Лабораторная работа 2

Шифры перестановки

Греков Максим Сергеевич

Содержание

List of Figures

1 Цель работы

- Ознакомиться с шифрами перестановки.
- Реализовать маршрутное шифрование.
- Реализовать шифрование с помощью решеток.
- Реализовать шифрование Таблица Виженера.

2 Описание методов

Шифры перестановки преобразуют открытый текст в криптограмму путем перестановки его символов.

Способ, каким при шифровании переставляются буквы открытого текста, и является ключом шифра.

Важным требованием евс является равенство длин ключа и исходного текста.

3 Маршрутное шифрование

3.1 Описание

Простейшим примеров перестановочного шифра являются так называемые «маршрутные перестановки», использующие некоторую геометрическую фигуру (плоскую или объемную).

Шифрование заключается в том, что текст записывается в такую фигуру по некоторой траектории, а выписывается по другой траектории. (рис. 3.1)

3.2 Реализация

```
8
9   def route_encryption(orig_string, pswd, n):
10      string = orig_string.replace(' ','')
11      m = len(string)//n+bool(len(string)%n)
12      string += 'a'*(m*n-len(string))
13      nums = [sorted(pswd).index(c) for c in pswd]
14      return ''.join(string[j*n + nums.index(i)] for i in range(n) for j in range(m)).upper()
15
16  print(route_encryption('нельзя недооценивать противника', 'пароль', 6))
17
```

Figure 3.1: Маршрутное шифрование

Результат: **ЕЕНПНЗОАТАЬОВОКННЕЬВЛДИРИЯЦТИА**

4 Шифрование с помощью решеток

4.1 Описание

Выбирается число *k*. Строим квадрат со стороной длины k и заполняем его клетки числами от *1* до *k^2*.

Поворачиваем квадрат на *90* градусов по часовой стрелке и приписываем справа от исходного квадрата.

Поворачивая на *90* градусов по часовой стрелки и добавляя полученный квадрат сначала снизу, а затем слева от предыдущего, получим квадрат со стороной *2k*.

В этом квадрате закрасим произвольным образом все цифры, причем каждая цифра может быть закрашена только один раз.

Это и будет решёткой для шифрования.

Код для шифрования представляет последовательность *k* цифр от *1* до *4*, *i*-тая цифра обозначает в каком подквадрате (нумеруются в порядке создания) закрашивать число *i*.

Открытый текст разбивается на блоки длины *4k^2*.

Каждый блок разбивается на подстроки длины *k^2*.

Решетка накладывается на пустой лист бумаги, закрашиваемые клетки вырезаются.

Для первой подстроки ее *i*-ый символ записывается в вырезанное *i*-ое число решетки.

Повторяем процесс еще *3* раза, поворачивая перед этим решетку на *90* градусов

по часовой стрелке.

В результате получаем таблицу, составляющую из символов открытого текста.

Криптограмма из этой таблицы получается путем построчного выписывания символов или применения приемов маршрутного шифрования. (рис. 4.1)

4.2 Реализация

```
17
18  def lattice_encryption(orig_string, pswd, k, xys):
19      string = orig_string.replace(' ','')
20      matr = []
21      for i in range(k*2):
22          matr.append(['.']*(k*2))
23
24      u = 0
25      for i in range(4):
26          for x, y in xys:
27              matr[x][y] = string[u]
28              u+=1
29          xys = [(y, 2*k-1-x) for x, y in xys]
30          # mprint(matr)
31
32      res = ''.join(''.join(c for c in matr[i]) for i in range(k*2))
33      return route_encryption(res, pswd, 2*k)
34
35  print(lattice_encryption('договор подписали', 'шифр', 2, [(0,3), (3,2), (2,3), (2,1)]))
36  print(lattice_encryption('ТЕКСТ ПОСЛЕ ШИФРОВАНИЯ СТАНЕТ НЕПОНЯТНЫМ', 'абвгде', 3,
37      [(0,5), (4,0), (2,5), (5,1), (4,4), (1,2), (2,0), (3,4), (3,2)]))
38
```

Figure 4.1: Шифрование с помощью решеток

Результат 1: **ДПОРДАГВЛИИОСПОО**

Результат 2: **ЕФОСЕИШАТТРСОПНЛАНВЫНМНИТЯОСТПТЯКЕНЕ**

5 Таблица Виженера

5.1 Описание

Шифр Виженера состоит из последовательности нескольких шифров Цезаря с различными значениями сдвига. (рис. 5.1)

Для зашифровывания может использоваться таблица алфавитов, называемая *tabula recta* или квадрат (таблица) Виженера.

Применительно к русскому алфавиту таблица Виженера составляется из строк по *33* символов, причём каждая следующая строка сдвигается на несколько позиций.

Таким образом, в таблице получается *33* различных шифров Цезаря.

На каждом этапе шифрования используются различные алфавиты, выбираемые в зависимости от символа ключевого слова.

5.2 Реализация

```
39
40  def vigenere_encryption(orig_string, pswd):
41      string = orig_string.replace(' ','')
42      pswd = (pswd*(len(string)//len(pswd)+1))[:len(string)]
43      alphabet = [chr(c) for c in range(ord('а'), ord('я') + 1)]
44      matr = [alphabet[i:] + alphabet[0:i] for i in range(32)]
45      return ''.join(matr[ord(s)-ord('а')][ord(p)-ord('а')] for p, s in zip(pswd, string)).upper()
46
47  print(vigenere_encryption('криптография серьезная наука', 'математика'))
48
```

Figure 5.1: Таблица Виженера

Результат: **ЦРЪФЮООХШКФФЯГКЬЬЧПЧАЛНТШЦА**

6 Вывод

- Ознакомились с шифрами перестановки.
- Реализовали маршрутное шифрование.
- Реализовали шифрование с помощью решеток.
- Реализовали шифрование Таблица Виженера.