PAGE NO.: 10
DATE : / / 20

Practical No. 5

Aim:- To write a program to implement Blowfish Algorithm.

Theory:- Blowfish is an encryption technique designed by Bruce Schneier in 1993 as an alternative to DES Encryption Technique. It is significantly faster than DES and provide a good encryption rate with no effective cryptanalysis technique found to date. It is one of the first secure block cipher not subject to any patents and hence freely available for anyone to use. It is symmetric block algorithm.

Algorithm:- Steps for encryption

Step 1:- Generation of Subkeys
These 18 subkeys are stored in a P-array with each array element being a 32-bit entry. It is initialized with digits of pi. The resultant P-array holds 18 subkeys that is used during the entire encryption process.

Step 2:- Initialize substitution boxes
4 substitution boxes are needed with each S-box having 256 entries where each entry is 32-bit. It is initialized with the digits of pi(?) after initializing the P-array.

Step 3:- Encryption
The encryption function consists of two parts =>
a) Rounds :- The encryption consists of 16 rounds with each round (Ri) taking inputs the plain text from previous round.
b) Post processing :- The output after the 16 rounds is ~~proper~~ processed.

Steps for Decryption =>

Teacher's Signature _____

Step 1 :- Generation of Subkeys
These 18 subkeys are stored in a P-array with each array element being a 32 bit entry. It is initialized with digits of pi (π).

Step 2 :- 4 substitution boxes initialization.
4 boxes are needed with each S-box having 256 entries where each entry is 32-bit. It is initialized with the digits of pi(π) after initializing the P-array.

Step 3 :- Decryption
It consists of two parts

a. Rounds :- The decryption also consists of 16 rounds with each round (Ri) taking inputs the cipher text from previous round.

b. Post-processing :- The output after the 16 round is processed.

Example.

1) Assume the message "Hi world" consists of seven characters plus a space, for a total of eight bytes or 64-bits.

2) It divides the input into 32 bits. Key expansion produces a value termed P_1, which is then XORed with the remaining 32 bits, "Hiw", to produce the desired result.

3) The 32 bits are then divided into 4 bytes each and delivered to four boxes after P1 has gone via a transformational F-function.

4) The first two value from the first two S-boxes are combined and then the third value from the third S-box.

5) To create 32 bits as the output, this result is added to the 4th S-box's output.

6) To create output F, the output of F in is XORed with the right 32 bits of the input message "old".

7) The left half is replaced by Fi' and right half by P1'

8) For a total of 16 rounds, the identical procedure is performed for each new message of P-array.

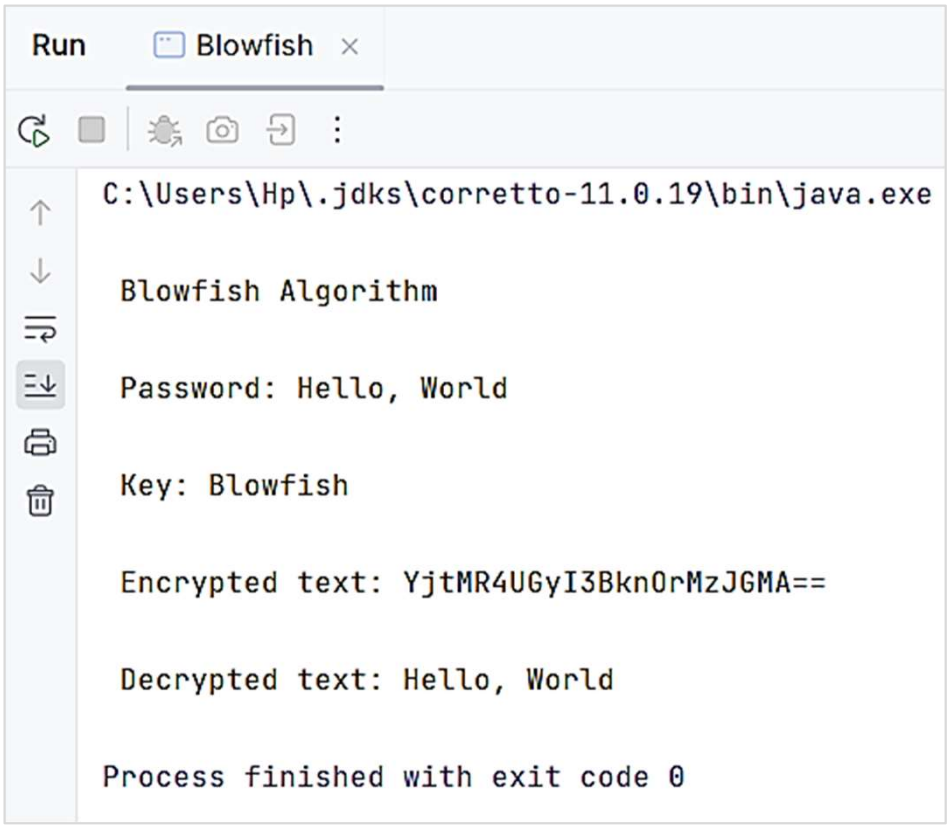
Amar KRISH

Teacher's Signature _____

Program:

```
import java.io.UnsupportedEncodingException;
import java.nio.charset.Charset;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.util.Base64;
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.spec.SecretKeySpec;

public class Blowfish {

   public String encrypt(String password, String key) throws
         NoSuchAlgorithmException, NoSuchPaddingException,
         InvalidKeyException, IllegalBlockSizeException,
         BadPaddingException, UnsupportedEncodingException {
      byte[] KeyData = key.getBytes();
      SecretKeySpec KS = new SecretKeySpec(KeyData, "Blowfish");
      Cipher cipher = Cipher.getInstance("Blowfish");
      cipher.init(Cipher.ENCRYPT_MODE, KS);
      String encryptedtext = Base64.getEncoder().
            encodeToString(cipher.doFinal(password.getBytes("UTF-8")));
      return encryptedtext;

   }

   public String decrypt(String encryptedtext, String key)
         throws NoSuchAlgorithmException, NoSuchPaddingException,
         InvalidKeyException, IllegalBlockSizeException,
         BadPaddingException {
      byte[] KeyData = key.getBytes();
      SecretKeySpec KS = new SecretKeySpec(KeyData, "Blowfish");
      byte[] ecryptedtexttobytes = Base64.getDecoder().
            decode(encryptedtext);
      Cipher cipher = Cipher.getInstance("Blowfish");
      cipher.init(Cipher.DECRYPT_MODE, KS);
      byte[] decrypted = cipher.doFinal(ecryptedtexttobytes);
      String decryptedString =
            new String(decrypted, Charset.forName("UTF-8"));
      return decryptedString;

   }

   public static void main(String[] args) throws Exception {
      final String password = "Hello, World";
      final String key = "Blowfish";
      System.out.println("\n Blowfish Algorithm");
      System.out.println("\n Password: " + password);
      Blowfish obj = new Blowfish();
      String enc_output = obj.encrypt(password, key);
      System.out.println("\n Key: " + key);
      System.out.println("\n Encrypted text: " + enc_output);
      String dec_output = obj.decrypt(enc_output, key);
      System.out.println("\n Decrypted text: " + dec_output);
   }
}
```

Output:


Run
Blowfish
C:\Users\Hp\.jdks\corretto-11.0.19\bin\java.exe
Blowfish Algorithm
Password: Hello, World
Key: Blowfish
Encrypted text: YjtMR4UGyI3BknOrMzJGMA==
Decrypted text: Hello, World
Process finished with exit code 0