**Theory:**

The Caesar Cipher is one of the oldest methods to convert data into a format such that it cannot be recognized by unauthorized users. Encryption is a process of converting data into secret code and decryption is the exact opposite process i.e. converting code to original data. It shifts each letter a few positions right to encrypt it. Further, this can be shifted from the same positions to the left to get the message.

Caesar gave a very simple method to encrypt and decrypt the information. It is also known as shift Caesar as the method shifts the character key positions ahead. If the current character is d and the key = 3, then Cipher text will be storing g i.e. 4 positions ahead of d. To decrypt the same, we go 4 positions behind for g which gives us d back.

Formula:

Encryption:

Cipher Text = (ch - 'a' + key) % 26 + 'a'

Decryption:

Plain Text = (ch - 'a' - key + 26) % 26 + 'a'

T(n) = O(n)

**Programming Language: C**

**IDE: DEV C++**

**Code :**

#include <stdio.h>

#include <ctype.h>

#define MAX\_SIZE 500

void encrypt() {

char text[MAX\_SIZE], ch;

int key, i;

printf("Enter a message to encrypt: ");

scanf("%s", text);

printf("Enter the key: ");

scanf("%d", &key);

for (i = 0; text[i] != '\0'; ++i) {

ch = text[i];

if (isalnum(ch)) {

if (islower(ch)) {

ch = (ch - 'a' + key) % 26 + 'a';

} else if (isupper(ch)) {

ch = (ch - 'A' + key) % 26 + 'A';

} else if (isdigit(ch)) {

ch = (ch - '0' + key) % 10 + '0';

}

} else {

printf("Invalid Message");

return;

}

text[i] = ch;

}

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

}

void decrypt() {

char text[MAX\_SIZE], ch;

int key, i;

printf("\n\n Enter a message to decrypt: ");

scanf("%s", text);

printf("Enter the key: ");

scanf("%d", &key);

for (i = 0; text[i] != '\0'; ++i) {

ch = text[i];

if (isalnum(ch)) {

if (islower(ch)) {

ch = (ch - 'a' - key + 26) % 26 + 'a';

} else if (isupper(ch)) {

ch = (ch - 'A' - key + 26) % 26 + 'A';

} else if (isdigit(ch)) {

ch = (ch - '0' - key + 10) % 10 + '0';

}

} else {

printf("Invalid Message");

return;

}

text[i] = ch;

}

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

}

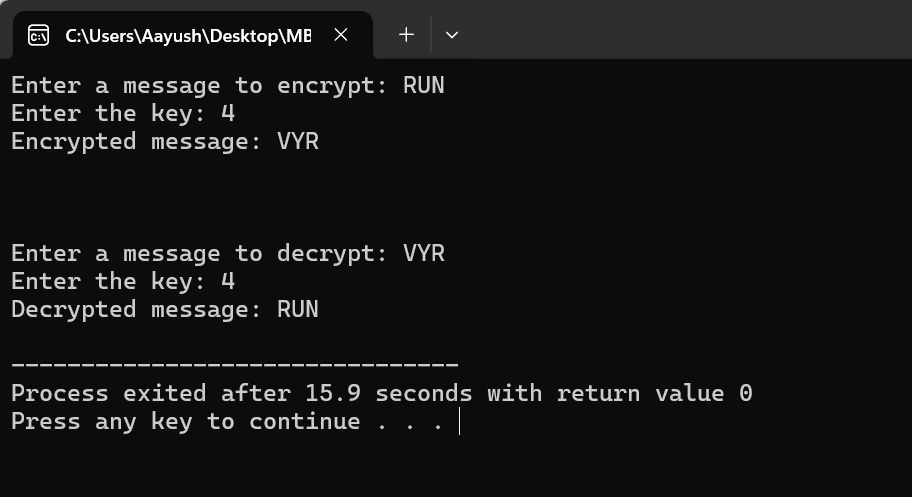
int main() {

encrypt();

decrypt();

return 0;

}



**Output**