МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

Інститут комп'ютерних наук та інформаційних технологій

Кафедра інформаційних систем та мереж



ЗВІТ

про виконання лабораторної роботи № 5

«Захист даних з допомогою мереж Фейстеля»

з дисципліни

«Технології захисту інформації»

Виконала:

студентка групи ІТ-31

Щербак Л. В. Прийняв викладач:

Досин Д.Г.

Львів 2022

**Мета роботи:** Вивчити принципи роботи мережі Фейстеля, навчитися

шифрувати інформацію за допомогою використання блокового

криптоалгоритму.

**Завдання роботи:**

індивідуальне завдання з детальним формулюванням розв’язуваної

задачі, використовувані (власні) теоретичні відомості;

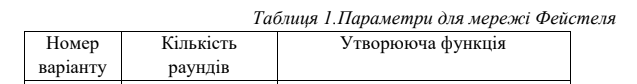
1. Написати програму на мові java яка виконує криптографічні перетворення відповідно до заданого варіанту

2. Розробити програму шифрування і дешифрування тексту. Провести

шифрування вихідного тексту, отримати шифрограму, здійснити її

дешифрування і порівняння з вихідним текстом.

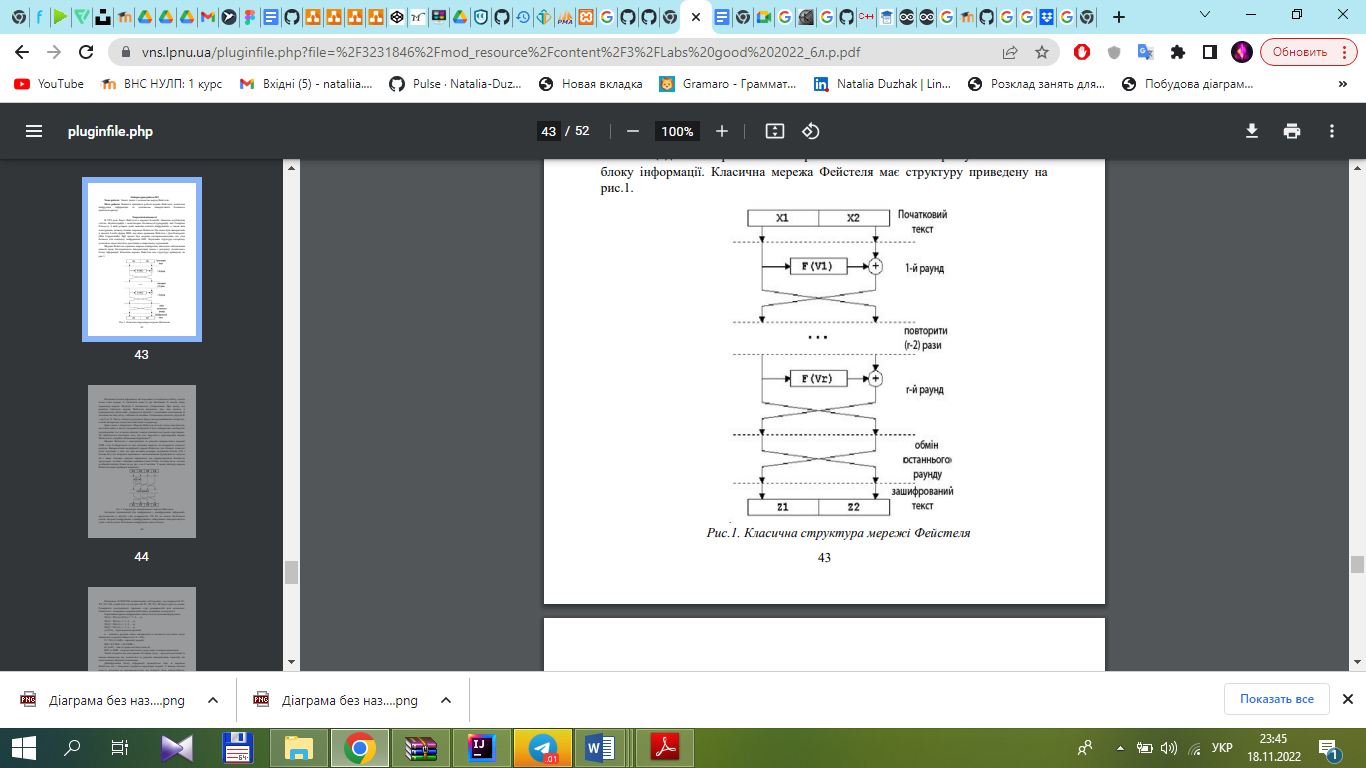
3. Оформити звіт згідно вимог.





**Відповіді на контрольні питання:**

1. Представте класичну структуру мережі Фейстеля.



2. Опишіть поняття раунда в мережі Фейстеля?

Під раундом (циклом) мережі Фейстеля розуміють дію, яка полягає в одноразовому обчисленні утворюючої функції з подальшим накладенням її значення на іншу вітку з обміном їх місцями. Часто, кількість раундів не фіксується розробниками алгоритму, а лише вказуються допустимі межі даного параметру

3. Якими властивостями володіє мережа Фейстеля?

Мережа Фейстеля володіє такою властивістю, що навіть якщо в якості утворюючої функції F буде використано необоротне перетворення, то і в цьому випадку можна відновити всі кроки перетворень. Це відбувається внаслідок того, що для зворотного перетворення мережі Фейстеля не потрібно обчислювати функцію F−1 .

4. Яким чином здійснюється використання ключа при шифруванні?

У явному вигляді ключ в алгоритмі не використовується, що підвищує його крипостійкість. При знанні ключа, але відсутності інформації щодо кількості раундів криптоаналітику буде досить складно дешифрувати зашифровану інформацію.

**Текст програми реалізації:**

package lab5;

import java.math.BigInteger;

import java.util.ArrayList;

import java.util.Random;

public class Feistel {

//Fill the message

public static String fillUp(String s, int diff) {

for (int i = 0; i < diff; i++) {

s = "0" + s.substring(0);

}

return s;

}

//Perform the final encryption by comparing the values of two binary numbers

//If two bits of the same position have the same value, the resulting bit will be 0, otherwise it will be 1.

public static String xor(String first, String second) {

int compare = first.length() - second.length();

if (compare < 0) {

first = fillUp(first, Math.abs(compare));

} else if (compare > 0) {

second = fillUp(second, compare);

}

StringBuilder builder = new StringBuilder();

char[] fir = first.toCharArray();

char[] sec = second.toCharArray();

for (int i = 0; i < first.length(); i++) {

if (fir[i] == sec[i]) {

builder.append("0");

} else {

builder.append("1");

}

}

return builder.toString();

}

//Shift the value of the message according to the key

public static String func(String s, String key) {

String token = xor(s, key);

//shift the code, respectively 0 or 1

String shiftedToken = token.substring(1) + token.substring(0, 1);;

return shiftedToken;

}

//Exchange the right and left parts with each other, several times, depending on the number of rounds

public static void alg(ArrayList<String> keys, String right, String left, int position) {

String token = func(right, keys.get(position));

String temp = right;

right = xor(left, token);

left = temp;

}

//Encrypt the message by adding the left part to the right part

public static String encrypt(ArrayList<String> keys, String msg) {

String left = msg.substring(0, msg.length() / 2);

String right = msg.substring(msg.length() / 2);

int compare = left.length() - right.length();

if (compare < 0) {

left = fillUp(left, Math.abs(compare));

} else if (compare > 0) {

right = fillUp(right, compare);

}

for (int i = 0; i < keys.size(); i++) {

alg(keys, right, left, i);

}

return right + left;

}

//Decode the message, determining the position of the data in the message

public static String decrypt(ArrayList<String> keys, String msg) {

int length = msg.length();

String left = msg.substring(0, msg.length() / 2);

String right = msg.substring(msg.length() / 2);

int compare = left.length() - right.length();

if (compare < 0) {

left = fillUp(left, Math.abs(compare));

} else if (compare > 0) {

right = fillUp(right, compare);

}

for (int i = keys.size() - 1; i >= 0; i--) {

alg(keys, right, left, i);

}

String r = right + left;

return r.substring(r.length() - length);

}

//Creating list of Keys to encode/decode

public static ArrayList<String> createKeys(int numRounds) {

ArrayList<String> keys = new ArrayList<>();

Random rnd = new Random();

for ( int i = 0; i < numRounds; i++) {

BigInteger key = new BigInteger(8, rnd);

keys.add(key.toString(2));

}

return keys;

}

public static void print(ArrayList<String> keys) {

System.out.print("Keys = [");

for ( int i = 0; i < keys.size()-1; i++ ){

System.out.print(keys.get(i) +", ");

}

System.out.println(keys.get(keys.size()-1) +"]");

}

public static void main(String[] args) {

int numRounds = 30;// Number of Rounds

String msg = "11111000";//Original Message in binary format

System.out.println("Number of Rounds : " + numRounds);

System.out.println("Original Message in binary format : " + msg);

//Creating list of Keys to encode/decode

ArrayList<String> keys = createKeys(numRounds);

//Print List of Keys

print(keys);

//Encode

String encryptedText = encrypt(keys, msg);

System.out.println("Encrypted message in binary format : " + encryptedText);

//Decode

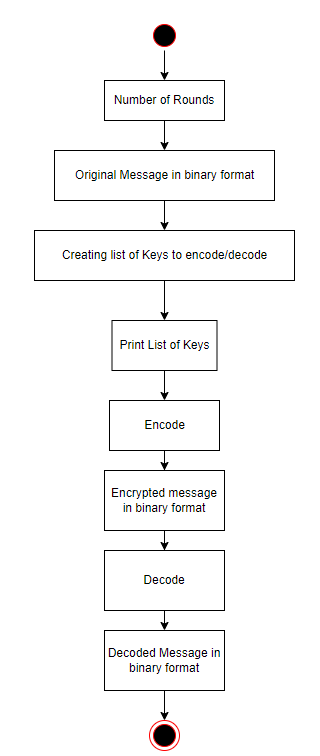
String decryptedText = decrypt(keys, encryptedText);

System.out.println("Decrypted message in binary format : " + decryptedText);

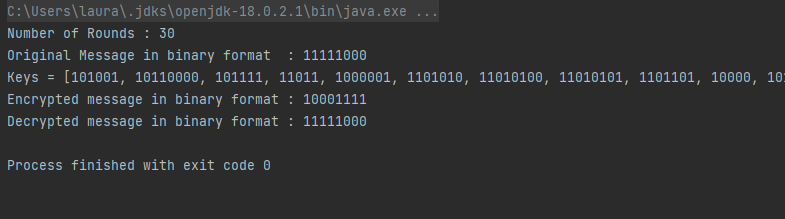
}

}

**Блок-схема:**



**Результати роботи програми:**



**Висновки:** Вивчила принципи роботи мережі Фейстеля, навчилася

шифрувати інформацію за допомогою використання блокового

криптоалгоритму.