

Отчёт по лабораторной работе 2

Шифры перестановки

Гебриал Ибрам Есам Зекри НФИ-02-22

Содержание

1	Цель работы	5
2	Теоретические сведения	6
3	Выполнение лабораторной работы	9
4	Выводы	15
5	Список литературы	16

List of Tables

List of Figures

3.1	Функция для кодирования текста шифром Маршрутного шифрования	9
3.2	Блок кода для вывода результат в соответствии с шифром Маршрутного шифрования	10
3.3	Функция для определения индекс буквы в ключе	10
3.4	Получение шифрования текста методом Маршрутного шифрования	11
3.5	Блок кода для шифрования с помощью решеток	12
3.6	Блок для выполнения матрицы	13
3.7	Получение шифрования текста методом решётки	13
3.8	Функция для кодирования текста шифром Фиженера	14
3.9	Получение шифрования текста методом Фиженера	14

1 Цель работы

Реализация маршрутного шифра, решетчатого шифра и таблицы Виженера.

2 Теоретические сведения

1- Маршрутное шифрование

Этот способ шифрования изобрел выдающийся французский математик и криптограф Франсуа Виет (1540-1603).

Пусть m и n – некоторые натуральные (т.е. целые положительные) числа, каждое больше 1. Открытый текст последовательно разбивается на части (блоки) с длиной, равной произведению mn (если в последнем блоке не хватает букв, можно дописать до нужной длины произвольный их набор). Блок вписывается построчно в таблицу размерности $m \times n$ (т.е. m строк и n столбцов). Криптограмма получается выписыванием букв из таблицы в соответствии с некоторым маршрутом. Этот маршрут вместе с числами m и n составляет ключ шифра.

Чаще всего буквы выписывают по столбцам, которые упорядочиваются в соответствии с паролем: под таблицей подписывается слово, состоящее из n неповторяющихся букв, и столбцы таблицы нумеруются по алфавитному порядку букв пароля. Например, для шифрования открытого текста, выражающего один из главных принципов криптологии: нельзя недооценивать противника, добавим к его 29 буквам еще одну, скажем а, возьмем $m=5$, $n=6$, впишем текст в таблицу 5×6 и выберем в качестве пароля слово п а р о л ь:

нельзя недооценивать противника пароль

Выписывая теперь буквы по столбцам в соответствии с алфавитным порядком букв в пароле, получаем следующую криптограмму: ЕЕНПНЗОАТАЬОВОКННЕЬ-ВЛДИРИЯЦТИА (истинные пробелы в криптографии не выставляются).

Выберите другой пароль и посмотрите, как изменится криптограмма.

Рассмотренный способ шифрования (столбцовая перестановка) в годы первой мировой войны использовала легендарная немецкая шпионка Мата Хари.

2- Шифрование с помощью решеток

Этот способ шифрования предложил в 1881 году австрийский криптограф Эдуард Флейснер. Выбирается натуральное число $k > 1$, и квадрат размерности $k \times k$ построчно заполняется числами 1, 2, ..., k . Для примера возьмем $k = 2$.

Квадрат поворачивается по часовой стрелке на 90° и размещается вплотную к предыдущему квадрату. Аналогичные действия совершаются еще два раза, так чтобы в результате из четырех малых квадратов образовался один большой с длиной стороны $2k$.

Далее из большого квадрата вырезаются клетки с числами от 1 до k^2 , для каждого числа одна клетка. Процесс шифрования происходит следующим образом. Сделанная решетка (квадрат с прорезями) накладывается на чистый квадрат $2k \times 2k$ и в прорези по строчкам (т.е. слева направо и сверху вниз) вписываются первые буквы открытого текста. Затем решетка поворачивается на 90° по часовой стрелке и накладывается на частично заполненный квадрат, вписывание продолжается.

После третьего поворота, наложения и вписывания все клетки квадрата будут заполнены. Правило выбора прорезей гарантирует, что при заполнении квадрата буква на букву никогда не попадет. Из заполненного квадрата буквы можно выписать по столбцам, выбрав подходящий пароль. Например, с использованием изображенной выше решетки и пароля ш и ф р открытый текст договор подписали переводится в криптограмму за пять шагов.

Итоговая криптограмма: ОВОРДЛГПАПИОСДОИ.[1]

3- Шифр Виженера

Шифр Виженера (фр. Chiffre de Vigenère) — метод полиалфавитного шифрования буквенного текста с использованием ключевого слова.

Этот метод является простой формой многоалфавитной замены. Шифр Виженера изобретался многократно. Впервые этот метод описал Джован Баттиста

Беллазо (итал. Giovan Battista Bellaso) в книге *La cifra del. Sig. Giovan Battista Bellaso* в 1553 году, однако в XIX веке получил имя Блеза Виженера, французского дипломата. Метод прост для понимания и реализации, он является недоступным для простых методов криптоанализа.

В шифре Цезаря каждая буква алфавита сдвигается на несколько строк; например в шифре Цезаря при сдвиге +3, А стало бы D, В стало бы Е и так далее. Шифр Виженера состоит из последовательности нескольких шифров Цезаря с различными значениями сдвига. Для зашифровывания может использоваться таблица алфавитов, называемая *tabula recta* или квадрат (таблица) Виженера. Применительно к латинскому алфавиту таблица Виженера составляется из строк по 26 символов, причём каждая следующая строка сдвигается на несколько позиций. Таким образом, в таблице получается 26 различных шифров Цезаря. На каждом этапе шифрования используются различные алфавиты, выбираемые в зависимости от символа ключевого слова. Например, предположим, что исходный текст имеет вид:

ATTACKATDAWN

Человек, посылающий сообщение, записывает ключевое слово («LEMON») циклически до тех пор, пока его длина не будет соответствовать длине исходного текста:

LEMONLEMONLE

Первый символ исходного текста А зашифрован последовательностью L, которая является первым символом ключа. Первый символ L шифрованного текста находится на пересечении строки L и столбца А в таблице Виженера. Точно так же для второго символа исходного текста используется второй символ ключа; то есть второй символ шифрованного текста Х получается на пересечении строки Е и столбца Т. Остальная часть исходного текста шифруется подобным способом.

Исходный текст: ATTACKATDAWN

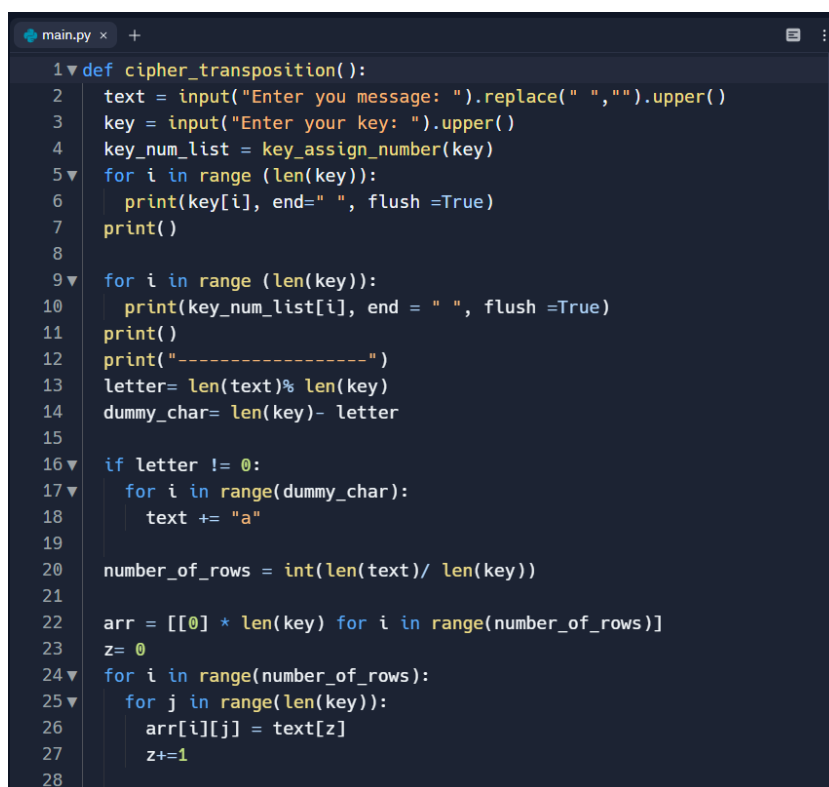
Ключ: LEMONLEMONLE

Зашифрованный текст: LXFOPVEFRNHR [2]

3 Выполнение лабораторной работы

1. Написал функцию cipher_transposition для шифрования текста. (рис. 3.1) (рис. 3.2) (рис. 3.3)

Написал функции для определения индекс буквы в нашем ключе, а затем пополнил таблицу с сообщением. в конце распечатал шифрование в порядке индекса ключа



```
1 def cipher_transposition():
2     text = input("Enter you message: ").replace(" ", "").upper()
3     key = input("Enter your key: ").upper()
4     key_num_list = key_assign_number(key)
5     for i in range (len(key)):
6         print(key[i], end=" ", flush =True)
7     print()
8
9     for i in range (len(key)):
10        print(key_num_list[i], end = " ", flush =True)
11    print()
12    print("-----")
13    letter= len(text)% len(key)
14    dummy_char= len(key)- letter
15
16    if letter != 0:
17        for i in range(dummy_char):
18            text += "a"
19
20    number_of_rows = int(len(text)/ len(key))
21
22    arr = [[0] * len(key) for i in range(number_of_rows)]
23    z= 0
24    for i in range(number_of_rows):
25        for j in range(len(key)):
26            arr[i][j] = text[z]
27            z+=1
28
```

Figure 3.1: Функция для кодирования текста шифром Маршрутного шифрования

```

9 for i in range(number_of_rows):
10     for j in range(len(key)):
11         print(arr[i][j], end = " ", flush=True)
12     print()
13 num_loc =get_location(key,key_num_list)
14 cipher_transposition =""
15 counter= 0
16 for i in range(number_of_rows+1):
17     if counter ==len(key):
18         break
19     else:
20         d= int(num_loc[counter])
21         for j in range(number_of_rows):
22             cipher_transposition += arr[j][d]
23         counter+=1
24
25 print("Cipher Text: {} ".format(cipher_transposition))

```

Figure 3.2: Блок кода для вывода результат в соответствии с шифром Маршрутного шифрования

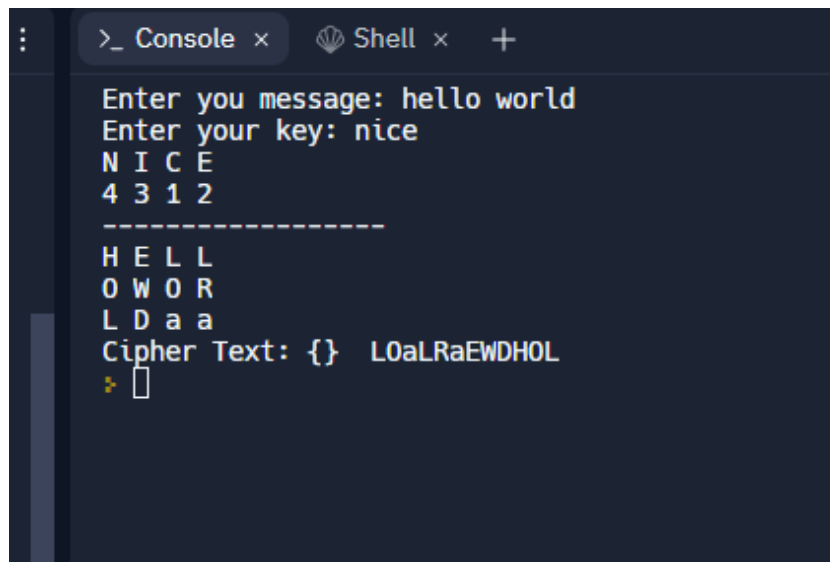
```

47 def get_location(key, key_num_list):
48     num_loc=""
49     for i in range(len(key)+1):
50         for j in range(len(key)):
51             if key_num_list[j] ==i:
52                 num_loc += str(j)
53     return num_loc
54 def key_assign_number(key):
55     alphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
56     key_num_list=list(range(len(key)))
57     init =0
58     for i in range(len(alphabet)):
59         for j in range(len(key)):
60             if alphabet[i]== key[j]:
61                 init+=1
62                 key_num_list[j]= init
63     return key_num_list
64
65 cipher_transposition()
66

```

Figure 3.3: Функция для определения индекс буквы в ключе

Получил результат. (рис. 3.4)

A terminal window with a dark background and light-colored text. The window has two tabs: ">_ Console" and "Shell". The text in the terminal shows a user entering a message and a key, followed by the key being converted to uppercase and its length. Then, the message is displayed in a grid format, and the resulting ciphertext is shown.

```
>_ Console x Shell x +
Enter you message: hello world
Enter your key: nice
N I C E
4 3 1 2
-----
H E L L
O W O R
L D a a
Cipher Text: {} LOaLRaEWDHOL
>
```

Figure 3.4: Получение шифрования текста методом Маршрутного шифрования

2. Писал блок для шифрования с помощью решеток.

Генерировал ключ с помощью random в соответствии с длиной текста. Затем проверяем что ключ не повторяется, так как мы не можем поставить две буквы в одной позиции.

Пополнял свободное место в матрице случайным образом из списка алфавита.
(рис. 3.5)

```

67 from random import choice, randint
68 from collections import Counter
69
70 text = "HelloWorld"
71 alphabet = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k',
              'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y',
              'z', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm',
              'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
72
73 while True:
74     key= [randint(1,20) for x in range(len(text))]
75     count = Counter(key)
76     switch =0;
77     n=0
78     for l in count:
79         if count[key[n]] > 1:
80             switch = 1
81             break
82         n += 1
83     if switch == 0:
84         break
85
86     key.sort()
87     print("key: ",key)
88
89     grille = [choice(alphabet) for x in range(0,21)]
90

```

Figure 3.5: Блок кода для шифрования с помощью решеток

Писал код для выполнения матрицы(рис. 3.6)

```

91 n=0
92 for i in range(len(grille)):
93     if n < len(text):
94         if i == key[n]:
95             grille[i] = text[n]
96             n+=1
97     print()
98 for j in range(0,6):
99     if j == 0:
100         print(" ", end = " ")
101     else:
102         print(j, end = " ")
103     print()
104
105 n = 1
106
107 for j in range(1,len(grille)):
108     if j == 1:
109         print(n, end = " | "); n += 1
110         print(grille[j], end = " | ")
111     elif j % 5 != 0:
112         print(grille[j], end = " | ")
113     else:
114         print(grille[j], end = " | ")
115         print()
116         if n < 5:
117             print(n, end = " | "); n += 1
118     print()
119
120

```

Figure 3.6: Блок для выполнения матрицы

Получил результат. (рис. 3.7)

```

key: [3, 5, 6, 7, 11, 13, 15, 17, 19, 20]

  1  2  3  4  5
1 | e | q | H | f | e |
2 | l | l | v | m | w |
3 | o | s | W | s | o |
4 | l | r | t | l | d |

```

Figure 3.7: Получение шифрования текста методом решётки

3. Написал функцию `vingere` для шифрования Виженера. (рис. 3.8)

Сначала написал алфавит в виде списка.

Потом определил индекс каждой буквы в сообщении, аналогично ключу.

Как определил позицию, сложил на него позиции ключа. потом распечатал зашифрованный текст.

```
123
124 alphabet = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k',
               'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y',
               'z', 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm',
               'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z']
125
126 def vigenere(start, key):
127     key *= len(start)
128
129     end_text = ""
130     for i, j in enumerate(start):
131         position = alphabet.index(j)
132         position_key = alphabet.index(key[i])
133         new_posotion = (position + position_key) % 26
134         end_text += alphabet[new_posotion]
135     print(f"our result: {end_text}")
136
137 vigenere("hello ".replace(' ', ''), "bic")
```

Figure 3.8: Функция для кодирования текста шифром Виженера

Получил результат. (рис. 3.9)

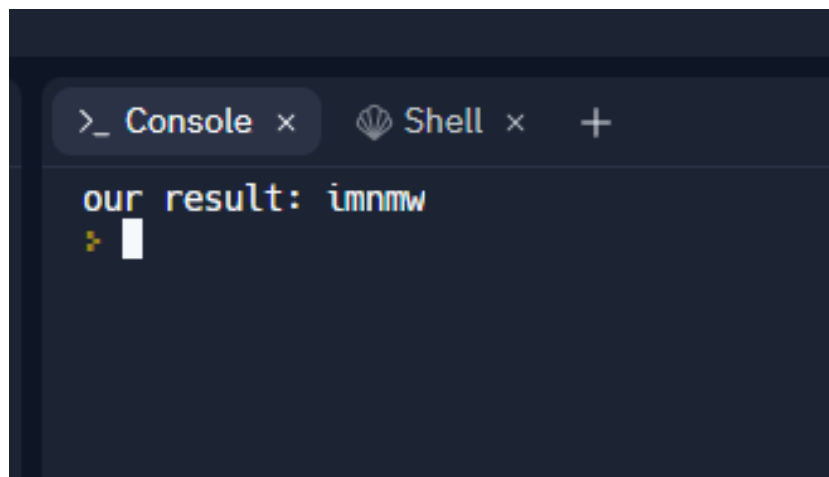
A screenshot of a terminal window with a dark background. At the top, there are tabs labeled ">_ Console x" and "Shell x" with a plus sign to the right. The main area of the terminal displays the text "our result: imnmw" in a light-colored font. Below this text, there is a small yellow cursor icon and a white rectangular block.

Figure 3.9: Получение шифрования текста методом Виженера

4 Выводы

Реализовал шифрование с помощью решеток, маршрутное шифрование и шифр Виженера

5 Список литературы

1. Перестановочные шифры.— URL: https://it.rfei.ru/course/_k017/7mdCpor7/~c5kOtaHYinformatika/shifry_prostoy_zameny.
2. Шифр Виженера. — URL: <https://www.sites.google.com/site/kriptografics/sifr-vizenera/>.