Лабораторная работа №3

Дисциплина: Математические основы защиты информации и информационной безопасности

Алгайли Абдулазиз Мохаммед

Содержание

Цель работы **5**

Задание **6**

Выполнение лабораторной работы **7**

Реализация функции шифрования XOR 7
Тестирование шифрования и расшифровки 7
Реализация LCG 9
Тестирование генерации ключей 10

Выводы **11**

Список иллюстраций

1 Результат шифрования 8
2 Результат расшифровки 9
3 Результат генерации ключей LCG 10

Список таблиц

Цель работы

Познакомиться с шифрованием с помощью XOR и генерацией ключей с использованием линейного конгруэнтного генератора (LCG).

Задание

1. Программно реализовать шифрование с помощью XOR.
2. Программно реализовать расшифровку с помощью XOR.
3. Программно реализовать генерацию ключей с использованием линейного конгруэнтного генератора (LCG).

Выполнение лабораторной работы

1) Все шифрования были реализованы на языке Julia. Сначала я создал функцию `xor_encrypt`, которая реализует побитовое сложение (XOR) между символами текста и ключа. Для расшифровки текста используется та же функция, так как операция XOR обратима.

Реализация функции шифрования XOR

```
function xor_encrypt(plaintext::String, key::String)
    if length(key) < length(plaintext)
        error("Key must be as long as or longer than the plaintext.")
    end

    encrypted = [Char(codepoint(plaintext[i]) ⊻ codepoint(key[i])) for i
    return join(encrypted)
end
```

Тестирование шифрования и расшифровки

Шаг 1: Шифрование

Пример 1:

```
Текст для шифрования: Привет
Ключ для шифрования: Ключик
```

Зашифрованный текст: {vu

```
Choose an operation:
1. Encrypt text using XOR
2. Decrypt text using XOR
3. Generate key using Linear Congruential Generator (LCG)
4. Exit
1
Enter the plaintext to encrypt:
Привет
Enter the key (should be as long as or longer than the text):
Ключик
xncrypted text: {vu
```

Рис. 1: Результат шифрования

Шаг 2: Расшифровка

Пример 2:

Зашифрованный текст: {vu

Ключ для расшифровки: Ключик

Расшифрованный текст: Привет

```
Choose an operation:
1. Encrypt text using XOR
2. Decrypt text using XOR
3. Generate key using Linear Congruential Generator (LCG)
4. Exit
1
Enter the plaintext to encrypt:
Привет
Enter the key (should be as long as or longer than the text):
Ключик
xncrypted text: {vu
Choose an operation:
1. Encrypt text using XOR
2. Decrypt text using XOR
3. Generate key using Linear Congruential Generator (LCG)
4. Exit
2
Enter the encrypted text:
{vu
Enter the key used for decryption:
Ключик
Decrypted text: wэл
Choose an operation:
1. Encrypt text using XOR
2. Decrypt text using XOR
3. Generate key using Linear Congruential Generator (LCG)
4. Exit
```

Рис. 2: Результат расшифровки

2) Далее я реализовал генерацию ключей с использованием линейного конгруэнтного генератора (LCG). Для этого была создана функция `lcg`, которая генерирует последовательность псевдослучайных чисел на основе параметров a, b, m и seed.

Реализация LCG

```
function lcg(a, b, m, seed, length)
    random_sequence = Int[]
    yi = seed
```

```
    for i in 1:length
        yi = (a * yi + b) % m
        push!(random_sequence, yi)
    end
    return random_sequence
end
```

Тестирование генерации ключей

Пример 3:

Параметры LCG: a = 5, b = 3, m = 16, seed = 7, длина = 6

Сгенерированная последовательность: [6, 1, 8, 11, 10, 5]

```
Choose an operation:
1. Encrypt text using XOR
2. Decrypt text using XOR
3. Generate key using Linear Congruential Generator (LCG)
4. Exit
3
Enter LCG parameters:
Enter value for a:
5
Enter value for b:
3
Enter value for m:
16
Enter the seed (initial value):
7
Enter the length of the key:
6
Generated key sequence: [6, 1, 8, 11, 10, 5]
```

Рис. 3: Результат генерации ключей LCG

5) Для удобства пользователя был создан интерактивный интерфейс с меню, позволяющим выбрать операцию: шифрование, расшифровка или генерация ключа.

Выводы

Я успешно реализовал шифрование с использованием XOR и генерацию ключей с помощью линейного конгруэнтного генератора (LCG). Все функции были протестированы на примерах с использованием русского текста. Результаты тестов показали, что шифрование и расшифровка работают корректно, а генерация ключей выдает ожидаемые результаты.