# 

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

### Факультет физико-математических и естественных наук

### Кафедра прикладной информатики и теории вероятностей

# ОТЧЕТ

# ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

### дисциплина: Математические основы защиты информации и информационной безопасности

### Студент: Пиняева Анна Андреевна

### Группа: НПИмд-01-24

### МОСКВА

### 2025

# 

# Теоретическое введение

**Шифрование гаммирванием**

Гаммирование - процедура наложения при помощи некоторой функции F на исходный текст гаммы шифра, т.е. псевдослучайной последовательности (ПСП) с выходов генератора G. Псевдослучайная последовательность по своим статистическим свойствам неотличима от случайной последовательности , но является детерминированной, т.е. известен алгоритм ее формирования. Чаще Обычно в качестве функции F берется операция поразрядного сложения по модулю два или по модулю N (N - число букв алфавита открытого текста).

В задании лабораторной предлагается рассмотреть альтернативный случай шифрования гаммированием – шифрованием гаммированием с конечной гаммой.

Исходный код написан на языке Julia [@doc-julia]. Код функции, осуществляющей шифрование гаммированием с конечной гаммой, представлен ниже.

# Цель работы

Изучение и реализация шифрования гаммированием на языке Julia.

# Ход работы

1. Функция получения номера буквы

alphabet=collect("АБВГДЕЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ")  
  
function letter(c::Char)  
 pos = findfirst(isequal(c), alphabet)  
 if pos === nothing  
 error("Буквы в алфавите нет")  
 end  
 return pos  
end

Что происходит в функции: - Поиск позиции буквы в алфавите - Проверка, найдена ли буква в алфавите (если не найдена - вывод сообщения об ошибке) - Возврат порядкового номера буквы

1. Функция получения буквы по номеру

function num(n::Int)  
 if n ==0  
 n=32  
 elseif n<1 || n>32  
 error("Номер не в диапазоне")  
 end  
 return alphabet[n]  
end

Что происходит в функции: - Обработка случая, когда результат операции mod дает 0 - Обработка случая 0 как последней буквы - Проверка корректности номера (если неверный - выаод об ошибке) - Возврат буквы по указаному номеру

1. Функция шифрования

function gamma(message::String, key::String, mod\_val::Int=32)  
 p\_nums = [letter(c) for c in message]  
 k\_nums=[letter(c) for c in key]  
 k\_len=length(k\_nums)  
  
 c\_nums=Vector{Int}(undef, length(p\_nums))  
 for i in 1:length(p\_nums)  
 ki=k\_nums[(i-1) % k\_len+1]  
 c\_nums[i]=(p\_nums[i] + ki) % mod\_val  
 if c\_nums[i] ==0  
 c\_nums[i]=mod\_val  
 end  
 end  
  
 return join([num(n) for n in c\_nums])  
end

Что происходит в функции: - Преобразование сообщения в числовую последовательность - Сохранение длины ключа для цикличесого повторения - Создание массива для числового представления шифротекста - Цикл по всем символам сообщения - Вычисление индекса ключа с циклическим повторением - Шифрование (сложение по модулю) - Корректирвка нулевого результата - Преобразрвание числовой последовательности в текст

1. Функция дешифрования

function gamma\_de(ciphertext::String, key::String, mod\_val::Int=32)  
 c\_nums=[letter(c) for c in ciphertext]  
 k\_nums=[letter(c) for c in key]  
 k\_len=length(k\_nums)  
  
 p\_nums=Vector{Int}(undef, length(c\_nums))  
 for i in 1:length(c\_nums)  
 ki=k\_nums[(i-1) % k\_len+1]  
 p\_nums[i]=(c\_nums[i] - ki) % mod\_val  
 if p\_nums[i] ==0  
 p\_nums[i]=mod\_val  
 end  
 end  
 return join([num(n) for n in p\_nums])  
end

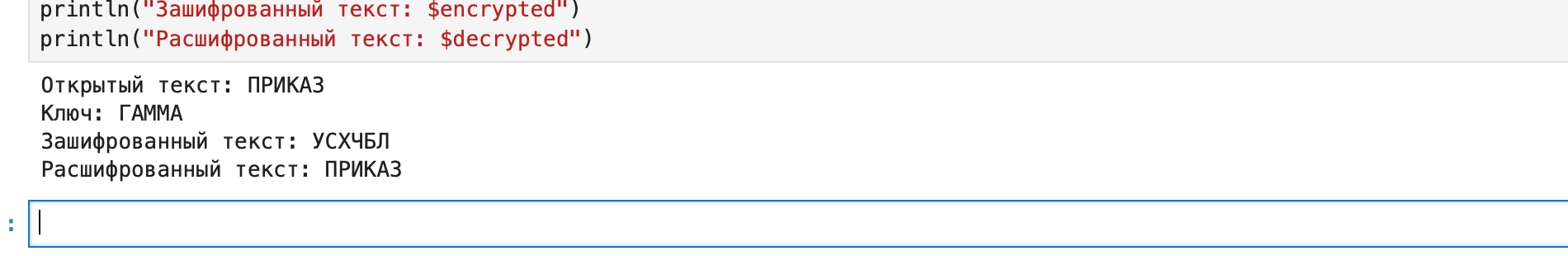
Что происходит в функции: - Преобразование шифротекста в числовую последовательность - Сохранение длины ключа для циклического повторения - Создание массива для числового представления открытого текста - Цикл по всем символам шифротекста - Вычисление индекса ключа с циклическим повторением - Дешифрование (вычитание по модулю) - Корректировка нулевого результата - Преобразование числовой последовательности в текст

1. Вывод результатов

message="ПРИКАЗ"  
key="ГАММА"  
encrypted=gamma(message,key)  
decrypted=gamma\_de(encrypted, key)  
  
println("открытый текст: $message")  
println("ключ: $key")  
println("зашифрованный текст: $encrypted")  
println("расшифрованный текст: $decrypted")

Результат тестирования представлен на рис.1

*Рис. 1 Тестирование шифрования гаммированием:*



Вывод: В ходе данной работы мной был изучен и реализован метод шифрования гаммированием. Написан программный код на языке Julia и протестирован.