

CAESAR CIPHER**Problem Statement:**

Julius Caesar protected his confidential information by encrypting it using a cipher. Caesar's cipher shifts each letter by a number of letters. If the shift takes you past the end of the alphabet, just rotate back to the front of the alphabet. In the case of a rotation by 3, w, x, y, and z would map to z, a, b and c.

Original alphabet: abcdefghijklmnopqrstuvwxyz

Alphabet rotated +3: defghijklmnopqrstuvwxyzabc

Aim:

To implement encryption and decryption in Caesar Cipher technique.

Algorithm:

1. Declare two arrays to store plaintext and ciphertext
2. Prompt the user to enter plaintext
3. Loop till the end-of line marker comes
 - a. get one plaintext character & put the same in plaintext[] array and increment i
 - b. apply caesar 3 key shift cipher on the character and store in ciphertext[] array and increment x.
4. Print the ciphertext

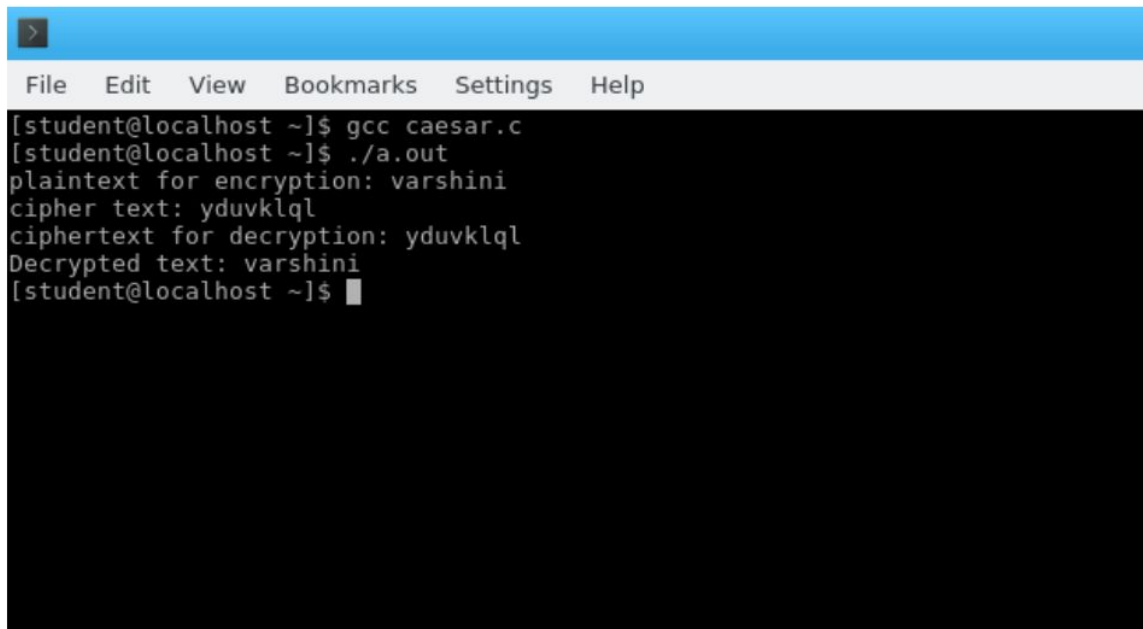
Program Code:

```
#include <stdio.h>

int main()
{
    char plaintext[100]={0},
    ciphertext[100]={0};    int c;
    printf("Plaintext:");
    while((c=getchar()) != '\n')
    {
        static int x=0, i=0;
        plaintext[i++]=(char)c;
        ciphertext[x++]=(char)(c+3);
    }
```

```
    printf("Cipher text:");  
printf("%s\n",ciphertext);  
return 0;  
  
}
```

Output:



The screenshot shows a terminal window with a blue title bar and a menu bar containing 'File', 'Edit', 'View', 'Bookmarks', 'Settings', and 'Help'. The terminal output is as follows:

```
[student@localhost ~]$ gcc caesar.c  
[student@localhost ~]$ ./a.out  
plaintext for encryption: varshini  
cipher text: yduvklql  
ciphertext for decryption: yduvklql  
Decrypted text: varshini  
[student@localhost ~]$
```

Result: