Отчёт по лабораторной работе №1

Шифры простой замены

Артур Арменович Давтян

Содержание

Список таблиц

Список иллюстраций

1 Цель работы

Ознакомиться с шифрами простой замены и обучиться их программной реализации.

2 Задание

- Реализовать шифр Цезаря с произвольным ключом k;
- Реализовать шифр Атбаша.

3 Теоретическое введение

При подготовке использовалась методичка со страницы курса в ТУИС.[1]

Шифр Цезаря является примером метода подстановки. Дальнейшее усовершенствование оригинального сдвига символа на три позиции в шифре Цезаря состоит в использовании арифметики по модулю двадцать шесть для ключа шифрования, который больше двадцати шести.

$$E_n(x) = (x + n)mod26$$

,

где x - значение открытого текста, n - номер сдвига.

Шифр Цезаря со сдвигом 1 (рис. 3.1):

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
B C D E F G H I J K L M N O P Q R S T U V W X Y Z A

Рис. 3.1: Шифр Цезаря со сдвигом 1

Шифр Атбаш – шифр простой замены. Шифрование происходит заменой первой буквы алфавита на последнюю, второй на предпоследнюю, и так далее. По сути, это шифр сдвига на всю длину. Шифр Атбаш для русского алфавита (рис. 3.2):

а б в г д е ж з и й к л м н о п р с т у ф х ц ч ш щ ъ ы ь э ю я ␣
␣ я ю э ь ы ъ щ ш ч ц х ф у т с р п о н м л к й и з ж е д г в б а

Рис. 3.2: Шифр Атбаш

4 Выполнение лабораторной работы

Работа была выполнена на языке программирования Python.

Сначала реализуем шифр Цезаря (рис. 4.1):

```
def caesar(plaintext, shift):
    alph = 'abcdefghijklmnopqrstuvwxyz'
    if shift > 26 or shift < -26:
        sh2 = shift%26
        shalph = alph[sh2:] + alph[:sh2]
    else:
        shalph = alph[shift:] + alph[:shift]
    table = str.maketrans(alph, shalph)
    print(plaintext.translate(table))
```

Рис. 4.1: Программная реализация шифра Цезаря

В переменную alph помещаем латинский алфавит. В переменную shalph задаём алфавит, который начинается с буквы, соответствующей числу сдвига, и прибавляем начало алфавита до этой буквы. Так как в таком случае при числе сдвига больше 26 и меньше -26 программа работать не будет, задаём условие, что в этом случае за число сдвига берется остаток от деления числа на 26. После этого создаём таблицу, в которой каждой букве исходного алфавита сопоставляется буква нового алфавита. В конце выводим зашифрованный текст с помощью метода str.translate, в который передаём таблицу.

Реализация шифра Атбаш (рис. 4.2):

```
def atbash(plaintext):
    alph = 'abcdefghijklmnopqrstuvwxyz '
    shalph = alph[::-1]
    table = str.maketrans(alph,shalph)
    print(plaintext.translate(table))
```

Рис. 4.2: Программная реализация шифра Атбаш

В переменную alph помещаем алфавит, но в этом случае, опираясь на [1] добавляем к нему пробел. В переменную shalph помещаем тот же алфавит, но перевёрнутый с помощью функционала python. Создаём таблицу и выводим зашифрованный текст.

Ввод исходного текста и числа сдвига (рис. 4.3):

```
while True:
    plainText = input("What is your plaintext? ")

    if any(char.isnumeric() for char in plainText):
        continue
    else:
        break

while True:
    try:
        shift = int(input("What is your shift? "))
    except ValueError:
        continue
    break

print('\033[1m' + '\nCaesars:' + '\033[0m')

caesar(plainText, shift)

print('\033[1m' + '\nAtbash:' + '\033[0m')

atbash(plainText)
```

Рис. 4.3: Программная реализация шифрования

Для ввода исходного текста вводим правило, что не может быть чисел, в противном случае просьба ввести текст будет выведена заново. Для ввода числа сдвига вводим правило, что не может быть букв, в противном случае просьба

ввести число сдвига будет выведена заново. Если всё введено правильно, то будет выведен текст, зашифрованный с помощью шифра Цезаря и Атбаш (рис. 4.4):

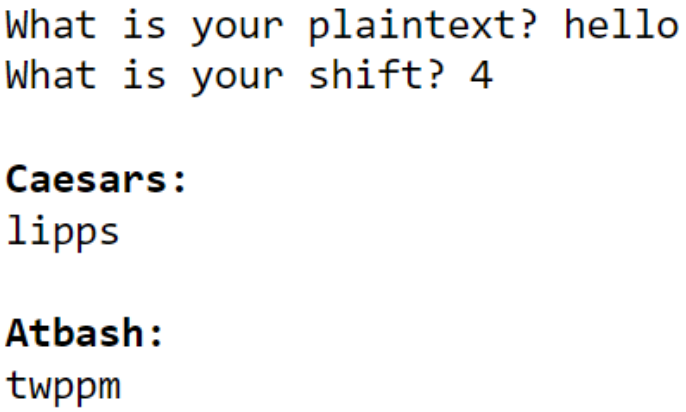

Рис. 4.4: Вывод программы

5 Выводы

Ознакомился с шифрами простой замены и обучился их программной реализации.

Список литературы

1. ТУИС: Математические основы защиты информации и информационной безопасности (02.04.02) [Электронный ресурс]. РУДН, 2022. URL: https://esystem.rudn.ru/course/view.php?id=2084.