Московский авиационный институт   
(государственный технический университет)   
  
Факультет прикладной математики   
  
Кафедра вычислительной математики и программирования

Лабораторная работа №4 по курсу

«Криптография»:

8 факультет, 3 курс

Студент: Стрыгин Д.Д.

Группа: М8О-306Б-19, №25

Преподаватель: Борисов А.В.

**Задача:**

Сравнить

* два осмысленных текста на естественном языке
* осмысленный текст и текст из случайных букв
* осмысленный текст и текст из случайных слов
* два текста из случайных букв
* два текста из случайных слов

Как сравнивать: считать процент совпадения букв в сравниваемых текстах – получить дробное значение от 0 до 1 как результат деления количества совпадений на общее число букв. Расписать подробно в отчёте алгоритм сравнения и приложить сравниваемые тексты в отчёте хотя бы для одного запуска по всем пяти подпунктам. Осознать какие значения получаются в этих пяти подпунктах. Привести свои соображения о том почему так происходит. Длина сравниваемых текстов должна совпадать. Привести соображения о том какой длины текста должно быть достаточно для корректного сравнения.

**Описание работы:**

Для решения поставленной задачи я взял два текста: “Оно” и “Граф Монте-Кристо” на английском языке. Затем, с помощью написанной функции препроцессинга над текстами, я удалил из текста лишние знаки препинания и знаки “\n”. После чего сгенерировал два текста из букв с пробелом и переводом на другую строку и два текста из 1000 самых популярных английских слов, также с пробелом и переводом на другую строку.

После препроцесса над текстами, я преступил к сравнению букв в тексте. После прохода по всем буквам текста (сложность алгоритма O(n)) и вывожу результат с стандартный поток вывода терминала.

**Листинг программы:**

import sys

import random

import string

letters = "0123456789 \n"

def errorMessage():

    print("Использование:\n"

    "\tpython3 {0} name\_of\_text\_file1 name\_of\_text\_file2 name\_of\_dictunary".format(sys.argv[0]))

    quit()

def preprocess\_of\_text(nameText, count\_of\_text):

    try:

        text = open(nameText, 'r')

    except FileNotFoundError:

        print("Файл(ы) не найден(ы). Исключение типа FileNotFoundError.")

        quit()

    text2 = open("text" + str(count\_of\_text) + ".txt", 'w')

    for line in text:

        out\_line = ""

        for letter in line:

            if line[0] == "\n":

                continue

            if letter.isalpha() or (letter in letters):

                out\_line += letter

        text2.write(out\_line)

    text2.close()

    text.close()

def equalazing\_texts():

    out\_str = ""

    text1 = open("text1.txt", 'r')

    data = text1.read()

    length\_of\_text1 = len(data)

    text2 = open("text2.txt", 'r')

    data = text2.read()

    length\_of\_text2 = len(data)

    if length\_of\_text1 < length\_of\_text2:

        text2.close()

        text2 = open("text2.txt", "w")

        for i in range(length\_of\_text1):

            text2.write(data[i])

        length\_of\_texts = length\_of\_text1

    elif length\_of\_text1 > length\_of\_text2:

        text1.close()

        text1 = open("text1.txt", "r")

        data = text1.read()

        text1.close()

        text1 = open("text1.txt", "w")

        for i in range(length\_of\_text2):

            text1.write(data[i])

        length\_of\_texts = length\_of\_text2

    text1.close()

    text2.close()

    return length\_of\_texts

def preprocess\_of\_letters(length\_of\_texts, n):

    rand\_text = open("text" + str(n) + ".txt", "w")

    ltrs = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789 \n'

    text = ''.join(random.choice(ltrs) for i in range(length\_of\_texts))

    rand\_text.write(text)

    rand\_text.close()

def preprocess\_of\_text\_of\_word(length\_of\_texts, n):

    spnl = [" ", "\n"]

    text = open(sys.argv[3], "r")

    data = text.read()

    data = data.split("\n")

    text2 = open("text" + str(n) + ".txt", "w")

    word\_text = ""

    while(len(word\_text) != length\_of\_texts):

        word = random.choice(data)

        word += random.choice(spnl)

        if len(word\_text) + len(word) < length\_of\_texts:

            word\_text += word

        else:

            for i in range(length\_of\_texts - len(word\_text)):

                word\_text += word[i]

    text2.write(word\_text)

    text2.close()

def preprocess():

    if len(sys.argv) != 4:

        errorMessage()

    count\_of\_text = 1

    preprocess\_of\_text(sys.argv[1], count\_of\_text)

    count\_of\_text += 1

    preprocess\_of\_text(sys.argv[2], count\_of\_text)

    length\_of\_texts = equalazing\_texts()

    print("Длина текстов - " + str(length\_of\_texts))

    preprocess\_of\_letters(length\_of\_texts, 3)

    preprocess\_of\_letters(length\_of\_texts, 4)

    preprocess\_of\_text\_of\_word(length\_of\_texts, 5)

    preprocess\_of\_text\_of\_word(length\_of\_texts, 6)

def compare(text1, text2):

    file1 = open(text1, "r")

    file2 = open(text2, "r")

    data1 = file1.read()

    data2 = file2.read()

    size = len(data1)

    suc = 0

    for i in range(size):

        if data1[i] == data2[i]:

            suc += 1

    k = suc / size \* 100

    file1.close()

    file2.close()

    return(round(k, 2))

def action():

    k = compare("text1.txt", "text2.txt")

    print("Процент совпадения двух осмысленных текстов - " + str(k) + "%")

    k = compare("text2.txt", "text3.txt")

    print("Процент совпадения осмысленного текста и из рандомных букв - " + str(k) + "%")

    k = compare("text1.txt", "text5.txt")

    print("Процент совпадения осмысленного текста и из рандомных слов - " + str(k) + "%")

    k = compare("text3.txt", "text4.txt")

    print("Процент совпадения текста из рандомных букв и из рандомных букв - " + str(k) + "%")

    k = compare("text5.txt", "text6.txt")

    print("Процент совпадения текста из рандомных слов и из рандомных слов - " + str(k) + "%")

def main():

    preprocess()

    action()

if \_\_name\_\_ == "\_\_main\_\_":

    main()

**Пример входных данных:**

**Часть от текста “Оно”:**

CHAPTER 1

After the Flood 1957

1

The terror which would not end for another twentyeight yearsif it ever did endbegan so far as I know or can tell with a boat made from a sheet of newspaper floating down a gutter swollen with rain

The boat bobbed listed righted itself again dived bravely through treacherous whirlpools and continued on its way down Witcham Street toward the traffic light which marked the intersection of Witcham and Jackson The three vertical lenses on all sides of the traffic light were dark this afternoon in the fall of 1957 and the houses were all dark too There had been steady rain for a week now and two days ago the winds had come as well Most sections of Derry had lost their power then and it was not back on yet

A small boy in a yellow slicker and red galoshes ran cheerfully along beside the newspaper boat The rain had not stopped but it was finally slackening It tapped on the yellow hood of the boys slicker sounding to his ears like rain on a shed roof a comfortable almost cozy sound The boy in the yellow slicker was George Denbrough He was six His brother William known to most of the kids at Derry Elementary School and even to the teachers who would never have used the nickname to his face as Stuttering Bill was at home hacking out the last of a nasty case of influenza In that autumn of 1957 eight months before the real horrors began and twentyeight years before the final showdown Stuttering Bill was ten years old

**Часть сгенерированного текста из букв:**

smx9cJjXFD8Y

orTQx2cFdVM7rAZ4Ajo36h7Bd6Xgzp3TO1ivQtQ4bgYv 59xTQzCptDotUpZFF0UFhFSCxg3Ra2zil

9BgryN

g7XnwqjA

**Часть сгенерированного текста из слов:**

usually sell film show

simply analysis chair court

figure

where

answer

support program it

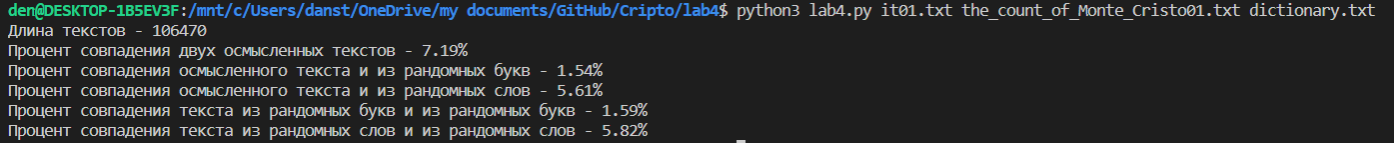
hit

nice

five

significant election

**Результат работы программы:**

****

**Выводы:**

После тестирования моей программы на различных входных данных и различном их количестве, я пришел к тому, что с увеличением количества символов текста процент совпадения букв естественно падает, однако лишь до определённого значения. К примеру: рандомно сгенерированный текст из букв при увеличении числа символом, уже после 1000 символов имеет процент совпадения с таким же случайным текстом из таких же букв – 1,5%, и это значение почти не изменяется.

Ну и проанализировав результаты работы программы, наглядно видно, что процент совпадения осмысленных текстов всё же выше, чем проценты совпадений остальных тестов, от сюда можно сделать вывод, что последовательность букв в осмысленных текстах, имеющая закономерности, будет требовать те алгоритмы шифрования, которые будут её максимально маскировать.