Яндекс Репетитор

Богданов Дмитрий Валериевич,
ГБОУ «Школа № 1474 г. Москвы»
BogdanovDV1@edu.mos.ru

Криптография и криптоанализ

- **Криптография** – наука о методах обеспечения конфиденциальности, целостности данных и аутентификации
- **Криптоанализ** – наука о методах дешифровки зашифрованной информации без предназначенного для этого ключа
- **Симметричный шифр** – способ шифрования, в котором для шифрования и расшифрования используется один и тот же ключ

Теорема (основная криптографии)
Криптографическая стойкость шифра не превышает стойкости ключа

Задание №7 ОГЭ по информатике и ИКТ

(Источник: СтатГрад, Яндекс.Репетитор #Т5717)

Вася и Петя играли в шпионов и кодировали сообщение собственным шифром. Фрагмент кодовой таблицы приведён ниже:

Шифр симметричный?

Н	М	Л	И	Т	О
~	*	* @	@ ~ *	@ *	~ *

Определите, из скольких букв состоит сообщение, если известно, что буквы в нём не повторяются.

* @ @ ~ * * ~ * ~

Яндекс Репетитор

yandex.ru/tutor

Шифрование методом Цезаря

- **Шифр Цезаря** – один из первых и наиболее простых методов симметричного шифрования
- Разновидность **шифра подстановки**, в котором каждый символ в открытом тексте заменяется другим, отстоящем от него в алфавите на фиксированное число позиций

ВАСЯ → **ЗЕЦД**

Реализация шифра Цезаря

Как расшифровать?!

```
power = 32 # Без учёта буквы Ё


def caesar(source, key=1):
    global power
    cypher = ''.join([chr(ord('A') + (ord(c) - ord('A') + key) % power)
                      for c in source.upper()])
    return cypher


print('\ncypher:', caesar(input('source: '),
```

**kwargs

```
if 'mode' in kwargs and kwargs['mode'] == 'decrypt':
    key = power – key % power
```

```
source: Информатика
key: 20

cypher: ЬБИВДАФЖЬЮФ
```

Криптоанализ шифра Цезаря

Brute Force

```
def caesar_attack(cypher):
    global power
    import pymorphy2
    morph = pymorphy2.MorphAnalyzer()
    for key in range(power):
        source = caesar(cypher, key, mode='decrypt')
        print(key, source)
        if morph.word_is_known(source):
            m = morph.parse(source)[0]
            if m.tag.POS == 'NOUN' and \
               m.inflect({'sing', 'nomn'}).word.upper() == source:
                return source, key
    return 'None', -1
```

Сколько возможных ключей?

Морфологический анализ

5

ЗЕЦД → **ВАСЯ**

```
source, key = caesar_attack(input())
print(f'\nsource: {source}\nkey: {key}')
```

Python3 Codes

ЗАПРОГРАММИРОВАТЬ ДОМА!

Шифр простой замены

```
from random import shuffle

alphabet = [chr(x) for x in range(ord('А'), ord('Я') + 1)]
shuffle(alphabet)
print(*alphabet)
```

Перемешивание списка

Что делает программа?

Сколько возможных ключей?

Ключом служит перемешанный случайным образом алфавит. При шифровании первая буква алфавита замещается первой буквой ключа и т.д.

Криптоанализ простых замен

Что такое словарь (dict) в Python?

Напишите программу, определяющую частоты вхождения в текст отдельных букв, морфем и слов.

Всегда ли такие частоты?

- **Поиск восхождения к вершине** – пошаговая оптимизация коэффициента, характеризующего вероятность принадлежности текста к естественному языку
- **Частотный анализ** – распределение букв в криптотексте сравнивается с распределением букв в алфавите исходного сообщения

Идеальный шифр

- **Криптографические алгоритмы** направлены на изменение частот букв в криптосообщении
- **Блочные замены** – одно из возможных направлений развития рассмотренных алгоритмов
- **Идеальный шифр** полностью скрывает в криптотексте все статистические закономерности открытого текста

Какое физическое явление?

БЕЛЫЙ ШУМ

В чём сложность реализации?

Используемые источники

- **Фомичев В.М.** Дискретная математика и криптография
- **Цыганов А.В.** Криптография и криптоанализ
- **НОУ ИНТУИТ** Методы криптоанализа https://www.intuit.ru/studies/courses/600/456/lecture/10198
- **Морфологический анализатор Pymorphy2** Официальная документация https://pymorphy2.readthedocs.io