Experiment 02

Design and Implement a product cipher using Substitution ciphers and Transposition Cipher..

Roll No.	70
Name	Mayuri Shridatta Yerande
Class	D15-B
Subject	Internet Security Lab
LO Mapped	LO1: To apply the knowledge of symmetric cryptography to implement classical ciphers.

Aim: Write a program to understand Implementation of a product cipher using Substitution ciphers and Transposition Cipher.

Introduction:

Product cipher, data encryption scheme in which the ciphertext produced by encrypting a plaintext document is subjected to further encryption. By combining two or more simple transposition ciphers or substitution ciphers, a more secure encryption may result.

Substitution cipher

Substitution technique is a classical encryption technique where the characters present in the **original message** are **replaced** by the **other characters or numbers or by symbols.** If the plain text (original message) is considered as the string of bits, then the substitution technique would replace bit pattern of plain text with the bit pattern of cipher text.

Substitution Technique:

1. Caesar Cipher: This the simplest substitution cipher by Julius Caesar. In this substitution

technique, to encrypt the plain text, each alphabet of the plain text is replaced by the alphabet three places further it. And to decrypt the cipher text each alphabet of cipher text is replaced by the alphabet three places before it.

2. Monoalphabetic Cipher: he cipher alphabet for each plain text alphabet is fixed, for the entire encryption. In simple words, if the alphabet ‘p’ in the plain text is replaced by the cipher alphabet ‘d’. Then in the entire plain text wherever alphabet ‘p’ is used, it will be replaced by the alphabet ‘d’ to form the ciphertext.
3. Playfair Cipher: Playfair cipher is a substitution cipher which involves a 5X5 matrix.

Create a 5X5 matrix and place the key in that matrix row-wise from left to right. Then put the remaining alphabets in the blank space.

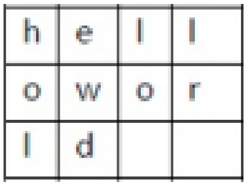
Break the plain text into a pair of alphabets.Convert plain text into ciphertext. For that, take the first pair of plain text and check for cipher alphabets for the corresponding in the matrix.

4. Hill Cipher
5. Polyalphabetic Cipher: Polyalphabetic cipher is far more secure than a monoalphabetic cipher. As monoalphabetic cipher maps a plain text symbol or alphabet to a ciphertext symbol and uses the same ciphertext symbol wherever that plain text occurs in the message. But polyalphabetic cipher, each time replaces the plain text with the different ciphertext.
6. One-Time Pad: The one-time pad cipher suggests that the **key length** should be **as long as the plain text** to prevent the repetition of key. Along with that, the **key** should be **used** only **once** to encrypt and decrypt the single message after that the key should be discarded.

Transposition cipher

Transposition Cipher is a cryptographic algorithm where the order of alphabets in the plaintext is rearranged to form a cipher text. In this process, the actual plain text alphabets are not included.

A simple example for a transposition cipher is columnar transposition cipher where each character in the plain text is written horizontally with specified alphabet width. The cipher is written vertically, which creates an entirely different cipher text.



h	e	l	l
o	w	o	r
l	d		

The plain text characters are placed horizontally and the cipher text is created with vertical format as : holewdlo lr.

Algorithm:

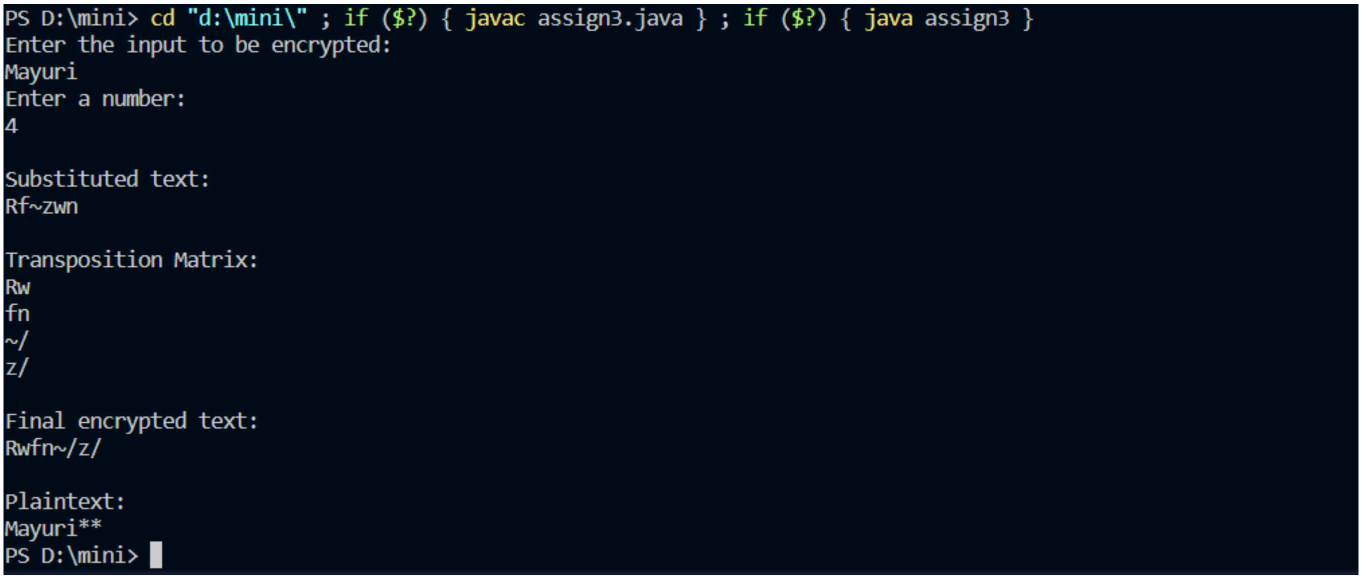
- Input a plain text and encrypt using substitution cipher technique
  The result obtained from step 1 shall be used as a plain text input to transposition cipher
- Perform encryption using transposition cipher
- Perform decryption using transposition cipher
- The result obtained from previous step shall be used as a input to substitution cipher decryption
- We get our original plain text as the output

Results:

```
import java.util.*;

public class Main {
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();
}



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);
```

```
}
}
```

Input: Mayuri



```
PS D:\mini> cd "d:\mini\" ; if ($?) { javac assign3.java } ; if ($?) { java assign3 }
Enter the input to be encrypted:
Mayuri
Enter a number:
4

Substituted text:
Rf~zwn

Transposition Matrix:
Rw
fn
~/
z/

Final encrypted text:
Rwfn~/z/

Plaintext:
Mayuri**
PS D:\mini> 
```

Conclusion: Hence we Designed and Implemenedt a product cipher using Substitution ciphers and Transposition Cipher..