Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Пермский национальный исследовательский политехнический университет»

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Направление 09.03.01 – «Информатика и вычислительная техника»

Дисциплина: «Защита информации»

Профиль: «Программная инженерия»

Семестр 5

ОТЧЕТ

по лабораторной работе №1

Тема: «Шифры перестановки и замены»

Вариант 7

Выполнил: студент группы РИС-23-1б

Гордеев В. А. \_\_\_\_\_\_\_\_\_\_\_

Проверил: доцент кафедры ИТАС

Шереметьев В. Г. \_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_

Пермь, 2025

**ЦЕЛЬ РАБОТЫ**

Получить практические навыки по применению шифров перестановки и шифров простой замены.

**ЗАДАНИЕ**

Реализовать шифрование текстового сообщения, используя систему шифрования Полибианский квадарат.

**ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

**Полибианский квадрат**

Одним из первых шифров простой замены считается так называемый *полибианский квадрат.* За два века до нашей эры греческий писатель и историк Полибий изобрел для целей шифрования квадратную таблицу размером 5х5, заполненную буквами греческого алфавита в случайном порядке.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| q | p | w | o | e |
| i | r | u | t | y |
| z | m | x | n | c |
| b | v | a | l | s |
| k | d | j | f | h |

Полибианский квадрат, заполненный случайным образом буквами латинского алфавита и пробелом

При шифровании в этом полибианском квадрате находили очередную букву открытого текста и записывали в шифртекст букву, расположенную ниже ее в том же столбце. Если буква текста оказывалась в нижней строке таблицы, то для шифртекста брали самую верхнюю букву из того же столбца. Например, для слова

square

получается шифртекст

hixjmy

Концепция полибианского квадрата оказалась плодотворной и нашла применение в криптосистемах последующего времени.

**ХОД РАБОТЫ**

На рисунке 1 представлены входные данные и полибианский квадрат

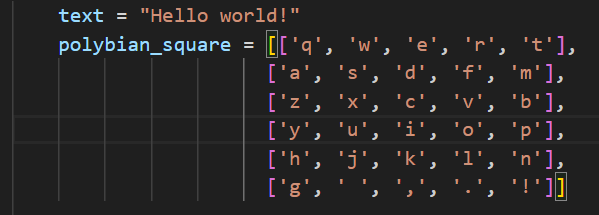


Рисунок 1 – Входные данные

Пример работы программы представлен на рисунке 2.

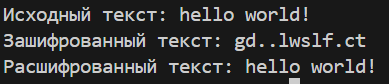


Рисунок 2 – Пример работы программы.

**ПРИЛОЖЕНИЕ А**

**Листинг программы**

import string

def text\_decryption(text: str, polybian\_square) -> str:

new\_text = ""

for symbol in text:

for i in range(len(polybian\_square)):

for j in range(len(polybian\_square[0])):

if symbol == polybian\_square[i][j]:

new\_text += polybian\_square[(i-1)%len(polybian\_square)][j]

return new\_text

def text\_encryption(text: str, polybian\_square) -> str:

new\_text = ""

for symbol in text:

for i in range(len(polybian\_square)):

for j in range(len(polybian\_square[0])):

if symbol == polybian\_square[i][j]:

new\_text += polybian\_square[(i+1)%len(polybian\_square)][j]

return new\_text

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

text = "Hello world!"

polybian\_square = [['q', 'w', 'e', 'r', 't'],

['a', 's', 'd', 'f', 'm'],

['z', 'x', 'c', 'v', 'b'],

['y', 'u', 'i', 'o', 'p'],

['h', 'j', 'k', 'l', 'n'],

['g', ' ', ',', '.', '!']]

# Первичная обработка текста

simbols = string.ascii\_letters + " "+ "," + "." + "!"

text = "".join(x for x in text if x in (simbols)).lower().strip()

print("Исходный текст: " + text)

text = text\_encryption(text, polybian\_square)

print("Зашифрованный текст: " + text)

text = text\_decryption(text, polybian\_square)

print("Расшифрованный текст: " + text)