

Отчёт по лабораторной работе №8

**Элементы криптографии. Шифрование (кодирование) различных
исходных текстов одним ключом**

Артамонов Тимофей Евгеньевич

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выводы	9

Список иллюстраций

3.1	Код	7
3.2	Каждую итерацию смотрим на то, что получилось и добавляем еще части к известному тексту	8

Список таблиц

1 Цель работы

Освоить на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.

2 Теоретическое введение

Два текста кодируются одним ключом (однократное гаммирование). Требуется не зная ключа и не стремясь его определить, прочесть оба текста. Необходимо разработать приложение, позволяющее шифровать и дешифровать тексты P_1 и P_2 в режиме однократного гаммирования. Приложение должно определить вид шифротекстов C_1 и C_2 обоих текстов P_1 и P_2 при известном ключе. Необходимо определить и выразить аналитически способ, при котором злоумышленник может прочесть оба текста, не зная ключа и не стремясь его определить.

3 Выполнение лабораторной работы

Напишем функции на python и зададим переменные. (рис. [3.1])

```
import random
import string
✓ 0.0s

def xor_text(text, key):
    result = ""
    for i in range(len(text)):
        result += chr(ord(text[i]) ^ ord(key[i]))
    return result
✓ 0.0s

def key_gen(text):
    key = ""
    for i in range(len(text)):
        key += random.choice(string.ascii_letters + string.digits)
    return key
✓ 0.0s

def part_key(fragment, encrypted):
    first = xor_text(fragment, encrypted[:len(fragment)])
    return first + key_gen(encrypted[len(fragment):])
✓ 0.0s

text = "Новый формат, друзья!"
fragment = "Новый"
```

Рис. 3.1: Код

Применим написанные функции для создания ключа, шифрования текста и дешифрования. (рис. [3.2])

```
fragment = "Всего"
c1, c2 = c1, c2
msg2 = fragment
l = len(msg2)
while l < len(P1):
    c12 = xor_text(c1[1:], c2[1:])
    msg1 = xor_text(c12, msg2)
    print("расшифровали\n", msg1 + c1[1:])
    print("введите продолжение")
    msg1 += input()
    l = len(msg1)
    print(msg1 + c1[1:])
    msg1, msg2 = msg2, msg1
    c1, c2 = c2, c1

[9] 42.9%
...
расшифровали
ВзапныйКфеОЗуК
введите продолжение
ВзапныйКфеОЗуК
расшифровали
ВсепныйКфеОЗуК
введите продолжение
ВсепныйКфеОЗуК
расшифровали
ВзапныйКфеОЗуК
введите продолжение
ВзапныйКфеОЗуК
расшифровали
ВсепныйКфеОЗуК
введите продолжение
ВсепныйКфеОЗуК
расшифровали
ВзапныйКфеОЗуК
введите продолжение
ВзапныйКфеОЗуК
расшифровали
ВсепныйКфеОЗуК
введите продолжение
ВсепныйКфеОЗуК
расшифровали
...
ВсепныйКфеОЗуК
расшифровали
ВзапныйКфеОЗуК
```

Рис. 3.2: Каждую итерацию смотрим на то, что получилось и добавляем еще части к известному тексту

4 Выводы

Освоили на практике применение режима однократного гаммирования на примере кодирования различных исходных текстов одним ключом.