**ФИО: Зотов Марк Владимирович  
Группа: СКБ252  
Подгруппа: 1**

1. **Задание:**

* Создать трекер книг

1. **Описание алгоритма:**

* Функция загрузки книг из файла:  
  Читает данные из файла и копирует из временных переменных в постоянные
* Функция записи в файл:  
  Записывает данные в конец файла
* Функция очистки буфера:  
  Не до конца понимаю, как она работает
* Функция добавления книги:  
  Добавляет в список книгу по структуре: «Название», «Автор», «Год прочтения»,  
  «Оценка»
* Функция показа всех прочитанных книг:  
  В основном отвечает за визуальное представление. Выравнивает текст по   
  левому краю и задает ширину каждому столбцу, а под каждым столбцом  
  выводит соответствующие ему данные
* Функция статистики:  
  Находит максимальную оценку среди всех книг, а также сохраняет ее индекс.  
  Считает среднее арифметическое значение оценки среди всех прочитанных  
  книг
* Функция меню:  
  Выводит тестовую обложку меню
* Функция main:  
  Загружает книги из файла и выводит функцию меню, далее в зависимости от выбора пользователя показывает вышеперечисленные функции, это будет до того момента, пока пользователь не выберет «Выход»

1. **Код:**#include <iostream>

#include <fstream>

#include <cstring>

#include <iomanip>

using namespace std;

const int MAX\_BOOKS=1000;

struct Book

{

char name\_book[101];

char author [51];

int date;

float rating;

};

case 1:

addBook();

break;

case 2:

showAllBooks();

break;

case 3:

showStats();

case 4:

cout<<"Выход"<<endl;

default:

cout<<"Неверный выбор"<<endl;

}

}while (ch!=4);

return 0;

}



Book books[MAX\_BOOKS];

int book\_count=0;

void loadBooks(const char \*filename)

{

ifstream fin(filename);

if (!fin)

{

return;

}

char temp\_nb[101];

char temp\_a[51];

int temp\_d;

float temp\_r;

while (fin>>books[book\_count].name\_book>>books[book\_count].author>>books[book\_count].date>>books[book\_count].rating)

{

book\_count++;

}

fin.close();

}

void SaveBook(const char \*filename, Book b)

{

ofstream fout(filename, ios::app);

if (!fout)

{

cout <<"error open"<<endl;

return;

}

fout<<b.name\_book<<" "<<b.author<<" "<<b.date<<""<<b.rating<<endl;

fout.close();

}

void addBook()

{

if (book\_count >= MAX\_BOOKS)

{

cout<<"LIMIT MEMORY"<<endl;

return;

}

Book newBook;

cout<<"Напишите название книги"<<endl;

cin>>newBook.name\_book;

cout<<"Напишите ФИО автора в формате - 'И.И. Иванов'"<<endl;

cin>>newBook.author;

cout<<"Напишите дату прочтения книги в формате - 'гггг'"<<endl;

cin>>newBook.date;

cout<<"Оцените книгу по 10-балльной шкале"<<endl;

cin>>newBook.rating;

if(newBook.rating<0 || newBook.rating>10)

{

cout<<"Оценка от 1 до 10!!!"<<endl;

return;

}

books[book\_count++]=newBook;

SaveBook("books.txt", newBook);

cout<<"Книга добавлена"<<endl;

}

void showAllBooks()

{

if (book\_count==0)

{

cout<<"Нет прочитанных книг"<<endl;

return;

}

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

cout<< left <<setw(12)<<"Название"

<<setw(15)<<"Автор"

<<setw(10)<<"Дата"

<<"Оценка"<<endl;

cout << "----------------------------------------------------" << endl;

for (int i=0; i<book\_count; i++)

{

cout<< left <<setw(12)<<books[i].name\_book

<<setw(15)<<books[i].author

<<setw(10)<<books[i].date

<<books[i].rating<<"/10"<<endl;

}

}

void showStats()

{

if (book\_count==0)

{

cout<<"Библиотека пуста"<<endl;

return;

}

float totalR=0;

float maxR=0;

int bestB=0;

for (int i=0; i<book\_count; i++)

{

totalR+=books[i].rating;

if (books[i].rating>maxR)

{

maxR=books[i].rating;

bestB=i;

}

}

float midR=totalR/book\_count;

cout<<"\n--- Статистика ---"<<endl;

cout<<"Количество книг:"<<book\_count<<endl;

cout<<"Средняя оценка:"<<fixed<<setprecision(1)<<midR<<"/10"<<endl;

cout<<"Самая лучшая книга:"<<books[bestB].name\_book<<books[bestB].author<<books[bestB].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"<<endl;

}

int main()

{

loadBooks("books.txt");

int ch;

do

{

showMenu();

cin>>ch;

switch (ch)

{

case 1:

addBook();

break;

case 2:

showAllBooks();

break;

case 3:

showStats();

case 4:

cout<<"Выход"<<endl;

default:

cout<<"Неверный выбор"<<endl;

}

}while (ch!=4);

return 0;

}

<<setw(10)<<"Дата"

<<"Оценка"<<endl;

cout << "----------------------------------------------------" << endl;

for (int i=0; i<book\_count; i++)

{

cout<< left <<setw(12)<<books[i].name\_book

<<setw(15)<<books[i].author

<<setw(10)<<books[i].date

<<books[i].rating<<"/10"<<endl;

}

}

void showStats()

{

if (book\_count==0)

{

cout<<"Библиотека пуста"<<endl;

return;

}

float totalR=0;

float maxR=0;

int bestB=0;

for (int i=0; i<book\_count; i++)

{

totalR+=books[i].rating;

if (books[i].rating>maxR)

{

maxR=books[i].rating;

bestB=i;

}

}

float midR=totalR/book\_count;

cout<<"\n--- Статистика ---"<<endl;

cout<<"Количество книг:"<<book\_count<<endl;

cout<<"Средняя оценка:"<<fixed<<setprecision(1)<<midR<<"/10"<<endl;

cout<<"Самая лучшая книга:"<<books[bestB].name\_book<<books[bestB].author<<books[bestB].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"<<endl;

}

int main()

{

loadBooks("books.txt");

int ch;

do

{

showMenu();

cin>>ch;

switch (ch)

{

case 1:

addBook();

break;

case 2:

showAllBooks();

break;

case 3:

showStats();

case 4:

cout<<"Выход"<<endl;

default:

cout<<"Неверный выбор"<<endl;

}

}while (ch!=4);

return 0;

}

**4. Ссылка на репозиторий:  
 https://github.com/pulldthe3ger/study**