33_IT_A_MOHIT_GUPTA

Cryptography & Network Security

TE IT

Experiment No 1 (Part A)

Implementation of Ceaser Cipher Algorithm

Aim: To write a Python program to perform encryption using conventional cryptography technique – Ceaser Cipher and its cryptanalysis using Brute Force attack.

Learning Objectives:

- To understand the substitution cipher.
- To describe the Ceaser cipher algorithm and encrypt the plain text.
- To describe cryptanalysis and do the cryptanalysis Ceaser Cipher algorithm by Brute Force attack.

Theory:

Substitution Cipher Techniques:

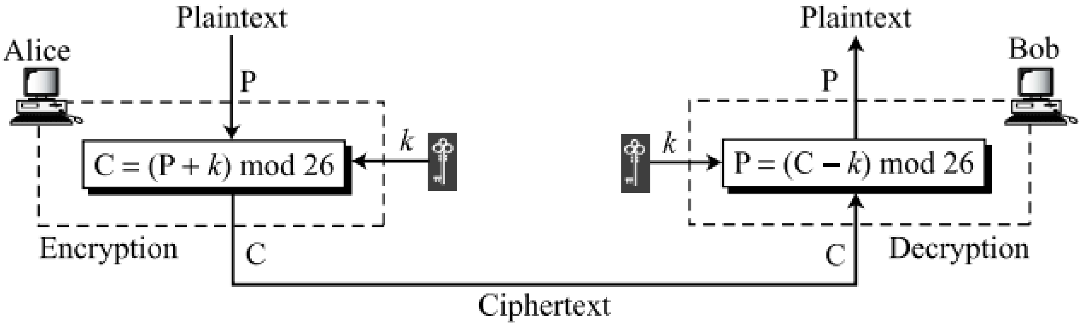
In the substitution cipher technique, the characters of plain text messages are replaced by other characters, numbers, or symbols.

Algorithm of Caesar Cipher:

The algorithm of Caesar cipher holds the following features −

- Caesar Cipher Technique is a simple and easy method of encryption technique.
- It is a simple type of substitution cipher.
- Each letter of plain text is replaced by a letter with some fixed number of positions down with the alphabet.

The following diagram depicts the working of Caesar cipher algorithm implementation −



33_IT_A_MOHIT_GUPTA

Process:

- The plain text character is traversed one at a time.
- For each character in the given plain text, transform the given character as per the rule depending on the procedure of encryption and decryption of text.
- After the steps are followed, a new string is generated which is referred to as cipher text.

Caesar Cipher Cryptanalysis:

Caesar ciphers are vulnerable to ciphertext-only attacks using exhaustive key searches (brute-force attacks). This technique does not demand much effort and is relatively simple for a hacker.

The key space in the Ceaser Cipher algorithm is only 25. One can try keys from 1 to 25 to break the Ceaser cipher.

Results:

a. Encrypt your full name using Ceaser Substitution Cipher with key = 5. s. Ignore the space between words.
b. Hack the given Ceaser Cipher text by applying a Brute Force attack.

Conclusion:

In conclusion, the experiment on Caesar Cipher Cryptanalysis demonstrated that the Caesar cipher is vulnerable to simple attacks like brute force and frequency analysis due to its limited key space. Encrypting the message multiple times with different keys using the Caesar cipher does not provide significant security enhancements. For stronger security, it is essential to employ more robust encryption algorithms with larger key spaces and advanced cryptographic techniques.
Answer the following questions.

1. Why is the key space for the Ceaser Cipher algorithm 25?

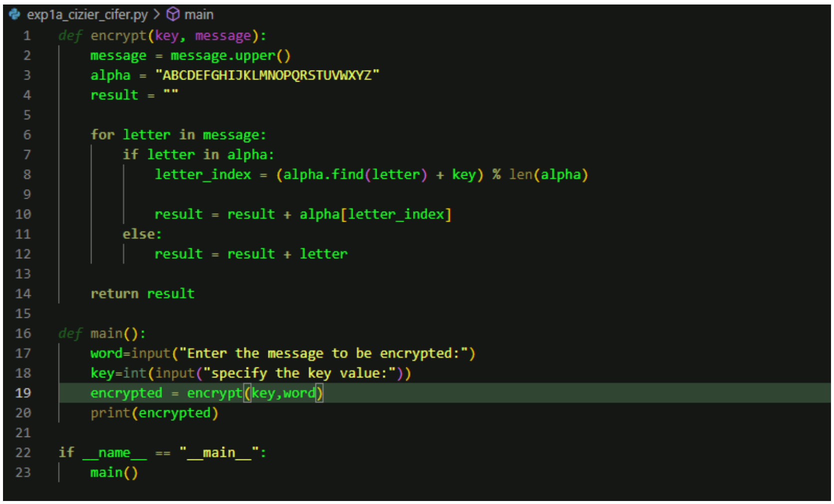
Ans: The key space for the Caesar Cipher algorithm is 25 because using a key value of 26 or 0 would have no effect on the plaintext. Thus, there are 25 possible effective key values for encryption and decryption.

2. Alice can use only the Ceaser cipher on her computer to send a message to a friend. She thinks that the message is more secure if she encrypts the message two times, each time with a different key. Is she right? Defend your answer.

Ans: Encrypting the message twice with different keys using the Caesar cipher does not significantly increase security. The Caesar cipher has a limited key space, making it vulnerable to attacks like frequency analysis. For better security, stronger encryption methods should be used.

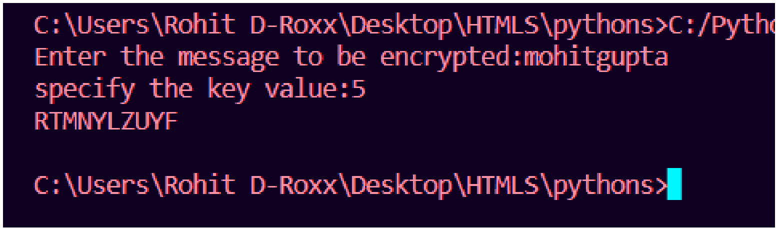
33_IT_A_MOHIT_GUPTA

c. Outputs:

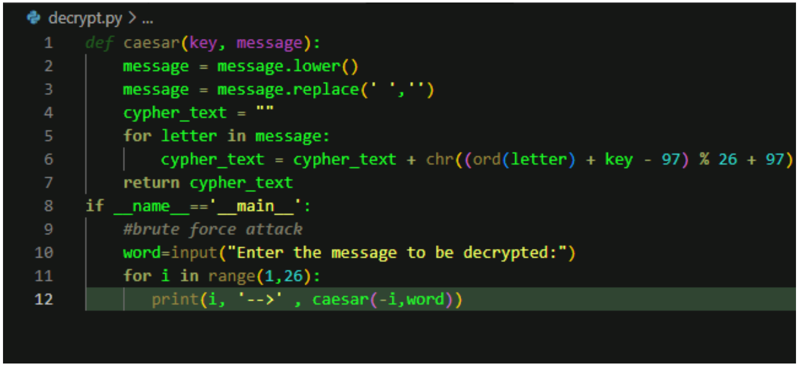


exp1a_cizier_cifer.py > main

```python
def encrypt(key, message):
    message = message.upper()
    alpha = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    result = ""

    for letter in message:
        if letter in alpha:
            letter_index = (alpha.find(letter) + key) % len(alpha)

            result = result + alpha[letter_index]
        else:
            result = result + letter

    return result

def main():
    word=input("Enter the message to be encrypted:")
    key=int(input("specify the key value:"))
    encrypted = encrypt(key,word)
    print(encrypted)

if __name__ == "__main__":
    main()
```

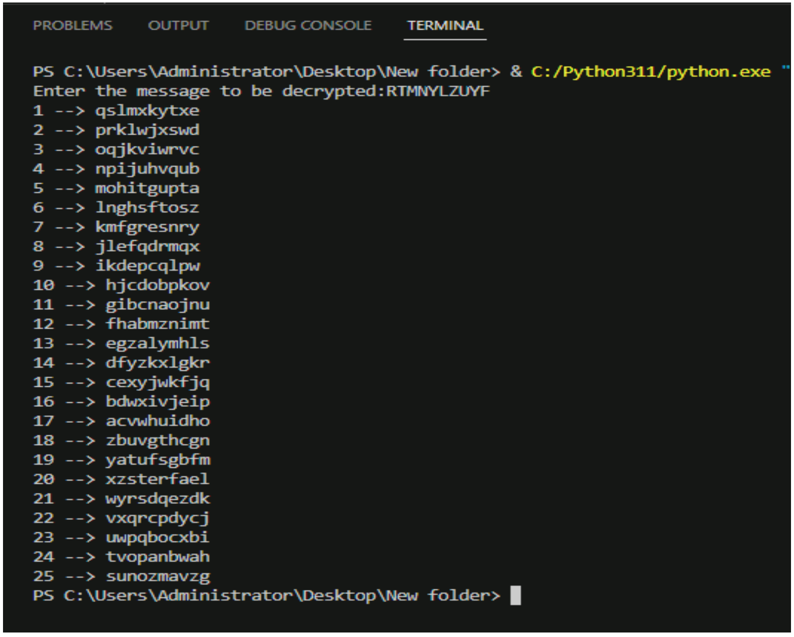


```
C:\Users\Rohit D-Roxx\Desktop\HTMLS\pythons>C:/Pytho
Enter the message to be encrypted:mohitgupta
specify the key value:5
RTMNYLZUYF

C:\Users\Rohit D-Roxx\Desktop\HTMLS\pythons>
```



decrypt.py > ...

```python
def caesar(key, message):
    message = message.lower()
    message = message.replace(' ','')
    cypher_text = ""
    for letter in message:
        cypher_text = cypher_text + chr((ord(letter) + key - 97) % 26 + 97)
    return cypher_text
if __name__=='__main__':
    #brute force attack
    word=input("Enter the message to be decrypted:")
    for i in range(1,26):
        print(i, '-->' , caesar(-i,word))
```



```
PS C:\Users\Administrator\Desktop\New folder> & C:/Python311/python.exe "
Enter the message to be decrypted:RTMNYLZUYF
1 --> qslmxkytxe
2 --> prklwjxswd
3 --> oqjkviwrvc
4 --> npijuhvqub
5 --> mohitgupta
6 --> lnghsftosz
7 --> kmfgresnry
8 --> jlefqdrmqx
9 --> ikdepcqlpw
10 --> hjcdobpkov
11 --> gibcnaojnu
12 --> fhabmznimt
13 --> egzalymhls
14 --> dfyzkxlgkr
15 --> cexyjwkfjq
16 --> bdwxivjeip
17 --> acvwhuidho
18 --> zbuvgthcgn
19 --> yatufsgbfm
20 --> xzsterfael
21 --> wyrsdqezdk
22 --> vxqrcpdycj
23 --> uwpqbocxbi
24 --> tvopanbwah
25 --> sunozmavzg
PS C:\Users\Administrator\Desktop\New folder>
```

33_IT_A_MOHIT_GUPTA

Practical Learning Outcomes:

After performing the practical, the learner is able to:	Marked √
i. Explain the concept of substitution cipher. ii. Describe the Ceaser cipher algorithm and encrypt the given plain text. iii. Describe cryptanalysis and do the cryptanalysis Ceaser Cipher algorithm by Brute Force attack	

Outcome	**PLO 1**	**PLO 2**	**PLO 3**	**Performance**	**Attendance**	**Total Score**	**IT DEPARTMENT- TCET**
Weight	**20**	**20**	**20**	**20**	**20**	**100**	Date of Performance: _______ Date of Correction: ___________ Roll No: _________ Marks: ______/100 Signature of Faculty:
Score							