

# **Лабораторная работа №1**

**Математические основы защиты информации и информационной безопасности**

Николаев Дмитрий Иванович, НПМмд-02-24

# Содержание

|   |                        |    |
|---|------------------------|----|
| 1 | Цель работы            | 5  |
| 2 | Теоретическое введение | 6  |
| 3 | Ход работы             | 7  |
| 4 | Выводы                 | 10 |
|   | Список литературы      | 11 |

## Список иллюстраций

|     |   |   |
|-----|---|---|
| 3.1 | Результат работы шифра Цезаря . . . . . | 8 |
| 3.2 | Результат работы шифра Атбаша . . . . . | 9 |

## Список таблиц

# 1 Цель работы

Изучить шифры простой замены, а именно шифр Цезаря и шифр Атбаша. Научиться программной реализации шифра Цезаря с произвольным ключом и шифра Атбаша на языке программирования Julia.

## 2 Теоретическое введение

Шифры простой замены представляют собой одни из первых и самых элементарных методов шифрования, применяемых для защиты данных. Эти шифры функционируют на основе замены символов оригинального текста (известного как открытый текст) другими символами, что делает информацию недоступной для несанкционированных пользователей. Классическими примерами такого рода шифрования являются шифр Цезаря и шифр Атбаша.

Шифр Цезаря, названный в честь Юлия Цезаря, который использовал его для защиты военных сообщений, — это метод, при котором каждый символ открытого текста сдвигается на определённое количество позиций в алфавите. Например, при смещении на три буквы 'А' превращается в 'D', 'В' — в 'Е' и так далее. Этот способ прост в применении, но подвержен частотному анализу, так как сохраняет особенности языка.

Шифр Атбаша является более элементарным вариантом шифра замены, в котором буквы меняются на их “отражения” в алфавите. Например, 'А' становится 'Z', 'В' — 'Y' и так далее.

### 3 Ход работы

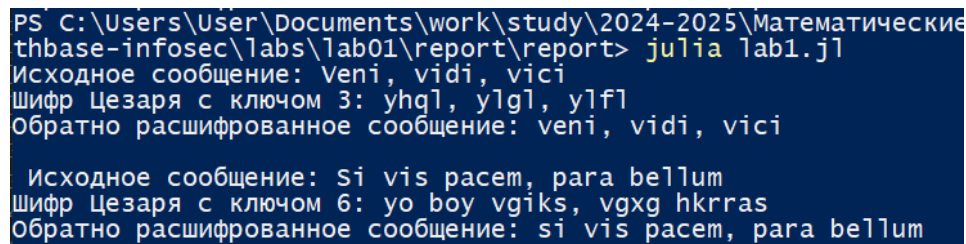
Следуем указаниям из [1]. Реализуем для начала функцию шифра Цезаря с произвольным ключом.

```
alphabet = 'a':'z' # Алфавит
function Caesar_Cipher(Input_Message::String, key::Int)::String
    # Зашифрованное сообщение
    Cipher = String[]
    for char in lowercase(Input_Message)
        if char in alphabet
            position = findfirst(x -> x == char, alphabet)
            # осуществляем сдвиг согласно ключу key
            new_position = mod1(position + key, length(alphabet))
            push!(Cipher, string(alphabet[new_position]))
        else
            # Символ не из алфавита остаётся неизменным
            push!(Cipher, string(char))
        end
    end
    return join(Cipher)
end
```

Дальше проверяем его работу на двух сообщениях ([3.1]):

```
Test_Message = "Veni, vidi, vici"
Test_Key = 3
```

```
println("Исходное сообщение: ", Test_Message)
println("Шифр Цезаря с ключом $(Test_Key): ",
        Caesar_Cipher(Test_Message, Test_Key))
println("Обратно расшифрованное сообщение: ",
        Caesar_Cipher(Caesar_Cipher(Test_Message, Test_Key), length(alphabet) - Test_Key))
Message_1 = "Si vis pacem, para bellum"
Key_1 = 6
println("\n Исходное сообщение: ", Message_1)
println("Шифр Цезаря с ключом $(Key_1): ",
        Caesar_Cipher(Message_1, Key_1))
println("Обратно расшифрованное сообщение: ",
        Caesar_Cipher(Caesar_Cipher(Message_1, Key_1), length(alphabet) - Key_1))
```



```
PS C:\Users\User\Documents\work\study\2024-2025\Математические thbase-infosec\labs\lab01\report\report> julia lab1.jl
Исходное сообщение: Veni, vidi, vici
Шифр Цезаря с ключом 3: yhq1, ylg1, ylf1
Обратно расшифрованное сообщение: veni, vidi, vici

Исходное сообщение: Si vis pacem, para bellum
Шифр Цезаря с ключом 6: yo boy vgiks, vgxg hkrras
Обратно расшифрованное сообщение: si vis pacem, para bellum
```

Рис. 3.1: Результат работы шифра Цезаря

Реализуем далее функцию шифра Атбаша.

```
function Atbash_Cipher(Input_Message::String)::String
    # Зашифрованное сообщение
    Cipher = String[]
    Reversed_alphabet = reverse(alphabet)
    for char in lowercase(Input_Message)
        if char in alphabet
            position = findfirst(x -> x == char, alphabet)
            # осуществляем сдвиг на весь алфавит
```



```

        push!(Cipher, string(Reversed_alphabet[position]))
    else
        # Символ не из алфавита остаётся неизменным
        push!(Cipher, string(char))
    end
end
end
return join(Cipher)
end

```

Дальше проверяем его работу на тех же двух сообщениях ([3.2]):

```

println("\n\nИсходное сообщение: ", Test_Message)
println("Шифр Атбаша: ", Atbash_Cipher(Test_Message))
println("Обратно расшифрованное сообщение: ",
        Atbash_Cipher(Atbash_Cipher(Test_Message)))
Message_1 = "Si vis pacem, para bellum"
Key_1 = 6
println("\n Исходное сообщение: ", Message_1)
println("Шифр Атбаша: ", Atbash_Cipher(Message_1))
println("Обратно расшифрованное сообщение: ",
        Atbash_Cipher(Atbash_Cipher(Message_1)))

```

```

Исходное сообщение: Veni, vidi, vici
Шифр Атбаша: evmr, erwr, erxr
Обратно расшифрованное сообщение: veni, vidi, vici

Исходное сообщение: Si vis pacem, para bellum
Шифр Атбаша: hr erh kzxvn, kziz yvoofn
Обратно расшифрованное сообщение: si vis pacem, para bellum

```

Рис. 3.2: Результат работы шифра Атбаша

## 4 Выводы

В ходе выполнения лабораторной работы я изучил шифры простой замены, а именно шифры Цезаря и Атбаша, а также написал программную реализацию шифра Цезаря с произвольным ключом и шифра Атбаша на языке программирования Julia.

## Список литературы

1. Лабораторная работа № 1. Шифры простой замены [Электронный ресурс]. Саратовский государственный университет имени Н.Г. Чернышевского.