Цель работы

Изучение алгоритмов маршрутной перестановки, решеток и Виженера. И создание программ для их реализации.

Теоретические сведения

Шифр маршрутной перестановки

Широкое распространение получили шифры перестановки, использующие некоторую геометрическую фигуру. Преобразования из этого шифра состоят в том, что в фигуру исходный текст вписывается по ходу одного «маршрута», а затем по ходу другого выписывается с нее. Такой шифр называют **маршрутной перестановкой**.

Например, можно вписывать исходное сообщение в прямоугольную таблицу, выбрав такой маршрут: по горизонтали, начиная с левого верхнего угла поочередно слева направо и справа налево. Выписывать же сообщение будем по другому маршруту: по вертикали, начиная с верхнего правого угла и двигаясь поочередно сверху вниз и снизу вверх

Зашифруем, например, указанным способом фразу:

ПРИМЕРМАРШРУТНОЙПЕРЕСТАНОВКИ

используя прямоугольник размера 7x4:

П	**Р**	**И**	**М**	**Е**	**Р**	**М**
Н	Т	У	Р	Ш	Р	А
О	Й	П	Е	Р	Е	С
И	К	В	О	Н	А	Т

Зашифрованная фраза выглядит так:

МАСТАЕРРЕШРНОЕРМИУПВКЙТРПНОИ

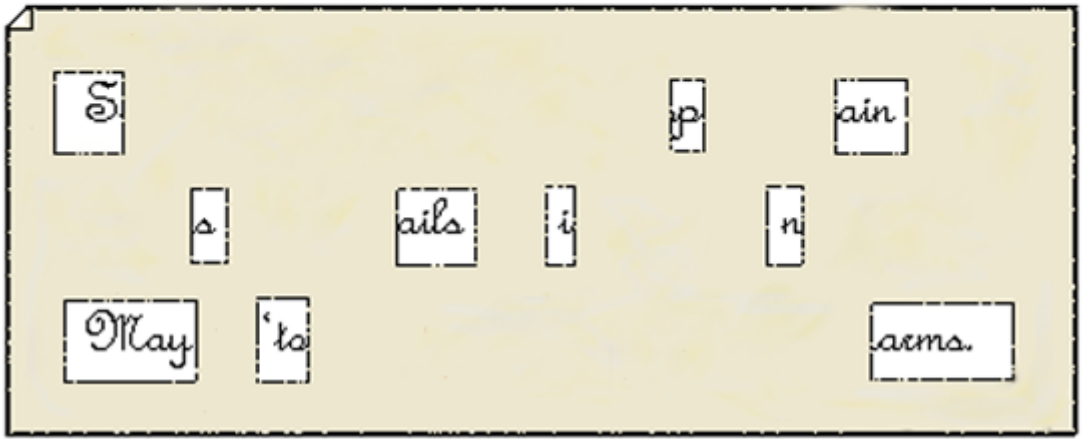
Теоретически маршруты могут быть значительно более изощренными, однако запутанность маршрутов усложняет использование таких шифров.

Шифр Кардано

Часто так бывает, что талантливый человек входит в историю как автор какого-то одного открытия, а прочие его заслуги при этом остаются в тени. Наверное, то же самое можно сказать про Джероламо Кардано.

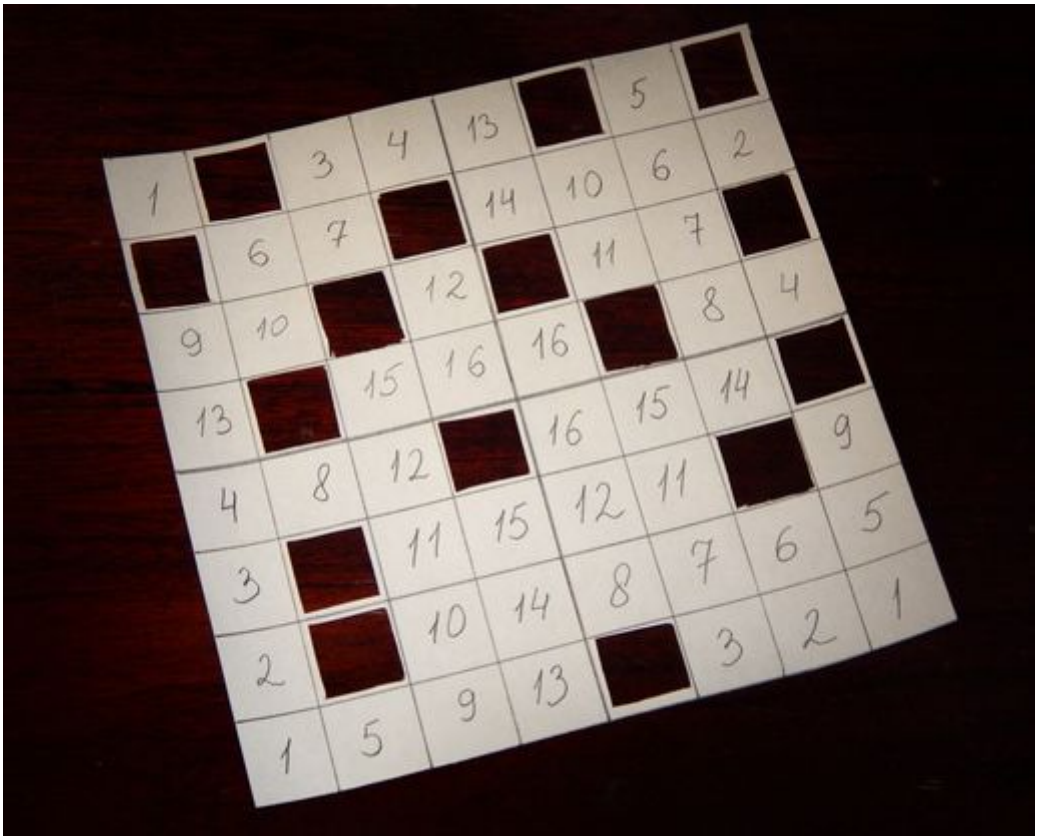
Даже если вы не разбираетесь в автомобилестроении, то наверняка слышали о каком-то карданном вале. Это такая деталь, которая передает крутящий момент от коробки передач или раздаточной коробки к редуктору переднего или заднего моста. Джероламо придумал этот шарнирный механизм, но, помимо “автомобильного” изобретения, у Кардано было много других блестящих идей, например о пользе переливания крови. Еще одно изобретение Кардано — шифрование по трафарету или решетке.

Sir John regards you well and spekes again that
all as rightly 'vails him is yours now and ever.
May he 'tone for past d'lays with many charms.



> **Текст записки:**
> Сэр Джон высоко ценит Вас и снова повторяет, что все, что доступно ему, теперь ваше, навсегда. Может ли он заслужить прощение за свои прежние промедления посредством своего обаяния.
> **Шифрованное послание:**
> В мае Испания направит свои корабли на войну.

Решетка Кардано может быть двух видов — простая и симметрично-поворотная. В первом случае для шифрования применяется трафарет с отверстиями, через которые "фильтруется" полезный текст. Другой вариант решетки, более интересный, состоит в том, чтобы использовать симметричный (квадратный) трафарет, который можно применять несколько раз, просто поворачивая его вокруг центра. Поворотная решетка Кардано позволяет записать текст массивом символов так, что результат будет выглядеть совершенно нечитаемым.



Решетка Кардано была очень практичной и удобной. Чтобы прочитать секретный текст, не нужно было "решать кроссворд" или тратить время на обучение секретному языку. Этим шифром предпочитали пользоваться многие известные личности, например кардинал Ришелье и русский драматург и дипломат Александр Грибоедов.

Таблица накладывается на носитель, и в вырезанные ячейки вписываются буквы, составляющие сообщение. После переворачивания таблицы вдоль вертикальной оси, процесс вписывания букв повторяется. Затем то же самое происходит после переворачивания вдоль горизонтальной и снова вдоль вертикальной осей.

В частном случае квадратной таблицы, для получения новых позиций для вписывания букв, можно поворачивать квадрат на четверть оборота.

Чтобы прочитать закодированное сообщение, необходимо наложить решётку Кардано нужное число раз на закодированный текст и прочитать буквы, расположенные в вырезанных ячейках.

Такой способ шифрования сообщения был предложен математиком Джероламо Кардано в 1550 году, за что и получил своё название.

Шифр Виженера

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

Интересно, что человек, давший имя этому шифру, никакого отношения к нему не имел. На самом деле его автором был итальянский математик Джованни Батиста Беллазо. Его труды и изучил Блез де Виженер во время своей двухлетней дипломатической миссии в Риме. Вникнув в простой, но эффективный принцип шифрования, дипломат сумел преподнести эту идею, показав ее комиссии Генриха III во Франции.

Суть нового принципа шифрования заключалась в том, что величина сдвига для замещения букв была переменной и определялась ключевым словом или фразой. Долгое время этот метод считался неуязвимым для разгадывания, и даже авторитеты в области математики признавали его надежность. Так, легендарный автор приключений “Алисы в зазеркалье” и “Алисы в стране

чудес", писатель-математик Льюис Кэрролл в своей статье «Алфавитный шифр» прямо и категорично называет шифр Виженера "невзламываемым". Эта статья вышла в детском журнале в 1868 году, но даже спустя полвека после статьи Чарльза Латуиджа Доджсона (это настоящее имя автора сказок про Алису) научно-популярный американский журнал Scientific American продолжал утверждать, что шифр Виженера невозможно взломать.

Для расшифровки шифра Виженера использовалась специальная таблица, которая называлась tabula recta.

	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	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
B	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
C	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
D	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
E	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
F	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
G	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
H	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
I	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
J	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
K	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
L	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
M	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
N	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
O	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
P	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
Q	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
R	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
S	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
T	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
U	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
V	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
W	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
X	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
Y	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
Z	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

> Квадрат Виженера, или таблица Виженера, также известная как tabula recta, может быть использована для шифрования и расшифровывания.

В шифре Цезаря каждая буква алфавита сдвигается на несколько позиций; например в шифре Цезаря при сдвиге +3, A стало бы D, B стало бы E 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 такой 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 («A») зашифрован последовательностью L, которая является первым символом ключа. Первый символ зашифрованного текста («L») находится на пересечении строки L и столбца A в таблице Виженера. Точно так же для второго символа исходного текста используется второй символ ключа; то есть второй символ зашифрованного текста («X») получается на пересечении строки E и столбца T. Остальная часть исходного текста шифруется подобным способом.

```
Исходный текст:       ATTACKATDAWN
Ключ:                 LEMONLEMONLE
Зашифрованный текст:  LXFOPVEFRNHR
```

Выполнение работы

Реализация шифра маршрутной перестановки на языке Python

```
# задание 1. Маршрутное шифрование.
def marhsrutshifr():
    text = input("Ввведите текст: ").replace(' ', '')
    print("Длина текста без пробелов: ", len(text))
    n = int(input("Введите число столбцов n "))
    m = int(input("Введите число строк m "))
    parol = input("Введите слово-пароль: ")
    lists = [['a' for i in range(0, n)] for j in range(m)]
    it = 0
    for i in range(m):
        for j in range(n):
            if it < len(text):
                lists[i][j] = text[it]
                it += 1
    lis = list()
    for i in range(n):
        lis.append(parol[i])
    lists.append(lis)
    prrint(lists)
    result = ""
    spisok = sorted(lists[len(lists) - 1])
    for i in spisok:
        print(i, " = ", lists[len(lists)-1].index(i))
        for j in range(len(lists)):
            if j==len(lists)-1:
                continue
            result += lists[j][lists[len(lists)-1].index(i)]
    print(result)
```

Реализация шифра решеткой на языке Python

```
# функция для поворота матрицы
def rot90(matrix):
    return[list(reversed(col)) for col in zip(*matrix)]

# функция удаления чисел из матрицы
def udalenie(largelist, inn, k):
    for i in range(k * 2):
        for j in range(k * 2):
            if largelist[i][j] == inn:
                largelist[i][j] = " "
                return

# задание 1. Шифр Кардано
def cardangrille():
```

```
k = int(input("Введите число k: "))
s=1
lists = [[i for i in range(k)] for i in range(k)]
for i in range(k):
    for j in range(k):
        lists[i][j] = s
        s += 1
print(lists)
lists1 = rot90(lists)
lists2 = rot90(lists1)
lists3 = rot90(lists2)
largelist = [[1 for i in range(2*k)] for i in range(2*k)]
for i in range(k):
    for j in range(k):
        largelist[i][j] = lists[i][j]
i1 = 0
j1 = 0
for i in range(0, k):
    for j in range(k, k*2):
        largelist[i][j] = lists1[i1][j1]
        j1 += 1
    j1 = 0
    i1 += 1
i1 = 0
j1 = 0
for i in range(k, k*2):
    for j in range(k, k * 2):
        largelist[i][j] = lists2[i1][j1]
        j1 += 1
    j1 = 0
    i1 += 1
i1 = 0
j1 = 0
for i in range(k, k * 2):
    for j in range(0, k):
        largelist[i][j] = lists3[i1][j1]
        j1 += 1
    j1 = 0
    i1 += 1
prrint(largelist)
text = "криптографиянаукаозащитеинформации"
largelist_a = [[" " for i in range(2*k)] for i in range(2*k)]
s = 0
li = [i for i in range(1,k**2+1)]
for inn in li:
    udalenie(largelist, inn, k)
ind = 0

for i in range(k * 2):
    for j in range(k * 2):
        if largelist[i][j] == largelist_a[i][j] and len(text) > 0:
            largelist_a[i][j] = text[0]
            text = text[1:]
largelist = rot90(largelist)
for i in range(k * 2):
    for j in range(k * 2):
        if largelist[i][j] == largelist_a[i][j] and len(text) > 0:
            largelist_a[i][j] = text[0]
```

```
                text = text[1:]
    if len(text) > 0:
        largelist = rot90(largelist)
        for i in range(k * 2):
            for j in range(k * 2):
                if largelist[i][j] == largelist_a[i][j] and len(text) > 0:
                    largelist_a[i][j] = text[0]
                    text = text[1:]
    if len(text) > 0:
        largelist = rot90(largelist)
        for i in range(k * 2):
            for j in range(k * 2):
                if largelist[i][j] == largelist_a[i][j] and len(text) > 0:
                    largelist_a[i][j] = text[0]
                    text = text[1:]
    prrint(largelist_a)
    stri = input("Введите пароль: ")

    if len(stri) > k*2:
        stri = stri[:k*2]
    elif len(stri) < k*2:
        while len(stri) != k*2:
            stri += "я"
    largelist_a.append(list(stri))
    prrint(largelist_a)
    result = ""

    spisok = sorted(largelist_a[len(largelist_a) - 1])
    for i in spisok:
        print(i, " = ", largelist_a[len(largelist_a) - 1].index(i))
        for j in range(len(largelist_a)):
            if j==len(largelist_a)-1:
                continue
            result += largelist_a[j][largelist_a[len(largelist_a) - 1].index(i)]
    print(result.replace(" ", ""))
```

Реализация шифра Виженера на языке Python

```
# задание 1. Шифр Вижинер
def form_dict():
    d = {}
    iter = 0
    for i in range(0,127):
        d[iter] = chr(i)
        iter = iter +1
    return d

def encode_val(word):
    list_code = []
    lent = len(word)
    d = form_dict()

    for w in range(lent):
        for value in d:
            if word[w] == d[value]:
               list_code.append(value)
    return list_code
```

```
def comparator(value, key):
    len_key = len(key)
    dic = {}
    iter = 0
    full = 0

    for i in value:
        dic[full] = [i,key[iter]]
        full = full + 1
        iter = iter +1
        if (iter >= len_key):
            iter = 0
    return dic


def full_encode(value, key):
    dic = comparator(value, key)
    print('Compare full encode', dic, "\n")
    lis = []
    d = form_dict()

    for v in dic:
        go = (dic[v][0]+dic[v][1]) % len(d)
        lis.append(go)
    return lis


def decode_val(list_in):
    list_code = []
    lent = len(list_in)
    d = form_dict()

    for i in range(lent):
        for value in d:
            if list_in[i] == value:
               list_code.append(d[value])
    return list_code

def full_decode(value, key):
    dic = comparator(value, key)
    print('Deshifre=', dic,)
    d = form_dict()
    lis =[]

    for v in dic:
        go = (dic[v][0]-dic[v][1]+len(d)) % len(d)
        lis.append(go)
    return lis

def vijer():
    word = "Hello world"
    key = "key"
    print("Слово: ", word)
    print("Ключ: ", key, "\n")
    key_encoded = encode_val(key)
    value_encoded = encode_val(word)
```

```
    print("Value= ",value_encoded)
    print("Key= ", key_encoded)

    shifre = full_encode(value_encoded, key_encoded)
    print("Шифр= ", "".join(decode_val(shifre)),  "\n")

    decoded = full_decode(shifre, key_encoded)
    print("Decode list=", decoded, "\n")
    decode_word_list = decode_val(decoded)
    print("Дешифровка= ","".join(decode_word_list))
```

Контрольный пример

```
Ввод [7]: marhsrutshifr()

          Ввведите текст: пусть будет так как мы хотим
          Длина текста без пробелов:  23
          Введите число столбцов n 6
          Введите число строк m 4
          Введите слово-пароль: защита
            п   у   с   т   ь   б
            у   д   е   т   т   а
            к   к   а   к   м   ы
            х   о   т   и   м   а
            з   а   щ   и   т   а


          а  =  1
          а  =  1
          з  =  0
          и  =  3
          т  =  4
          щ  =  2
          удкоудкопукхтткиьтммсеат
```

```
Ввод [8]: cardangrille()

Введите число k: 3
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
  1   2   3   7   4   1
  4   5   6   8   5   2
  7   8   9   9   6   3
  3   6   9   9   8   7
  2   5   8   6   5   4
  1   4   7   3   2   1


  к   р   и   п   т   ф
  о   о   г   р   и   я
  р   м   а   н   а   у
  а   ц   и   з   к   а
  и       а   щ   и   о
      т   е   и   н   ф


Введите пароль: защита
  к   р   и   п   т   ф
  о   о   г   р   и   я
  р   м   а   н   а   у
  а   ц   и   з   к   а
  и       а   щ   и   о
      т   е   и   н   ф
  з   а   щ   и   т   а


а  =  1
а  =  1
з  =  0
и  =  3
т  =  4
щ  =  2
ромцтромцткораипрнзщитиакинигаиае
```

```
Ввод [8]: vijer()

Слово:  Hello world
Ключ:  key

Value=  [72, 101, 108, 108, 111, 32, 119, 111, 114, 108, 100]
Key=  [107, 101, 121]
Compare full encode {0: [72, 107], 1: [101, 101], 2: [108, 121], 3: [108, 107], 4: [111, 101], 5: [32, 121], 6: [11
9, 107], 7: [111, 101], 8: [114, 121], 9: [108, 107], 10: [100, 101]}

Шифр=  4KfXUcUlXJ

Deshifre= {0: [52, 107], 1: [75, 101], 2: [102, 121], 3: [88, 107], 4: [85, 101], 5: [26, 121], 6: [99, 107], 7: [8
5, 101], 8: [108, 121], 9: [88, 107], 10: [74, 101]}
Decode list= [72, 101, 108, 108, 111, 32, 119, 111, 114, 108, 100]

Дешифровка=  Hello world
```

Выводы

Составлены программы шифрования с использованием алгоритмов:

- маршрутное шифрование
- шифрование с помощью решеток
- таблица Вижнера

Список литературы{.unnumbered}

1. Маршрутная перестановка
2. Шифры из прошлого: тайнопись и загадки докомпьютерной эпохи
3. Решётка Кардано
4. Шифр Виженера