	D_n: (24 - 3) mod 26	Plaintext: 21 → V
Plaintext: A → 00	E_n: (00 + 3) mod 26	Ciphertext: 3 → D
Plaintext: T → 19	E_n: (19 + 3) mod 26	Ciphertext: 22 → W
Plaintext: P → 15	E_n: (15 + 3) mod 26	Ciphertext: 18 → S
Plaintext: O → 14	E_n: (14 + 3) mod 26	Ciphertext: 17 → R
Plaintext: I → 08	E_n: (08 + 3) mod 26	Ciphertext: 11 → L
Plaintext: N → 13	E_n: (13 + 3) mod 26	Ciphertext: 16 → Q
Plaintext: T → 19	E_n: (19 + 3) mod 26	Ciphertext: 22 → W

The decrypted message is "JAVATPOINT".


A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
t u v w x y z a b c d e f g h i j k l m n o p q r s
PLAIN TEXT
cipher text
Key = 19

IMPLEMENTATION

For example, look at this encrypted text:

```
L PZ AOL TVZA MYLXBLUA SLAALY PU AOPZ ZLUALUJL
```

If you count the letters, you will notice that L appears more often than any other letter (9 times). It is therefore a safe guess that L stands for E if this is a substitution cipher and the original message was in English. L is 7 spaces away from E in the alphabet. What happens if you work backwards to decrypt this message using a key of 7 (L becomes E, M becomes F, and so on)?

Doing a brute-force attack or frequency analysis by hand can be easy for very short messages, but can become time-consuming for entire paragraphs or pages of text. This is where writing a computer program to do the work for you comes in handy. In the procedure of this project, you will write your own programs that can first encrypt plaintext using a Caesar cipher, and then attempt to decrypt the text using both a brute-force attack and frequency analysis.

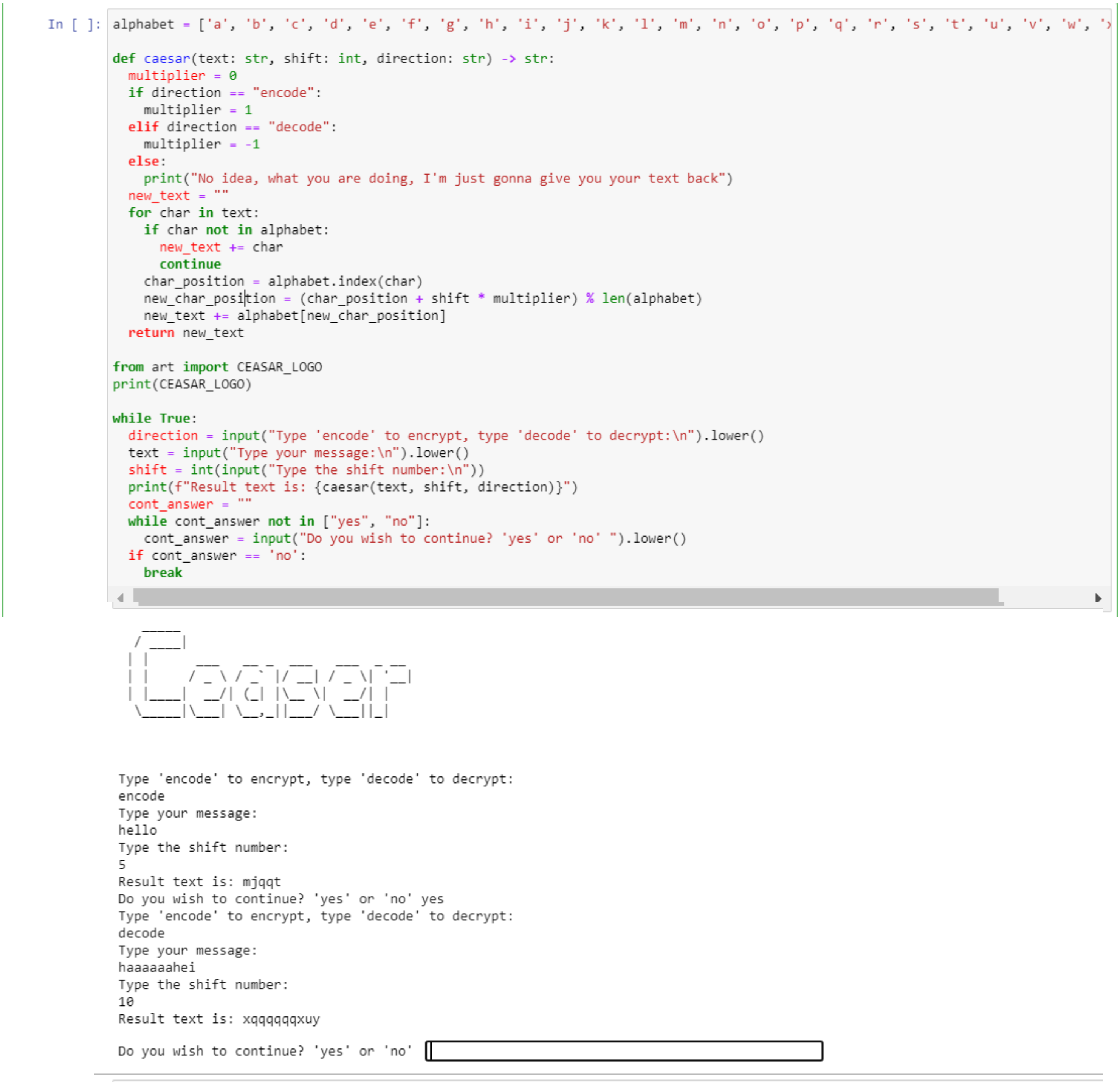
RESULTS AND DISCUSSION

Avalanche effect Avalanche effect determines the strength of any cryptography technique. Avalanche effect is the ratio of number of flipped bits to the total number of bits in the cipher text. High avalanche effect is produced by good encryption technique. $\boldsymbol{Avalanche\ effect\ No.of\ flipped\ bits\ in\ the\ ciphertext\ No.of\ bits\ in\ the\ ciphertext} * 100$ (1) TABLE VII. AVALANCHE EFFECT ANALYSIS Fig. 3 Comparison of different algorithms based on Avalanche Effects Figure demonstrates the avalanche effect of various encryption techniques. Encryption techniques like basic Caesar cipher, XOR Caesar Cipher and Delta Caesar Cipher are compared on the basis of avalanche effect. As the graph shown that the Delta Caesar cipher have higher avalanche effect. b) Brute Force Attack In case of brute force attack, an attacker attempts all the feasible set of combinations until plaintext is retrieved from the cipher text. Traditional Caesar cipher is mostly affected because of less number of keys. Brute force attack can be executed by thoroughly scanning all possible keys until the accurate plaintext is found. Brute force attacks can be minimized by making complex data that cannot be easily breached. One way to find the strength of an encryption system is how long it would theoretically take an attacker to conduct a successful brute force attack on it. Brute force attack is difficult to carry in Advance Caesar cipher than Basic Caesar cipher

CONCLUSION AND FUTURE ENHANCEMENT

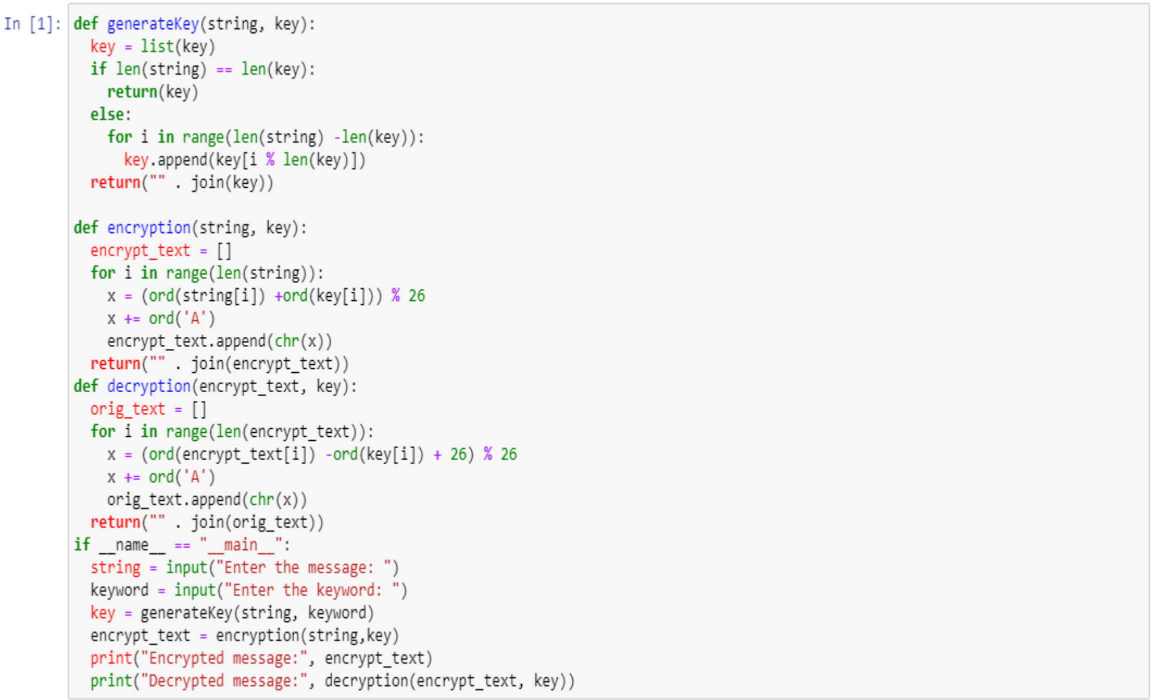
Security plays a major role in wireless type of medium because when we transmit our data wirelessly, it can be access by the third party or an outsider. Cryptography plays an important role for safe transmission of data. Data is encrypted and decrypted by many techniques. Caesar cipher is important technique which has less complex, limited power consumption and less memory consumption. Many advancement is done in Caesar cipher to make it more secure. Delta formation Caesar cipher and XOR Caesar cipher are the example of advanced Caesar cipher. Main factor which effect the Caesar cipher is brute force attack. Attacker tries all the possible set of key to recover the data. But the new techniques have substitution in key which make the Caesar cipher more secure.

PROGRAM CODES



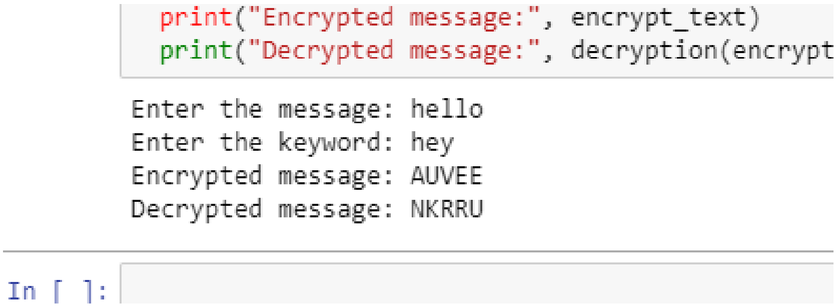
```
In [ ]: alphabet = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x

        def caesar(text: str, shift: int, direction: str) -> str:
          multiplier = 0
          if direction == "encode":
            multiplier = 1
          elif direction == "decode":
            multiplier = -1
          else:
            print("No idea, what you are doing, I'm just gonna give you your text back")
          new_text = ""
          for char in text:
            if char not in alphabet:
              new_text += char
              continue
            char_position = alphabet.index(char)
            new_char_position = (char_position + shift * multiplier) % len(alphabet)
            new_text += alphabet[new_char_position]
          return new_text

        from art import CEASAR_LOGO
        print(CEASAR_LOGO)

        while True:
          direction = input("Type 'encode' to encrypt, type 'decode' to decrypt:\n").lower()
          text = input("Type your message:\n").lower()
          shift = int(input("Type the shift number:\n"))
          print(f"Result text is: {caesar(text, shift, direction)}")
          cont_answer = ""
          while cont_answer not in ["yes", "no"]:
            cont_answer = input("Do you wish to continue? 'yes' or 'no' ").lower()
          if cont_answer == 'no':
            break
```

```
   _____
  / ____|
 | |        ___   __ _  ___   ___  _ __
 | |       / _ \ / _` |/ __| / _ \| '__|
 | |____|  __/| (_| |\__ \|  __/| |
  \_____|\___| \__,_||___/ \___||_|



Type 'encode' to encrypt, type 'decode' to decrypt:
encode
Type your message:
hello
Type the shift number:
5
Result text is: mjqqt
Do you wish to continue? 'yes' or 'no' yes
Type 'encode' to encrypt, type 'decode' to decrypt:
decode
Type your message:
haaaaaahei
Type the shift number:
10
Result text is: xqqqqqqxuy

Do you wish to continue? 'yes' or 'no'
```



In [1]:

```
def generateKey(string, key):
  key = list(key)
  if len(string) == len(key):
    return(key)
  else:
    for i in range(len(string) -len(key)):
      key.append(key[i % len(key)])
  return("" . join(key))

def encryption(string, key):
  encrypt_text = []
  for i in range(len(string)):
    x = (ord(string[i]) +ord(key[i])) % 26
    x += ord('A')
    encrypt_text.append(chr(x))
  return("" . join(encrypt_text))
def decryption(encrypt_text, key):
  orig_text = []
  for i in range(len(encrypt_text)):
    x = (ord(encrypt_text[i]) -ord(key[i]) + 26) % 26
    x += ord('A')
    orig_text.append(chr(x))
  return("" . join(orig_text))
if __name__ == "__main__":
  string = input("Enter the message: ")
  keyword = input("Enter the keyword: ")
  key = generateKey(string, keyword)
  encrypt_text = encryption(string,key)
  print("Encrypted message:", encrypt_text)
  print("Decrypted message:", decryption(encrypt_text, key))
```



```
  print("Encrypted message:", encrypt_text)
  print("Decrypted message:", decryption(encrypt
```

```
Enter the message: hello
Enter the keyword: hey
Encrypted message: AUVEE
Decrypted message: NKRRU
```

In [ ]:

REFERENCES

[1] L .C HAN, N.M. MAHYUDDIN, ”AN IMPLEMENTATION OF CAESAR CIPHER AND XOR ENCRYPTION TECHNIQUE IN A SECURE WIRELESS COMMUNICATION”,IEEE CONFERENCE, PP.111-116, 2014.

[2] A. Rajan, D. Balakumaran, “Advancement in Caesar cipher by randomization and delta formation”, ICICES, 2014.

[3] P. Garg1, J. Singh Dilawari, “A Review Paper on Cryptography and Significance of Key Length”, International Journal of Computer Science and Communication Engineering, pp.88-91, 2012

[4] O. Abraham, “An improved Caesar cipher (ICC) algorithm”, International Journal Of Engineering Science & Advanced Technology, pp-1199-1202, 2012.

[5]nD. Thakral, “A Review on Security Issues in Wireless Sensor Networks”, International Journal of Advanced Research in Computer Science and Software Engineering, pp-269-273, 2012.

[6] M. Roopak, “Review of Threats in Wireless Sensor Networks”, International Journal of Computer Science and Information Technologies, pp-25-31, 2014.

[7] P. Patni, “Implementation and Result Analysis of Polyalphabetic Approach to Caesar Cipher”, IOSR Journal of Computer Engineering (IOSR-JCE), vol. 16, pp. 100-106, 2014.

[8] B. Purnama, H. Rohayani. AH, “A New Modified Caesar Cipher Cryptography Method With Legible Ciphertext From A Message To Be Encrypted”, International Conference on Computer Science and Computational Intelligence, pp.195 – 204, 2015.

[9] Fahad NaimNife, A New Modified Cesar Cipher Cryptographic Method Along With Rail Fence to Encrypt Message, International Journal of Engineering Research and Development e-ISSN: 2278- 067X, p-ISSN: 2278-800X, www.ijerd.com Volume 11, Issue 02 (February 2015)

[10]Goyal,Khasis. Kinger, Supriya.Modified Caesar Cipherfor Better Security Enhancement. International Journal of Computer Aplications (0975- 8887) Volume 73 – No.3 July 2013.

[11]Senthil, K., K. Prasanthi, and R. Rajaram. "A modern avatar of Julius Caesar and Vigenere cipher." Computational Intelligence and Computing Research (ICCIC), 2013 IEEE International Conference on. IEEE, 2013.

[12]Purnama, Benni, and AH Hetty Rohayani. "A New Modified Caesar Cipher Cryptography Method with Legible Ciphertext from a Message to be Encrypted." Procedia Computer Science 59 (2015): 195- 204