Отчёт по лабораторной работе №3

Шифр гаммирования

Гердт Ольга НФИмд-02-21"

Содержание"

# Цель работы

Изучение алгоритма шифрования гаммированием

# Теоретические сведения

## Шифр гаммирования

Рассматрим шифры, который относятся к шифрам замены, но выделяются в собственный класс в связи со своими характерными свойствами и особенностями. Эти шифры получили название шифров гаммирования.

Гаммирование – это наложение (снятие) на открытые (зашифрованные) данные криптографической гаммы, т.е. последовательности элементов данных, вырабатываемых с помощью некоторого криптографического алгоритма, для получения зашифрованных (открытых) данных.

В алфавите любого естественного языка буквы следуют друг за другом в определенном порядке. Это дает возможность присвоить каждой букве алфавита ее естественный порядковый номер. Так, в английском алфавите букве A присваивается порядковый номер 1, букве Q - порядковый номер 17, а букве Z - порядковый номер 26. Аналогичное отождествление можно осуществить и для русского алфавита, например для RUS30 (где Ё=Е, Й=И, Ъ=Ь). Буква А будет иметь порядковый номер 1, О - номер 14, Я - 30. Если в открытом сообщении каждую букву заменить ее естественным порядковым номером в рассматриваемом алфавите, то преобразование числового сообщения в буквенное позволяет однозначно восстановить исходное открытое сообщение. Например, числовое сообщение

1 11 20 1 3 9 18

в алфавите RUS30 преобразуется в буквенное сообщение:

АЛФАВИТ

| А | Б | В | Г | Д | Е | Ё | Ж | З | И |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 6 | 7 | 8 | 9 |
| Й | К | Л | М | Н | О | П | Р | С | Т |
| 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
| У | Ф | Х | Ц | Ч | Ш | Щ | Ы | Ь | Ъ |
| 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 27 |
| Э | Ю | Я |  |  |  |  |  |  |  |
| 28 | 29 | 30 |  |  |  |  |  |  |  |

Зададим теперь преобразования зашифрования $f$ и преобразования расшифрования $g$ для произвольного шифра гаммирования. Пусть:

* необходимо зашифровать сообщение $X=x\_1,...x\_t$ в алфавите Ω={$a\_1,...a\_n$}.
* $n$ - мощность алфавита.
* Каждая буква отождествляется со своим порядковым номером в алфавите.
* Выберем некоторую последовательность, составленную из букв Ω:$y\_1,...,y\_t$ - данная последовательность называется гаммой шифра, или *ключевой последовательностью*.

Тогда преобразованием зашифрования $f\_{k\_i}$ будет являться преобразование, при котором $i$-ая буква шифртекста $y\_i$ равна:

$y\_i=f\_{k\_i}=r\_n(x\_i+y\_i),$

где $k\_i=y\_i$ - используемый знак гаммы последовательности для шифрования $i$ -той буквы сообщения $x\_i$; $r\_n(b)$ - остаток от деления числа $b$ на $n$ (полагаем, что $r\_n=n$). Итак, зашифрование шифром гаммирования означает «сложение» или, как говорят, «наложение» некоторой последовательности (гаммы) на знаки (буквы) открытого текста. Очевидно, что в таком случае для расшифрования нужно вычесть из букв шифртекста знаки гаммы:

$x\_i=g\_{k\_i}(y\_i)=r\_n(x\_i-y\_i),$

Соответственно, в силу сказанного, весь отрезок гаммы (то есть вся последовательность) является ключом данного шифра, именно поэтому ее называют ключевой последовательностью.

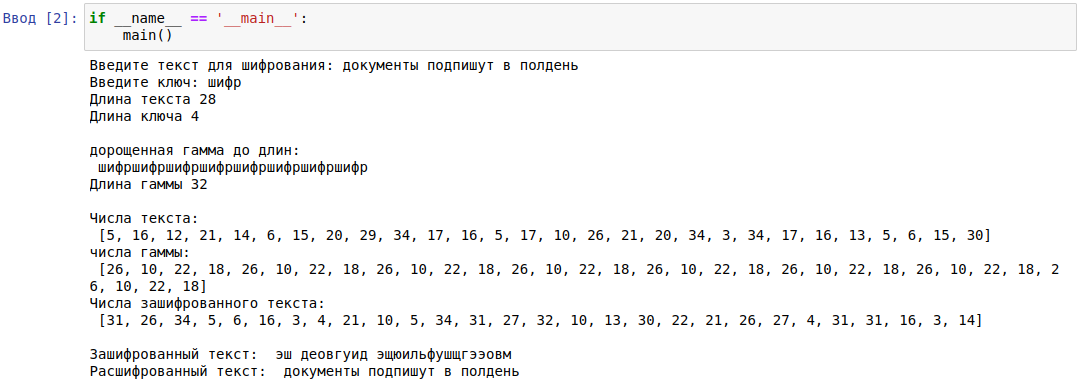
Принцип шифрования гаммированием заключается в генерации гаммы шифра с помощью датчика псевдослучайных чисел и наложении полученной гаммы шифра на открытые данные обратимым образом. Процесс дешифрования сводится к повторной генерации гаммы шифра при известном ключе и наложении такой же гаммы на зашифрованные данные.

# Выполнение работы

## Реализация шифратора и дешифратора Python

def main():
  
 #создаем алфавит
  
 dict = {"а" :1, "б" :2 , "в" :3 ,"г" :4 ,"д" :5 ,"е" :6 ,"ё" :7 ,"ж": 8, "з": 9, "и": 10, "й": 11, "к": 12, "л": 13,
  
 "м": 14, "н": 15, "о": 16, "п": 17,
  
 "р": 18, "с": 19, "т": 20, "у": 21, "ф": 22, "х": 23, "ц": 24, "ч": 25, "ш": 26, "щ": 27, "ъ": 28,
  
 "ы": 29, "ь": 30, "э": 31, "ю": 32, "я": 32
  
 }
  
 # меняем местами ключ и значение, такой словарь понадобится в будущем
  
 dict2 = {v: k for k, v in dict.items()}
  
 gamma = input("Введите гамму(на русском языке! Да и пробелы тоже нельзя! Короче, только символы из dict").lower()
  
 text = input("Введите текст для шифрования").lower()
  
 listofdigitsoftext = list() #сюда будем записывать числа букв из текста
  
 listofdigitsofgamma = list() #для гаммы
  
 #запишем числа в список
  
 for i in text:
  
 listofdigitsoftext.append(dict[i])
  
 print("Числа текста", listofdigitsoftext)
  
 #то же самое сделаем с гаммой
  
 for i in gamma:
  
 listofdigitsofgamma.append(dict[i])
  
 print("числа гаммы", listofdigitsofgamma)
  
 listofdigitsresult = list() #сюда будем записывать результат
  
 ch = 0
  
 for i in text:
  
 try:
  
 a = dict[i] + listofdigitsofgamma[ch]
  
 except:
  
 ch=0
  
 a = dict[i] + listofdigitsofgamma[ch]
  
 if a>=33:
  
 a = a%33
  
 ch+=1
  
 listofdigitsresult.append(a)
  
 print("Числа зашифрованного текста", listofdigitsresult)
  
 # теперь обратно числа представим в виде букв
  
 textencrypted=""
  
 for i in listofdigitsresult:
  
 textencrypted+=dict2[i]
  
 print("Зашифрованный текст: ", textencrypted)
  
 #теперь приступим к реализации алгоритма дешифровки
  
 listofdigits = list()
  
 for i in textencrypted:
  
 listofdigits.append(dict[i])
  
 ch = 0
  
 listofdigits1 = list()
  
 for i in listofdigits:
  
 a = i - listofdigitsofgamma[ch]
  
 #проблемы тут могут быть
  
 if a < 1:
  
 a = 33 + a
  
 listofdigits1.append(a)
  
 ch+=1
  
 textdecrypted = ""
  
 for i in listofdigits1:
  
 textdecrypted+=dict2[i]
  
 print("Decrypted text", textdecrypted)

## Контрольный пример



# Выводы

Изучили алгоритмы шифрования на основе гаммирования

# Список литературы{.unnumbered}

1. [Гаммирование](https://ru.wikipedia.org/wiki/%D0%93%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)
2. [Методы гаммирования](https://intuit.ru/studies/courses/691/547/lecture/12373?page=4)