

Темы, необходимые для выполнения задачи "Трекер книг":

1. **Структуры (struct)**
→ для хранения информации о книге в одном объекте.
2. **Массивы структур**
→ чтобы хранить много книг (`Book books[1000]`).
3. **Циклы (for, while)**
→ для перебора всех книг при выводе и подсчёте статистики.
4. **Ветвление (if, switch)**
→ для выбора действий в меню и проверки условий.
5. **Функция main()**
→ как точка входа и главный цикл управления программой.
6. **Функции**
→ разбиение кода на логические части: `addBook()`, `showAllBooks()`, `showStats()` и т.д.
7. **Работа с файлами (fstream)**
→ загрузка и сохранение данных (`ifstream`, `ofstream`).
8. **Строки как массивы символов (char[])**
→ для хранения названия книги и автора (без `string!`).
9. **Форматированный вывод (iomanip)**
→ красивый вывод таблиц: `setw`, `left`, `fixed`, `setprecision`.
10. **Базовый ввод/вывод (cin, cout)**
→ для общения с пользователем через консоль.

Задание

"Трекер прочитанных книг" — консольная программа на C++.

Создайте программу, в которой пользователь может вести учёт прочитанных книг.

Программа должна уметь:

1. Добавлять информацию о книге:
 - Название (до 100 символов),
 - Автор (до 50 символов),
 - Год прочтения,
 - Оценка по 10-балльной шкале (float).
 2. Сохранять данные в файл `books.txt`.
 3. Загружать данные из файла при запуске.
 4. Показывать список всех книг.
 5. Показывать статистику:
 - Количество прочитанных книг;
 - Средняя оценка;
 - Самая высоко оценённая книга.
-

Требования:

- Не использовать классы и STL (только struct, массивы, fstream, cstring, iomanip, iostream);
- Программа должна быть написана в одном файле;
- Интерфейс через консольное меню;
- Максимальное количество книг — 1000.

Пример работы программы:

```
=== Трекер книг ===
1. Добавить книгу
2. Показать все книги
3. Показать статистику
4. Выход
Выберите пункт: 1
Введите название книги: Преступление и наказание
Введите автора: Достоевский
Год прочтения: 2023
Ваша оценка (0-10): 9.5
Книга добавлена!
```