**УКРАЇНИ МІНІСТЕРСТВО ОСВІТИ І НАУКИ**

**СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**

**ЦЕНТР ЗАОЧНОЇ, ДИСТАНЦІЙНОЇ ТА ВЕЧІРНЬОЇ ФОРМ НАВЧАННЯ**

**КАФЕДРА КОМП’ЮТЕРНИХ НАУК**

**ПРАКТИЧНА РОБОТА № 6**

з дисципліни "Захист інформації"

Тема: **"Шифри простої заміни. Шифр Цезаря"**

Варіант № 14

**Перевірив Коробченко О. В.**

**Виконала студентка гр. Індн – 31с Пархомчук А. К.**

Суми – 2017

**Мета**

Виробити вміння та навички шифрування інформації за допомогою шифру Цезаря, а також проводити криптоаналіз даного шифру.

**Завдання**

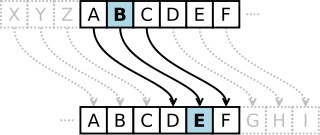
|  |  |  |  |
| --- | --- | --- | --- |
| **№ в-та** | А1 | Key | A2 |
| 14 | Цифри, спецсимвол (#) і заголовні літери російського алфавіту | 13 | Російський |

Розподіл частот появи символів англійського алфавіту

|  |  |
| --- | --- |
| Російський алфавіт | Частота |
| А | 0.065 |
| Б | 0.014 |
| В | 0.038 |
| Г | 0.013 |
| Д | 0.025 |
| Е | 0.072 |
| Ж | 0.007 |
| З | 0.016 |
| И | 0.062 |
| Й | 0.010 |
| К | 0.028 |
| Л | 0.035 |
| М | 0.026 |
| Н | 0.053 |
| О | 0.090 |
| П | 0.023 |
| Р | 0.040 |
| С | 0.045 |
| Т | 0.053 |
| У | 0.021 |
| Ф | 0.002 |
| Х | 0.009 |
| Ц | 0.004 |
| Ч | 0.012 |
| Ш | 0.006 |
| Щ | 0.003 |
| Ы | 0.016 |
| Ъ | 0.014 |
| Ь | 0,019 |
| Э | 0.003 |
| Ю | 0.006 |
| Я | 0.018 |
| └┘ | 0.175 |

**Теоретичний матеріал**

**Шифр Цезаря**

[](file:///C:\wiki\РР·РѕР±СЂР°Р¶РµРЅРёРµ:Caesar3.svg)

## Шифр Цезаря - один з найдавніших шифрів. При шифруванні кожен символ замінюється іншим, віддаленим від нього в алфавіті на фіксоване число позицій. Шифр Цезаря можна класифікувати як шифр підстановки, при більш вузької класифікації - шифр простої заміни.

## Математична модель

Якщо зіставити кожному символу алфавіту його порядковий номер (нумеруючи з 0), то зашифровування виражається формулою (1):

 (1)

а розшифрування виражається формулою:

 (2)

де  — номер в алфавіті символу відкритого тексту,

— номер в алфавіті символу шифрованого тексту,

— потужність алфавіту,

— ключ(),

, *length*-довжина відкритого/зашифрованого тексту

**Криптоаналіз шифра Цезаря**

**Алгоритм криптоаналізу:**

1. Визначити частоти символів алфавіту. Занести їх у масив *FiL*. (З Таблиці 2 або Таблиці 3)
2. Визначити частоти символів шифртекста. Занести їх у масив *FiМ*.
   * Обчислити скільки разів зустрічається той чи інший символ в шифртексту (*занести в масив freq*);
   * Визначити кількість символів в шифртексту (*l*);
   * Нормувати частоти символів обчислені в пункті а). Результат помістити в масив *FiМ* : 
3. Знайти таке значення *k* при якому сума однойменних різниць *d* була б мінімальною

* обчислити для всіх значень  суми різниць

, де , *n –*потужність алфавіту.

* + Знайти значення *k*, при якому сума різниць *d* буде мінімальною.
  + Розшифрувати шифртекст з ключем 

**Вихідні дані**

В ході виконання першого завдання (шифрування та дешифрування тексту) були використані наступні данні:

* Алфавіт потужністю 43 що містить наступні символи:

0123456789АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ#

* Ключ: 13
* Текст:

ТЕСТ123#ШИФРА#ЦЕЗАРЯ#ТАКЖЕ#ИЗВЕСТНЫЙ#КАК#ШИФР#СДВИГА4567

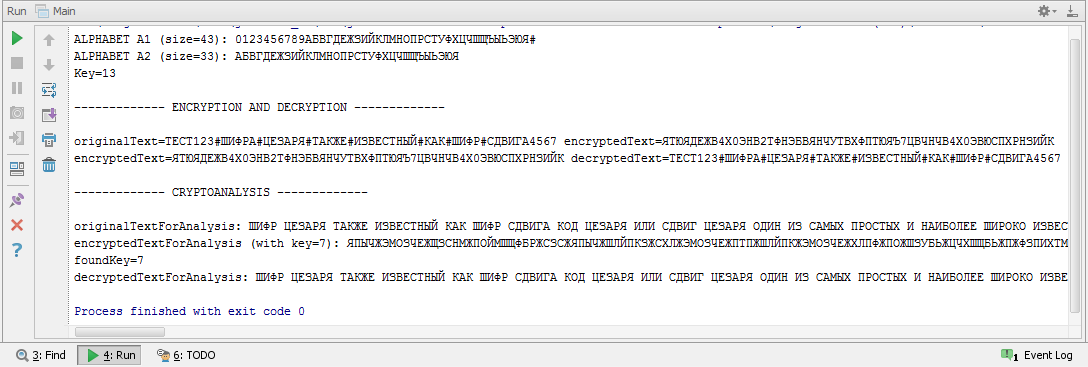
В ході виконання другого завдання (криптоаналiз зашифрованого тексту) були використані наступні данні:

* Алфавіт потужністю 33 що містить наступні символи: АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ└┘
* Ключ – 7
* Текст:

ШИФР ЦЕЗАРЯ ТАКЖЕ ИЗВЕСТНЫЙ КАК ШИФР СДВИГА КОД ЦЕЗАРЯ ИЛИ СДВИГ ЦЕЗАРЯ ОДИН ИЗ САМЫХ ПРОСТЫХ И НАИБОЛЕЕ ШИРОКО ИЗВЕСТНЫХ МЕТОДОВ ШИФРОВАНИЯ ШИФР ЦЕЗАРЯ ЭТО ВИД ШИФРА ПОДСТАНОВКИ В КОТОРОМ КАЖДЫЙ СИМВОЛ В ОТКРЫТОМ ТЕКСТЕ ЗАМЕНЯЕТСЯ СИМВОЛОМ НАХОДЯЩИМСЯ НА НЕКОТОРОМ ПОСТОЯННОМ ЧИСЛЕ ПОЗИЦИЙ ЛЕВЕЕ ИЛИ ПРАВЕЕ НЕГО В АЛФАВИТЕ НАПРИМЕР В ШИФРЕ СО СДВИГОМ ВПРАВО НА А БЫЛА БЫ ЗАМЕНЕНА НА Г Б СТАНЕТ Д И ТАК ДАЛЕЕ ШИФР НАЗВАН В ЧЕСТЬ РИМСКОГО ИМПЕРАТОРА ГАЯ ЮЛИЯ ЦЕЗАРЯ ИСПОЛЬЗОВАВШЕГО ЕГО ДЛЯ СЕКРЕТНОЙ ПЕРЕПИСКИ СО СВОИМИ ГЕНЕРАЛАМИ ШАГ ШИФРОВАНИЯ ВЫПОЛНЯЕМЫЙ ШИФРОМ ЦЕЗАРЯ ЧАСТО ВКЛЮЧАЕТСЯ КАК ЧАСТЬ БОЛЕЕ СЛОЖНЫХ СХЕМ ТАКИХ КАК ШИФР ВИЖЕНЕРА И ВСЕ ЕЩЕ ИМЕЕТ СОВРЕМЕННОЕ ПРИЛОЖЕНИЕ В СИСТЕМЕ КАК И ВСЕ МОНОАЛФАВИТНЫЕ ШИФРЫ ШИФР ЦЕЗАРЯ ЛЕГКО ВЗЛАМЫВАЕТСЯ И НЕ ИМЕЕТ ПОЧТИ НИКАКОГО ПРИМЕНЕНИЯ НА ПРАКТИКЕ

* Таблиця співвідношення символів та їх частот у тексті (приведена у главі «Завдання»)

**Скриншоти роботи програми**



**Вихідний текст програми**

*CaesarCipher.java*

package com.infoprotect.task6;

import java.io.\*;

public class CaesarCipher {

public static final String ALPHABET\_A1 = "0123456789АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ#";

public static final String ALPHABET\_A2 = "АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ ";

private final double[] freqA2 = new double[]{

0.065, 0.014, 0.038, 0.013, 0.025, 0.072, 0.007, 0.016, 0.062, 0.010, 0.028, 0.035, 0.026, 0.053, 0.090,

0.023, 0.040, 0.045, 0.053, 0.021, 0.002, 0.009, 0.004, 0.012, 0.006, 0.003, 0.016, 0.014, 0.019, 0.003,

0.006, 0.018, 0.175

};

public String encrypt(String text, int key, final String alphabet) {

return encryptDecrypt(text, key, alphabet, new CryptographicFunction() {

@Override

public int execute(int currentPosition, int key, int alphabetSize) {

return (currentPosition + key) % alphabet.length();

}

});

}

public String decrypt(String text, int key, final String alphabet) {

return encryptDecrypt(text, key, alphabet, new CryptographicFunction() {

@Override

public int execute(int currentPosition, int key, int alphabetSize) {

return (currentPosition - key + alphabet.length()) % alphabet.length();

}

});

}

public String encryptDecrypt(String text, int key, final String alphabet, CryptographicFunction function) {

StringBuilder decodedText = new StringBuilder();

for (int i = 0; i < text.length(); i++) {

int currentPosition = alphabet.indexOf(text.charAt(i));

if (currentPosition == -1) {

throw new IllegalArgumentException("Char '" + text.charAt(i) + "' at position " + i

+ " is not present in current alphabet [" + alphabet + "]!");

}

int newPosition = function.execute(currentPosition, key, alphabet.length());

decodedText.append(alphabet.charAt(newPosition));

}

return decodedText.toString();

}

public int cryptoAnalysis(String encryptedText) {

double[] freq = new double[ALPHABET\_A2.length()];

int count = 0;

for (int i = 0; i < encryptedText.length(); i++) {

int currentSymbolPosition = ALPHABET\_A2.indexOf(encryptedText.charAt(i));

if (currentSymbolPosition == -1) {

throw new IllegalArgumentException("Char '" + encryptedText.charAt(i)

+ "' at position " + i + " is not present in current alphabet [" + ALPHABET\_A2 + "]!");

}

freq[currentSymbolPosition]++;

count++;

}

for (int i = 0; i < freq.length; i++) {

freq[i] /= count;

}

double[] sumDifferences = new double[ALPHABET\_A2.length()];

for (int i = 0; i < ALPHABET\_A2.length(); i++) {

sumDifferences[i] = 0;

for (int k = 0; k < ALPHABET\_A2.length(); k++) {

int index = (k - i + ALPHABET\_A2.length()) % ALPHABET\_A2.length();

sumDifferences[i] += Math.abs(freqA2[k] - freq[index]);

}

}

int foundKey = 0;

double minD = sumDifferences[0];

for (int i = 0; i < ALPHABET\_A2.length(); i++)

if (sumDifferences[i] < minD) {

minD = sumDifferences[i];

foundKey = ALPHABET\_A2.length() - i;

}

return foundKey;

}

public String readFile(String fileName) {

try {

BufferedReader br = new BufferedReader(new FileReader(fileName));

String buffer = br.readLine();

br.close();

return buffer;

} catch(IOException ex) {

System.out.println(ex.toString());

return "";

}

}

public void writeFile(String fileName, String message){

try {

BufferedWriter bw = new BufferedWriter(new FileWriter(fileName));

bw.write(message);

bw.close();

} catch(IOException ex){

System.out.println(ex.toString());

}

}

}

*Main.java*

package com.infoprotect.task6;

public class Main {

public static final String FILE\_INPUT\_TEXT = "res/input.txt";

public static final String FILE\_ENCRYPTED\_TEXT = "res/encrypted.txt";

public static final String FILE\_DECRYPTED\_TEXt = "res/decrypted.txt";

public static final String FILE\_CRYPTOANALYSIS\_INPUT\_TEXT = "res/cryptoanalysis\_input\_text.txt";

public static final String FILE\_CRYPTOANALYSIS\_ENCRYPTED\_TEXT = "res/cryptoanalysis\_encrypted\_text.txt";

public static final String FILE\_CRYPTOANALYSIS\_DECRYPTED\_TEXT = "res/cryptoanalysis\_decrypted\_text.txt";

public static void main(String[] args) {

CaesarCipher caesarCipher = new CaesarCipher();

System.out.println("ALPHABET A1 (size=" + CaesarCipher.ALPHABET\_A1.length() + "): " + CaesarCipher.ALPHABET\_A1);

System.out.println("ALPHABET A2 (size=" + CaesarCipher.ALPHABET\_A2.length() + "): " + CaesarCipher.ALPHABET\_A2);

int key = 13;

System.out.println("Key=" + key);

String testPhrase = "ТЕСТ123#ШИФРА#ЦЕЗАРЯ#ТАКЖЕ#ИЗВЕСТНЫЙ#КАК#ШИФР#СДВИГА4567";

// String originalText = caesarCipher.readFile(FILE\_INPUT\_TEXT);

String originalText = testPhrase;

System.out.println("\n------------- ENCRYPTION AND DECRYPTION -------------");

String encryptedText = caesarCipher.encrypt(originalText, key, CaesarCipher.ALPHABET\_A1);

System.out.println("originalText=" + originalText + " encryptedText=" + encryptedText);

caesarCipher.writeFile(FILE\_ENCRYPTED\_TEXT, encryptedText);

String decryptedText = caesarCipher.decrypt(encryptedText, key, CaesarCipher.ALPHABET\_A1);

System.out.println("encryptedText=" + encryptedText + " decryptedText=" + decryptedText);

caesarCipher.writeFile(FILE\_DECRYPTED\_TEXt, encryptedText);

System.out.println("\n------------- CRYPTOANALYSIS -------------");

String originalTextForAnalysis = caesarCipher.readFile(FILE\_CRYPTOANALYSIS\_INPUT\_TEXT);

System.out.println("originalTextForAnalysis: " + originalTextForAnalysis);

String encryptedTextForAnalysis = caesarCipher.encrypt(originalTextForAnalysis, 7, CaesarCipher.ALPHABET\_A2);

System.out.println("encryptedTextForAnalysis (with key=7): " + encryptedTextForAnalysis);

caesarCipher.writeFile(FILE\_CRYPTOANALYSIS\_ENCRYPTED\_TEXT, encryptedTextForAnalysis);

int foundKey = caesarCipher.cryptoAnalysis(encryptedTextForAnalysis);

System.out.println("foundKey=" + foundKey);

String decryptedTextForAnalysis = caesarCipher.decrypt(encryptedTextForAnalysis, foundKey, CaesarCipher.ALPHABET\_A2);

System.out.println("decryptedTextForAnalysis: " + decryptedTextForAnalysis);

caesarCipher.writeFile(FILE\_CRYPTOANALYSIS\_DECRYPTED\_TEXT, decryptedTextForAnalysis);

}

}

*CryptographicFunction.java*

package com.infoprotect.task6;

public interface CryptographicFunction {

public int execute(int currentPosition, int key, int alphabetSize);

}

**Результати роботи програми**

ALPHABET A1 (size=43): 0123456789АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ#

ALPHABET A2 (size=33): АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ

Key=13

------------- ENCRYPTION AND DECRYPTION -------------

originalText=ТЕСТ123#ШИФРА#ЦЕЗАРЯ#ТАКЖЕ#ИЗВЕСТНЫЙ#КАК#ШИФР#СДВИГА4567 encryptedText=ЯТЮЯДЕЖВ4Х0ЭНВ2ТФНЭБВЯНЧУТВХФПТЮЯЪ7ЦВЧНЧВ4Х0ЭВЮСПХРНЗИЙК

encryptedText=ЯТЮЯДЕЖВ4Х0ЭНВ2ТФНЭБВЯНЧУТВХФПТЮЯЪ7ЦВЧНЧВ4Х0ЭВЮСПХРНЗИЙК decryptedText=ТЕСТ123#ШИФРА#ЦЕЗАРЯ#ТАКЖЕ#ИЗВЕСТНЫЙ#КАК#ШИФР#СДВИГА4567

------------- CRYPTOANALYSIS -------------

originalTextForAnalysis: ШИФР ЦЕЗАРЯ ТАКЖЕ ИЗВЕСТНЫЙ КАК ШИФР СДВИГА КОД ЦЕЗАРЯ ИЛИ СДВИГ ЦЕЗАРЯ ОДИН ИЗ САМЫХ ПРОСТЫХ И НАИБОЛЕЕ ШИРОКО ИЗВЕСТНЫХ МЕТОДОВ ШИФРОВАНИЯ ШИФР ЦЕЗАРЯ ЭТО ВИД ШИФРА ПОДСТАНОВКИ В КОТОРОМ КАЖДЫЙ СИМВОЛ В ОТКРЫТОМ ТЕКСТЕ ЗАМЕНЯЕТСЯ СИМВОЛОМ НАХОДЯЩИМСЯ НА НЕКОТОРОМ ПОСТОЯННОМ ЧИСЛЕ ПОЗИЦИЙ ЛЕВЕЕ ИЛИ ПРАВЕЕ НЕГО В АЛФАВИТЕ НАПРИМЕР В ШИФРЕ СО СДВИГОМ ВПРАВО НА А БЫЛА БЫ ЗАМЕНЕНА НА Г Б СТАНЕТ Д И ТАК ДАЛЕЕ ШИФР НАЗВАН В ЧЕСТЬ РИМСКОГО ИМПЕРАТОРА ГАЯ ЮЛИЯ ЦЕЗАРЯ ИСПОЛЬЗОВАВШЕГО ЕГО ДЛЯ СЕКРЕТНОЙ ПЕРЕПИСКИ СО СВОИМИ ГЕНЕРАЛАМИ ШАГ ШИФРОВАНИЯ ВЫПОЛНЯЕМЫЙ ШИФРОМ ЦЕЗАРЯ ЧАСТО ВКЛЮЧАЕТСЯ КАК ЧАСТЬ БОЛЕЕ СЛОЖНЫХ СХЕМ ТАКИХ КАК ШИФР ВИЖЕНЕРА И ВСЕ ЕЩЕ ИМЕЕТ СОВРЕМЕННОЕ ПРИЛОЖЕНИЕ В СИСТЕМЕ КАК И ВСЕ МОНОАЛФАВИТНЫЕ ШИФРЫ ШИФР ЦЕЗАРЯ ЛЕГКО ВЗЛАМЫВАЕТСЯ И НЕ ИМЕЕТ ПОЧТИ НИКАКОГО ПРИМЕНЕНИЯ НА ПРАКТИКЕ

encryptedTextForAnalysis (with key=7): ЯПЫЧЖЭМОЗЧЕЖЩЗСНМЖПОЙМШЩФБРЖСЗСЖЯПЫЧЖШЛЙПКЗЖСХЛЖЭМОЗЧЕЖПТПЖШЛЙПКЖЭМОЗЧЕЖХЛПФЖПОЖШЗУБЬЖЦЧХШЩБЬЖПЖФЗПИХТММЖЯПЧХСХЖПОЙМШЩФБЬЖУМЩХЛХЙЖЯПЫЧХЙЗФПЕЖЯПЫЧЖЭМОЗЧЕЖЖГЩХЖЙПЛЖЯПЫЧЗЖЦХЛШЩЗФХЙСПЖЙЖСХЩХЧХУЖСЗНЛБРЖШПУЙХТЖЙЖХЩСЧБЩХУЖЩМСШЩМЖОЗУМФЕМЩШЕЖШПУЙХТХУЖФЗЬХЛЕ ПУШЕЖФЗЖФМСХЩХЧХУЖЦХШЩХЕФФХУЖЮПШТМЖЦХОПЭПРЖТМЙММЖПТПЖЦЧЗЙММЖФМКХЖЙЖЗТЫЗЙПЩМЖФЗЦЧПУМЧЖЙЖЯПЫЧМЖШХЖШЛЙПКХУЖЙЦЧЗЙХЖФЗЖЗЖИБТЗЖИБЖОЗУМФМФЗЖФЗЖКЖИЖШЩЗФМЩЖЛЖПЖЩЗСЖЛЗТММЖЯПЫЧЖФЗОЙЗФЖЙЖЮМШЩВЖЧПУШСХКХЖПУЦМЧЗЩХЧЗЖКЗЕЖДТПЕЖЭМОЗЧЕЖПШЦХТВОХЙЗЙЯМКХЖМКХЖЛТЕЖШМСЧМЩФХРЖЦМЧМЦПШСПЖШХЖШЙХПУПЖКМФМЧЗТЗУПЖЯЗКЖЯПЫЧХЙЗФПЕЖЙБЦХТФЕМУБРЖЯПЫЧХУЖЭМОЗЧЕЖЮЗШЩХЖЙСТДЮЗМЩШЕЖСЗСЖЮЗШЩВЖИХТММЖШТХНФБЬЖШЬМУЖЩЗСПЬЖСЗСЖЯПЫЧЖЙПНМФМЧЗЖПЖЙШМЖМ МЖПУММЩЖШХЙЧМУМФФХМЖЦЧПТХНМФПМЖЙЖШПШЩМУМЖСЗСЖПЖЙШМЖУХФХЗТЫЗЙПЩФБМЖЯПЫЧБЖЯПЫЧЖЭМОЗЧЕЖТМКСХЖЙОТЗУБЙЗМЩШЕЖПЖФМЖПУММЩЖЦХЮЩПЖФПСЗСХКХЖЦЧПУМФМФПЕЖФЗЖЦЧЗСЩПСМ

foundKey=7

decryptedTextForAnalysis: ШИФР ЦЕЗАРЯ ТАКЖЕ ИЗВЕСТНЫЙ КАК ШИФР СДВИГА КОД ЦЕЗАРЯ ИЛИ СДВИГ ЦЕЗАРЯ ОДИН ИЗ САМЫХ ПРОСТЫХ И НАИБОЛЕЕ ШИРОКО ИЗВЕСТНЫХ МЕТОДОВ ШИФРОВАНИЯ ШИФР ЦЕЗАРЯ ЭТО ВИД ШИФРА ПОДСТАНОВКИ В КОТОРОМ КАЖДЫЙ СИМВОЛ В ОТКРЫТОМ ТЕКСТЕ ЗАМЕНЯЕТСЯ СИМВОЛОМ НАХОДЯЩИМСЯ НА НЕКОТОРОМ ПОСТОЯННОМ ЧИСЛЕ ПОЗИЦИЙ ЛЕВЕЕ ИЛИ ПРАВЕЕ НЕГО В АЛФАВИТЕ НАПРИМЕР В ШИФРЕ СО СДВИГОМ ВПРАВО НА А БЫЛА БЫ ЗАМЕНЕНА НА Г Б СТАНЕТ Д И ТАК ДАЛЕЕ ШИФР НАЗВАН В ЧЕСТЬ РИМСКОГО ИМПЕРАТОРА ГАЯ ЮЛИЯ ЦЕЗАРЯ ИСПОЛЬЗОВАВШЕГО ЕГО ДЛЯ СЕКРЕТНОЙ ПЕРЕПИСКИ СО СВОИМИ ГЕНЕРАЛАМИ ШАГ ШИФРОВАНИЯ ВЫПОЛНЯЕМЫЙ ШИФРОМ ЦЕЗАРЯ ЧАСТО ВКЛЮЧАЕТСЯ КАК ЧАСТЬ БОЛЕЕ СЛОЖНЫХ СХЕМ ТАКИХ КАК ШИФР ВИЖЕНЕРА И ВСЕ ЕЩЕ ИМЕЕТ СОВРЕМЕННОЕ ПРИЛОЖЕНИЕ В СИСТЕМЕ КАК И ВСЕ МОНОАЛФАВИТНЫЕ ШИФРЫ ШИФР ЦЕЗАРЯ ЛЕГКО ВЗЛАМЫВАЕТСЯ И НЕ ИМЕЕТ ПОЧТИ НИКАКОГО ПРИМЕНЕНИЯ НА ПРАКТИКЕ

**Висновки**

В ході виконання роботи я здобув навички та вміння шифрування інформації за допомогою шифру Цезаря, а також навчився проводити криптографічний аналіз даного шифру. Під час виконання роботи мною була розроблена програмна реалізація на мові Ruby шифру Цезаря, а також програмна реалізація алгоритму його статистичного криптографічного аналізу.