Elgamal Algorithm

```
public class Main {

    public static void main(String[] args) {

        int p = 19;
        int g = 10;
        int x = 5;
        int m = 17;
        int k = 6;

        double Y = (Math.pow(g, x)) % p;

        System.out.println("Y:" + Y);

        double key = (Math.pow(Y, k)) % p;

        System.out.println("Key generated:" + key);

        double C1 = (Math.pow(g, k)) % p;
        double C2 = (key * m) % p;

        double Deckey = (Math.pow(C1, x)) % p;

        System.out.println("Dec key:" + Deckey);

        Double keyinv = (double) calmodInv(Deckey, p);

        System.out.println(keyinv);

        Double Decmsg = (C2 * keyinv) % p;

        System.out.println(Decmsg);

        if (key == Deckey) {
            System.out.println("Key matches:" + Deckey);
        }

        if (m == Decmsg) {
            System.out.println("Message matches:");
        }
    }

    static int calmodInv(double a, int b) {
        a = a % b;

        for (int x = 1; x < b; x++)
            if (((a * x) % b) == 1)
                return x;

        return 1;
    }
}
```

Blowfish Algorithm

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.security.Key;
import java.util.Base64;
import javax.crypto.Cipher;
import javax.crypto.CipherOutputStream;
import javax.crypto.KeyGenerator;

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

      KeyGenerator keyGenerator= KeyGenerator.getInstance("Blowfish");
      keyGenerator.init(128);
      Key secretKey = keyGenerator.generateKey();
      Cipher cipherOut = Cipher.getInstance("Blowfish/CFB/NoPadding");
      cipherOut.init(Cipher.ENCRYPT_MODE,secretKey);

      Base64.Encoder encoder = Base64.getEncoder();
      byte iv[] = cipherOut.getIV();
      if(iv!=null){
         System.out.println("Initialization vector of the Cipher:" +
encoder.encodeToString(iv));
      }
      FileInputStream fin = new FileInputStream("inputFile.txt");
      FileOutputStream fout = new FileOutputStream("outputFile.txt");
      CipherOutputStream cout = new CipherOutputStream(fout,cipherOut);
      int input;
      while((input = fin.read()) != -1) {
         cout.write(input);
      }

      fin.close();
      cout.close();
   }
}
```

----

inputFile.txt (Create this file in project root directory, type the below text and save it)

Welcome to Security Laboratory

Tiny Encryption Algorithm(TEA)

```
import java.io.*;

public class tea {
    int delta = 0x9e3779b9; // (2^32 golden ratio, key scheduling constant)
    int sum = 0;
    int k[] = new int[4];
    int p[] = new int[2];
    int L, R;

    public void encrypt() {
        // Split 32 bits
        L = p[0];
        R = p[1];

        for (int i = 1; i <= 32; i++) {
            sum += delta;

            // L += ((R << 4) + K[0]) XOR (R + sum) XOR ((R >> 5) + K[1])
            L += (((R << 4) + (k[0])) ^ (R + sum) ^ ((R >> 5) + (k[1])));

            // R += ((L << 4) + K[2]) XOR (L + sum) XOR ((L >>5 ) + K[3])
            R += (((L << 4) + (k[2])) ^ (L + sum) ^ ((L >> 5) + (k[3])));
        }

        System.out.println("Ciphertext is L: " + L + ", R: " + R);
    }

    public void decrypt() {
    }

    public void getKey() {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        int count = 0, idx = 0;

        try {
            String str = br.readLine();
            while (count <= 3) {
                k[count++] = Integer.parseInt(str.substring(idx, idx + 2));
                idx += 2;
            }
        } catch (IOException ex) {
            ex.printStackTrace();
        } catch (NumberFormatException ex) {
            ex.printStackTrace();
        }
    }

    public void getPlainText() {
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        int count = 0, idx = 0;

        try {
            String number = br.readLine();
            while (count <= 1) {
                p[count++] = Integer.parseInt(number.substring(idx, idx + 2));
                idx += 2;
            }
        } catch (IOException ex) {
            ex.printStackTrace();
        } catch (NumberFormatException ex) {
            ex.printStackTrace();
        }
    }

    public static void main(String args[]) {
        System.out.println("TEA ENCRYPTION: ");
        tea t = new tea();
        System.out.println("Enter the key: ");
        t.getKey();
        System.out.println("Enter the number: ");
        t.getPlainText();
        t.encrypt();
    }
}
```