Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Лабораторная работа №7**

Студент: Бобрович Г.С.

ФИТ 3 курс 7 группа

Преподаватель: Савельева М.Г.

Минск 2023

**Разработать приложение для реализации указанных преподавателем методов блочного зашифрования/расшифрования.**

****

Рисунок 1.1 — Условие

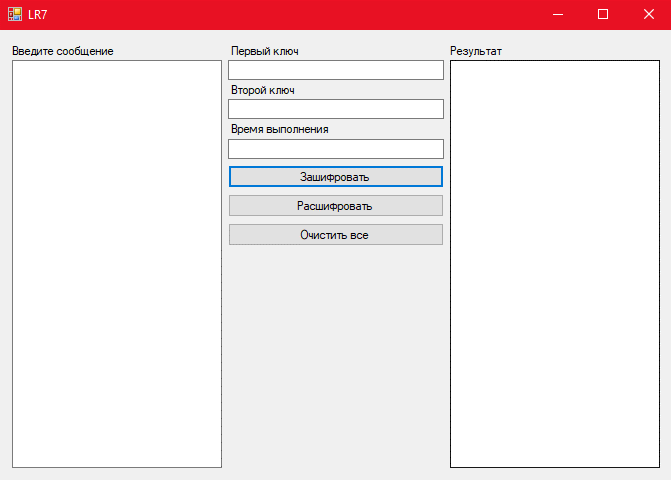
****

Рисунок 1.2 — Приложение

Для проверки работы приложения, введем какой-нибудь текст.

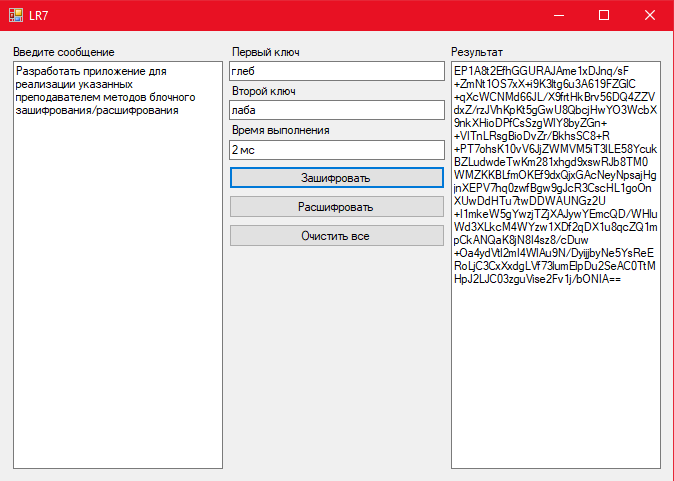


Рисунок 1.3 — Результат зашифрования

Теперь расшифруем полученный текст:

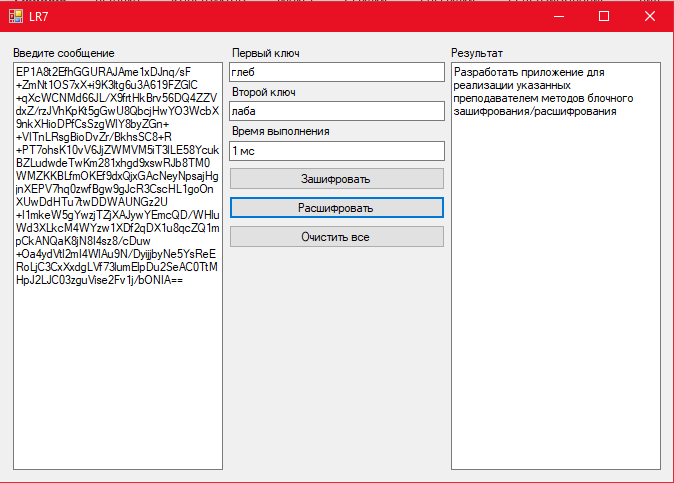


Рисунок 1.4 — Результат расшифрования

Зашифровка происходит с помощью классов, взятых из System.Security.Cryptography. Само зашифрование работает следующим образом: исходная строка (сообщение) разбивается на блоки с фиксированной длиной и дополнительным последним блоком. DES-EEE2 представляет собой операции шифрования, расшифрования, шифрования. Его отличие от других в том, что на первом и третьих шагах используется один и тот же ключ. То есть, сначала исходное сообщение зашифруется с помощью первого ключа, затем результат зашифруется с помощью второго ключа и в конце концов результат второй операции зашифруется первым ключом. Расшифровка выполняется в обратном порядке: зашифрованный текст расшифруется с помощью первого ключа, затем результат расшифруется с помощью второго и этот результат расшифруется первым ключом. В итоге, мы получим исходное сообщение

Из полученных результатов видно, что зашифровка исходного сообщения заняла 2 мс. При расшифровке зашифрованного сообщения время выполнения ­­­– 1 мс. Расшифровка происходит быстрее, так как используются данные кэша, полученные при зашифровке сообщения

**Лавинный эффект**

Для оценки лавинного эффекта зашифруем текст, затем изменим 1 символ и сравним количество символов.

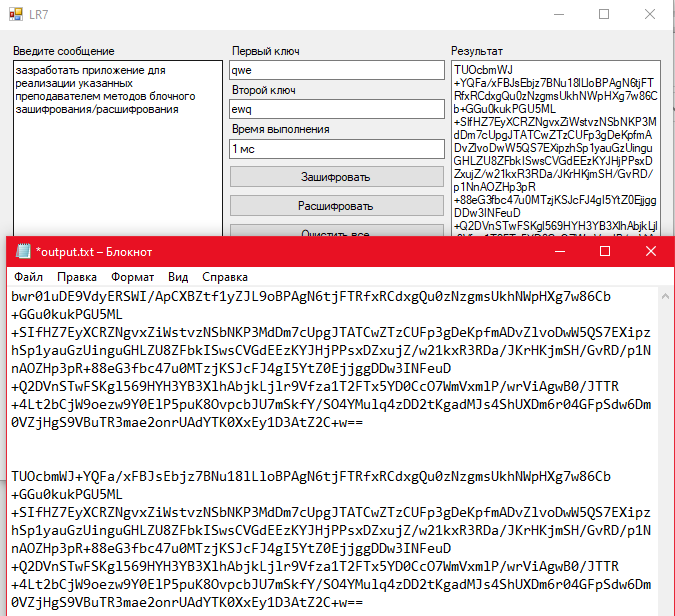


Рисунок 1.5 — Результат лавинного эффекта

Мы заменили один символ при втором шифровании текста, в следствие чего было изменено 75 битов текста.

**Слабые/полуслабые ключи**

Первоначальные ключи являются слабыми в виду того, что они изменяются при получении подключа для каждого раунда алгоритма. Если все биты каждой половины будут равны 0/1, то для всех раундов будет использован один ключ.

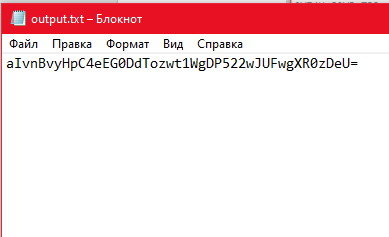


Рисунок 1.6 — Использование слабого ключа

Шифрование получилось более слабым из-за того, что ключи состоят из 0 и 1 и при использовании операции XOR с текстом он не будет изменяться, в следствие чего изменяется меньшее количество битов.

При полуслабых ключах, один из ключей может расшифровать сообщение, которое было зашифровано другим ключом из пары. При генерации полуслабых ключей генерируется два различных подключа, затем каждый из ключей используется 8 раз.

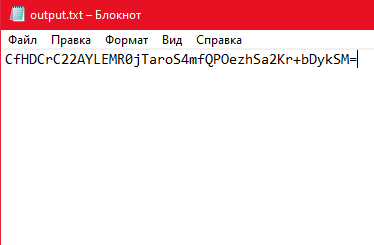


Рисунок 1.7 — Использование полуслабого ключа

**Оценка времени выполнения**

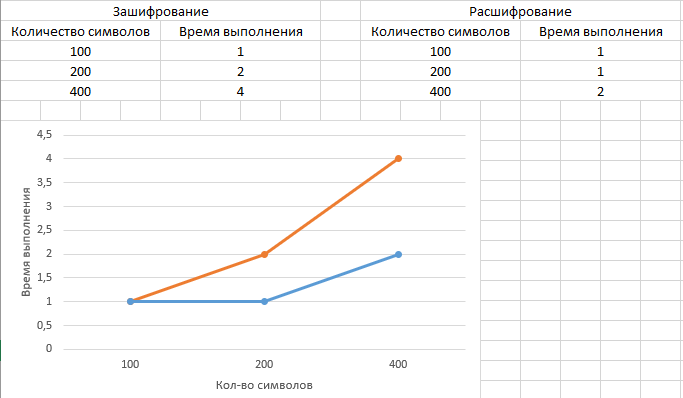
****

Рисунок 1.8 — Время выполнения

**Оценка степени сжатия текста**

Также была оценена степень сжатия файлов исходного и зашифрованного сообщений. Для исходного – 221 байт, для зашифрованного – 16,2 Кбайт. Большая разница в размере вызывается из-за лавинного эффекта, возникающего при каждом зашифровании, когда растет зависимость всех битов результата от битов исходных данных и ключа вместе с этим растет и количество символов в зашифрованном сообщении по отношении к количеству символов в исходном сообщении.

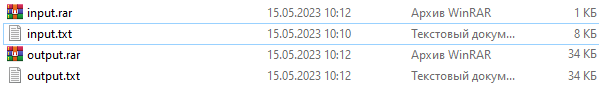


Рисунок 1.9 — Оценка степени сжатия

**Вывод**

В ходе выполнения лабораторной работы были изучены и приобретены практические навыки разработки и использования приложения для реализации блочных шифров.