# Лабораторная работа 3

Шифрование гаммированием конечной гаммы

Климин Никита Денисович
Российский университет дружбы народов

# Содержание

# 1 Цель работы

Реализовать алгоритм шифрования гаммированием конечной гаммы на языке julia.
Проверить работу программы на примере расшифровки и зашифровки текста.

# 2 Задание

Реализовать алгоритм шифрования гаммированием конечной гаммы на языке julia.
Проверить работу программы на примере расшифровки и зашифровки текста.

# 3 Теоретическое введение

**Гаммирование** — это метод симметричного шифрования текста, при котором каждая буква исходного текста преобразуется с помощью ключа (гаммы) по модулю длины алфавита.

- Простейший вариант — **одноразовое использование ключа**, где длина ключа равна длине текста.
- Шифрование выполняется операцией сложения по модулю 2 (XOR) для двоичных данных или по модулю N для букв (N — длина алфавита).
- Дешифрование осуществляется повторным применением той же операции.

  Формулы для шифрования и расшифровки букв русского алфавита (N = 33): C_i = (P_i + K_i) mod 33 # шифрование P_i = (C_i - K_i) mod 33 # расшифровка
- P_i — порядковый номер буквы исходного текста
- K_i — порядковый номер буквы ключа
- C_i — порядковый номер зашифрованной буквы

# 4 Выполнение лабораторной работы

Программа была написана на Julia.

```
    kn = [num.(collect(k))...] # преобразуем ключ в масив номеров букв
    kn = repeat(kn, ceil(Int, length(tn)/length(kn)))[1:length(tn)] # если ключ короче текста, то повторяем
    op = decrypt ? (-) : (+) # определяем операцию шифрования или расшифрования

    r = [ # основный цикл шифрования и расшифрования
        ch == ' ' ? ' ' : lettr(mod1(op(num(ch) ,y), length(alphabet))) # если символ пробел, то оставляем
        for (ch, y) in zip(tn, kn) # иначе берём номер буквы и применяем сложение/вычитание по модулю и пер
    ]
    return join(r)
end

println("введите текст")
text = readline()
println("введите гамму")
key = readline()
println("выберите действие, 1- шифрование, 2 - дешифрование")
choice = readline()

if choice == "1"
    result = gamma(text, key)
    println("зашифрованный текст $result")
elseif choice == "2"
    result = gamma(text, key, decrypt=true)
    println("расшифрованный текст $result")
else
    println("введите 1 или 2")
end
```

# Пример работы программы в терминале

```
ndklimin@ndklimin:~/labib/lab3$ julia lab3.jl
введите текст
приказ
введите гамму
гамма
выберите действие, 1- шифрование, 2 - дешифрование
1
зашифрованный текст УСХЧБЛ
ndklimin@ndklimin:~/labib/lab3$ julia lab3.jl
введите текст
усхчбл
введите гамму
гамма
выберите действие, 1- шифрование, 2 - дешифрование
2
расшифрованный текст ПРИКАЗ
```

Рисунок 1: Пример работы программы

# 5 Выводы

Реализован алгоритм шифрования гаммированием конечной гаммой. Программы коректно шифрует и расшифровывает текст на примере слова ПРИКАЗ

# Список литературы

Speaker notes