Лабораторная работа №3

Математические основы защиты информации и информационной безопасности

Данилова Анастасия Сергеевна

Содержание

Цель работы

Изучить шифрование гаммированием, реализовать алгоритм шифрования гаммированием конечной гаммой на языке программирования Julia.

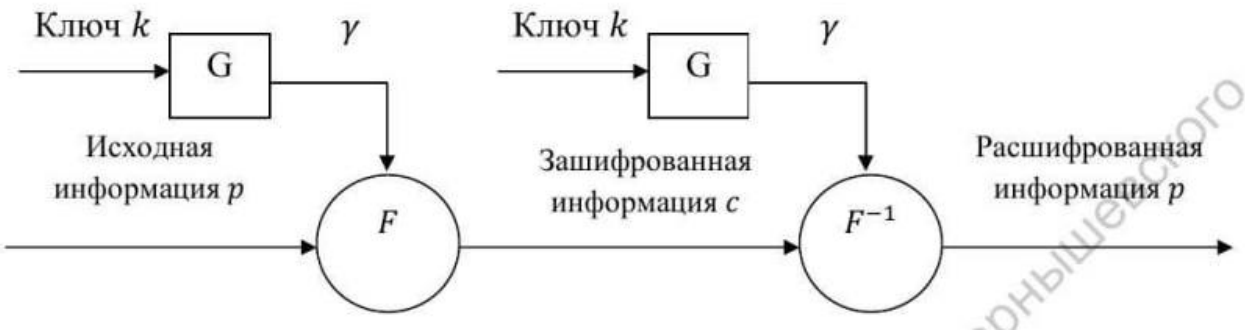
Задание

- Изучить теоретическую часть о шифровании гаммированием
- Реализовать алгоритм шифрования гаммированием конечной гаммой

Теоретическое введение

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).

Ниже представлена схема, которая называется гаммированием



Гаммирование

Стойкость шифров, основанных на процедуре гаммирования, зависит от характеристик гаммы - длины и равномерности распределения вероятностей появления знаков гаммы. При использовании генератора ПСП получаем бесконечную гамму. Однако, возможен режим шифрования конечной гаммы. В роли конечной гаммы может выступать фраза. Как и ранее, используется алфавитный порядок букв, т.е. буква «а» имеет порядковый номер 1, «б» - 2 и т.д.

Выполнение лабораторной работы

Итак, реализуем схему шифрования и дешифрования текста, используя ключ-гамму:

Функция encrypt принимает на вход исходный текст и гамму-ключ и возвращает зашифрованный текст:

- Она преобразует исходный текст в массив числовых кодов
- Затем шифрует каждый символ текста, используя соответствующий символ из гаммы-ключа
- Шифрование производится путем сложения числовых кодов символа текста и символа гаммы по модулю 32
- Полученные символы объединяются в единую строку

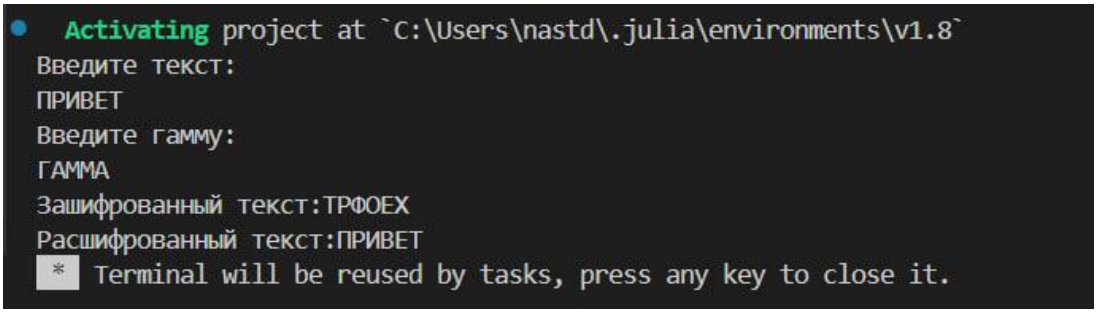
Функция decrypt принимает на вход зашифрованный текст и гамму-ключ и возвращает расшифрованный текст:
- Она проходит по каждому символу зашифрованного текста
- Для каждого символа производится вычитание числового кода соответствующего символа гаммы-ключа по модулю 32
- Полученные символы объединяются в одну строку

```
using Dates

function encrypt(text, gamma)
    shtext = ""
    key = iterate(gamma)
    values = [Int(c) for c in text]
    for p in values
        if key === nothing
            key = iterate(gamma)
        end
        k, state = key
        c = Char(((p - 1040) + (Int(k) - 1040)) % 32 + 1040)
        shtext *= string(c)
        key = iterate(gamma, state)
    end
    return shtext
end

function decrypt(shtext, gamma)
    text = ""
    key = iterate(gamma)
    for c in shtext
        if key === nothing
            key = iterate(gamma)
        end
        k, state = key
        p = Char(((Int(c) - 1040) - (Int(k) - 1040)) % 32 + 1040)
        text *= string(p)
        key = iterate(gamma, state)
    end
    return text
end

println("Введите текст: ")
text = readline()
println("Введите гамму: ")
gamma = readline()
```

Программа

```
shtext = encrypt(text, gamma)
println("Зашифрованный текст:", shtext)
descr_text = decrypt(shtext, gamma)
println("Расшифрованный текст:", descr_text)
```

Программа



Результат

Выводы

Мы изучили шифрование гаммированием, а также реализовали алгоритм шифрования гаммированием конечной гаммой на языке программирования Julia.

Список литературы

1. Mathematics // Julia URL: https://docs.julialang.org/en/v1/base/math/