

Лабораторная работа 1

Попов Дмитрий Павлович, НФИмд-01-23

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	9
4	Список литературы	10

List of Figures

2.1	cesar1	6
2.2	cesar2	7
2.3	cesar_out	7
2.4	atbash	8
2.5	atbash_out	8

List of Tables

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра математического моделирования и искусственного интеллекта

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

дисциплина: Математические основы защиты информации и информацион-
ной безопасности

Преподаватель: Кулябов Дмитрий Сергеевич

Студент: Попов Дмитрий Павлович

Группа: НФИмд-01-23

МОСКВА

2023 г.

1 Цель работы

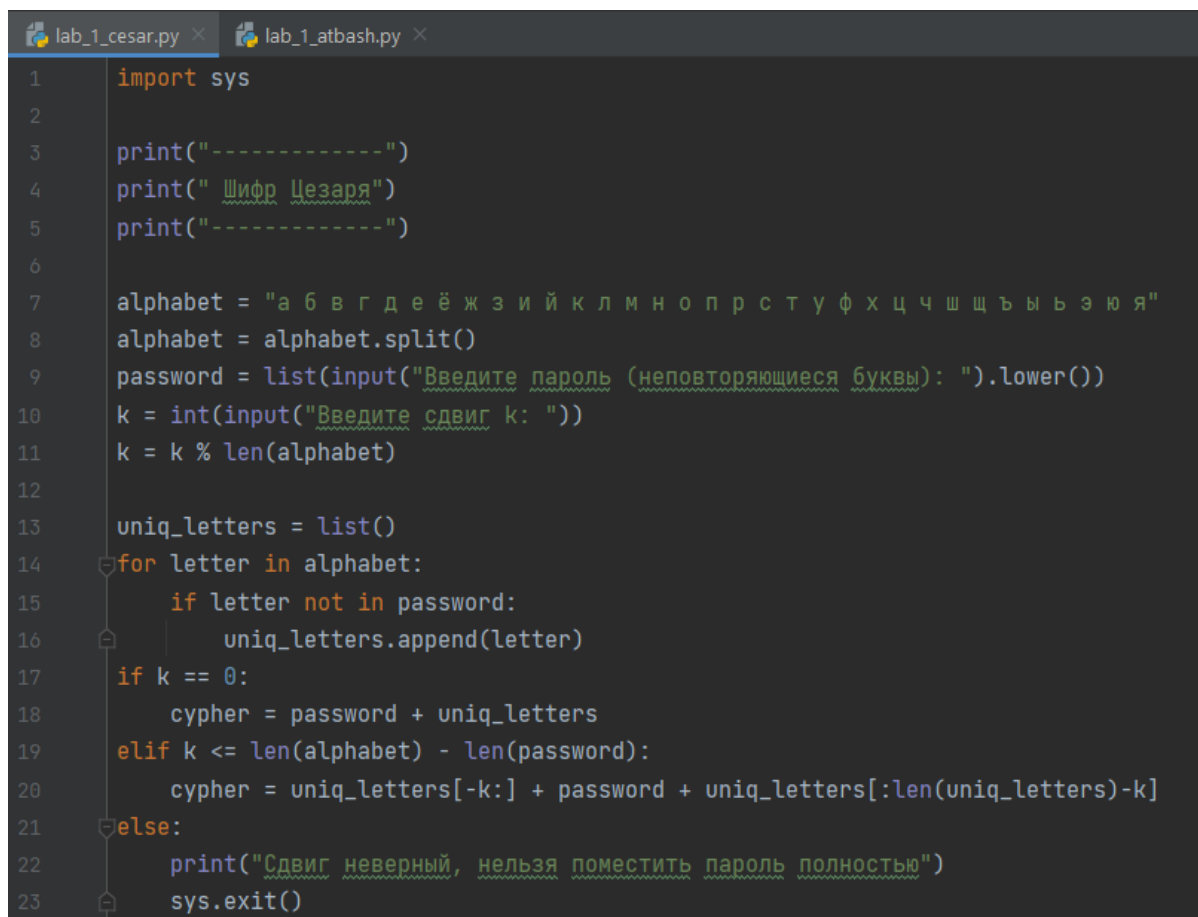
Целью данной работы является приобретение практических навыков шифрования простой замены.[1]

2 Выполнение лабораторной работы

Требуется реализовать шифр Цезаря с произвольным ключом k и Реализовать шифр Атбаш.

Для этого я реализовал две программы на языке Python

Первая программа для шифра Цезаря(fig. 2.1)(fig. 2.2).

The image shows a code editor with two tabs: 'lab_1_cesar.py' and 'lab_1_atbash.py'. The 'lab_1_cesar.py' tab is active, displaying a Python script for a Caesar cipher. The script imports the 'sys' module, prints a title 'Шифр Цезаря', and prompts the user for a password and a shift key 'k'. It then processes the password based on the shift key, either concatenating unique letters, rotating the alphabet, or printing an error if the shift is invalid. The script ends with 'sys.exit()'.

```
1 import sys
2
3 print("-----")
4 print(" Шифр Цезаря")
5 print("-----")
6
7 alphabet = "а б в г д е ё ж з и й к л м н о п р с т у ф х ц ч ш щ ъ ы ь э ю я"
8 alphabet = alphabet.split()
9 password = list(input("Введите пароль (неповторяющиеся буквы): ").lower())
10 k = int(input("Введите сдвиг k: "))
11 k = k % len(alphabet)
12
13 uniq_letters = list()
14 for letter in alphabet:
15     if letter not in password:
16         uniq_letters.append(letter)
17 if k == 0:
18     cypher = password + uniq_letters
19 elif k <= len(alphabet) - len(password):
20     cypher = uniq_letters[-k:] + password + uniq_letters[:len(uniq_letters)-k]
21 else:
22     print("Сдвиг неверный, нельзя поместить пароль полностью")
23 sys.exit()
```

Figure 2.1: cesar1

```

24
25 print("Таблица шифрования")
26 print(alphabet)
27 print(cypher)
28
29 while True:
30     mess = str(input("Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): "))
31     if mess == '0':
32         print("Выход из шифрования...")
33         break
34
35     cypher_mess = str()
36     for symbol in mess:
37         if symbol == ' ':
38             cypher_mess += ' '
39         else:
40             cypher_mess += cypher[alphabet.index(symbol)]
41
42     print("    Введенное предложение: ", mess)
43     print("    Зашированное предложение: ", cypher_mess)

```

Figure 2.2: cesar2

Затем я запустил программу, ввел пароль и сдвиг. Получил таблицу шифрования. Затем ввел предложение, которое нужно закодировать и получил зашифрованное сообщение. Вывод работы программы (fig. 2.3)

```

lab_1_cesar
C:\Users\79119\AppData\Local\Programs\Python\Python310\python.exe C:\Users\79119\PycharmProjects\Inform_security_labs\src\lab_1_cesar.py
-----
Шифр Цезаря
-----
Введите пароль (неповторяющиеся буквы): пароль
Введите сдвиг k: 3
Таблица шифрования
['a', 'b', 'v', 'g', 'd', 'e', 'b', 'j', 'z', 'n', 'a', 'k', 'l', 'm', 'n', 'o', 'p', 'r', 'c', 't', 'y', 'f', 'x', 'q', 'c', 'h', 'w', 's', 'b', 'y', 'b', 'z', 'e', 'a', 'я']
['m', 'z', 'ю', 'я', 'n', 'a', 'p', 'o', 'l', 'b', 'b', 'v', 'r', 'd', 'e', 'b', 'j', 'z', 'n', 'a', 'k', 'l', 'm', 'n', 'o', 'p', 'r', 'c', 't', 'y', 'f', 'x', 'q', 'c', 'h', 'w', 's', 'b', 'y']
Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): я люблю помидоры
Введенное предложение: я люблю помидоры
Зашированное предложение: ь гззгз жёдьпёзц
Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): 0
Выход из шифрования...
Process finished with exit code 0

```

Figure 2.3: cesar_out

Вторая программа для шифра Атбаш(fig. 2.4).

```
lab_1_cesar.py x lab_1_atbash.py x
1 print("-----")
2 print(" Шифр Атбаш")
3 print("-----")
4
5 alphabet = "а б в г д е ё ж з и й к л м н о п р с т у ф х ц ч щ ъ ы ъ э ю я"
6 alphabet = alphabet.split()
7 alphabet.append(' ')
8 cypher = alphabet.copy()
9 cypher.reverse()
10
11
12 print("Таблица шифрования")
13 print(alphabet)
14 print(cypher)
15
16 while True:
17     mess = str(input("Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): "))
18     if mess == '0':
19         print("Выход из шифрования...")
20         break
21
22     cypher_mess = str()
23     for symbol in mess:
24         cypher_mess += cypher[alphabet.index(symbol)]
25
26     print("    Введенное предложение: ", mess)
27     print("    Зашированное предложение: ", cypher_mess)
```

Figure 2.4: atbash

Вывод работы программы (fig. 2.5)

```
lab_1_atbash
C:\Users\79119\AppData\Local\Programs\Python\Python310\python.exe C:/Users/79119/PycharmProjects/Inform_security_Labs/src/Lab_1_atbash.py
-----
Шифр Атбаш
-----
Таблица шифрования
['а', 'б', 'в', 'г', 'д', 'е', 'ё', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с', 'т', 'у', 'ф', 'х', 'ц', 'ч', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я', ' ']
[' ', 'я', 'ю', 'э', 'ы', 'ь', 'щ', 'ш', 'ч', 'ц', 'х', 'ф', 'у', 'т', 'с', 'р', 'п', 'о', 'н', 'м', 'л', 'к', 'й', 'и', 'з', 'ж', 'ё', 'е', 'д', 'г', 'в', 'б', 'а']
Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): я люблю огурцы
    Введенное предложение: я люблю огурцы
    Зашированное предложение: бафвяфвасмпйе
Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): пароль плохой
    Введенное предложение: пароль плохой
    Зашированное предложение: р псфдарфсксц
Введите предложение, которое нужно зашифровать (0 - для завершения шифрования): 0
Выход из шифрования...

Process finished with exit code 0
```

Figure 2.5: atbash_out

3 Выводы

В результате выполнения работы я освоил на практике шифрование простой замены. Шифр Цезаря и Атбаш.

4 Список литературы

1. Методические материалы курса