

Лабораторная работа №2

Традиционные шифры с симметричным ключом

Дугаева Светлана Анатольевна, НФИИМД-02-22

Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Выполнение лабораторной работы	8
Реализация маршрутного шифрования	8
Реализация шифрования с помощью решеток	9
Реализация таблицы Виженера	9
Выводы	12

Список таблиц

Список иллюстраций

0.1	код1	8
0.2	код2	9
0.3	код3	10
0.4	код4	11

Цель работы

Цель данной работы — изучить и программно реализовать шифры перестановки.

Задание

Заданием является:

- Реализовать маршрутное шифрование
- Реализовать шифрование с помощью решеток
- Реализовать таблицу Виженера

Теоретическое введение

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

Выполнение лабораторной работы

Для реализации шифров мы будем использовать Python, так как его синтаксис позволяет быстро реализовать необходимые нам алгоритмы.

Реализация маршрутного шифрования

Код маршрутного шифрования реализуем в виде функции следующего вида:

```
def split_text(text, length):  
    return [text[i:i+length] for i in range(0, len(text), length)]  
  
def encode(key, text):  
    order = {int(val): num for num, val in enumerate(key)}  
  
    res = ''  
    for ind in sorted(order.keys()):  
        for t in split_text(text, len(key)):  
            try:  
                res += t[order[ind]]  
            except IndexError:  
                continue  
    return res  
print(encode('41253', 'HEYEVERYONEHERE'))
```

ERHYEVENEEEEOR

Рис. 0.1: код1

Для проверки ввели текст как в лабораторной работе, получили тот же результат.

Реализация шифрования с помощью решеток

Шифрование с помощью решеток реализуем в виде функции следующего вида:

```
def vig(text, key):
    key_len = len(key)
    key_i = [ord(i) for i in key]
    text_i = [ord(i) for i in text]
    res = ''
    for i in range(len(text_i)):
        val = (text_i[i] + key_i[i % key_len]) % 26
        res += chr(val + 65)
    return res
print(vig('HEYAREYOUSTILLHERE', 'DUCK'))
```

KYAKUYAYXMVSOFJOUY

Рис. 0.2: код2

Для проверки ввели текст как в лабораторной работе, получили тот же результат.

Реализация таблицы Виженера

Таблицу Виженера реализуем в виде функций следующего вида:

```

import numpy as np

def resh(text, key):
    ru_letters = 'абвгдеёжзийклмнопрстуфхцчщъыьэюя'
    k = 2
    k_2 = [x+1 for x in range(k*k)]

    matr = [[0 for x in range(2*k)] for y in range(2*k)]
    matr = np.array(matr)
    for x in range(k*k):
        cou = 0
        for x in range(k):
            for y in range(k):
                matr[x][y] = k_2[cou]
                cou += 1
        matr = np.rot90(matr)

    d_s = {k: 0 for k in k_2}
    d_ss = {1:2, 2:4, 3:3, 4:3}
    for x in range(k*k):
        for y in range(k*k):
            d_s[matr[x][y]] += 1
            if d_s[matr[x][y]] != d_ss[matr[x][y]]:
                matr[x][y] = -1
            else:
                matr[x][y] = 0

```

Рис. 0.3: код3

```

cou_p = 0
text1 = iter(text)
matr_p = [['0' for y in range(k*k)] for x in range(k*k)]
for d in range(4):
    for x in range(k*k):
        for y in range(k*k):
            if matr[x][y] == 0:
                matr_p[x][y] = text[cou_p]
                cou_p += 1
        matr = np.rot90(matr, -1)
    key_3 = [ru_letters.index(x) for x in key]
    key_sort = sorted(key_3)
    res = ''
    for i in key_sort:
        for x in range(k*k):
            res += matr_p[x][key_3.index(i)]

    return(res)
print(resh('приветпокаконец', 'беги'))

```

натоексвкерппцио

Рис. 0.4: код4

Выводы

В рамках выполненной лабораторной работы мы изучили и реализовали следующие шифры: маршрутное шифрование, шифрование с помощью решеток, таблицу Виженера.