

# Cryptography---19CS412-classical-techniques

## Caesar Cipher

Caesar Cipher using with different key values

### AIM:

To encrypt and decrypt the given message by using Caesar Cipher encryption algorithm.

### DESIGN STEPS:

#### Step 1:

Design of Caesar Cipher algorithm

#### Step 2:

Implementation using C or python code

#### Step 3:

1. In Caesar Cipher each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet.
2. For example, with a left shift of 3, D would be replaced by A, E would become B, and so on.
3. 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.
4. Encryption of a letter x by a shift n can be described mathematically as,  $En(x) = (x + n) \bmod 26$
5. Decryption is performed similarly,  $Dn(x) = (x - n) \bmod 26$

### PROGRAM:

Developed By: Muhammad Afshan A

Ref No.: 212223100035

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <string.h>
```

```
int main() {
    char message[100]; // Array to store the message
    int key;

    printf("Enter the message to encrypt: ");
    fgets(message, sizeof(message), stdin); // Read input from the user

    // Remove trailing newline from fgets
    message[strcspn(message, "\n")] = '\0';

    printf("Enter the Caesar Cipher key (an integer): ");
    scanf("%d", &key); // Read the key from the user

    // Encryption logic (directly in main)
    for (int i = 0; message[i] != '\0'; i++) {
        char c = message[i];
```

```
    if (c >= 'A' && c <= 'Z') {
        message[i] = ((c - 'A' + key) % 26 + 26) % 26 + 'A';
    } else if (c >= 'a' && c <= 'z') {
        message[i] = ((c - 'a' + key) % 26 + 26) % 26 + 'a';
    }
}

printf("Encrypted Message: %s\n", message);

// Decryption logic (directly in main)
for (int i = 0; message[i] != '\0'; i++) {
    char c = message[i];

    if (c >= 'A' && c <= 'Z') {
        message[i] = ((c - 'A' - key) % 26 + 26) % 26 + 'A';
    } else if (c >= 'a' && c <= 'z') {
        message[i] = ((c - 'a' - key) % 26 + 26) % 26 + 'a';
    }
}

printf("Decrypted Message: %s\n", message);

return 0;
}
```

## OUTPUT:

**Output**

Clear

Enter the message to encrypt: MUHAMMAD AFSHAN A  
Enter the Caesar Cipher key (an integer): 22  
Encrypted Message: IQDWIIWZ WBODWJ W  
Decrypted Message: MUHAMMAD AFSHAN A

=== Code Execution Successful ===

Enter the message to encrypt: MUHAMMAD AFSHAN A  
Enter the Caesar Cipher key (an integer): 22  
Encrypted Message: IQDWIIWZ WBODWJ W  
Decrypted Message: MUHAMMAD AFSHAN A

## RESULT:

The program is executed successfully