**Aim:** Design and implement a product cipher using Substitution ciphers and Transportation cipher.

**Program:**

import java.util.Scanner;

public class ProductCipher {

public static String substitutionCipher(String plaintext, int shift) {

StringBuilder result = new StringBuilder();

for (char c : plaintext.toCharArray()) {

if (Character.isUpperCase(c)) {

result.append((char) ((c + shift - 65) % 26 + 65));

} else if (Character.isLowerCase(c)) {

result.append((char) ((c + shift - 97) % 26 + 97));

} else {

result.append(c);

}

}

return result.toString();

}

public static String transpositionCipher(String plaintext, int key) {

StringBuilder result = new StringBuilder();

int n = plaintext.length();

char[][] rail = new char[key][n];

// Initializing the rail matrix

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

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

rail[i][j] = '\n';

}

}

boolean direction\_down = false;

int row = 0, col = 0;

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

if (row == 0 || row == key - 1) {

direction\_down = !direction\_down;

}

rail[row][col++] = plaintext.charAt(i);

if (direction\_down) {

row++;

} else {

row--;

}

}

// Constructing the result

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

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

if (rail[i][j] != '\n') {

result.append(rail[i][j]);

}

}

}

return result.toString();

}

public static String productCipher(String plaintext, int shift, int transpositionKey) {

String substitutionResult = substitutionCipher(plaintext, shift);

return transpositionCipher(substitutionResult, transpositionKey);

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the original text: ");

String plaintext = scanner.nextLine();

int shift = 4; // Example shift for Caesar Cipher

int transpositionKey = 3; // Example key for Rail Fence Cipher

String encryptedText = productCipher(plaintext, shift, transpositionKey);

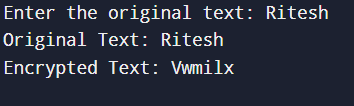
System.out.println("Original Text: " + plaintext);

System.out.println("Encrypted Text: " + encryptedText);

}

}}

**Output:**



**Conclusion:** Hence, we have successfully designed and implemented a product cipher using Substitution ciphers and Transportation cipher.