Вторая лабораторная работа. Шифры перестановки

НПИмд-01-23

Бармина Ольга Константиновна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Цель данной работы - ознакомиться с шифрами перестановки, а также научиться применять их на практике.

2 Задание

1. Реализовать маршрутное шифрование
2. Реализовать шифрование с помощью решеток
3. Реализовать шифрование с использованием таблицы Вижинера

3 Теоретическое введение

Шифры перестановки преобразуют открытый текст в криптограмму путем перестановки его символов. Способ, каким при шифровании переставляются буквы открытого текста и является ключом шифра. Важным требования является равенство длин ключа исходного текста [1].

4 Выполнение лабораторной работы

1. Произведено ознакомление с шифрами перестановки по методическим материалам курса
2. Прописан код для маршрутного шифрования на языке программирования Python. Выводим на экран результат применения.

```
function marsh(text::String, n::Int, m::Int, pas::String)
    text = lowercase(replace(text, " " => ""))
    alphabet = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"

    if length(text) < n * m
        text *= alphabet[1:(m * n - length(text))]
    end

    c = Iterators.flatten(Iterators.repeated(text, 1))
    matrix = [collect(c)[(i - 1) * m + 1:i * m] for i in 1:n]

    mat_pas = [findfirst(==(x), alphabet) for x in pas]
    mat_pas_sort = sort(mat_pas)

    res = ""
    for c1 in mat_pas_sort
        for i in 1:n
            res *= matrix[i][findfirst(==(c1), mat_pas)]
        end
    end
    return res
end
```

```
marsh (generic function with 1 method)
```

```
test_text = "Нельзя недооценивать противника"
marsh(test_text, 5, 6, "пароль")
```

```
"еенпнзоатаьовокннеьвлдирияцтиа"
```

Рис. 4.1: Маршрутное шифрование

4. Прописан код для шифрования с помощью решеток на языке программирования Python. Выводим на экран результат применения.

```
using LinearAlgebra

function resh(text, pas, k=2)
    text = lowercase(replace(text, " " => ""))
    k_2 = [x + 1 for x in 0:(k^2 - 1)]
    matr = zeros(Int, 2*k, 2*k)

    for x in 1:k^2
        c = 1
        for x in 1:k
            for y in 1:k
                matr[x, y] = k_2[c]
                c += 1
            end
        end
        matr = rotr90(matr)
    end

    mv = Dict(k => 0 for k in k_2)
    mv_2 = Dict(1 => 2, 2 => 4, 3 => 3, 4 => 3)

    for x in 1:k^2
        for y in 1:k^2
            mv[matr[x, y]] += 1
            if mv[matr[x, y]] != mv_2[matr[x, y]]
                matr[x, y] = -1
            else
                matr[x, y] = 0
            end
        end
    end
```

Рис. 4.2: Шифрование с помощью решеток

```
    ct = 1
    t = Iterators.flatten(Iterators.repeated(text, 1))
    matr2 = fill('0', 2*k, 2*k)

    for v in 1:4
        for x in 1:k^2
            for y in 1:k^2
                if matr[x, y] == 0
                    matr2[x, y] = text[ct]
                    ct += 2
                end
            end
        end
        matr = rotr90(matr, -1)
    end

    alphabet = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"
    password = [findfirst(==(x), alphabet) for x in pas]
    pas_sort = sort(password)
    res = ""
    for c1 in pas_sort
        for i in 1:k^2
            res *= string(matr2[i, findfirst(==(c1), password)])
        end
    end

    return res
end
```

```
resh (generic function with 2 methods)
```

```
test_text = "договор подписали"
resh(test_text, "шифр", 2)
```

```
"осолдргиопповдаи"
```

Рис. 4.3: Результат применения 2

6. Прописан код для шифрования с использованием таблицы Вижинера на языке программирования Python. Выводим на экран результат применения.

```
function gen_key(m, pas)
    m = lowercase(replace(m, " " => ""))
    pas = lowercase(replace(pas, " " => ""))
    pas = collect(pas)
    if length(m) == length(pas)
        return pas
    else
        for i in 1:(length(m) - length(pas))
            push!(pas, pas[(i - 1) % length(pas) + 1])
        end
    end
    return join(pas)
end
```

```
gen_key (generic function with 1 method)
```

```
function vigion(text, pas)
    v = Char[]
    text = lowercase(replace(text, " " => ""))
    for i in 1:2:length(text)*2
        x = (Int(text[i]) + Int(pas[i]) - 2*Int('a')) % 32 + Int('a')
        push!(v, Char(x))
    end
    return join(v)
end
```

```
vigion (generic function with 1 method)
```

```
test_text = "криптография серьезная наука"
key = "математика"
gen_key(test_text, key)
```

```
"математикаматематикаматема"
```

```
vigion(test_text, gen_key(test_text, key))
```

```
"цръфюохшкффягкььчпчалнтшца"
```

Рис. 4.4: Таблица Вижинера

5 Выводы

В рамках данной лабораторной работы было произведено ознакомление с шифрами перестановки. Шифры были реализованы на языке программирования Python.

Список литературы

1. Кулябов Д.С. Методические материалы курса. РУДН, 2024. 354 с.