**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики - процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**ОТЧЕТ**

**по лабораторной работе №4**

**по дисциплине «Функциональное программирование»**

**на тему «Разработка рекомендательной системы для выбора книг»**

**Студент гр. 23Б15-пу**

**Кубякин Н.А.**

**Преподаватель**

**Киямов Ж. У.**

**Санкт-Петербург**

**2024 г.**

***Оглавление***

[1. Цель работы 3](#_Toc178797423)

[2. Описание задачи (формализация задачи) 3](#_Toc178797424)

[3. Теоретическая часть 4](#_Toc178797431)

[4. Основные шаги программы 4](#_Toc178797442)

[5. Описание программы 6](#_Toc178797442)

[6. Рекомендации пользователя 7](#_Toc178797443)

[7. Рекомендации программиста 7](#_Toc178797444)

[8. Исходный код программы 7](#_Toc178797445)

[9. Контрольный пример 8](#_Toc178797447)

[10. Вывод 9](#_Toc178797447)

[11. Источники 9](#_Toc178797448)

# Цель работы

# Создать рекомендательную систему книг, которая сможет подобрать книги по заданным предпочтениям по жанрам, авторам, годам выпуска. Программа должна анализировать заранее заданный список книг и выводить результаты в графический интерфейс и/или отдельный файл.

# Описание задачи (формализация задачи)

Система должна:

1. Принимать входные данные, включая:
   * Любимые жанры
   * Любимых авторов
   * Ключевые слова
2. Иметь базу данных книг, представленную в формате JSON или CSV,:
   * Название книги
   * Автор книги
   * Жанр книги
   * Краткое описание
   * Год публикации
3. Осуществлять обработку предпочтений пользователя:
   * Формировать словарь предпочтений, основанный на введенных пользователем данных.
   * Сравнивать введенные предпочтения с метаданными в базе данных и оценивать степень соответствия каждого элемента.
4. Применять алгоритм рекомендаций:
   * Для каждой книги из базы данных вычислять рейтинг соответствия.
   * Сортировать книги по убыванию рейтинга.
5. Реализовать фильтры и сортировку:
   * Пользователь может фильтровать рекомендации по:
     + Жанру
     + Году публикации
     + Автору
   * Пользователь может сортировать список книг по:
     + Рейтингу соответствия
     + Алфавиту
     + Году публикации

# Теоретическая часть

#### **Система рекомендаций:**

В данной задаче мы используем **контентную фильтрацию**, где рекомендации основаны на характеристиках книг, таких как жанр, автор, ключевые слова и другие метаданные.

#### **Представление предпочтений пользователя:**

* **Жанры книг**, которые пользователю интересны
* **Авторы**, чьи книги пользователь предпочитает.
* **Ключевые слова** из описаний книг, которые соответствуют интересам пользователя.

#### **Метод расчёта оценки соответствия:**

* **Жанр** книги имеет высокий вес, так как часто жанр является основным критерием.
* **Автор** имеет средний вес, поскольку некоторые пользователи предпочитают книги определённых авторов, но не ограничиваются в вариантах.
* **Ключевые слова** – это более тонкая настройка, которая позволяет учитывать более специфические интересы пользователя.

Каждое из этих полей анализируется, и, если оно совпадает с предпочтениями пользователя, добавляется соответствующая сумма баллов в общий результат.

# 

# Основные шаги программы

#### 1. **Загрузка и обработка данных о книгах**

Для каждой книги собираются следующие характеристики:

* Название книги
* Автор
* Жанр
* Ключевые слова
* Год публикации

#### 2**. Получение пользовательских предпочтений**

Предпочтения пользователя могут включать:

* **Жанры**
* **Авторы**
* **Ключевые слова**

#### 3. **Оценка соответствия книг предпочтениям**

* **Проверка жанра**
* **Проверка автора**
* **Проверка ключевых слов**

#### 4. **Фильтрация и сортировка книг**

* **Фильтрация**: Книги, которые не соответствуют предпочтениям пользователя, исключаются из списка.
* **Сортировка**: Оставшиеся книги сортируются по убыванию оценки соответствия.

Сортировка может включать дополнительные критерии, такие как год публикации или алфавитный порядок.

#### 5. **Предоставление результатов пользователю**

После того как список книг отсортирован, программа выводит результаты пользователю. Это может быть выполнено через интерфейс, в котором отображаются названия книг, их авторы, жанры, а также оценка соответствия.

#### 6. **Сохранение результатов**

Если пользователь хочет сохранить результат для дальнейшего использования, программа предоставляет возможность экспортировать список рекомендованных книг в формат json.

# Описание программы

Описание функций программы представлено в таблице 5.

*Таблица 5. Описание функций*

|  |  |  |
| --- | --- | --- |
| Имя | Параметры | Описание |
| load\_books | filepath: Строка содержащая путь к файлу с книгами | Метод для загрузки списка книг |
| process\_preferences | genres: Строка, содержащая жанр.  authors: Строка, содержащая автора.  keywords: Строка, содержащая ключевые слова. | Метод для сбора информации из списка книг. |
| calculate\_score | book: Название книги.  preferences: Предпочтения пользователя. | Метод для рассылки вычисления рейтинга книги. |
| get\_recommendations | preferences: Предпочтения пользователя.  year\_filter: Год, меньше которого книги не показываются.  sort\_by: Ключ сортировки | Метод для поиска рекомендаций на основе предпочтений пользователя. |
| save\_selected\_recomendations | - | Метод сохранения выбранных книг в json файл. |

# Рекомендации пользователя

Для использования программы:

1. Выбор жанров – введите жанры через запятую.
2. Выбор годов публикации – введите год, книги старше которого не будут   
   отображаться.
3. Выбор авторов – введите авторов через запятую
4. Указание ключевых слов – введите слова через запятую
5. Сортировка книг – выберите желаемый ключ сортировки.
6. Получение рекомендаций – рекомендации выведутся в интерфейс.
7. Сохранение списка книг – выберите номера желаемых книг.

# Рекомендации программиста

# Для запуска программы необходим Python, а также 64-битная операционная система Windows, или Linux, или macOS.

# Библиотеки: tkinter [[1]](#tkinter), json [[2]](#asyncio).

# Исходный код программы

# Исходный код программы и необходимые текстовые файлы доступны по ссылке: <https://github.com/NikiTaku1/spbu_funcprog/tree/main/lab4>

# Контрольный пример

В данном разделе представлены контрольные примеры работы программы.

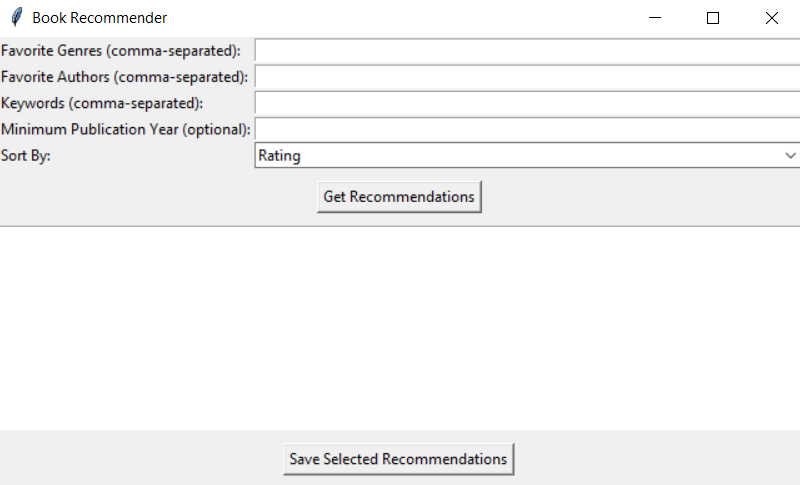


Рис. 9.1 Окно интерфейса

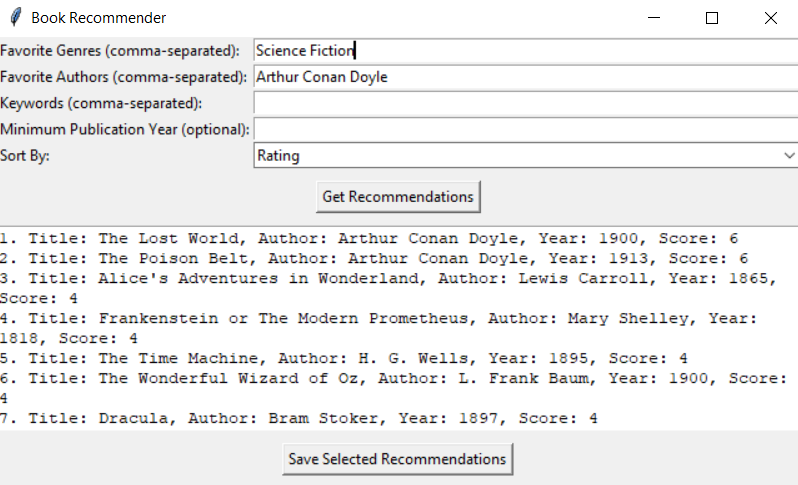


Рис. 9.2 Вывод результатов

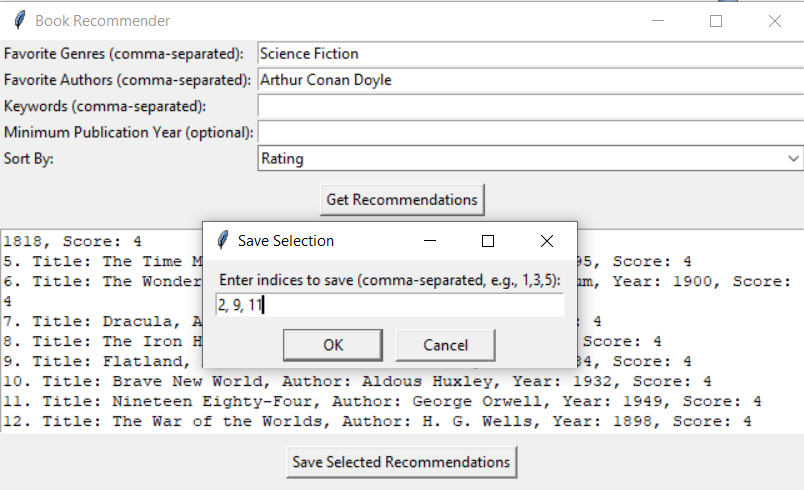


Рис. 9.3 Выбор книг для сохранения

# Вывод

# В ходе разработки была успешно создана программа для предоставления рекомендаций книг. Программа имеет графический интерфейс и умеет рекомендовать книги основываясь на предпочтении в жанре, авторе и ключевых словах в описаниях книг. Есть возможность сохранить рекомендации в отдельный файл для дальнейшего использования.

# Источники

# 1. tkinter — tkinter documentation // Documentation URL: <https://docs.python.org/3/library/tkinter.html> (дата обращения: 15.12.2024).

# 2. json — json documentation // Documentation URL: <https://docs.python.org/3/library/json.html> (дата обращения: 15.12.2024).