

# Cryptography---19CS412-classical-techqniques

Name:SARWESHVARAN A

Regno:212223230198

## Caeser Cipher

Caeser Cipher using with different key values

### AIM:

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

### DESIGN STEPS:

#### Step 1:

Design of Caesar Cipher algorithm

#### Step 2:

Implementation using C or pyhton code

#### Step 3:

1. In Ceaser 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:

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

int main()
{
    int key;
    char s[1000];

    printf("Enter a plaintext to encrypt:\n");
    fgets(s, sizeof(s), stdin);
    printf("Enter key:\n");
    scanf("%d", &key);

    int n = strlen(s);

    for (int i = 0; i < n; i++)
    {
        char c = s[i];
        if (c >= 'a' && c <= 'z')
```



```

    {
        s[i] = 'a' + (c - 'a' + key) % 26;
    }
    else if (c >= 'A' && c <= 'Z')
    {
        s[i] = 'A' + (c - 'A' + key) % 26;
    }
}
printf("Encrypted message: %s\n", s);

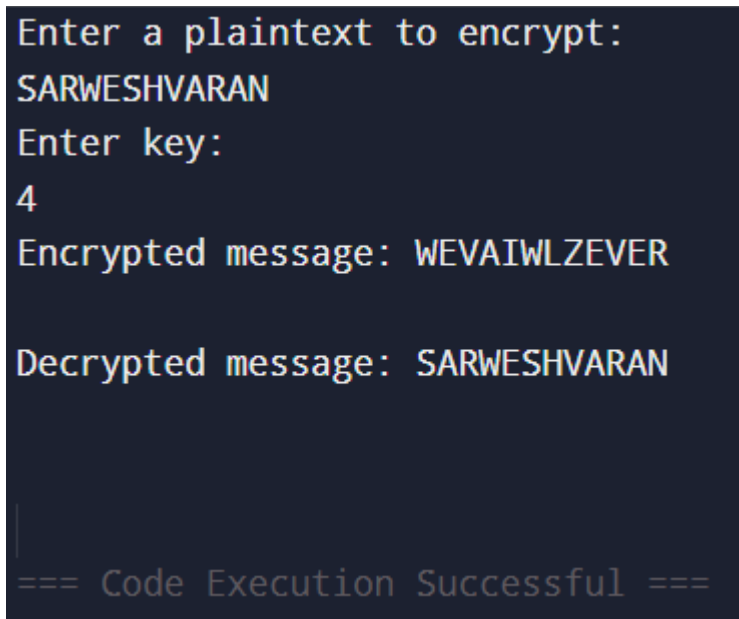
for (int i = 0; i < n; i++)
{
    char c = s[i];
    if (c >= 'a' && c <= 'z')
    {
        s[i] = 'a' + (c - 'a' - key + 26) % 26;
    }
    else if (c >= 'A' && c <= 'Z')
    {
        s[i] = 'A' + (c - 'A' - key + 26) % 26;
    }
}
printf("Decrypted message: %s\n", s);

return 0;
}

```

## OUTPUT:

### Simulating Caesar Cipher



```

Enter a plaintext to encrypt:
SARWESHVARAN
Enter key:
4
Encrypted message: WEVAIWLZEVER

Decrypted message: SARWESHVARAN

|
=== Code Execution Successful ===

```

## RESULT:

The program is executed successfully