Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный технологический университет»

Факультет информационных технологий

Кафедра информатики и веб-дизайна

**Отчет к лабораторной работе**:

«Исследование криптографических шифров на основе подстановки (замены) символов»

Выполнила:

студентка 3 курса 9 группы

специальности ДЭИВИ

Змитревич Д. А.

Минск 2023

**Цель**: изучение и приобретение практических навыков разработки и использования приложений для реализации подстановоч-ных шифров.

**Задачи**:

1. Закрепить теоретические знания по алгебраическому описа- нию, алгоритмам реализации операций зашифрования/расшифро- вания и оценке криптостойкости подстановочных шифров.

2. Ознакомиться с особенностями реализации и свойствами различных подстановочных шифров на основе готового программ- ного средства (L\_LUX).

3. Разработать приложение для реализации указанных препода- вателем методов подстановочного зашифрования/расшифрования.

4. Выполнить исследование криптостойкости шифров на ос- нове статистических данных о частотах появления символов в ис- ходном и зашифрованном сообщениях.

5. Оценить скорость зашифрования/расшифрования реализо- ванных способов шифров.

6. Результаты выполнения лабораторной работы оформить в виде описания разработанного приложения, методики выполне- ния экспериментов с использованием приложения и результатов эксперимента.

1. Теоретические сведения

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

Если исходить из того, что используемые алфавиты являются конечными множествами, то в общем случае каждой букве ax алфавита AM (ax ∈ AM) для создания сообщения Мi (Мi ∈ M) соответствует буква ay или множество букв {АхС} для создания шифртекста Сi (Сi ∈ С). Важно, чтобы во втором случае любые два множества (например, {АхС}b и {АхС}n, b≠n, 1≤b, n, x, y≤N, N – мощность алфавита), используемые для замены разных букв открытого текста, не пересекались:

{АхС}b ∩ {АхС}n = 0.

Если в сообщении Мi содержится несколько букв ax, то каждая из них заменяется на символ ay либо на любой из символов {АхС}. За счет этого с помощью одного ключа можно сгенерировать различные Сi для одного и того же Мi. Так как множества {АхС}b и {АхС}n попарно не пересекаются, то по каждому символу Сi можно однозначно определить, какому множеству он принадлежит, и, следовательно, какую букву открытого сообщения Мi он заменяет. В силу этого открытое сообщение восстанавливается из зашифрованного однозначно.

Приведенные утверждения справедливы для следующих типов подстановочных шифров:

– Моноалфавитных (шифры однозначной замены или простые подстановочные). В данных шифрах операция замены производится раздельно над каждым одиночным символом сообщения Мi.

– Полиграммных. В таких шифрах одна подстановка соответствует сразу нескольким символам исходного текста.

– Омофонических (однозвучные шифры или шифры многозначной замены). Создавались с целью увеличить сложность частотного анализа шифртекстов путем маскировки реальных частот появления символов текста с помощью омофонии.

– Полиалфавитных. Состоят из нескольких шифров однозначной замены. Выбор варианта алфавита для зашифрования одного символа зависит от особенностей метода шифрования.

2. Практическое задание

Разработанное авторское приложение выполняет шифрование и расшифрование текстовых документов, созданных на основе алфавита английского языка, при помощи шифра Цезаря с ключевым словом и таблицы Трисемуса с ключевым словом.

*function* caesarCipher($text, $keyword, $encrypt=true) {

$start\_time = microtime(true);

$text = strtoupper($text); // переводим текст в верхний регистр

$keyword = strtoupper($keyword); // переводим ключевое слово в верхний регистр

$shifts = []; // массив сдвигов

for ($i = 0; $i < strlen($keyword); $i++) {

$shifts[] = ord($keyword[$i]) - 65;

}

$result = '';

$j = 0; // счетчик сдвигов

for ($i = 0; $i < strlen($text); $i++) {

$char = $text[$i];

if (ctype\_alpha($char)) { // проверяем, является ли символ буквой

$shift = $shifts[$j];

if (!$encrypt) {

$shift = 26 - $shift; // если расшифровываем текст, инвертируем сдвиг

}

$ascii = ord($char);

if ($ascii >= 65 && $ascii <= 90) { // шифруем только буквы

$ascii = (($ascii + $shift - 65) % 26) + 65; // шифруем символ

$result .= chr($ascii);

$j = ($j + 1) % strlen($keyword); // переходим к следующему сдвигу

}

} else {

$result .= $char; // если символ не является буквой, оставляем его без изменений

}

}

$end\_time = microtime(true);

$time = ($end\_time - $start\_time) \* 1000;

return ['text' => $result, 'time' => $time, 'sym' => characterFrequencyHistogram($result)];

}

Рисунок 1.1 – Реализация шифра Цезаря

В качестве параметров функция принимает текст, ключевое слово и флаг $encrypt для обозначения шифрования/расшифрования текста. Результатом данной функции является массив из полученной строки, затраченного времени и частоты появления символов в виде html-таблицы.

*function* encryptTrisemus($text, $keyword) {

$start\_time = microtime(true);

$alphabet = range('a', 'z'); // cоздаем массив с алфавитом

$keyArr = str\_split(str\_replace(' ', '', $keyword)); // разбиваем ключевое слово на массив символов

$keyLength = count($keyArr); // определяем длину ключа

$textArr = str\_split(str\_replace(' ', '', $text)); // разбиваем исходный текст на массив символов

$textLength = count($textArr); // определяем длину текста

$keyIndex = 0; // устанавливаем начальный индекс ключа

$result = '';

for ($i = 0; $i < $textLength; $i++) { // проходимся по всем символам текста

$letter = $textArr[$i]; // получаем текущий символ

if (!in\_array($letter, $alphabet)) { // если символ не находится в алфавите, то добавляем его в результат и переходим к следующему символу

$result .= $letter;

continue;

}

$shift = array\_search($keyArr[$keyIndex], $alphabet); // ищем сдвиг символа по ключу

$keyIndex = ($keyIndex + 1) % $keyLength; // обновляем индекс ключа

// ищем индекс зашифрованного символа в алфавите и добавляем его к результату

$index = (array\_search($letter, $alphabet) + $shift) % 26;

$result .= $alphabet[$index];

}

$end\_time = microtime(true);

$time = ($end\_time - $start\_time) \* 1000;

return ['text' => $result, 'time' => $time, 'sym' => characterFrequencyHistogram($result)];

}

Рисунок 1.2 – Реализация шифрования текста при помощи таблицы Трисемуса

Функция для расшифрования текста отличается лишь строчкой, где

$index = (array\_search($letter, $alphabet) - $shift + 26) % 26;

мы ищем индекс зашифрованного символа в алфавите и добавляем его к результату. Поскольку расшифрование – операция обратная шифрованию, то сдвиг мы считаем следующим образом.

Обе функции принимают текст и ключевое слово, а их результатом является массив из полученной строки, затраченного времени и частоты появления символов в виде html-таблицы.

Функция characterFrequencyHistogram() создает таблицу с частотами появления символов в строке.

*function* characterFrequencyHistogram($string) {

$char\_count = array();

$string\_length = strlen($string);

// подсчитываем количество вхождений каждого символа

for ($i = 0; $i < $string\_length; $i++) {

$char = $string[$i];

if (!isset($char\_count[$char])) {

$char\_count[$char] = 0;

}

$char\_count[$char]++;

}

// создаем HTML таблицу

$table\_html = "<table class='sym-table'><tbody><tr>";

foreach ($char\_count as $char => $count) {

$table\_html .= "<td>" . htmlspecialchars($char) . "</td>";

}

$table\_html .= "</tr><tr>";

foreach ($char\_count as $char => $count) {

$table\_html .= "<td>" . bcdiv($count / 26, 1, 2) . "</td>";

}

$table\_html .= "</tr></tbody></table>";

return $table\_html;

}

Рисунок 1.3 – Реализация подсчета частоты появления символов

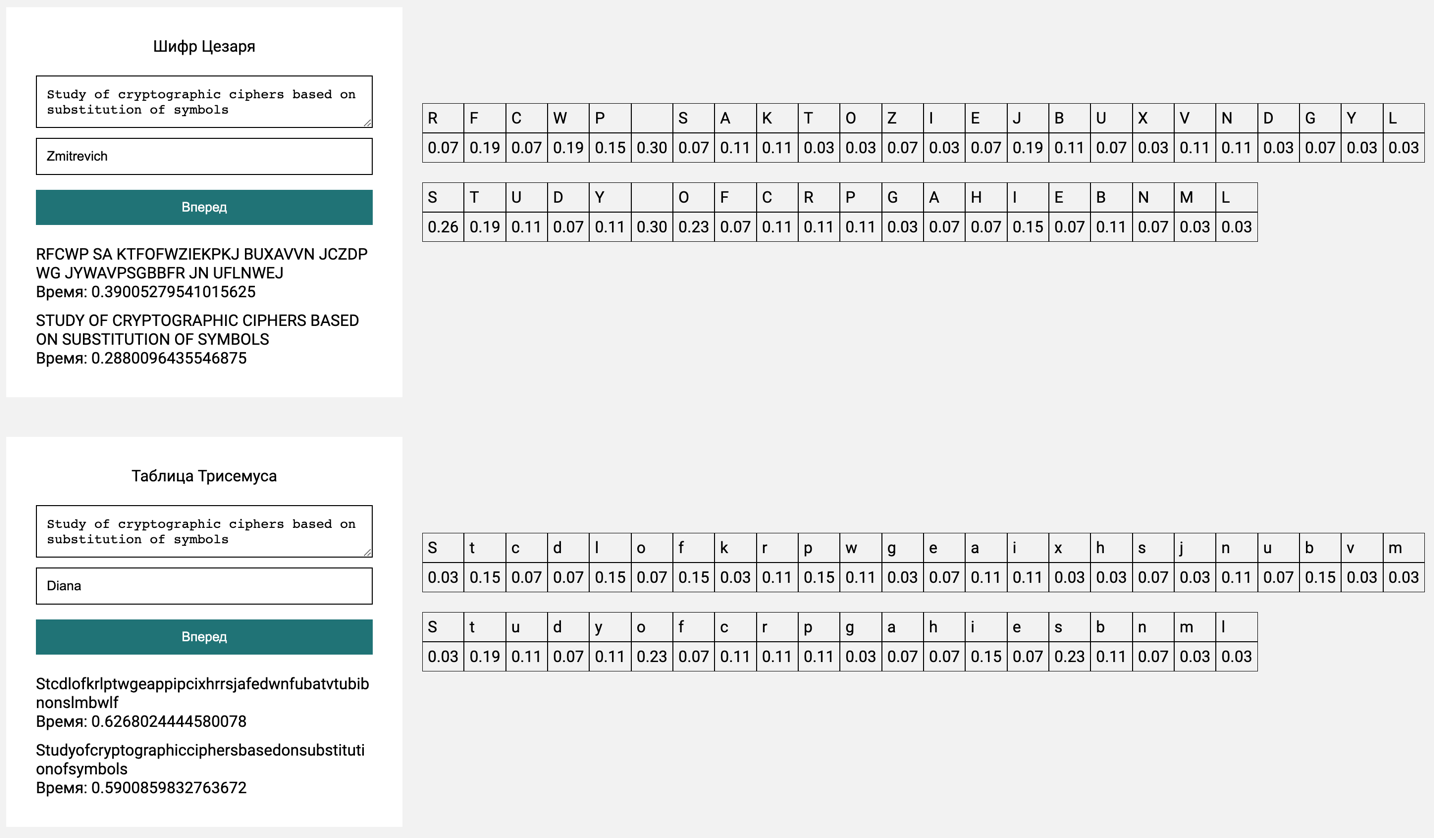


Рисунок 1.4 – Пример использования разработанных функций с текстом «Study of cryptographic ciphers based on substitution of symbols» и ключевыми словами из имени и фамилии

**Вывод**

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

**Контрольные вопросы**

1. В чем заключается основная идея криптографических преобразований на основе шифров замены?

Она состоит в замене одних символов, битов или блоков данных на другие символы, биты или блоки данных в соответствии с определенным алгоритмом или ключом. Это позволяет обеспечить конфиденциальность информации.

2. Привести классификационные признаки и дать сравнительную характеристику разновидностям подстановочных шифров.

Однобуквенные и многобуквенные шифры; открытые и закрытые шифры; шифры с использованием ключа и без; моноалфавитные и полиалфавитные шифры.

Однобуквенные просты в реализации, но уязвимы к атакам, в то время как многобуквенные более сложны в реализации, но более устойчивы к атакам, например частотному анализу.

Открытые шифры просты в реализации, так как алгоритм шифрования известен всем. Закрытые же шифры более сложные, но более надежны в защите информации.

Шифры с использованием ключей более безопасны, так как замена символов осуществляется в соответствии с ключом, который может быть сложным и длинным, но требуют обмена ключами между коммуникантами. Шифры без использования ключа требуют генерации и обмена одноразовыми ключами, что может быть сложно в реализации и управлении.

Моноалфавитные просты в реализации, поскольку каждому символу открытого текста соответствует одна и та же замена символа. Символу в полиалфавитном шифре может соответствовать несколько различных замен символов в зависимости от контекста или ключа, поэтому такой вид шифрования требует большего внимания, но более устойчив к атакам.

3. Сколько разновидностей шифров, подобных шифру Цезаря, можно составить для алфавитов русского и белорусского языков?

Оба алфавита состоят из 33 символов, поэтому количество разновидностей шифров будет схожее. К примеру, моноалфавитный шифр (1 разновидность, так как каждая буква заменяется на одну и ту же букву с фиксированным смещением), полиалфавитный шифр (более 33 разновидностей, так как каждая буква может заменяться на одну из 33 букв с разными смещениями, и количество возможных комбинаций смещений составляет более 33), круговой шифр (32 разновидности, так как каждая буква может быть смещена на любое из 32 других положений в алфавите, исключая саму себя), комбинированный шифр (более 33 разновидностей, так как можно комбинировать моноалфавитный и полиалфавитный шифры, а также использовать разные алгоритмы замены для разных букв алфавита).

Тем самым, общее количество разновидностей шифров, подобных шифру Цезаря, для алфавитов русского и белорусского языков составляет более 100.

4. Найти ключ шифра, с помощью которого получен шифртекст: «byajhvfwbjyyfzgjcktljdfntkmyjcnm».

Для нахождения ключа можно воспользоваться методом анализа частотности букв.

Буква "b" встречается 1 раз.

Буква "y" встречается 4 раза.

Буква "a" встречается 1 раз.

Буква "j" встречается 4 раза.

Буква "h" встречается 1 раз.

Буква "v" встречается 1 раз.

Буква "f" встречается 2 раза.

Буква "w" встречается 1 раз.

Буква "z" встречается 1 раз.

Буква "g" встречается 1 раз.

Буква "c" встречается 2 раза.

Буква "k" встречается 1 раз.

Буква "t" встречается 2 раза.

Буква "l" встречается 1 раз.

Буква "d" встречается 1 раз.

Буква "n" встречается 2 раза.

Буква "m" встречается 1 раз.

Наиболее часто встречающаяся буква в данном шифртексте – это буква "y", которая встречается 4 раза. Предположим, что буква "y" заменяет букву "e" в исходном тексте.

Предполагаемый ключ шифра Цезаря можно найти, используя смещение между позициями букв "y" и "e" в английском алфавите. Позиция буквы "y" в английском алфавите - это 24, а позиция буквы "e" – это 4.

Смещение ключа = 24-4 = 20.

Таким образом, предполагаемый ключ шифра Цезаря для данного шифртекста составляет 20.

5. Расшифровать (с демонстрацией каждого шага алгоритма) текст Сi=«qrscqcocqclc», зашифрованный аффинным шифром Цезаря при N=26, а=3, b=5.

Найдем обратное число "a" по модулю N. Для этого используем алгоритм нахождения обратного элемента по модулю.

Для данного случая, a=3 и N=26, обратное число для 3 по модулю 26 равно 9, так как 3\*9=27, и 27%26=1, то есть 3 и 9 являются обратными числами друг другу.

Теперь, используя найденное обратное число и параметр "b", расшифруем каждую букву шифртекста по формуле обратное\_число\*(Ci-b)%N.

Пример расчета для первой буквы – 9\*(16-5)%26=99%26=21.

Тем самым, расшифрованный текст – "VENISONBEEF".

6. Зашифровать и расшифровать свою фамилию (на основе ки- риллицы), используя аффинный шифр Цезаря.

C=(a\*P+b)modN – формула для шифрования, a=3, b=7, N=33, P – буква алфавита.

Применим формулу выше для каждого символа сообщения «Диана Змитревич», зашифрованное сообщение – «Пягтпяйгтпефпуця».

P=(a^(-1)\*(C-b))modN – формула для расшифрования, где a^(-1) это обратное a по модулю N. В данном случае, a=3, поэтому a^(-1)=11, так как 3\*11=33+1=N+1. После подсчета формулы для каждого символа зашифрованного сообщения, получим исходное сообщение.

7. Можно ли использовать в качестве ключевого в шифре Виженера слово, равное по длине открытому тексту? Обосновать ответ.

Да, можно. Если ключевое слово имеет такую же длину, как и открытый текст, то каждая буква открытого текста будет использоваться для генерации соответствующего символа ключа. Это означает, что каждая буква открытого текста будет использоваться только один раз в качестве ключа перед зашифровкой следующей буквы. Таким образом, в ключе не будет повторяющихся символов, и шифрование будет более безопасным.

8. По какому признаку можно определить, что текст зашифрован шифром Плейфера?

Текст, зашифрованный шифром Плейфера, может быть определен по признакам, таким как равномерное распределение символов (шифр использует таблицу или матрицу для замены букв на пары символов), частотный анализ биграмм (шифр заменяет буквы на пары символов, а, например, в русском языке можно заменить высокую частоту встречаемости некоторых пар), отсутствие пробелов (так как шифр оперирует парами символов) и сохранение длины текста.

9. Имеются ли предпочтения в выборе размеров таблицы Трисемуса для виртуального алфавита мощностью 40: 4×10; 10×4; 5×8; 8×5; 2×20; 20×2?

Из целей удобства и безопасности, размер таблицы Трисемуса следует выбирать такой, который позволит равномерно распределить символы по таблице, а именно с большим количеством строк и столбцов, так как она создает больше возможных комбинаций для замены символов и является удобной в шифровании.

На основе вышеуказанных факторов, таблицы размером 5x8 или 8x5 могут быть более предпочтительными для виртуального алфавита мощностью 40 символов.

10. Охарактеризовать основные виды атак на шифры.

Частотный анализ – проводится анализ частоты встречаемости символов в зашифрованном тексте, что может помочь в определении соответствующих символов открытого текста.

По выбору открытого текста – злоумышленник имеет возможность выбирать открытый текст и/или зашифрованный текст и наблюдать за соответствующими шифрованным и открытым текстами.

Брутфорс – злоумышленник перебирает все возможные ключи (или их комбинации) для расшифровки зашифрованного текста.

Социальная инженерия – злоумышленник использует манипуляции социального характера, чтобы получить доступ к ключу или открытому тексту. Например, это может включать подбор пароля с помощью личных данных, фишинговые атаки или взлом аккаунтов через социальные сети.

Криптоанализ – атака, основанная на математическом анализе шифров.

Атаки на физическую безопасность – таки, направленные на физические аспекты шифрования, такие как украденные или подделанные ключи, взлом физических устройств шифрования или снижение безопасности среды, в которой происходит шифрование.

11. Сравнить криптостойкость шифра Цезаря и шифра Виженера.

Шифр Цезаря основан на простой замене символов в тексте с фиксированным сдвигом (сдвигом на определенное количество позиций в алфавите). Он имеет всего 25 возможных ключей (сдвигов) для английского алфавита, что делает его криптостойкость низкой. Атаки перебором (brute-force) и анализом частотности могут эффективно раскрыть текст, зашифрованный шифром Цезаря. Криптостойкость этого шифра низка, и он не рекомендуется для использования в современных криптографических приложениях.

Шифр Вижинера использует ключевое слово в качестве ключа, и его особенностью является использование повторяющихся ключевых слов на разных позициях текста, что делает его более криптостойким по сравнению с шифром Цезаря. Он представляет собой комбинацию нескольких шифров Цезаря с различными сдвигами, что усложняет анализ частотности и атаки перебором. Однако, шифр Виженера все равно подвержен атакам, таким как известный открытый текст и атаки на длину ключа, если длина ключа недостаточно длинная. В целом, криптостойкость шифра Виженера также относительно низка, особенно при использовании коротких ключей.

12. Охарактеризовать основные методы взлома подстановочных шифров.

Подстановочные шифры, такие как шифры замены и перестановки, представляют собой классические методы шифрования, которые могут быть подвержены различным методам взлома. К основным методам взлома подстановочных шифров можно отнести: брутфорс, анализ частотности, известный открытый текст, использование словарей и баз данных.