Отчёт по лабораторной работе №2. Шифры перестановки

Дисциплина: Математические основы защиты информации и информационной безопасности

Студент: Агеева Анастасия Сергеевна, 1032212304
Группа: НФИмд-02-21
Преподаватель: д-р.ф.-м.н., проф. Кулябов Дмитрий Сергеевич

11 ноября, 2021, Москва

Прагматика

Прагматика данной лабораторной работы

- В рамках дисциплины “Математические основы защиты информации и информационной безопасности” нам необходимо изучить ее разделы. Данная лабораторная работа входит в раздел “Шифрование”.
- Данная работа необходима для более глубоко и детального понимания работы алгоритмов шифрования.

Цель

Цель выполнения данной лабораторной работы

- Цель данной лабораторной работы изучение реализация трех алгоритмов шифрования: маршрутное шифрование, шифрование с помощью решеток и шифрование при помощи таблицы Виженера.

Задачи

Задачи выполнения данной лабораторной работы

1. Реализовать программно маршрутное шифрование;
2. Реализовать программно шифрование с помощью решеток;
3. Реализовать программно шифрование при помощи таблицы Виженера.

Результаты выполнения данной лабораторной работы

Маршрутное шифрование

In [3]:

```
def path(alp, k, mes):
    alp = list(alp)
    k = list(k)
    mes = list(mes)
    n = len(k)
    m = len(mes)//n
    if (len(mes)//n != 0):
        m+=1
    matr = [[np.random.choice(alp) for i in range(0, n)] for j in range (m)]
    c = 0
    for i in range(m):
        for j in range(n):
            if c < len(mes):
                matr[i][j] = mes[c]
                c += 1
    matr.append(k)
    way = sorted(matr[len(matr) - 1])
    new_mes = []
    for i in way:
        for j in range(len(matr)):
            if j == len(matr) - 1:
                continue
            new_mes += matr[j][matr[len(matr) - 1].index(i)]
    new_mes = ''.join(new_mes)
    return new_mes
```

Figure 1: Маршрутное шифрование

Шифрование с помощью решеток (1)

In [7]:
```
def delete(b_m, i_1, k_1):
    for i in range(2*k_1):
        for j in range(2*k_1 - i):
            if b_m[j][i] == i_1:
                b_m[j][i] = ' '
                return
```

Figure 2: Шифрование с помощью решеток (1)

In [8]:
```
def lattice(alp, key, mes):
    key = list(key)
    mes = list(mes)
    alp = list(alp)
    k = int(np.sqrt(len(key)))
    matr1 = [[i for i in range(k)] for i in range(k)]
    c = 1
    for i in range(k):
        for j in range(k):
            matr1[i][j] = c
            c += 1
    matr2 = np.rot90(matr1, k = 1, axes = (1, 0))
    matr3 = np.rot90(matr2, k = 1, axes = (1, 0))
    matr4 = np.rot90(matr3, k = 1, axes = (1, 0))
    bigmatr_n = [[0 for i in range(2*k)] for i in range(2*k)]
    for i in range(k):
        for j in range(k):
            bigmatr_n[i][j] = matr1[i][j]
            bigmatr_n[i][j + k] = matr2[i][j]
            bigmatr_n[i + k][j + k] = matr3[i][j]
            bigmatr_n[i + k][j] = matr4[i][j]
    bigmatr_l = [[' ' for i in range(2*k)] for i in range(2*k)]
    print(bigmatr_n)
    list1 = [i for i in range(1, k**2+1)]
    for i in list1:
        delete(bigmatr_n, i, k)
    print(bigmatr_n)
```

Figure 3: Шифрование с помощью решеток (2)

Шифрование с помощью решеток (2)

```
    list1 = [i for i in range(1, k**2+1)]
    for i in list1:
        delete(bigmatr_n, i, k)
    print(bigmatr_n)
    for i in range(4):
        for i in range(k*2):
            for j in range(k*2):
                if bigmatr_n[i][j] == bigmatr_l[i][j] and len(mes) > 0:
                    bigmatr_l[i][j] = mes[0]
                    mes = mes[1:]
        bigmatr_n = np.rot90(bigmatr_n, k = 1, axes = (1, 0))
    bigmatr_l.append(key)
    way = sorted(bigmatr_l[len(bigmatr_l) - 1])
    new_mes = []
    for i in way:
        for j in range(len(bigmatr_l)):
            if j == len(bigmatr_l) - 1:
                continue
            new_mes += bigmatr_l[j][bigmatr_l[len(bigmatr_l) - 1].index(i)]
    print(bigmatr_l)
    new_mes = ''.join(new_mes)
    return new_mes
```

Figure 4: Шифрование с помощью решеток (3)

Таблица Виженера

In [12]:

```
def vigenere(a, k, m):
    a = list(a)
    m = list(m)
    while len(k) < len(m):
        k += k
    k = k[:len(m)]
    k = list(k)
    new_a = []
    for i in range(len(a)):
        tmp = a[i:] + a[:i]
        new_a.append(tmp)
    new_a = np.array(new_a)
    new_m = []
    for i,j in zip(m, k):
        x = [idx for idx,q in enumerate(new_a[0,:]) if q == i][0]
        y = [idx for idx,q in enumerate(new_a[:,0]) if q == j][0]
        new_m += new_a[x,y]
    new_m = ''.join(new_m)
    return new_m
```

Figure 5: Таблица Виженера

Результаты (1)

- Маршрутное шифрование:

```
In [2]:
key = 'пароль'
message = 'нельзянедооцениватьпротивника'
alphabet = 'абвгдеёжзийклмнопрстуфхцчшщъыьэюя'
```

Figure 6: Входные данные маршрутное шифрование

```
In [4]:
new_message = path(alphabet, key, message)

In [5]:
print(message, "- сообщение")
print(new_message, "- зашифрованное собщение")

нельзянедооцениватьпротивника - сообщение
еенпнзоатаьовокннеьвлдирияцтив - зашифрованное собщение
```

Figure 7: Результат программы маршрутное шифрование

Результаты (2)

- Шифрование с помощью решеток:

```
In [6]:
key = 'шифр'
message = 'договорподписали'
alphabet = 'абвгдеёжзийклмнопрстуфхцчшщъыьэюя'
```

Figure 8: Входные данные шифрование с помощью решеток

```
In [9]:
new_message = lattice(alphabet, key, message)

[[1, 2, 3, 1], [3, 4, 4, 2], [2, 4, 4, 3], [1, 3, 2, 1]]
[[' ', 2, 3, 1], [' ', ' ', 4, 2], [' ', 4, 4, 3], [1, 3, 2, 1]]
[['д', 'в', 'о', 'р'], ['о', 'г', 'п', 'о'], ['о', 'с', 'д', 'п'], ['а', 'л', 'и', 'и'], ['ш', 'и', 'ф', 'р']]

In [10]:
print(message, "- сообщение")
print(new_message, "- зашифрованное собщение")

договорподписали - сообщение
вгслропиопдидооа - зашифрованное собщение
```

Figure 9: Результат программы шифрование с помощью решеток

Результаты (3)

- Таблица Виженера:

```
In [11]:
alphabet = 'абвгдежзийклмнопрстуфхцчшщыьэюя'
key = 'математика'
message = 'криптографиясерьезнаянаука'
```

Figure 10: Входные данные таблица Виженера

```
In [13]:
new_message = vigenere(alphabet, key, message)

In [14]:
print(message, "- сообщение")
print(new_message, "- зашифрованное собщение")

криптографиясерьезнаянаука - сообщение
црыфяохшкффядкэьчпчалнтшца - зашифрованное собщение
```

Figure 11: Результат программы таблица Виженера

Выводы

- Исходя из теоретических сведений, программы выполнены без ошибок, чему свидетельствуют полученные результаты.
- В ходе выполнения данной работы были выполнены поставленные цели и задачи.