MARRI LAXMAN REDDY
GROUP OF INSTITUTIONS

MLR INSTITUTE OF TECHNOLOGY
(UGC AUTONOMOUS)
Affiliated to JNTUH,Approved by AICTE
Laxman Reddy Avenue, Dundigal, Hyderabad-500 043, Telangana, India

NATIONAL BOARD OF ACCREDITATION

ACCREDITED WITH GRADE A NAAC

NAME	**N.AASRITH**
ROLL NO	**22R21A05P6**
CLASS AND YEAR	**CSE-D 4th YEAR 1st SEM**
WEEK NUMBER	**WEEK 3**

PROBLEM STATEMENT: Write a Java program to perform encryption and decryption using the following algorithms: a)Ceaser Cipher b)Substitution Cipher c)Hill Cipher.

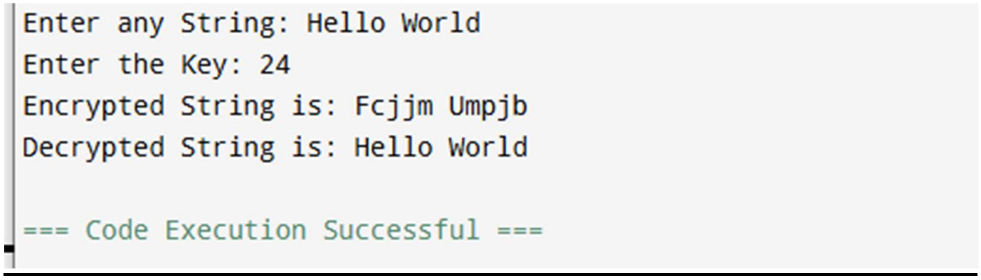
PROGRAM:

a)Ceasar Cipher

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.Scanner;
public class CeaserCipher {
  static Scanner sc = new Scanner(System.in);
  static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
  public static void main(String[] args) throws IOException {
    System.out.print("Enter any String: ");
    String str = br.readLine();
    System.out.print("Enter the Key: ");
    int key = sc.nextInt();
    String encrypted = encrypt(str, key);
    System.out.println("Encrypted String is: " + encrypted);
    String decrypted = decrypt(encrypted, key);
    System.out.println("Decrypted String is: " + decrypted);
  }
  public static String encrypt(String str, int key) {
    String encrypted = "";
    for (int i = 0; i < str.length(); i++) {
      int c = str.charAt(i);
      if (Character.isUpperCase(c)) {
        c = c + (key % 26);
        if (c > 'Z') {
          c = c - 26;
        }
      } else if (Character.isLowerCase(c)) {
        c = c + (key % 26);
        if (c > 'z') {
```

```
                    c = c - 26;
                }   }
            encrypted += (char) c;
        }
        return encrypted;
    }
    public static String decrypt(String str, int key) {
        String decrypted = "";
        for (int i = 0; i < str.length(); i++) {
            int c = str.charAt(i);
            if (Character.isUpperCase(c)) {
                c = c - (key % 26);
                if (c < 'A') {
                    c = c + 26;
                }
            } else if (Character.isLowerCase(c)) {
                c = c - (key % 26);
                if (c < 'a') {
                    c = c + 26;
                }
            }
            decrypted += (char) c;
        }
        return decrypted;
    }
}
```

Output:



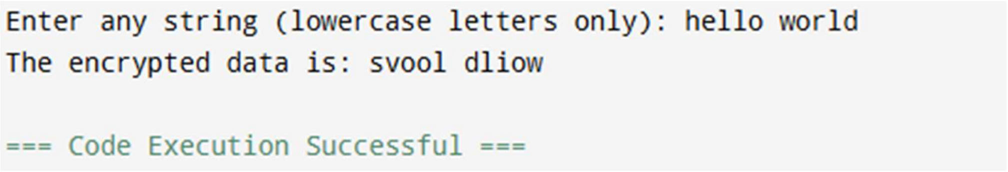
```
Enter any String: Hello World
Enter the Key: 24
Encrypted String is: Fcjjm Umpjb
Decrypted String is: Hello World

=== Code Execution Successful ===
```

b)Substitution Cipher

```
import java.io.*;
import java.util.*;
public class SubstitutionCipher {
    static Scanner sc = new Scanner(System.in);
    static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    public static void main(String[] args) throws IOException {
        String plain = "abcdefghijklmnopqrstuvwxyz";
        String cipher = "zyxwvutsrqponmlkjihgfedcba";
        System.out.print("Enter any string (lowercase letters only): ");
        String str = br.readLine();
        String encrypted = "";
        char c;
        for (int i = 0; i < str.length(); i++) {
            c = str.charAt(i);
            int index = plain.indexOf(c);
            if (index != -1) {
                encrypted += cipher.charAt(index);
            } else {
                encrypted += c;
            }
        }
        System.out.println("The encrypted data is: " + encrypted);
    }
}
```

<u>Output:</u>



```
Enter any string (lowercase letters only): hello world
The encrypted data is: svool dliow

=== Code Execution Successful ===
```

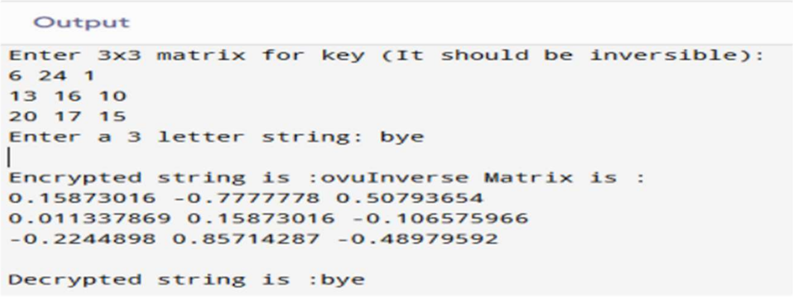
c)Hill Cipher

```
import java.io.*;
import java.util.*;
public class HillCipher {
    static float[][] decrypt = new float[3][1];
    static float[][] a = new float[3][3];    static float[][] b = new float[3][3];
    static float[][] mes = new float[3][1];    static float[][] res = new float[3][1];
    static BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) throws IOException {
        getkeymes();
        for (int i = 0; i < 3; i++)
            for (int j = 0; j < 1; j++)
                for (int k = 0; k < 3; k++)
                    res[i][j] = res[i][j] + a[i][k] * mes[k][j];
        System.out.print("\nEncrypted string is :");
        for (int i = 0; i < 3; i++) {
            System.out.print((char) ((res[i][0] % 26 + 26) % 26 + 97));
            res[i][0] = res[i][0];
        }
        inverse();
        for (int i = 0; i < 3; i++)
            for (int j = 0; j < 1; j++)
                for (int k = 0; k < 3; k++)
                    decrypt[i][j] = decrypt[i][j] + b[i][k] * res[k][j];
        System.out.print("\nDecrypted string is:");
        for (int i = 0; i < 3; i++) {
            System.out.print((char) ((Math.round(decrypt[i][0]) % 26 + 26) % 26 + 97));
        }
        System.out.print("\n");
    }
    public static void getkeymes() throws IOException {
        System.out.println("Enter 3x3 matrix for key (It should be inversible): ");
        for (int i = 0; i < 3; i++)
            for (int j = 0; j < 3; j++)
                a[i][j] = sc.nextFloat();
        System.out.print("Enter a 3 letter string: ");
        String msg = br.readLine();

        for (int i = 0; i < 3; i++)
```

```
      mes[i][0] = msg.charAt(i) - 97;
  }
  public static void inverse() {
    float p, q;
    float[][] c = new float[3][3];
    for (int i = 0; i < 3; i++)
      for (int j = 0; j < 3; j++) {
        c[i][j] = a[i][j];
        b[i][j] = (i == j) ? 1 : 0;
      }
    for (int k = 0; k < 3; k++) {
      for (int i = 0; i < 3; i++) {
        if (i != k) {
          p = c[i][k];
          q = c[k][k];
          for (int j = 0; j < 3; j++) {
            c[i][j] = c[i][j] * q - p * c[k][j];
            b[i][j] = b[i][j] * q - p * b[k][j];
        }         }        }       }
    for (int i = 0; i < 3; i++) {
      float div = c[i][i];
      for (int j = 0; j < 3; j++)
        b[i][j] = b[i][j] / div;
    }
    System.out.println("Inverse Matrix is:");
    for (int i = 0; i < 3; i++) {
      for (int j = 0; j < 3; j++)
        System.out.print(b[i][j] + (j == 2 ? "" : " "));
      System.out.println();
    }   }   }
```

Output:



```
Enter 3x3 matrix for key (It should be inversible):
6 24 1
13 16 10
20 17 15
Enter a 3 letter string: bye
|
Encrypted string is :ovuInverse Matrix is :
0.15873016 -0.7777778 0.50793654
0.011337869 0.15873016 -0.106575966
-0.2244898 0.85714287 -0.48979592

Decrypted string is :bye
```