Encryption Decryption – Main Class

byte[] input;

byte[] keyBytes = "12345678".getBytes(); // our own key

byte[] ivBytes = "input123".getBytes();

SecretKeySpec key = new SecretKeySpec(keyBytes, "DES");

IvParameterSpec ivSpec = new IvParameterSpec(ivBytes);

Cipher cipher;

byte[] chiperText;

int ctLenght;

public Main() {

initComponents();

}

**private void btnEncryptActionPerformed(java.awt.event.ActionEvent evt) {**

try {

Security.addProvider(new org.bouncycastle.jce.provider.BouncyCastleProvider()); //type of security

input = type.getText().getBytes();

SecretKeySpec key = new SecretKeySpec(keyBytes, "DES");

IvParameterSpec ivSpec = new IvParameterSpec(ivBytes);

cipher = Cipher.getInstance("DES/CTR/NoPadding", "BC"); //CTR of encryption

cipher.init(Cipher.ENCRYPT\_MODE, key, ivSpec);

chiperText = new byte[cipher.getOutputSize(input.length)];

ctLenght = cipher.update(input, 0, input.length, chiperText, 0);

ctLenght += cipher.doFinal(chiperText, ctLenght);

encrypt.setText(new String(chiperText));

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

**private void btnDecryptActionPerformed(java.awt.event.ActionEvent evt) {**

try {

cipher.init(Cipher.DECRYPT\_MODE, key,ivSpec);

byte[] plainText = new byte[cipher.getOutputSize(ctLenght)];

int ptLength = cipher.update(chiperText, 0,ctLenght,plainText,0);

ptLength += cipher.doFinal(plainText, ptLength);

decrypt.setText(new String(plainText));

} catch (Exception e) {

JOptionPane.showMessageDialog(null, e);

}

}

{

namespace ConsoleApplication2

{

class Program

{

static void Main(string[] args)

{

//Int64 id = Encrypt("abacus");

string word=Dencrypt(1630866511392489104);

System.Console.WriteLine(word);

System.Console.ReadLine();

}

private static Int64 Encrypt(string p)

{

Int64 c = 31;

for (int i = 0; i < p.Length; i++)

{

c = c \* 47 + (byte)p[i] % 97;

}

return c;

}

private static string Dencrypt(Int64 id)

{

string word = "";

while (id > 31)

{

byte c= (byte) (id % 47 + 97) ; // remove the multiplied value and extract the character

char ch= Convert.ToChar(c); // get the charactor

word = ch + word; // create the word

id = id / 47;

}

return word;

}

}

}

/\*

The following Python function encrypts plain text to

cipher text. For example given the text 'abacus' the function returns a

the cypher text 334160560256. Your task is to reverse engineer the

algorithm and find the source text which results in the cypher text

15731535468486 in a language of your choosing. Remember to comment the

code with your reasoning.

def encrypt (p):

h = 31

for i in range(0, len(p)):

h = h \* 47 + ord(p[i]) % 97

return h

\*/