Лабораторная работа №1

Шифры простой замены

Ли Тимофей Александрович, НФИмд-02-22

Содержание

Список таблиц

Список иллюстраций

Цель работы

Цель данной работы — изучить и программно реализовать шифры простой замены.

Задание

Заданием является:

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

Теоретическое введение

Шифр простой замены представляет собой замену каждой буквы в исходном слове на определенное число, которому соответствует данная буква. В основе функционирования шифров простой замены лежит следующий принцип: для получения шифртекста отдельные символы или группы символов исходного алфавита заменяются символами или группами символов шифроалфавита.

Шифр Цезаря

Шифр Цезаря является моноалфавитной подстановкой, т.е. каждой букве открытого текста ставится соответствие одна буква шифротекста.

Математическая процедура шифрования описывается как

$$T_m = \left\{T^j\right\}, j = 0, 1, \cdots, m-1,$$

$$T^j(a) = (a + j) \mod m,$$

где m - длина алфавитаа, j - произвольный ключ (величина сдвига от изначальной позиции буквы), a - текущая позиция буквы в алфавите.

Для латинского алфавита длина составляет 26 символов, а формулу можно привести к виду:

$$T^k(i) = (i + k) \mod 26,$$

где i, k соответствуют a, j, а $m = 26$.

Сам же Цезарь обычно использовал подстановку T^3.

Шифр Атбаш

Шифр Атбаш является сдвигом на всю длину алфавита. Правило шифрования состоит в замене i-й буквы алфавита буквой с номером $n - i + 1$, где n — число букв в алфавите.

Выполнение лабораторной работы

Для реализации шифров мы будем использовать Python, так как его синтаксис позволяет быстро реализовать необходимые нам алгоритмы.

Реализация шифра Цезаря с произвольным ключом k

Шифр Цезаря реализуем в виде функции ceasar следующего вида:

```
In [14]: def ceasar(letter: chr, key: int, alphabet: list):
             def ceasar(letter:chr, key:int):
                 return alphabet.index(letter)+key
             if letter.lower() not in alphabet:
                 return letter
             t_letter=alphabet[ceasar(letter.lower(),key)%len(alphabet)]
             if letter.isupper():
                 t_letter=t_letter.upper()
             return t_letter
```

Рис. 0.1: функция шифра Цезаря

На вход она принимает переменные letter (один символ), key (произвольный ключ), alphabet (алфавит в виде списка).

В ходе обработке мы работаем с индексами элементов массива-строки, предварительно проверяя, является ли символ частью передаваемого алфавита. Если да, то мы вызываем вложенную функцию для расчета сдвига и выполняем к ней операцию деления с остатком (исходя из формулы в теоретическом введении).

В конце мы проверяем, является ли буква заглавной, и, после ситуативной обработки, возвращаем зашифрованную букву.

Реализация шифра Атбаша

Шифр Атбаш реализуем в виде функции atbash следующего вида:

```
def atbash(letter:chr, alphabet:list):
    if letter.lower() not in alphabet:
        return letter
    t_letter=alphabet[len(alphabet)-alphabet.index(letter.lower())-1]
    if letter.isupper():
        t_letter=t_letter.upper()
    return t_letter
```

Рис. 0.2: функция шифра Атбаш

На вход она принимает те же переменные, что и функция Шифра Цезаря, исключая произвольный ключ.

Шифруется символ засчет вычитания из длины алфавита индекс символа, над которым производится шифрование.

Возвращается также зашифрованный символ.

Тестирование

Для запуска шифрования мы создали следующие функции:

```
In [16]: def run_ceasar(message: str, key:int, alphabet:list):
             a=list(map(lambda letter: ceasar(letter,key,alphabet),message))
             return "".join(a)
```

```
In [20]: def run_atbash(message: str, alphabet:list):
             a=list(map(lambda letter: atbash(letter,alphabet),message))
             return "".join(a)
```

Рис. 0.3: функции запуска шифрования

Также создали два вида английского алфавита – без пробела, и с ним – для запуска шифров Цезаря и Атбаш соответственно.

Запустив наш программный код, получим следующие результаты:

```
In [17]: run_ceasar("lessss go first try",3,eng)
Out[17]: 'ohvvvv jr iluvw wub'

In [18]: run_ceasar("ohvvvv jr iluvw wub",23,eng)
Out[18]: 'lessss go first try'
```

Рис. 0.4: результат Цезаря

```
In [26]: run_atbash("abcdefghijklmnopqrstuvwxyz ",eng2)
Out[26]: ' zyxwvutsrqponmlkjihgfedcba'
```

Рис. 0.5: результат Атбаш

Видим, что шифрование проведено корректно.

Выводы

В рамках выполненной лабораторной работы мы изучили и реализовали следующие шифры простой замены: шифр Цезаря (с произвольным ключом k) и шифр Атбаш.