Учреждение образования

«Белорусский государственный технологический университет»

**Отчет по лабораторной работе №4**

по дисциплине: «Защита информации и надежность информационных систем»

Исполнитель:

Воликов Д. А., ФИТ 4-7

Руководитель:

Асс. Савельева М. Г.

Минск 2024

# Лабораторная работа №4. Изучение устройства и функциональных особенностей шифровальной машины «энигма»

**Цель**: изучение и приобретение практических навыков разработки и использования приложений для реализации перестановочных шифров (работа рассчитана на 4 часа аудиторных занятий).

**Задачи**:

1. Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости перестановочных шифров.
2. Изучить структуру, принципы функционирования, реализацию процедур зашифрования сообщений в машинах семейства «Энигма».
3. Изучить и приобрести практические навыки выполнения криптопреобразований информации на платформе «Энигма», реализованной в виде симуляторов.
4. Получить практические навыки оценки криптостойкости подстановочных и перестановочных шифров на платформе «Энигма».
5. Результаты выполнения лабораторной работы оформить в виде отчета о проведенных исследованиях, методике выполнения практической части задания и оценке криптостойкости шифров.

## Теоретические сведения

Идея создания шифровального устройства высказана голландцем Гуго Кох де Дельфтю (в некоторых источниках – Гуго Александр Кох, Hugo Alexander Koch) еще в 1919 г. В 1920 г. он же изобрел первую роторную шифровальную машинку. Параллельно с этим немец Артур Шербиус (Arthur Scherbius) изучал проблему криптостойкости (в нашем современном понимании) шифровальных машин. Он же получил патент на такую машину, которую назвали «Энигма». Основная особенность «Энигмы» – все знали в то время алгоритм шифрования, но никто не мог подобрать нужный ключ.

Первая шифровальная машина, «Enigma A», появилась на рынке в 1923 г. Это была большая и тяжелая машина со встроенной пишущей машинкой и весом около 50 кг. Вскоре после этого была представлена «Enigma B», очень похожая на «Enigma A». Вес и размеры этих машин сделали их непривлекательными для использования в военных целях.

По достоинству шифровальную машину оценили в немецкой армии. В 1925 г. ее принял на вооружение сначала военно-морской флот (модель С), а в 1930 г.– Вермахт («Enigma I»). Общее количество шифраторов, произведенных до и во время Второй мировой войны, превысило 100 тысяч. Применялись они всеми видами вооруженных сил Германии, а также военной разведкой и службой безопасности.

Машина «Энигма» – это электромеханическое устройство. Как и другие роторные машины, «Энигма» состоит из комбинации механических и электрических подсистем.

Механическая часть включает в себя клавиатуру, набор вращающихся дисков – роторов, которые расположены вдоль вала и прилегают к нему, и ступенчатого механизма, двигающего один или несколько роторов при каждом нажатии на клавишу.

Электрическая часть, в свою очередь, состоит из электрической схемы, соединяющей между собой клавиатуру, коммутационную панель, лампочки и роторы (для соединения роторов использовались скользящие контакты).

Конкретный механизм мог быть разным, но общий принцип был таков: при каждом нажатии на клавишу самый правый ротор сдвигался на одну позицию, а при определенных условиях сдвигались и другие роторы. Движение роторов приводило к различным криптографическим преобразованиям при каждом следующем нажатии на клавишу на клавиатуре, т. е. зашифрование/расшифрование сообщений основано на выполнении ряда замен (подстановок) одного символа другим. Идея А. Шербиуса состояла в том, чтобы добиться этих подстановок электрическими связями.

Механические части двигались и, замыкая контакты, образовывали меняющийся электрический контур. При нажатии на клавишу клавиатуры контур замыкается, ток проходит через созданную (для зашифрования/расшифрования одного конкретного символа сообщения) цепь и в результате включает одну из набора лампочек, отображающую искомую букву шифртекста (или расшифрованного сообщения).

## 1.2 Практическое задание

**Вариант 2**. По условию в машине используются роторы II (левый), III (центральный) и V (правый). Рефлектор типа *B*. С помощью разработанного приложения зашифровать сообщение, применив не менее 5 вариантов начальных установок роторов.

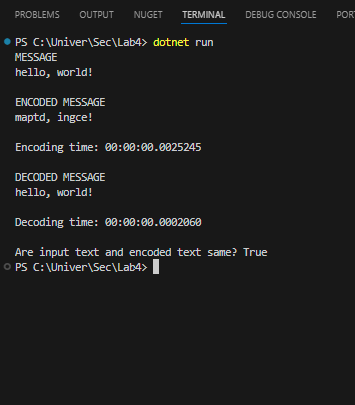


Рисунок 1 – Зашифрование/расшифрование сообщения с установленными значениями роторами 1-1-1

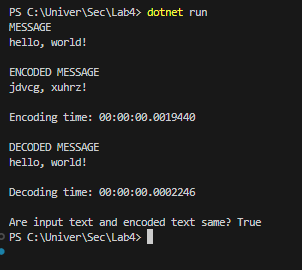


Рисунок 2 – Зашифрование/расшифрование сообщения с установленными значениями роторами 15-20-15

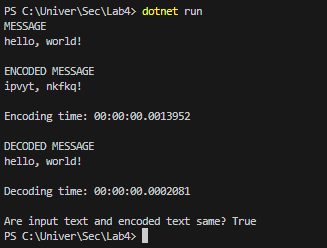


Рисунок 3 – Зашифрование/расшифрование сообщения с установленными значениями роторами 4-5-1

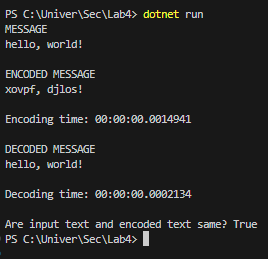


Рисунок 4 – Зашифрование/расшифрование сообщения с установленными значениями роторами 1-2-3

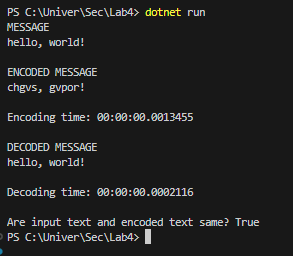


Рисунок 5 – Зашифрование/расшифрование сообщения с установленными значениями роторами 3-5-7

Как можно заметить, сообщение корректно зашифровывается и расшифровывается. Теперь оценим криптостойкость кода. Зная, что используется три ротора, соответственно комбинаций ключей будет = 26 ⋅ 26 ×  
× 26 = 17 576.

Количество различных перестановок роторов: всего имеется 10 роторов, из которых одномоментно может использоваться только 3, поэтому количество перестановок = 10 ⋅ 9 ⋅ 8 = 720.

Коммутационная панель обеспечивает самый большой набор возможных настроек. Для первого кабеля одна сторона может иметь любое из 26 положений, а другая сторона – любое из 25 оставшихся положений. Однако поскольку комбинация и ее обратная сторона идентичны (AB такая же, как BA), мы должны игнорировать все двойные числа во всех возможных комбинациях для одного кабеля, предоставляя (26 ⋅ 25) / (1! ⋅ 2­1), или 325 уникальных способов коммутаций одним кабелем. У нас в машине используется 10 кабелей, следовательно 26! / (26 – 2 ⋅ 10)! ⋅ 10! ⋅ 210 = 2.498 ⋅ 1035 комбинаций.

Таким образом наша модель машины Энигма, состоящая из 3 роторов (с выбором из 10), отражателя B Dunn и 10 штекерных кабелей для коммутационной панели может быть настроена на 2.498 ⋅ 1035 различных состояний, что сопоставимо с 117-битным ключом.

Вывод: в ходе этой лабораторной работы я изучил принцип работы шифровальной машины «Энигма». Реализовал программу, которая зашифровывает и расшифровывает сообщение. Рассчитал криптостойкость для машины «Энигма» с тремя роторами, рефлектором и 10 коммутаторами.

# Листинг «Лабораторная работа №4»

namespace Lab4.Rotors

{

public class Rotor

{

public Dictionary<char, char> Sequence { get; private set; } = [];

public int Position { get; private set; } = 1;

public int RotationNumber { get; private set; } = 1;

public string SequenceFileText { get; private set; }

public Rotor(string sequenceFile)

{

SequenceFileText = File.ReadAllText(sequenceFile).ToLower();

SetAlphabet();

}

public void SetPosition(int position)

{

if (position < 1 || position > 26) {

throw new Exception("Rotor positon must be in range from 1 to 26");

}

Position = 1;

SetAlphabet();

for (int i = 1; i < position; i++) {

Rotate();

}

}

public void SetRotationNumber(int rotationNumber)

{

if (rotationNumber < 1 || rotationNumber > 26) {

throw new Exception("Rotor rotation number must be in range from 1 to 26");

}

RotationNumber = rotationNumber;

}

public char GetChar(char input) => Sequence[input];

public void Rotate()

{

char keyChar = 'a';

char dictionaryFirstValue = Sequence[keyChar];

foreach (var item in Sequence)

{

Sequence[item.Key] = item.Key == 'z' ? dictionaryFirstValue : Sequence[++keyChar];

}

Position = Position == 26 ? 1 : Position++;

}

private void SetAlphabet()

{

char startChar = 'a';

SequenceFileText.ToCharArray().ToList().ForEach((ch) => Sequence[startChar++] = ch);

}

}

}

Листинг 4.1 – Код ротора

namespace Lab4.Reflectors

{

public class Reflector

{

public Dictionary<char, char> Pairs { get; private set; } = [];

public Reflector(string reflectorFile){

var reflectorPairs = File.ReadAllLines(reflectorFile);

foreach (var pair in reflectorPairs)

{

Pairs.Add(pair!.ToLower()[0], pair!.ToLower()[1]);

Pairs.Add(pair!.ToLower()[1], pair!.ToLower()[0]);

}

}

public char GetChar(char input) => Pairs[input];

}

}

Листинг 4.2 – Код рефлектора

using Lab4.Reflectors;

using Lab4.Rotors;

namespace Lab4.Enigma

{

public class Enigma(List<Rotor> rotors, Reflector reflector)

{

public readonly List<Rotor> rotors = rotors;

readonly Reflector reflector = reflector;

public void SetRotorsPositions(int[] positions)

{

if (positions.Length != rotors.Count) {

throw new Exception("Wrong numbers of rotors positions");

}

for (int i = 0; i < rotors.Count; i++) {

rotors[i].SetPosition(positions[i]);

}

}

public void SetRotorsRotationNumbers(int[] rotationNumbers)

{

if (rotationNumbers.Length != rotors.Count) {

throw new Exception("Wrong numbers of rotors rotation numbers");

}

for (int i = 0; i < rotors.Count; i++) {

rotors[i].SetRotationNumber(rotationNumbers[i]);

}

}

public string EncodeMessage(string message)

{

string alphabet = "abcdefghijklmnopqrstuvwxyz";

string encodedMessage = "";

foreach (var ch in message)

{

if (!alphabet.Contains(ch)) {

encodedMessage += ch;

continue;

}

char encodedChar = ch;

rotors.ForEach((rotor) => encodedChar = rotor.GetChar(encodedChar));

encodedChar = reflector.GetChar(encodedChar);

for (int i = rotors.Count - 1; i >= 0; i--)

{

encodedChar = rotors[i].Sequence.FirstOrDefault(item => item.Value == encodedChar).Key;

}

for (int i = 0; i < rotors.Count; i++) {

for (int j = 0; j < rotors[i].RotationNumber; j++) {

rotors[i].Rotate();

}

}

encodedMessage += encodedChar;

}

return encodedMessage;

}

public string DecodeMessage(string message) => EncodeMessage(message);

}

}

Листинг 4.3 – Код энигмы

using Lab4.Enigma;

using Lab4.Reflectors;

using Lab4.Rotors;

List<Rotor> rotors = [

new Rotor(@"./Rotors/RotorFive.txt"),

new Rotor(@"./Rotors/RotorThree.txt"),

new Rotor(@"./Rotors/RotorTwo.txt"),

]; // First - right rotor, Last - left rotor

Reflector reflector = new(@"./Reflectors/ReflectorC.txt");

Enigma enigma = new(rotors, reflector);

var rotorsPositions = new int[] {3, 5, 7};

enigma.SetRotorsPositions(rotorsPositions);

enigma.SetRotorsRotationNumbers([1, 2, 2]);

string message = File.ReadAllText(@"./file.txt").ToLower();

if (message == "") {

throw new Exception("Empty message");

}

Console.WriteLine($"MESSAGE\n{message}\n");

var watch = System.Diagnostics.Stopwatch.StartNew();

string encodedMessage = enigma.EncodeMessage(message);

watch.Stop();

Console.WriteLine($"ENCODED MESSAGE\n{encodedMessage}\n\nEncoding time: {watch.Elapsed}\n");

enigma.SetRotorsPositions(rotorsPositions);

watch = System.Diagnostics.Stopwatch.StartNew();

string decodedMessage = enigma.DecodeMessage(encodedMessage);

watch.Stop();

Console.WriteLine($"DECODED MESSAGE\n{decodedMessage}\n\nDecoding time: {watch.Elapsed}\n");

Console.WriteLine($"Are input text and encoded text same? {message == decodedMessage}");

Листинг 4.4 – Код входной точки программы