**Лабораторная №4**

**Выполнил: Ягодяк Павел Сергеевич АИСа-020303-о24**

**Поиск по тексту с использованием частотного анализа**

**Цель:**

Изучить методы обработки текстов и реализовать программу на Python, которая позволяет пользователю выполнять поиск слов в большом текстовом файле с использованием частотного анализа. Научиться работать с библиотеками для обработки строк и коллекциями для подсчета частоты слов.

**Задание:**

1. Загрузить текстовый файл с произведением Л.Н. Толстого «Война и мир».
2. Реализовать функционал, позволяющий пользователю вводить запрос, состоящий из не менее чем 3 символов.
3. Найти и отобразить первые 20 слов, содержащих введенное пользователем слово, отсортированных по частоте их вхождения в текст.
4. Обеспечить, чтобы время на начальную обработку текста не превышало 1 минуты, а время обработки каждого запроса не превышало 2 секунд.

**Листинг программы 1:**

print("Ягодяк Павел Сергеевич АИСа-020303-о24")

import re

from collections import Counter

def load\_text(file\_path):

    with open(file\_path, 'r', encoding='utf-8') as file:

        text = file.read()

    return text

def preprocess\_text(text):

    # Удаляем знаки препинания и переводим текст в нижний регистр

    text = re.sub(r'[^\w\s]', '', text.lower())

    words = text.split()

    return words

def build\_frequency(words):

    return Counter(words)

def search\_words(frequency, query):

    # Находим слова, содержащие запрос

    matching\_words = {word: freq for word, freq in frequency.items() if query in word}

    # Сортируем по частоте и берем первые 20

    sorted\_words = sorted(matching\_words.items(), key=lambda item: item[1], reverse=True)

    return sorted\_words[:20]

def main():

    file\_path = 'voina\_i\_mir.txt'  # Путь к файлу с текстом

    text = load\_text(file\_path)

    words = preprocess\_text(text)

    frequency = build\_frequency(words)

    while True:

        query = input("Введите запрос (не менее 3 символов): ")

        if len(query) < 3:

            print("Запрос должен содержать не менее 3 символов.")

            continue

        results = search\_words(frequency, query)

        print("Найденные слова:")

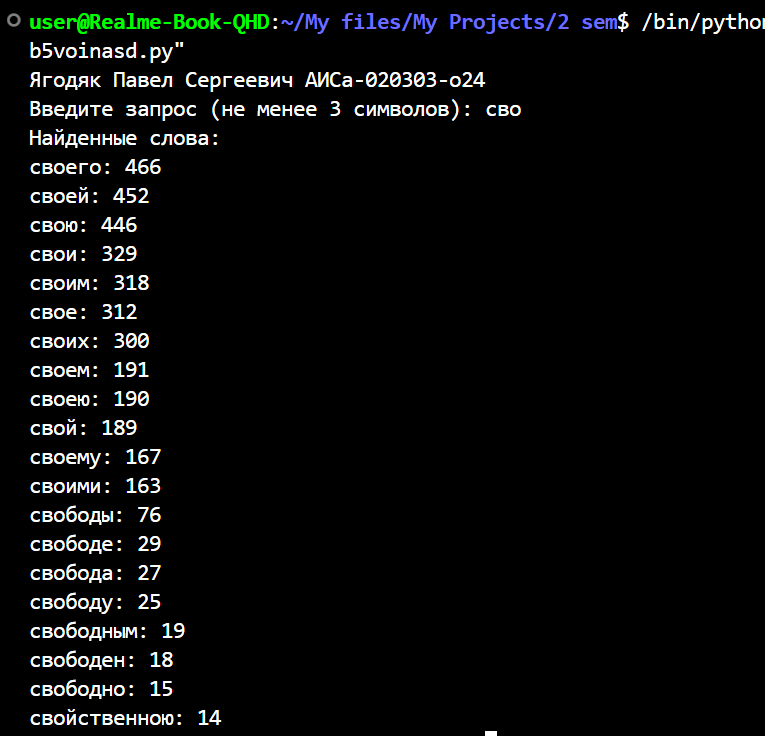
        for word, freq in results:

            print(f"{word}: {freq}")

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

    main()

**Результат выполнения программы:**



**Вывод:**

В результате выполнения лабораторной работы была разработана программа, которая эффективно обрабатывает текст и позволяет выполнять поиск по нему. Программа демонстрирует использование стандартных библиотек Python для работы с текстом и анализом данных. Реализованный алгоритм обеспечивает быструю обработку запросов, что делает его эффективным даже для больших объемов текста. Полученные результаты показывают, что частотный анализ может быть полезным инструментом для поиска информации в текстах.