Министерство образования Республики Беларусь

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Отчет к лабораторной работе № 2:

«Исследование криптографических шифров на основе подстановки символов»

Выполнил:

студент 4 курса 1 группы

Халалеенко А.Н.

Преподаватель:

Сазонова Д.В.

Минск 2024

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

**Задачи**:

* Закрепить теоретические знания по алгебраическому описанию, алгоритмам реализации операций зашифрования/расшифрования и оценке криптостойкости подстановочных шифров.
* Ознакомиться с особенностями реализации и свойствами различных перестановочных шифров на основе готового программного средства (L\_LUX).
* Разработать приложение для реализации указанных преподавателем методов подстановочного зашифрования/расшифрования.
* Выполнить исследование криптостойкости шифров на основе статистических данных о частотах появления символов в исходном и зашифрованном сообщениях.
* Оценить скорость зашифрования/расшифрования реализованных способов шифров.
* Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполнения эксперимента с использованием приложения и результатов эксперимента.

1 Теоретические сведения

Сущность подстановочного шифрования состоит в том, что исходный текст (М) и зашифрованный текст (С) основаны на использовании одного и того же или разных алфавитов, а тайной или ключевой информацией является алгоритм подстановки.

Одним из существенных недостатков моноалфавитных шифров является их низкая криптостойкость. Зачастую метод криптоанализа базируется на частоте встречаемости букв исходного текста.

Применяя одновременно операции сложения и умножения по модулю n над элементами множества (индексами букв алфавита), можно получить систему подстановок, которую называют аффинной системой подстановок Цезаря.

Омофонические шифры (омофоническая замена), или однозвучные шифры подстановки, создавались с целью увеличить сложность частотного анализа шифртекстов путем маскировки реальных частот появления символов текста с помощью омофонии.

Существует определенное сходство между подстановочными шифрами и шифрами на основе гаммирования. Последние рассматриваются как самостоятельный класс. Такие шифры схожи с подстановочными (и в определенном плане – с перестановочными) тем, что в обоих случаях можно использовать табличное представление выполняемых операций на основе ключей. В шифрах на основе гаммирования и в подстановочных шифрах при зашифровании происходит подмена одних символов другими.

2 Практическая часть

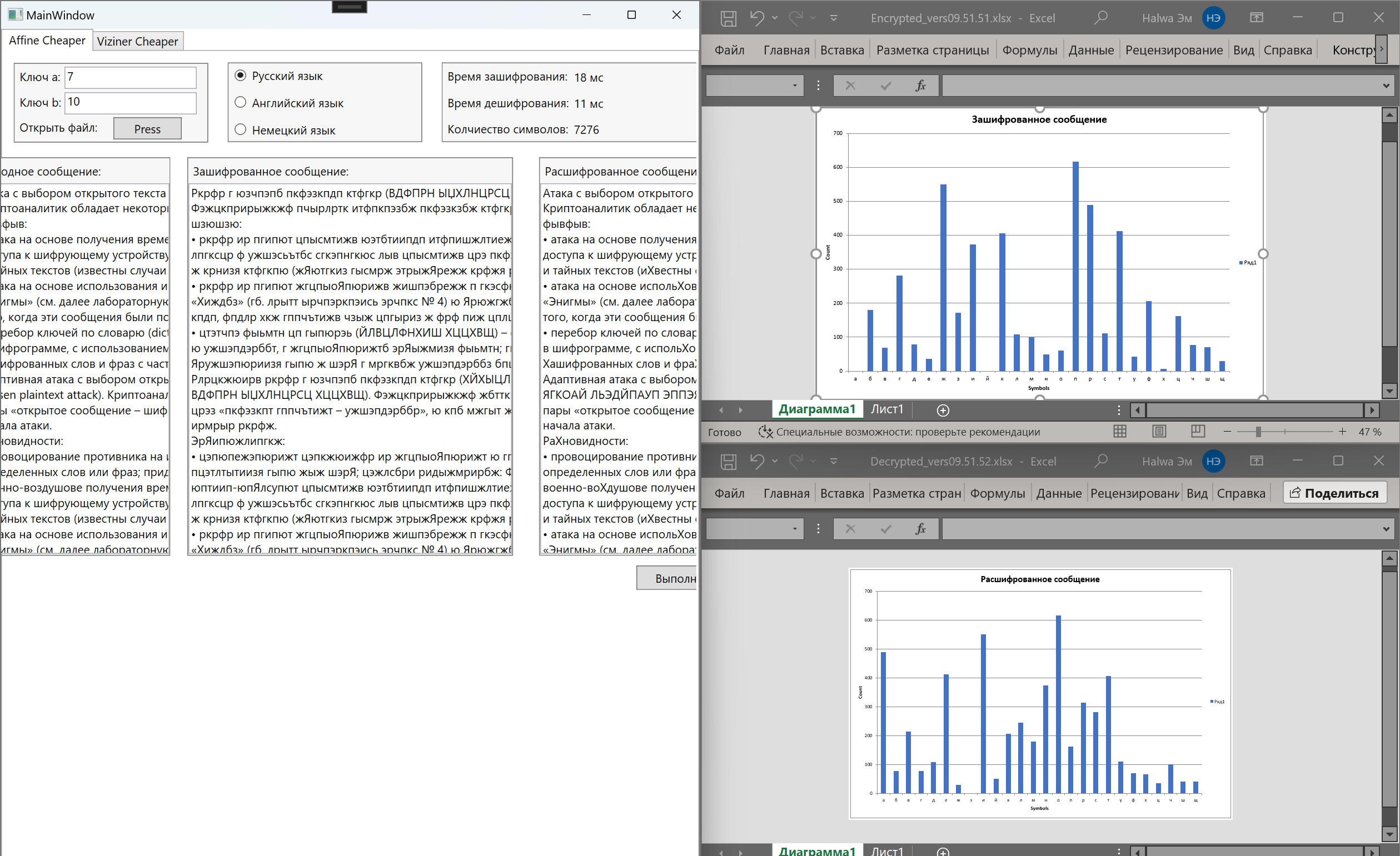


Рисунок 2.1 – Результат шифрования/дешифрования исходного сообщения методом аффиного преобразования

На данном скриншоте отображен процесс работы программы с шифрованием и дешифрованием исходного сообщения методом аффиного преобразования.

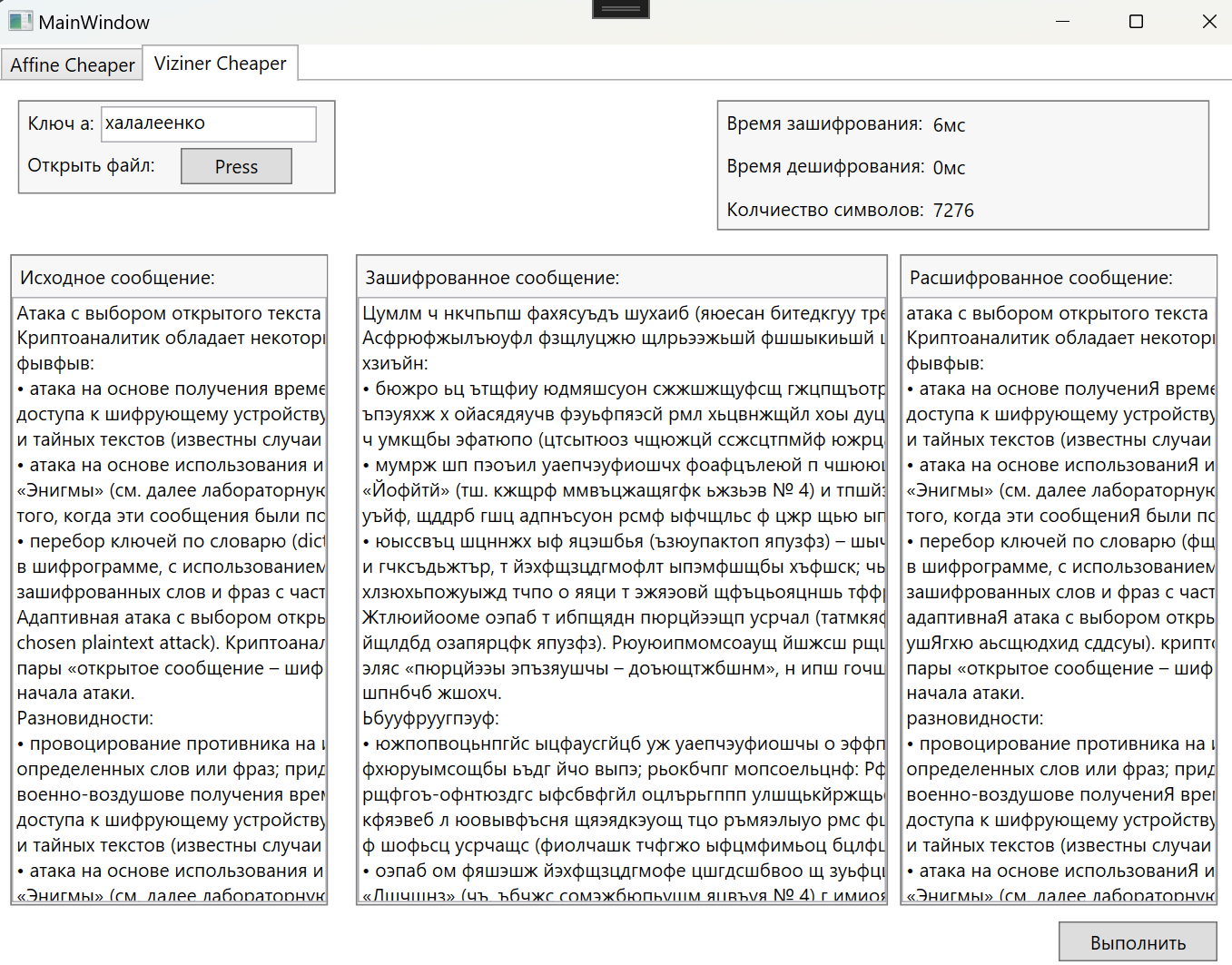


Рисунок 2.2 – результат шифрования/дешифрования исходного сообщения шифром Виженера

На данном скриншоте видна явная работа по преобразованию текста, используя собственную фамилию в качестве текста.

Основные функции программы представлены ниже.

public class affine\_cheaper\_ru

{

public static string AFFINE\_ENCRYPTED(string input\_msg, int key\_a, int key\_b)

{

string outputencr\_msg = "";

foreach (char c in input\_msg)

{

if (char.IsLetter(c))

{

int shiftedChar = (key\_a \* (char.ToLower(c) - 1071) + key\_b) % 33 + 1071;

outputencr\_msg += char.IsUpper(c) ? char.ToUpper((char)shiftedChar) : (char)shiftedChar;

}

else

{

outputencr\_msg += c;

}

}

return outputencr\_msg;

}

public static string AFFINE\_DECRYPTED(string inputencr\_msg, int key\_a, int key\_b)

{

string outputdecr\_msg = "";

int aInv = 0;

for (int i = 1; i < 33; i++)

{

if (key\_a \* i % 33 == 1)

{

aInv = i;

break;

}

}

foreach (char c in inputencr\_msg)

{

if (char.IsLetter(c))

{

int shiftedChar = aInv \* (char.ToLower(c) - 1071 - key\_b + 33) % 33 + 1071;

outputdecr\_msg += char.IsUpper(c) ? char.ToUpper((char)shiftedChar) : (char)shiftedChar;

}

else

{

outputdecr\_msg += c;

}

}

return outputdecr\_msg;

}

}

public static void ExcelPrint1(string filepath\_tosave, string filepath\_yomainmsg, string header)

{

Excel.Application excel = new Excel.Application();

excel.Visible = true;

Excel.Workbook workbook = excel.Workbooks.Add();

Excel.Worksheet worksheet = (Excel.Worksheet)workbook.ActiveSheet;

string fileContent = File.ReadAllText(filepath\_yomainmsg);

char[] chars = new char[] { 'а', 'б', 'в', 'г', 'д', 'е', 'ё', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ь', 'ы', 'ъ', 'э', 'ю', 'я' };

Dictionary<char, int> charCount = new Dictionary<char, int>();

foreach (char c in chars)

{

charCount.Add(c, 0);

}

foreach (char c in fileContent)

{

if (charCount.ContainsKey(c))

{

charCount[c]++;

}

}

Excel.Chart chart = (Excel.Chart)workbook.Charts.Add();

chart.ChartType = Excel.XlChartType.xlColumnClustered;

foreach (KeyValuePair<char, int> pair in charCount)

{

worksheet.Cells[1, pair.Key - 'а' + 1] = pair.Key.ToString();

worksheet.Cells[2, pair.Key - 'а' + 1] = pair.Value;

}

Excel.Range chartRange = worksheet.get\_Range("A1:Z2");

chart.SetSourceData(chartRange);

chart.HasTitle = true;

chart.ChartTitle.Text = header;

Excel.Axis xAxis = (Excel.Axis)chart.Axes(Excel.XlAxisType.xlCategory, Excel.XlAxisGroup.xlPrimary);

xAxis.HasTitle = true;

xAxis.AxisTitle.Text = "Symbols";

Excel.Axis yAxis = (Excel.Axis)chart.Axes(Excel.XlAxisType.xlValue, Excel.XlAxisGroup.xlPrimary);

yAxis.HasTitle = true;

yAxis.AxisTitle.Text = "Count";

string saveFilePath = filepath\_tosave;

workbook.SaveAs(saveFilePath);

workbook.Close();

excel.Quit();

}

}

Вывод

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