

# **Отчёт по лабораторной работе №1**

**Шифр простой замены**

Александр Дмитриев

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Теоретические сведения</b>	<b>5</b>
2.1	Шифр Цезаря . . . . .	5
2.2	Шифр Атбаш . . . . .	6
<b>3</b>	<b>Выполнение работы</b>	<b>7</b>
3.1	Реализация шифра Цезаря на языке Python . . . . .	7
3.2	Реализация шифра Атбаш на языке Python . . . . .	8
3.3	Контрольный пример . . . . .	9
<b>4</b>	<b>Выводы</b>	<b>10</b>
	<b>Список литературы</b>	<b>11</b>

# List of Figures

3.1	шифр Цезаря . . . . .	9
3.2	шифр Атбаш . . . . .	9

# **1 Цель работы**

Изучение алгоритмов шифрования Цезаря и Атбаш

## 2 Теоретические сведения

### 2.1 Шифр Цезаря

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 3 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 ещё имеет современное приложение в системе ROT13. Как и все моноалфавитные шифры, шифр Цезаря легко взламывается и не имеет практически никакого применения на практике.

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

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

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

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

С точки зрения математики шифр Цезаря является частным случаем аффинного шифра.

## 2.2 Шифр Атбаш

Атбаш — простой шифр подстановки, изначально придуманный для иврита. Правило шифрования состоит в замене  $i$ -й буквы алфавита буквой с номером  $n - i + 1$ , где  $n$  — число букв в алфавите.

## 3 Выполнение работы

### 3.1 Реализация шифра Цезаря на языке Python

Блок шифрования

```
def cesar(text, step):  
    liters = 'абвгдеёжзийклмнопрстуфхцчщъыьэюяабвгдеёжзийклмнопрстуфхцчщъыьэюя'  
    res = ''  
    for i in text:  
        index = liters.find(i)  
        new_index = index + step  
        if i in liters:  
            res += liters[new_index]  
        else:  
            res += i  
    return res
```

Блок дешифровки

```
def cesar_dec(text, step):  
    liters = 'абвгдеёжзийклмнопрстуфхцчщъыьэюяабвгдеёжзийклмнопрстуфхцчщъыьэюя'  
    res = ''  
    for i in text:  
        index = liters.find(i)  
        new_index = index - step
```

```

        if i in liters:
            res += liters[new_index]
        else:
            res += i
    return res

```

## 3.2 Реализация шифра Атбаш на языке Python

Блок шифрования

```

def atbash(text):
    liters = 'абвгдеёжзийклмнопрстуфхцчшщъыьэюя'
    liters_r = [x for x in liters]
    liters_r.reverse()
    res = ''
    for i in text:
        for j,l in enumerate(liters):
            if i==l:
                res += liters_r[j]
    return res

```

Блок дешифровки

```

def atbash_dec(text):
    liters = 'абвгдеёжзийклмнопрстуфхцчшщъыьэюя'
    liters_r = [x for x in liters]
    liters_r.reverse()
    res = ''
    for i in text:
        for j,l in enumerate(liters_r):
            if i==l:

```



```

        res += liters[j]

    return res

```

### 3.3 Контрольный пример

```

In [18]: 1 def cesar(text, step, p=0):
          2     liters = 'абвгдеёжзийклмнопрстуфхцщъыьэяабвгдеёжзийклмнопрстуфхцщъыьэя'
          3     res = ''
          4     if p==1:
          5         for i in text:
          6             index = liters.find(i)
          7             new_index = index + step
          8             if i in liters:
          9                 res += liters[new_index]
          10            else:
          11                res += i
          12        if p == 0:
          13            for i in text:
          14                index = liters.find(i)
          15                new_index = index - step
          16                if i in liters:
          17                    res += liters[new_index]
          18                else:
          19                    res += i
          20
          21        return res

In [19]: 1 t = 'физмат сила'

In [20]: 1 print(f'{t} - {cesar(t, 3, 1)} - {cesar(cesar(t, 3, 1), 3, 0)}')
          физмат сила - члкггх флог - физмат сила

```

Figure 3.1: шифр Цезаря

```

In [23]: 1 def atbash(text, p=0):
          2     liters = 'абвгдеёжзийклмнопрстуфхцщъыьэя'
          3     liters_r = [i for i in liters]
          4     liters_r.reverse()
          5     res = ''
          6     if p==1:
          7         for i in text:
          8             for j,l in enumerate(liters):
          9                 if i==l:
          10                    res += liters_r[j]
          11        if p==0:
          12            for i in text:
          13                for j,l in enumerate(liters_r):
          14                    if i==l:
          15                        res += liters[j]
          16
          17        return res

In [24]: 1 print(f'{t} - {atbash(t, 1)} - {atbash(atbash(t, 1), 0)}')
          физмат сила - лчшу наочф - физмат сила

```

Figure 3.2: шифр Атбаш

## 4 Выводы

Изучили алгоритмы шифрования Цезаря и Атбаш.

# Список литературы

1. Шифр Цезаря
2. Шифр Атбаш