

# **Лабораторная работа 1**

**Шифры простой замены**

Греков Максим Сергеевич

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Описание реализации</b>	<b>5</b>
<b>3</b>	<b>Реализация</b>	<b>6</b>
3.1	Шифр Цезаря с произвольным ключом k . . . . .	6
3.2	Шифр Атбаш . . . . .	7
<b>4</b>	<b>Вывод</b>	<b>8</b>

# List of Figures

2.1	Код генерации алфавитов . . . . .	5
3.1	Код Шифра Цезаря . . . . .	7
3.2	Код Шифра Атбаш . . . . .	7

# 1 Цель работы

Ознакомиться с шифрами простой замены.

Реализовать шифр Цезаря с произвольным ключом  $k$ .

Реализовать шифр Атбаш.

## 2 Описание реализации

Для реализации алгоритмов использовались средства языка Python.

Был предложен функционал генерации алфавитов и их добавления. (рис. 2.1)

Были реализованы как шифраторы, так и дешифраторы рассматриваемых алгоритмов.

```
1 def define_alphabet(c, alphabets):
2     for alphabet in alphabets:
3         if c in alphabet:
4             return alphabet, alphabet.index(c)
5     return None, None
6
7
8 def get_alphabets():
9     en_low = [chr(c) for c in range(ord('a'), ord('z')+1)]
10    en_up = [c.upper() for c in en_low]
11    ru_low = [chr(c) for c in range(ord('a'), ord('e')+1)] + \
12            ['ё'] + [chr(c) for c in range(ord('ж'), ord('я')+1)]
13    ru_up = [c.upper() for c in ru_low]
14    return en_low, en_up, ru_low, ru_up
15
```

Figure 2.1: Код генерации алфавитов

## 3 Реализация

### 3.1 Шифр Цезаря с произвольным ключом $k$

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. (рис. 3.1)

Если сопоставить каждому символу алфавита его порядковый номер (нумеруя с 0), то шифрование и дешифрование можно выразить формулами модульной арифметики:

$$y = (x + k) \bmod(n)$$

$$x = (y - k) \bmod(n)$$

где  $x$  — символ открытого текста,  $y$  — символ шифрованного текста,  $n$  — мощность алфавита, а  $k$  — ключ.

```

29 def caesar_encode(string, alphabets, key):
30     res = ''
31     for c in string:
32         alphabet, pos = define_alphabet(c, alphabets)
33         res += c if alphabet is None else alphabet[(pos+key) % len(alphabet)]
34     return res
35
36
37 def caesar_decode(string, alphabets, key):
38     return caesar_encode(string, alphabets, -key)
39

```

Figure 3.1: Код Шифра Цезаря

## 3.2 Шифр Атбаш

Шифр Атбаш — простой шифр подстановки для алфавитного письма. Правило шифрования состоит в замене  $i$ -й буквы алфавита буквой с номером  $n - i + 1$ , где  $n$  — число букв в алфавите. (рис. 3.2)

```

17 def atbash_encode(string, alphabets):
18     res = ''
19     for c in string:
20         alphabet, pos = define_alphabet(c, alphabets)
21         res += c if alphabet is None else alphabet[len(alphabet)-pos-1]
22     return res
23
24
25 def atbash_decode(string, alphabets):
26     return atbash_encode(string, alphabets)

```

Figure 3.2: Код Шифра Атбаш

## 4 Вывод

Ознакомились с шифрами простой замены.

Реализовали шифр Цезаря с произвольным ключом  $k$ .

Реализовали шифр Атбаш.