**Program Code –**

import java.util.\*;

class productCipher {

public static void main(String args[]) {

System.out.println("Enter the input to be encrypted:");

String substitutionInput = new Scanner(System.in).nextLine();

System.out.println("Enter a number:");

int n = new Scanner(System.in).nextInt();

// Substitution encryption

StringBuffer substitutionOutput = new StringBuffer();

for (int i = 0; i < substitutionInput.length(); i++) {

char c = substitutionInput.charAt(i);

substitutionOutput.append((char) (c + 5));

}

System.out.println("\nSubstituted text:");

System.out.println(substitutionOutput);

// Transposition encryption

String transpositionInput = substitutionOutput.toString();

int modulus;

if ((modulus = transpositionInput.length() % n) != 0) {

modulus = n - modulus;

// ‘modulus’ is now the number of blanks/padding (X) to be appended

for (; modulus != 0; modulus--) {

transpositionInput += "/";

}

}

StringBuffer transpositionOutput = new StringBuffer();

System.out.println("\nTransposition Matrix:");

for (int i = 0; i < n; i++) {

for (int j = 0; j < transpositionInput.length() / n; j++) {

char c = transpositionInput.charAt(i + (j \* n));

System.out.print(c);

transpositionOutput.append(c);

}

System.out.println();

}

for (int j = 0; j < transpositionInput.length() / n; j++) {

char c = transpositionInput.charAt(i + (j \* n));

System.out.print(c);

transpositionOutput.append(c);

}

System.out.println();

}

System.out.println("\nFinal encrypted text:");

System.out.println(transpositionOutput);

// Transposition decryption

n = transpositionOutput.length() / n;

StringBuffer transpositionPlaintext = new StringBuffer();

for (int i = 0; i < n; i++) {

for (int j = 0; j < transpositionOutput.length() / n; j++) {

char c = transpositionOutput.charAt(i + (j \* n));

transpositionPlaintext.append(c);

}

}

// Substitution decryption

StringBuffer plaintext = new StringBuffer();

for (int i = 0; i < transpositionPlaintext.length(); i++) {

char c = transpositionPlaintext.charAt(i);

plaintext.append((char) (c - 5));

}

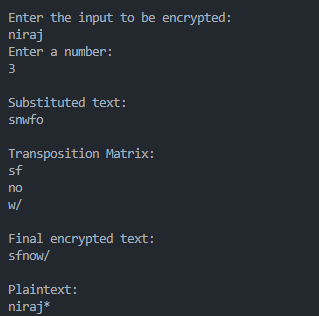
System.out.println("\nPlaintext:");

System.out.println(plaintext);

}

}

**Output –**

****