28.08.2020

Ex No: 3

IMPLEMENTATION OF DES

AIM:

To write a C program to implement Data Encryption Standard (DES) using C

Language.

DESCRIPTION:

DES is a symmetric encryption system that uses 64-bit blocks, 8 bits of which are

used for parity checks. The key therefore has a "useful" length of 56 bits, which means that

only 56 bits are actually used in the algorithm. The algorithm involves carrying out

combinations, substitutions and permutations between the text to be encrypted and the key,

while making sure the operations can be performed in both directions. The key is ciphered

on 64 bits and made of 16 blocks of 4 bits, generally denoted k₁ to k₁₆. Given that "only" 56

bits are actually used for encrypting, there can be 2^{56} different keys.

The main parts of the algorithm are as follows:

Fractioning of the text into 64-bit

blocks Initial permutation of blocks

Breakdown of the blocks into two parts: left and right, named L and

R Permutation and substitution steps repeated 16 times

Re-joining of the left and right parts then inverse initial permutation

ALGORITHM:

STEP-1: Read the 64-bit plain text.

- **STEP-2:** Split it into two 32-bit blocks and store it in two different arrays.
- **STEP-3:** Perform XOR operation between these two arrays.
- **STEP-4:** The output obtained is stored as the second 32-bit sequence and the originalsecond 32-bit sequence forms the first part.
- **STEP-5:** Thus the encrypted 64-bit cipher text is obtained in this way. Repeat the same process for the remaining plain text characters.

PROGRAM:

DES.java

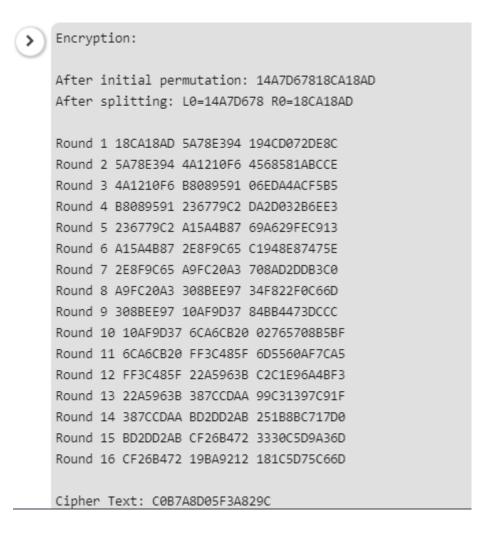
```
import javax.swing.*;
import java.security.SecureRandom;
import javax.crypto.Cipher;
import javax.crypto.KeyGenerator;
import javax.crypto.SecretKey;
import
javax.crypto.spec.SecretKeySpec;
import java.util.Random ; class DES {
     byte[] skey = new byte[1000];
     String skeyString;
     static byte[] raw;
     String inputMessage,encryptedData,decryptedMessage;
public DES()
{
try
{
```

```
generateSymmetricKey();
     inputMessage=JOptionPane.showInputDialog(null, "Enter
    message to encrypt");
    byte[] ibyte = inputMessage.getBytes();
    byte[] ebyte=encrypt(raw, ibyte);
     String encryptedData = new String(ebyte);
     System.out.println("Encrypted message "+encryptedData);
     JOptionPane.showMessageDialog(null, "Encrypted Data
     "+"\n"+encryptedData);
    byte[] dbyte= decrypt(raw,ebyte);
     String decryptedMessage = new String(dbyte);
     System.out.println("Decrypted message
     "+decryptedMessage);
     JOptionPane.showMessageDialog(null, "Decrypted Data
     "+"\n"+decryptedMessage);
}
catch(Exception e)
{
     System.out.println(e);
}
}
```

```
void generateSymmetricKey() {
try {
     Random r = new Random();
     int num = r.nextInt(10000);
     String knum = String.valueOf(num);
     byte[] knumb = knum.getBytes();
     skey=getRawKey(knumb);
     skeyString = new String(skey);
     System.out.println("DES Symmetric key = "+skeyString);
}
catch(Exception e)
     System.out.println(e);
}
}
private static byte[] getRawKey(byte[] seed) throws Exception
{
     KeyGenerator kgen = KeyGenerator.getInstance("DES");
     SecureRandom sr = SecureRandom.getInstance("SHA1PRNG");
     sr.setSeed(seed);
     kgen.init(56, sr);
     SecretKey skey =
     kgen.generateKey(); raw =
     skey.getEncoded(); return raw;
}
private static byte[] encrypt(byte[] raw, byte[] clear)
throws Exception {
          SecretKeySpec skeySpec = new
          SecretKeySpec(raw, "DES");
```

```
Cipher cipher =
          Cipher.getInstance("DES");
          cipher.init(Cipher.ENCRYPT MODE,
          skeySpec); byte[] encrypted =
          cipher.doFinal(clear); return
          encrypted;
}
private static byte[] decrypt(byte[] raw, byte[] encrypted)
throws Exception
{
          SecretKeySpec skeySpec = new
          SecretKeySpec(raw, "DES");
          Cipher cipher =
          Cipher.getInstance("DES");
          cipher.init(Cipher.DECRYPT MODE,
          skeySpec); byte[] decrypted =
          cipher.doFinal(encrypted); return
          decrypted;
}
public static void main(String
     args[]) { DES des = new DES();
}
}
```

OUTPUT:



RESULT:

Thus the data encryption standard algorithm had been implemented successfully using C language.

