PROGRAM:- 1

#include <stdio.h> #include<string.h> void main()

{

char str[]="hello world"; char str1[20];

int alpha=strlen(str); for(int i=0;i<alpha;i++)

{

str1[i]=str[i]^0; printf("%c",str1[i]);

}

printf("\n");

}

#include <stdio.h> #include<string.h> void main()

{

char str[]="hello world"; char str1[20];

char str2[20]; int l=strlen(str);

for(int i=0;i<l;i++)

{

str1[i]=str[i]&127; printf("%c",str1[i]);

}

printf("\n");

PROGRAM:-2

for (int j=0;j<l;j++)

{

str2[j]=str[j]^127; printf("%c",str2[j]);

}

printf("\n");

}

package pgm3A;

import java.io.\*; import java.util.\*; public class Pgm3a {

public static void main(String args[])throws IOException{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter a text:");

String str = br.readLine();

String alpha = "abcdefghijklmnopqrstuvwxyz"; String encrypted = encrypt(str,alpha);

System.out.println("Encrypted text: "+encrypted); String decrypted = decrypt(encrypted,alpha); System.out.println("Decrypted text: "+decrypted);

}

public static String encrypt(String pt, String alpha) { String ct="";

for(int i=0;i<pt.length();i++) { char c = pt.charAt(i);

int j = alpha.indexOf(c); j+=3;

if(j>=26) j=j%26;

ct+=alpha.charAt(j);

}

return ct;

}

public static String decrypt(String ct, String alpha) { String pt="";

for(int i=0;i<ct.length();i++) { char c = ct.charAt(i);

int j = alpha.indexOf(c); j-=3;

if(j<0) j=j+26;

pt+=alpha.charAt(j);

}

return pt;

}

}

PROGRAM:- 3B

package pgm3; import java.io.\*; import java.util.\*; public class Pgm3b {

public static void main(String args[])throws IOException{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in)); Scanner sc = new Scanner(System.in);

System.out.println("Enter PT:"); String pt = sc.nextLine();

String alpha="abcdefghijklmnopqrstuwxyz"; String encstr="mnbvcxzlkjhgfdsapoiuytrewq"; String encrypted=encrypt(pt,alpha, encstr);

String decrypted=decrypt(encrypted,alpha, encstr);

System.out.println("ENC:"+encrypted+"\tDEC:"+decrypted);

}

public static String encrypt(String pt,String alpha, String encstr) { String encrypted="";

for(int i=0;i<pt.length();i++) { char c = pt.charAt(i);

int j = alpha.indexOf(c); encrypted+=encstr.charAt(j);

}

return encrypted;

}

public static String decrypt(String enc, String alpha, String encstr) { String decrypted="";

for(int i=0;i<enc.length();i++) { char c = enc.charAt(i);

int j = encstr.indexOf(c); decrypted+=alpha.charAt(j);

}

return decrypted;

}

}

Program:- 3C

package pgm3; import java.io.\*; import java.util.\*; public class Pgm3c {

public static void main(String args[])throws IOException{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in)); Scanner sc = new Scanner(System.in);

int msg[][]=new int[1][2];

int key[][]=new int[2][2];

System.out.println("Enter 2x2 key matrix:"); for(int i=0;i<2;i++) {

for(int j=0;j<2;j++) { key[i][j]=sc.nextInt();

}

}

System.out.println("Enter the message:"); String str = br.readLine();

for(int i=0;i<str.length();i++) { msg[0][i]=str.charAt(i)-97;

}

encrypt\_decrypt(msg,key);

}

public static void encrypt\_decrypt(int msg[][],int key[][]) { int cipher[][]=new int[1][2];

int plain[][]=new int[1][2];

int decrypted[][]=new int[1][2]; int encrypted[][]=new int[1][2]; int inv[][]=new int[2][2];

for(int j=0;j<2;j++) { decrypted[0][j]=0; for(int k=0;k<2;k++) {

decrypted[0][j]+=msg[0][k]\*key[k][j];

}

cipher[0][j]=(decrypted[0][j]%26)+97;

}

System.out.println("Encrypted string:");

for(int i=0;i<2;i++)

System.out.println((char)cipher[0][i]); inv[0][0]=key[1][1];

inv[0][1]=-key[0][1];

inv[1][0]=-key[1][0];

inv[1][1]=key[0][0];

int det = (key[0][0]\*key[1][1])-(key[0][1]\*key[1][0]); for(int j=0;j<2;j++) {

encrypted[0][j]=0; for(int k=0;k<2;k++) {

encrypted[0][j]+=decrypted[0][k]\*inv[k][j];

}

plain[0][j]=((encrypted[0][j]/det)%26)+97;

}

System.out.println("Decrypted string:"); for(int i=0;i<2;i++)

System.out.println((char)plain[0][i]);

}

}

PROGRAM:- 4

package pgm4; import java.io.\*; import java.util.\*; public class RSA {

public static void main(String args[])throws IOException{ Scanner sc = new Scanner(System.in);

System.out.println("Enter p:"); int p=sc.nextInt();

System.out.println("Enter q:"); int q=sc.nextInt();

int n = p\*q;

int phi = (p-1)\*(q-1); int d=0,e=0;

double ct=0;

for(int i=2;i<phi;i++) { if(gcd(i,phi)==1) { if(i==3) {

continue;

}

else { e=i; break;

}

}

}

System.out.println("e="+e); for(int k=1;;k++) {

if(((k\*e)%phi)==1)

{

d=k;break;

}

}

System.out.println("d="+d); System.out.println("PU : "+e+","+n);

System.out.println("PR : "+d+","+n); System.out.println("Enter plaintext:"); int pt = sc.nextInt();

ct = Math.pow(pt,e)%n; System.out.println("CT:"+ct);

}

public static int gcd(int a, int b) { if(b==0)

return a;

return gcd(b,a%b);

}

}

Program:- 5

package pgm5; import java.io.\*;

import java.security.InvalidKeyException; import java.security.NoSuchAlgorithmException; import java.util.Scanner;

import javax.crypto.\*; public class DESAlgo {

public static void main(String args[])throws IOException{ try {

Scanner sc = new Scanner(System.in); System.out.println("Enter PT:");

String pt = sc.nextLine();

KeyGenerator kg = KeyGenerator.getInstance("DES"); SecretKey key = kg.generateKey();

Cipher c = Cipher.getInstance("DES/ECB/PKCS5Padding"); c.init(Cipher.ENCRYPT\_MODE,key);

byte[] text = pt.getBytes();

System.out.println("Text in bytes:"+text); System.out.println("Text :"+new String(text)); byte encrypted[] = c.doFinal(text);

System.out.println("ENC Text in bytes:"+encrypted); System.out.println("ENC Text :"+new String(encrypted));

c.init(Cipher.DECRYPT\_MODE,key);

byte decrypted[] = c.doFinal(encrypted); System.out.println("DEC Text in bytes:"+decrypted);

System.out.println("DEC Text :"+new String(decrypted));

}

catch(Exception e) {

System.out.println(e.getMessage());

}

}

}

PROGRAM:- 6

package pgm6; import java.util.\*; import javax.crypto.\*;

public class BlowfishAlgo {

public static void main(String[] args) throws Exception { Scanner sc = new Scanner(System.in);

System.out.println("Enter the message"); String msg = sc.nextLine();

KeyGenerator kg = KeyGenerator.getInstance("Blowfish"); SecretKey sk = kg.generateKey();

Cipher c = Cipher.getInstance("Blowfish"); c.init(Cipher.ENCRYPT\_MODE, sk);

byte[] encrypted =c.doFinal(msg.getBytes()); c.init(Cipher.DECRYPT\_MODE,sk );

byte[] decrypted = c.doFinal(encrypted);

System.out.println("Encrypted text: "+new String(encrypted)); System.out.println("Decrypted text: "+new String(decrypted));

}

}

PROGRAM:-7

package pgm7; import java.util.\*;

public class DiffieHellman {

public static void main(String[] args) { int q, a,Xa, Xb, Ya,Yb, Ka, Kb;

Scanner sc = new Scanner(System.in); System.out.println("Enter the value of q:"); q = sc.nextInt();

System.out.println("Enter the value of a:"); a = sc.nextInt();

System.out.println("Enter the value of Xa:"); Xa = sc.nextInt();

System.out.println("Enter the value of Xb:"); Xb = sc.nextInt();

Ya = (int)Math.pow(a, Xa)%q; Yb= (int)Math.pow(a, Xb)%q; Ka = (int)Math.pow(Yb,Xa)%q; Kb = (int)Math.pow(Ya,Xb)%q;

System.out.println("Value of Ya: "+Ya); System.out.println("Value of Yb: "+Yb); System.out.println("Value of Ka: "+Ka); System.out.println("Value of Kb: "+Kb); if(Ka==Kb){

System.out.println("A and B can communicate");

}

else

System.out.println("A and B can NOT communicate");

}}

PROGRAM:- 8

import javax.crypto.\*;

import javax.swing.JOptionPane; public class Blowfish1 {

public static void main(String args[])throws Exception{ KeyGenerator kg=KeyGenerator.getInstance("Blowfish"); SecretKey sk=kg.generateKey();

Cipher c=Cipher.getInstance("Blowfish"); c.init(Cipher.ENCRYPT\_MODE,sk);

String input=JOptionPane.showInputDialog("Input your text:"); byte[] encrypt=c.doFinal(input.getBytes());

c.init(Cipher.DECRYPT\_MODE,sk); byte[] decerypt=c.doFinal(encrypt);

JOptionPane.showMessageDialog(JOptionPane.getRootFrame(),"\n encrypted text:"+new String(encrypt)+"\n"+"\n decrypt text:"+new String(decerypt));

System.exit(0);

}

}

PROGRAM:-9

package pgm9;

import java.security.\*; import java.math.\*; public class MD5Algo {

public static void main(String[] args)throws Exception {

MessageDigest md = MessageDigest.getInstance("MD5"); System.out.println("Algorithm = " + md.getAlgorithm()); System.out.println("Provider = " + md.getProvider()); System.out.println("To String = " + md.toString());

String input=""; md.update(input.getBytes()); byte output[] = md.digest();

System.out.println("MD5("+input+")= "+ bytesToHex(output));

input="abc"; md.update(input.getBytes()); output = md.digest();

System.out.println("MD5("+input+")= "+ bytesToHex(output));

input="abcdefghijklmnoqrstuvwxyz"; md.update(input.getBytes());

output = md.digest();

System.out.println("MD5("+input+")= "+ bytesToHex(output));

}

public static String bytesToHex(byte output[]) { BigInteger n = new BigInteger(1,output);

String hashtext = n.toString(16); return hashtext;

}

}

PROGRAM:- 11

import java.util.\*; import javax.crypto.\*;

import javax.crypto.spec.SecretKeySpec; import java.math.BigInteger;

public class PGRO11 {

public static void main(String[] args)throws Exception { Scanner sc = new Scanner(System.in);

System.out.println("Enter the message:"); String msg = sc.nextLine();

KeyGenerator kg = KeyGenerator.getInstance("AES"); kg.init(128);

SecretKey sk = kg.generateKey(); byte[] raw = sk.getEncoded();

SecretKeySpec sks = new SecretKeySpec(raw,"AES"); Cipher c = Cipher.getInstance("AES"); c.init(Cipher.ENCRYPT\_MODE, sks);

byte[] encrypt=c.doFinal(msg.getBytes());

System.out.println("encrypted string: "+new String(encrypt)); System.out.println("encrypted string in hex: "+ bytesToHex(encrypt)); c.init(Cipher.DECRYPT\_MODE, sks);

byte[] decrypt=c.doFinal(encrypt);

System.out.println("decrypted string: "+new String(decrypt));

}

public static String bytesToHex(byte[] b) { BigInteger n = new BigInteger(1,b);

String s = n.toString(16); return s;

}}

PROGRAM:- 10

package pgm10; import java.security.\*; import java.math.\*; public class SHA1Algo {

public static void main(String[] args)throws Exception { MessageDigest md = MessageDigest.getInstance("SHA-1"); System.out.println("Algorithm = " + md.getAlgorithm()); System.out.println("Provider = " + md.getProvider()); System.out.println("To String = " + md.toString());

String input=""; md.update(input.getBytes()); byte output[] = md.digest();

System.out.println("SHA-1("+input+")= "+ bytesToHex(output)); input="abc";

md.update(input.getBytes()); output = md.digest();

System.out.println("SHA-1("+input+")= "+ bytesToHex(output));

input="abcdefghijklmnoqrstuvwxyz"; md.update(input.getBytes());

output = md.digest();

System.out.println("SHA-1("+input+")= "+ bytesToHex(output));

}

public static String bytesToHex(byte output[]) { BigInteger n = new BigInteger(1,output);

String hashtext = n.toString(16); return hashtext;

}

}