Ex:1 Caesar Cipher - Implementation and Cryptanalysis

Aim:

1. To implement the Caesar cipher algorithm
2. To cryptanalyse the Caesar cipher text by
    a. Brute Force Attack
    b. Frequency Analysis Attack

Algorithm:

The transformation can be represented by aligning two alphabets; the cipher alphabet is the plain alphabet rotated left or right by some number of positions.

Encryption: The encryption can also 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 *x* by a shift *n* can be described mathematically as,

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

Decryption: Decryption is performed similarly,

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

The replacement remains the same throughout the message, so the cipher is classed as a type of *mono-alphabetic substitution*, as opposed to *polyalphabetic substitution*.

Cryptanalysis :

The Caesar cipher can be easily broken even in a ciphertext-only scenario. Two situations can be considered:

- an attacker knows (or guesses) that some sort of simple substitution cipher has been used, but not specifically that it is a Caesar scheme; - **Frequency Analysis**
- an attacker knows that a Caesar cipher is in use, but does not know the shift value; - **Brute Force Attack**

Frequency Analysis:

If there is a sufficiently large ciphertext, it can be decrypted by comparing the frequency of letters in the cipher text against the frequency of letters in standard English. If the frequency of the letter in the cipher text is almost the same as the frequency of letters in standard English, we can find out which letter is substituted as there exists a one to one relationship between each letter.

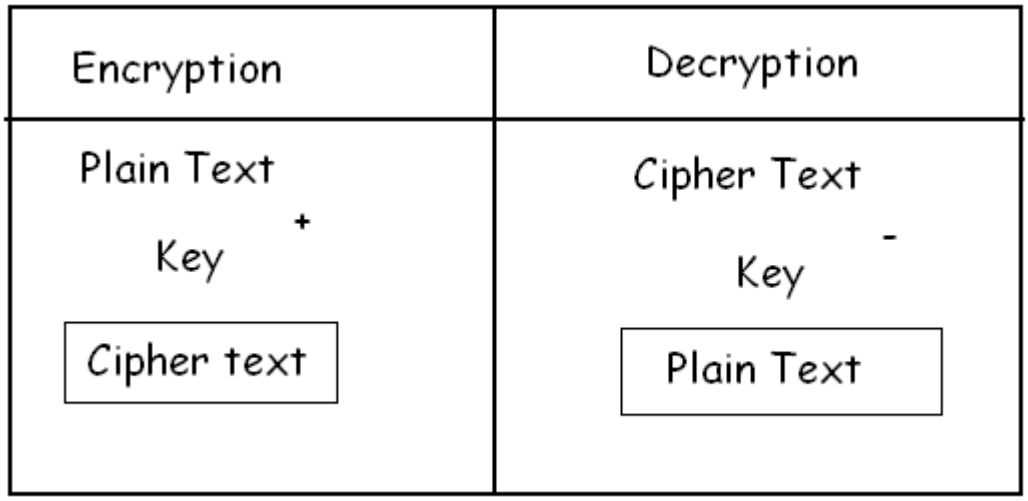
Letter	Frequency
E	0.127
T	0.097
I	0.075
A	0.073
O	0.068
N	0.067
S	0.067
R	0.064
H	0.049
C	0.045
L	0.040
D	0.031
P	0.030
Y	0.027
U	0.024
M	0.024
F	0.021
B	0.017
G	0.016
W	0.013
V	0.008
K	0.008
X	0.005
Q	0.002
Z	0.001
J	0.001

The frequency table of the letter appears in English

Brute Force Attack:

It involves systematically checking all possible keys until the correct key is found. The key length used in the encryption determines the practical feasibility of performing a brute force attack, with longer keys exponentially more difficult to crack than shorter ones.

Block Diagram :



Program : // To be implemented using Java GUI (Applet)

Sample Output:

The following example is with a shift of three, so that a B in the plaintext becomes E in the ciphertext.

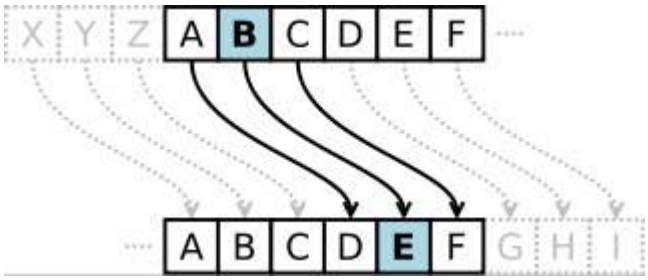
Encryption: Key = 3.

```
Plain:    ABCDEFGHIJKLMNOPQRSTUVWXYZ
Cipher:   DEFGHIJKLMNOPQRSTUVWXYZABC
```

Decryption:

```
Ciphertext: WKH TXLFN EURZQ IRA MXPSV RYHU WKH ODCB GRJ
Plaintext:  the quick brown fox jumps over the lazy dog
```

Reference :



Result :

Thus,

- Caesar Cipher Algorithm has been implemented successfully
- Cryptanalysis of the Caesar Cipher texts by Brute Force Attack and Frequency Analysis Attack is done successfully