Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информационных систем и технологий

**Лабораторная работа № 15**

«ИССЛЕДОВАНИЕ МЕТОДОВ ТЕКСТОВОЙ СТЕГАНОГРАФИИ»

Выполнил:

Студент: Лысков К.Е.

ФИТ 3 курс 4 группа

Преподаватель: Сазонова Д.В.

Минск 2023

1. **Описание приложения**

Приложение написано на языке программирования Python, в котором реализован один из методов текстовой стеганографии на основе модификации пространственно-геометрических параметров текста-контейнера. В данном случае было реализовано изменение длины строки, модификация апроша.

1. **Методика выполнения расчетов**

В данной лабораторной работе была поставлена цель создания приложения, которое изменение длины строки, модификация апроша.. На листинге 2.1 представлены функции, реализующие данную функциональность.

from support\_function import read\_file

import docx

def ModificationOfTheStringLength(message: str, container: str):

    message\_binary = ''.join([bin(ord(i))[2:].rjust(8, '0') for i in message])

*# Вставка в контейнер*

    list\_container = [i for i in container.split()]

    container\_modify = ' '.join([list\_container[i] + ' ' if message\_binary[i] == '1' else list\_container[i]

                                 for i in range(len(message\_binary))] + list\_container[len(message\_binary):])

    print(f'----- Вставка в контейнер -----\n'

          f'Сообщение:  {message}\n'

          f'Сообщение(2): {message\_binary}\n'

          f'Контейнер:\n{container}\n'

          f'Модифицированное сообщение:\n{container\_modify}\n')

*# Запись в docx*

    doc\_modify = docx.Document()

    doc\_modify.add\_paragraph(container\_modify)

    doc\_modify.save('result.docx')

*# Извлечение из контейнера*

    doc\_modify = docx.Document('result.docx')

    for paragraph\_ in doc\_modify.paragraphs:

        for run in paragraph\_.runs:

            extracted\_container = run.text

    list\_extracted\_container = ''.join(['1' if i == '' else '0'

                                        for i in extracted\_container.split(' ')]).replace('01', '1')

    extracted\_message = ''.join([chr(int(list\_extracted\_container[i \* 8: (i + 1) \* 8], 2))

                                 for i in range(len(list\_extracted\_container) // 8)]).rstrip('\x00')

    print(f'----- Извлечение из контейнера -----\n'

          f'Извлечённый контейнер: {extracted\_container}\n'

          f'Извлечённое сообщение: {extracted\_message}\n')

def main():

    container = read\_file('text.txt', -1)

    message = 'Dashchinskii Maksim Leonidovich'

    print(f'----- Изменение длины строки -----')

    ModificationOfTheStringLength(message, container)

if \_\_name\_\_ == '\_\_main\_\_':

    main()

def read\_file(name\_file: str, size\_buffer: int):

    file\_for\_only\_read = open(name\_file, 'r', encoding="utf8")

    return file\_for\_only\_read.read() if size\_buffer <= -1 else file\_for\_only\_read.read()[:size\_buffer]

def write\_file(name\_file: str, message: str, size\_buffer: int):

    file\_for\_only\_write = open(name\_file, 'w', encoding="utf8")

    if size\_buffer <= -1:

        file\_for\_only\_write.write(message)

    else:

        file\_for\_only\_write.write(message[:size\_buffer])

Листинг 2.1 –код программы, реализующие заданную ранее функциональность

**3. Результаты работы приложения**

Для выполнения расчетов достаточно необходимо запустить приложение. Рисунок 3.1 показывают требуемые в данной лабораторной работе результаты. Также будут созданые необходимые файлы во время работы программы.

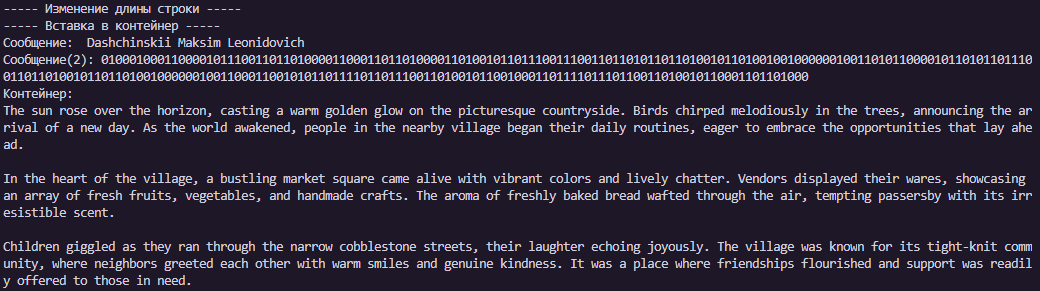


Рисунок 3.1 – Результат работы

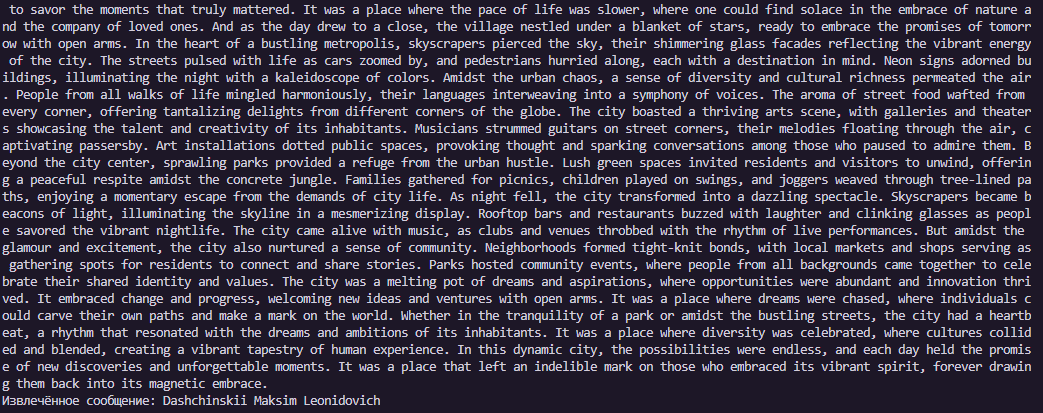


Рисунок 3.2 – Результат работы

**Вывод**

В ходе лабораторной работы было разработано приложение для реализации одного из методов текстовой стеганографии на основе модификации пространственно-геометрических параметров текста-контейнера.