Учреждение образования «Белорусский государственный технологический университет»

Факультет Информационных технологий

Кафедра Информационных систем и технологий

Отчёт по лабораторной работе №6

по дисциплине **«Криптографические методы защиты информации»**

Тема: изучение устройства и функциональных особенностей шифровальной машины “Энигма”

Руководитель: ст. преп. Блинова Е.А.

Выполнил: студент 3 курса,

ФИТ ПОИТ, 5 группы Храмых В.О.

Минск, 2020

**Цель:** изучение и приобретение практических навыков разработки и использования приложений для реализации перестановочных шифров.

**Задачи:**

1. Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости подстановочно-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 проведенных исследованиях, методики выполнения практической части задания и оценки криптостойкости шифров.
6. **Описание разработанного приложения**

В результате выполнения данной лабораторной работы было разработано приложение на языке программирования С#. Приложение создано для шифрования исходного текста (имени и фамилии студента) симулятором шифровальной машины «Энигма». Согласно варианту, симулятор шифровальной машины имеет ротор II, Gamma, ротор IV и отражатель C. Каждый из роторов при зашифровании одного символа перемещается на 1 шаг.

Данные для тестирования были взяты из методического пособия в соответствии с 6 (1) вариантом.

1. **Методика выполнения экспериментов**

Изначально в коде приложения были определены данные о роторах и рефлекторе, которые показаны на рисунке 2.1.

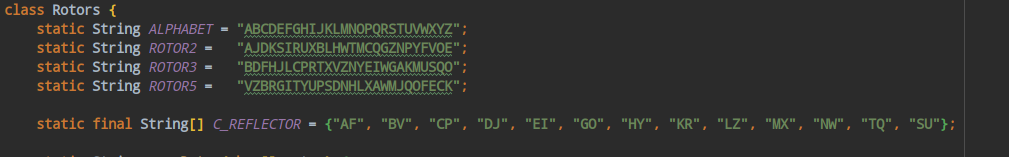


Рисунок 2.1 – роторы и рефлекторы, используемые в симуляторе

Так как по заданию 2-го варианта, начальные позиции роторов должны стоять в порядке 0-1-1, то мы сдвигаем 3 и 5 ротор по одному разу.

Следующим шагом происходит обработка каждого символа входных данных «KHRAMYKHVLAD». Каждый символ исходной строки последовательно обрабатывается в цикле, который выполняет следующий алгоритм:

Начинаем с правого ротора, а именно ротора 5.

1. Определяем символ, находящийся в роторе 5 под индексом исходного символа в алфавите;
2. Определяем символ, находящийся в роторе 3 роторе под индексом исходного символа в алфавите;
3. Определяем символ, находящийся в роторе 2 под индексом исходного символа в алфавите;
4. Находим в рефлекторе С пару символов, один из которых текущий, и заменяем его на другой символ в паре;
5. Выполняем последовательно пункты 3, 2, 1;
6. Символ, найденный в пункте 5, записываем в результирующую строку;
7. Выполняем сдвиг всех роторов на 1 шаг.

Программная реализация данного алгоритма представлена на рисунке 2.2.

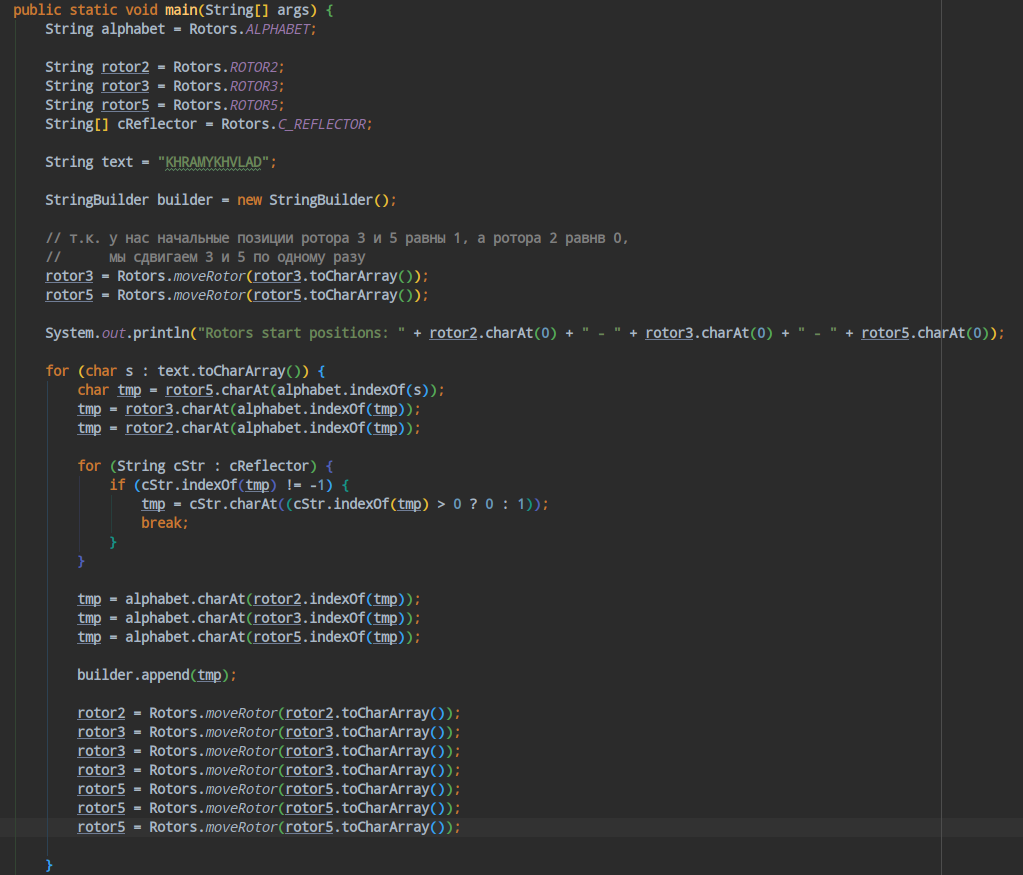


Рисунок 2.2 – программная реализация вышепреведенного алгоритма

Для работы данного алгоритма был разработан метод moveRotor. Он позволяет выполнить сдвиг ротора на 1 шаг. Листинг функции представлен на рисунке 2.3.

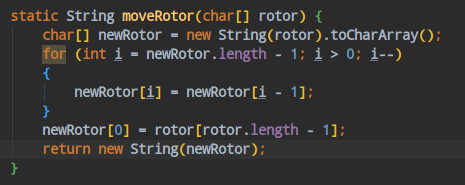


Рисунок 2.3 – функция Move\_Rotor\_1 (Program.cs)

Протестируем симулятор при различных начальных положениях роторов. Входную строку не изменяем.

Установим начальное положение роторов в A-B-V. Результат выполения программы с такими входными параметрами представлен на рисунке 2.4.

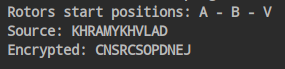


Рисунок 2.4 – тестирование программы при начальном положении роторов установленом в A-O-K

Установим начальное положение роторов в E-Q-K. Результат выполения программы с такими входными параметрами представлен на рисунке 2.5.

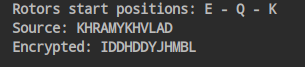


Рисунок 2.5 – тестирование программы при начальном положении роторов установленом в K-K-V

Установим начальное положение роторов в F-K-E. Результат выполения программы с такими входными параметрами представлен на рисунке 2.6.

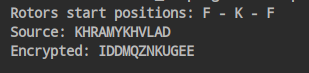


Рисунок 2.6 – тестирование программы при начальном положении роторов установленом в F-Q-E

Установим начальное положение роторов в O-U-F. Результат выполения программы с такими входными параметрами представлен на рисунке 2.7.

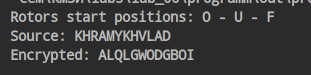


Рисунок 2.7 – тестирование программы при начальном положении роторов установленом в Z-A-J

Установим начальное положение роторов в A-U-F. Результат выполения программы с такими входными параметрами представлен на рисунке 2.8.

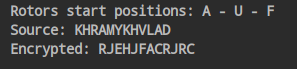


Рисунок 2.8 – тестирование программы при начальном положении роторов установленом в M-Y-H

Так как шифр симметричный, заменив исходную строку зашифрованной ранее, мы видим наше имя-фамилию:

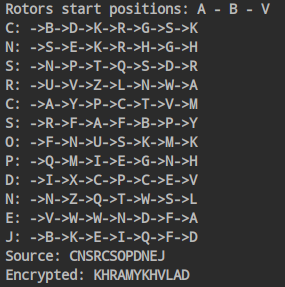


Рисунок 2.8 – пример расшифрования

Приложение соответсвует заявленым требованиям, выполняет функции симулятора машины Энигма и позволяет шифровать данные при различных установках начального положения роторов.