

# **ФИО: Голев Никита Владимирович**

## **Группа: СКБ252**

### **1. Задание:**

Создать программу под названием «Трекер книг», которая будет вести учёт прочитанных пользователем книг. В программе должен быть реализован функционал сохранения (записи) информации про книги, просмотра этой информации, а так же вывод статистики по сохранённым записям.

### **2. Описание алгоритма:**

#### **Главная функция main()**

- Запуск программы.
- Отображение меню.
- Запуск других функций в зависимости от выбора пользователя.

#### **Функция showmenu()**

- Выводит на экран основное меню программы с выбором действия.

#### **Функция newbook()**

- Запрашивает у пользователя данные книги: название, автора, год прочтения и оценку
- Добавляет новую запись в массив и сохраняет её в файл.

#### **Функция savebookinfile()**

- Получает на вход структуру Book с информацией о книге.
- Добавляет данные о книге в текстовый файл books.txt.

#### **Функция loadBooks()**

- Загружает ранее сохранённые книги из файла в массив структур.

#### **Функция showall()**

- Выводит таблицу со всеми сохранёнными книгами в удобном формате.

#### **Функция showstat()**

#### **Вычисляет общую статистику:**

- Количество книг
- Среднюю оценку
- Самую высоко оценённую книгу

### 3.Код

```
#include <iostream>
#include <fstream>
#include <cstring>
#include <iomanip>
using namespace std;

const int MAX = 1000;

struct Book
{
    char title[100];
    char author[50];
    int year;
    float rating;
};

Book books[MAX];
int k = 0 ;
//k это количество

void loadBooks(const char *filename)
{
    ifstream fin(filename);

    if (!fin)
        return;

    while (fin >> books[k].title
           >> books[k].author
           >> books[k].year
           >> books[k].rating)
    {
        k+=1;
    }
    fin.close();
}

// random это переменная какая-то , когда а,б,с уже надоели
void savebookinfile(const char *filename, Book random)
{
    ofstream fout(filename , ios::app);
    if (!fout)
    {
        cout << "Ошибка при сохранении файла!" << endl;
        return;
    }
    fout << random.title<< " " << random.author<< " " <<
    random.year<< " " << random.rating << endl;
```

```

        fout.close();
    }
    void newbook()
    {
        if (k >= MAX)
        {
            cout << "Превышен лимит книг!" << endl;
            return;
        }
        Book random;
        cin.ignore();
        cout << "Введите название книги: ";
        cin.getline(random.title, 100);

        cout << "Введите автора: ";
        cin.getline(random.author, 50);

        cout << "Год прочтения: ";
        cin >> random.year;

        cout << "Ваша оценка (0-10): ";
        cin >> random.rating;

        if (random.rating < 0 or random.rating > 10)
        {
            cout << "Ошибка: оценка должна быть от 0 до 10!" << endl;
            return;
        }

        books[k++] = random;
        savebookinfile("books.txt", random);

        cout << "Книга добавлена!\n";
    }

    void showall()
    {
        if (k == 0)
        {
            cout << "Нет данных о книгах.\n";
            return;
        }

        cout << "\n--- Все книги ---\n";
        cout << left << setw(40) << "Название"
            << setw(25) << "Автор"
            << setw(10) << "Год"
            << setw(8) << "Оценка" << endl;
    }

```

```

    cout <<
    "-----\n";
    << endl;

    for (int i = 0; i < k; i++)
    {
        cout << left << setw(40) << books[i].title
            << setw(25) << books[i].author
            << setw(10) << books[i].year
            << setw(8) << books[i].rating << endl;
    }
}

void showstat()
{
    if (k == 0)
    {
        cout << "Нет книг для анализа.\n";
        return;
    }
    float total = 0;
    float maxrating = -1345;
    int best;
    for (int i = 0; i < k; i++)
    {
        total += books[i].rating;
        if (books[i].rating > maxrating)
        {
            maxrating = books[i].rating;
            best = i;
        }
    }
    float sr = total / k;
    cout << "\n--- Статистика ---\n";
    cout << "Количество прочитанных книг: " << k << endl;
    cout << "Средняя оценка: " << sr << endl;
    cout << "Самая высоко оценённая книга: " << books[best].title
    << books[best].rating << endl;
}

void showmenu()
{
    cout << "\n=== Трекер книг ===" << endl;
    cout << "1. Добавить книгу" << endl;
    cout << "2. Показать все книги" << endl;
    cout << "3. Показать статистику" << endl;
    cout << "4. Выход" << endl;
    cout << "Выберите пункт: ";
}

int main()
{
    loadBooks("books.txt");
}

```

```
int choice;  
do  
{  
    showmenu();  
    cin >> choice;
```

```
    switch ( choice)  
    {  
        case 1:  
            newbook();  
            break;  
        case 2:  
            showall();  
            break;  
        case 3:  
            showstat();  
            break;  
        case 4:  
            cout << "Выход из программы.\n";  
            break;  
        default:  
            cout << "Неверный выбор.\n";  
    }
```

```
    } while ( choice != 4);
```

```
    return 0;
```

```
}
```

#### 4. Ссылка на репозиторий:

<https://github.com/nikgolic-h-ship-it/Nikita-Golev-SCB252.git>

## 5. Вывод:

В ходе выполнения работы я написал программу на языке **C++**, реализующая учёт прочитанных книг.

Я научился использовать:

- файловые потоки (ifstream, ofstream) для сохранения и загрузки данных;
- структуры (struct) для хранения информации о книгах;
- базовые функции ввода и вывода (cin, cout);
- форматированный вывод данных при помощи iomanip;
- работу с циклами, условиями и динамическими массивами.

Данная работа помогла закрепить базовые навыки программирования на **C++** и понять, как организовывать хранение и обработку пользовательских данных в файле.