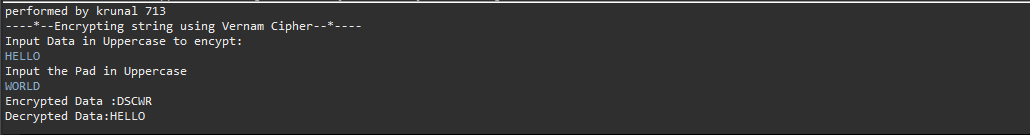
**Practical no 2**

**AIM:** Write program to implement the following Substitution Cipher Techniques

a)Vernam Cipher b)Playfair Cipher

**Code:**

|  |
| --- |
| import java.util.Scanner;  public class Vernam {  String encrypt(String str, String pad) {  String encrypted = "";  for (int i = 0; i < str.length(); i++) {  int c = str.charAt(i);  int p = pad.charAt(i);  c = (c + p);  if (c > 'Z') {    c = c%26;  c = c+65;  }  encrypted += (char) c;  }  return encrypted;  }  String decrypt(String str, String pad) {  String decrypted = "";  for (int i = 0; i < str.length(); i++) {  int c = str.charAt(i);  int p = pad.charAt(i);  c = (c - p)+26;  if (c < 'A') {  c = (c%26);  c = c+65;  }  decrypted += (char) c;  }  return decrypted;  }  public static void main(String[] args) {  System.***out***.println("performed by krunal 713");  System.***out***.println("----\*--Encrypting string using Vernam Cipher--\*----");  Vernam v = new Vernam();  Scanner s = new Scanner(System.***in***);  System.***out***.println("Input Data in Uppercase to encypt:");  String str = s.nextLine();  System.***out***.println("Input the Pad in Uppercase");  String pad = s.nextLine();  String encrypted = v.encrypt(str, pad);  System.***out***.println("Encrypted Data :" + encrypted);  String decrypted = v.decrypt(encrypted, pad);  System.***out***.println("Decrypted Data:" + decrypted);  }  } |



**b)Playfair Cipher**

|  |
| --- |
| import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStreamReader;  import java.util.Arrays;  public class PlayFair {  public static char *keymat*[][] = new char[5][5];  public static String *trans* = "J";  public static char *subs* = 'X';  private static int decrem(int pos) {  if (pos < 0) {  return pos + 5;  } else {  return pos;  }  }  private static int[] srch(char c) {  int i, j;  int[] pos = new int[2];  for (i = 0; i < 5; i++) {  for (j = 0; j < 5; j++) {  if (*keymat*[i][j] == c) {  pos[0] = i;  pos[1] = j;  break;  }  }  }  return pos;  }  private static String encrypt(char c1, char c2) {  int[] pos1 = new int[2];  int[] pos2 = new int[2];  String frag = "";  pos1 = *srch*(c1);  pos2 = *srch*(c2);  if (pos1[0] == pos2[0]) { //condition for same row  c1 = *keymat*[pos1[0]][(pos1[1] + 1) % 5];  c2 = *keymat*[pos2[0]][(pos2[1] + 1) % 5];  frag = ("" + c1 + c2 + "");  } else if (pos1[1] == pos2[1]) { //condition for same column  c1 = *keymat*[(pos1[0] + 1) % 5][pos1[1]];  c2 = *keymat*[(pos2[0] + 1) % 5][pos2[1]];  frag = ("" + c1 + c2 + "");  } else { //condition for different row & column  c1 = *keymat*[pos2[0]][pos1[1]];  c2 = *keymat*[pos1[0]][pos2[1]];  frag = ("" + c1 + c2 + "");  }  return frag;  }  private static String decrypt(char c1, char c2) {  int[] pos1 = new int[2];  int[] pos2 = new int[2];  String frag = "";  pos1 = *srch*(c1);  pos2 = *srch*(c2);  if (pos1[0] == pos2[0]) { //condition for same row  c1 = *keymat*[pos1[0]][*decrem*(pos1[1] - 1) % 5];  c2 = *keymat*[pos2[0]][*decrem*(pos2[1] - 1) % 5];  frag = ("" + c1 + c2 + "");  } else if (pos1[1] == pos2[1]) { //condition for same column  c1 = *keymat*[*decrem*(pos1[0] - 1) % 5][pos1[1]];  c2 = *keymat*[*decrem*(pos2[0] - 1) % 5][pos2[1]];  frag = ("" + c1 + c2 + "");  } else { //condition for different row & column  c1 = *keymat*[pos2[0]][pos1[1]];  c2 = *keymat*[pos1[0]][pos2[1]];  frag = ("" + c1 + c2 + "");  }  return frag;  }  public static void main(String[] args) throws IOException {  BufferedReader br = new BufferedReader(new InputStreamReader(System.***in***));  String key;  int p = 0, k = 0, c = 65;  System.***out***.print("Enter Key:\t");  key = br.readLine();  for (int i = 0; i < 5; i++) {  for (int j = 0; j < 5; j++) {  if (p < key.length()) {  *keymat*[i][j] = key.charAt(p);  p++;  } else {  if ((char) c == 'J') {  c++;  }  for (; k < key.length();) {  if ((char) c == key.charAt(k)) {  k = 0;  c++;  }  k++;  }  *keymat*[i][j] = (char) c;  c++;  k = 0;  }  }  }  System.***out***.println("\nMatrix of characters:");  for (int i = 0; i < 5; i++) {  for (int j = 0; j < 5; j++) {  System.***out***.print(*keymat*[i][j] + "\t");  }  System.***out***.println();  }  String etext = "", dtext = "";  System.***out***.print("\nEnter Text: \t");  String text = br.readLine();  text = text.toUpperCase();  text = text.replaceAll("\\s", ""); //removes whitespaces  text = text.replace(*trans*, "I"); //replaces J with I  text = text.replaceAll("([A-Z])\\1+", "$1" + *subs* + "$1");  if (text.length() % 2 != 0) {  text += *subs*;  }  char[] PTC = text.toCharArray();  System.***out***.println("Padded Text:\t" + text);  for (int i = 0; i < text.length(); i += 2) {  etext += *encrypt*(PTC[i], PTC[i + 1]);  }  System.***out***.println("Encrypted Text:\t" + etext);  char[] OTC = etext.toCharArray();  System.***out***.println("P: " + Arrays.*toString*(OTC));  for (int i = 0; i < etext.length(); i += 2) {  dtext += *decrypt*(OTC[i], OTC[i + 1]);  }  System.***out***.println("Decrypted Text:\t" + dtext);  System.***out***.println("Performed by: 713 Krunal Dhavle");  }  } |

