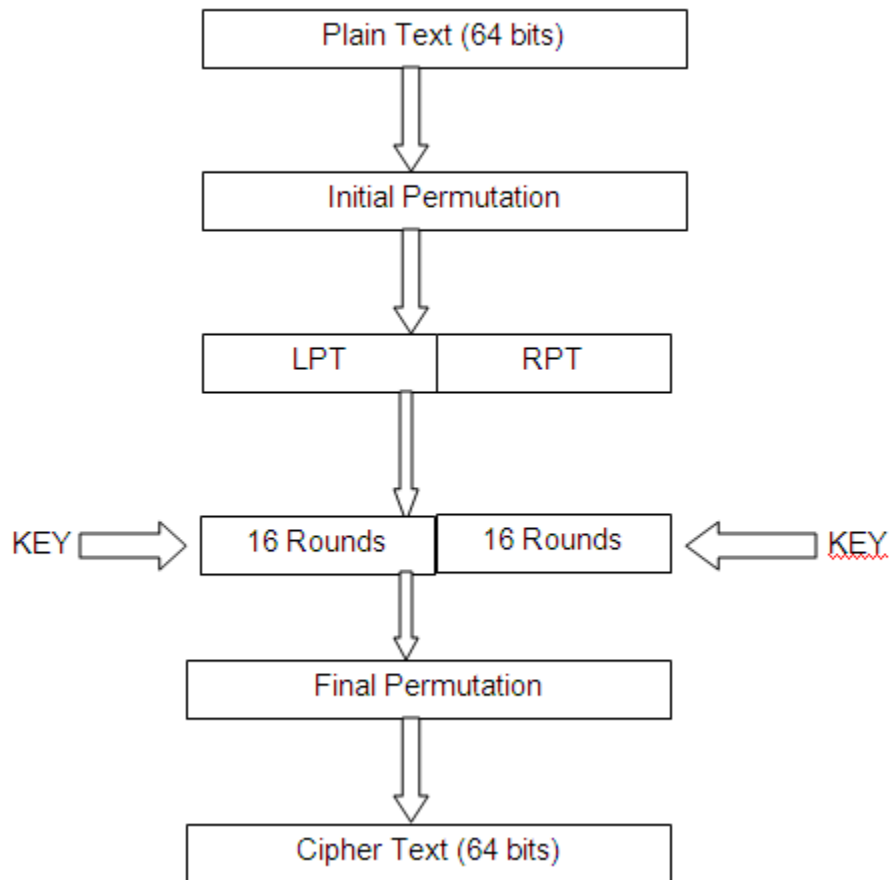


AIM: To implement DES Algorithm

DES(Data Encryption Standard) is a symmetric key algorithm. It works by using the same key for encryption & decryption of messages so both the parties must know & use same private key for the process of encryption & decryption. It uses 16 rounds structure



Program

```
packagedesalgorithm;
import java.security.spec.KeySpec;
import javax.crypto.Cipher;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.DESKeySpec;
import javax.swing.JOptionPane;
import sun.misc.BASE64Decoder;
import sun.misc.BASE64Encoder;
public class DESAlgorithm
{
    Cipher ecipher;
    Cipher dcipher;
    KeySpec keysp;
    SecretKeyFactory keyf;
    byte[] keyasbyte[];
    String enkey;
    Cipher cipher;
```

```

SecretKeyskey;
DESAlgorithm() throws Exception
{
enkey="hi.i.am.chakresh";
keyasbyte=enkey.getBytes("UTF8");           //utf8 is unicode format
keysp=new DESKeySpec(keyasbyte);
keyf=SecretKeyFactory.getInstance("DES");    //DES is encryption scheme
cipher=Cipher.getInstance("DES");
skey=keyf.generateSecret(keysp);
}

public String encrypt(String plain) throws Exception
{
String enstring="";
cipher.init(Cipher.ENCRYPT_MODE, skey);
byte[] plaintext=plain.getBytes("UTF8");
byte[] encrypttext=cipher.doFinal(plaintext);
BASE64Encoder base64en=new BASE64Encoder();
enstring=base64en.encode(encrypttext);
returnenstring;
}

public String decrypt(String enstring)throws Exception
{
String destring="";
cipher.init(Cipher.DECRYPT_MODE, skey);
BASE64Decoder base64de=new BASE64Decoder();
byte[] encntext=base64de.decodeBuffer(enstring);
byte plain[]=cipher.doFinal(encntext);
destring=byte2string(plain);
returndestring;
}

public String byte2string(byte plain[])
{
StringBuffer str=new StringBuffer();
for(int i=0;i<plain.length;i++)
str.append((char)plain[i]);
returnstr.toString();
}

public static void main(String[] args)throws Exception
{
// TODO code application logic here
{

/*Generate an encrypter*/
DESAlgorithm ds=new DESAlgorithm();
String plain="",ciphertext="";
plain = JOptionPane.showInputDialog("Enter the plain text");
ciphertext=ds.encrypt(plain);
JOptionPane.showMessageDialog(null,ciphertext);
System.out.println("\nCipher text: "+ciphertext);
plain=ds.decrypt(ciphertext);
JOptionPane.showMessageDialog(null,plain);
//System.out.println("\nPlain text: "+plain);
}
}

```

```
}  
}
```

Output

