**Лабораторная работа №6**

**по курсу**

**«Введение в программирование»**

**Вариант 8**

**Выполнил:**

Студент группы ЭР-05-20

Волчков Дмитрий

**Задание:**

Разработать программу, составив для своего задания таблицу, подобную таблице:

| Группа | фамилия, имя, отчество | Год  рождения | Пол | Оценка  по  физике | Оценка по математике | Оценка по ин­форматике | Стипендия |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |

Решить задачу с использованием файлов.

Предусмотреть проверку правильности вводимых данных в соответствии с условием задачи

О каждой книге имеются следующие сведения: фамилия и инициалы автора, название, год издания и число запросов за последние три года. Вывести фамилии авторов и названия книг, изданных до 1980 г., по которым нет запросов. Определить число таких книг.

**1.Постановка задачи:**

Разработать программу, считывающую данные из файла или вручную на выбор пользователя в структуру и возвращающая книги, изданные до 1980, по которым нет запросов в виде таблицы. В случаях ошибок предлагать пользователю изменить ошибку. В случае отказа оставить данные о книгах без изменений.

*Входные данные:* данные о названии книги (name типа string), фамилия и инициалы автора (author типа string), год издания книги (public\_year типа unsigned int), количество запросов за 3 года (request типа unsigned int) из файла.

*Выходные данные:* количество книг, удовлетворяющих условию (booknum типа unsigned int), данные о названии книги (name типа string), фамилия и инициалы автора (author типа string), год издания книги (public\_year типа unsigned int), количество запросов за 3 года (request типа unsigned int) книг, удовлетворяющих условию отбора.

*Ограничения:* в файле может быть максимум 100 книг, количество вводится вручную перед работой программы. Если ввести количество книг больше, чем в файле то будет предложено ввести книги дополнительно, если ввести количество книг меньше, чем в файле, то он прочитает заданное количество книг с начала файла. Остальные книги не будут прочитаны. Если ввести неверный год или количество запросов, то программа предложит исправить ошибку. Год книги не может быть меньше 0 и больше 2021. Количество упоминаний не может быть меньше нуля.

Данные в файле представлены в виде:

|  |
| --- |
| [Название книги 1]  [Фамилия и инициалы автора книги 1]  [Год публикации книги 1]  [Количество запросов за 3 года книги 1]  [Название книги 2]  …  [Количество запросов за 3 года книги m] |

Вид приложения – консольное приложение на языке C++

Среда разработки – Microsoft Visual Studio 2019

**2.Разработка программы:**

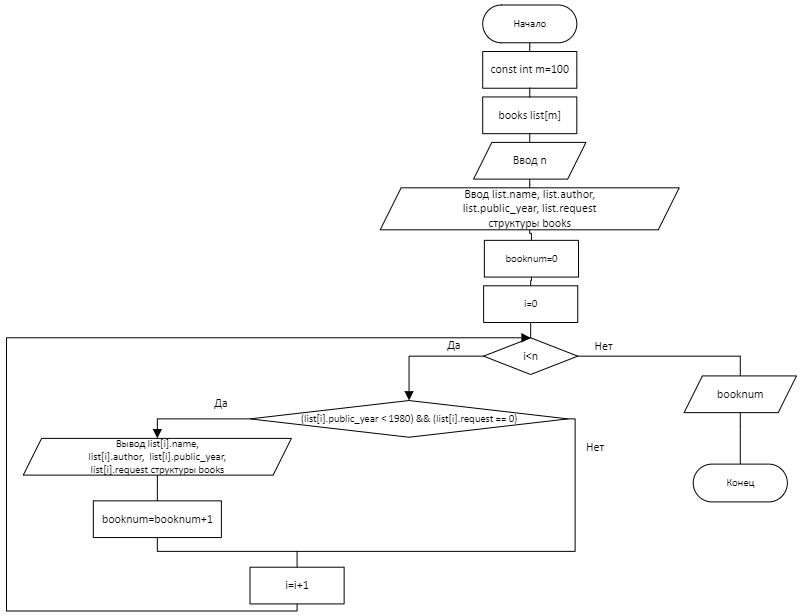
Для решения задачи было решено создать структуру books с четыремя информационными полями и использовать статический массив элементов структуры из-за простоты реализации и обработки. Также решено было ввести проверку на ввод и предложить пользователю возможности считать данные из файла или ввести вручную, исправить ошибки в файле или оставить их. В соответствии с условием вывести необходимые книги в виде таблицы:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер | Название | Автор | Год издания | Количество запросов за 3 года |

***3.Состав данных***

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Смысл** | **Тип** | **Структура** |
| Исходные данные | | | |
| books | структура с 4-мя информационными полями |  | структура |
| name | информационное поле данных о названии книги | символьный | символьная строка |
| author | информационное поле данных об авторе и его инициалах книги | символьный | символьная строка |
| public\_year | информационное поле данных о годе издания книги | целый | простая переменная |
| request | информационное поле данных о количестве запросов книги | целый | простая переменная |
| n | количество вводимых книг | целый | простая переменная |
| m | максимальное количество книг | целый | постоянная |
| Промежуточные данные | | | |
| i | счетчик цикла | целый | простая переменная |
| list | одномерный массив элементов структуры |  | одномерный массив |
| Выходные данные | | | |
| booknum | количество книг, удовлетворяющих условию отбора | целый | простая переменная |

**4.Алгоритм программы**

****

**5.Листинг программы**

#include <iostream>

#include <string>

#include <fstream>

#include <windows.h>

#include <iomanip>

/\* О каждой книге имеются следующие сведения : фамилия и инициалы автора, название, год издания и число запросов за последние три года.

Вывести фамилии авторов и названия книг, изданных до 1980 г., по которым нет запросов. Определить число таких книг. Решить задачу с использованием файлов. \*/

struct books { // Создание структуры книг

std::string name; // Название книги

std::string author; // Фамилия и инициалы автора

int public\_year; // Год публикации книги

int request; // Количество запросов книги за 3 года

};

int main()

{

setlocale(LC\_ALL, "rus");

SetConsoleCP(1251); // Устанваливаем кодировку для ввода и вывода

SetConsoleOutputCP(1251);

const int m = 100;

books list[m]; // Инициализируем массив книг

int n=0;

while ((n > 0) || (n <= m)) {

std::cout << "Введите корректное количество вводимых книг: ";

std::cin >> n;

}

char input;

std::cout << "Ввести книги из файла или вручную?(F-из файла/H-вручную)" << std::endl;

std::cin >> input;

// Ввод с клавиатуры

if (input == 'H') {

for (int i = 0; i < n; i += 1) {

while (list[i].name.length() == 0) {

std::cout << "Введите корректное название книги:" << std::endl;

std::cin.ignore();

getline(std::cin, list[i].name);

}

while (list[i].author.length() == 0) {

std::cout << "Введите корректные фамилию и инициалы автора:" << std::endl;

getline(std::cin, list[i].author);

}

while ((list[i].public\_year <= 0) || (list[i].public\_year > 2021)) {

std::cout << "Введите корректный год публикации книги:" << std::endl;

std::cin >> list[i].public\_year;

}

while (list[i].request < 0) {

std::cout << "Введите корректное количество запросов за 3 года:" << std::endl;

std::cin >> list[i].request;

}

}

}

// Чтение из открытого файла

else if (input == 'F') {

std::ifstream file("C:\\Users\\Dmitrii\\Desktop\\Важное\\3 сем\\Инфа\\6 лаба\\books.txt"); // Открываем файл с информацией о книгах

for (int i = 0; i < n; i += 1) {

getline(file, list[i].name);

getline(file, list[i].author);

file >> list[i].public\_year >> list[i].request;

file.ignore(256, '\n');

if ((list[i].name.length() == 0) && (list[i].author.length() == 0)) {

std::cout << "Книги №" << i + 1 << " не существует. Хотите ввести данные?(Y-да/N-нет)" << std::endl;

char flag;

std::cin >> flag;

if (flag == 'Y') {

while (list[i].name.length() == 0) {

std::cout << "Введите корректное название книги:" << std::endl;

std::cin.ignore();

getline(std::cin, list[i].name);

}

while (list[i].author.length() == 0) {

std::cout << "Введите корректные фамилию и инициалы автора:" << std::endl;

getline(std::cin, list[i].author);

}

while ((list[i].public\_year <= 0) || (list[i].public\_year > 2021)) {

std::cout << "Введите корректный год публикации книги:" << std::endl;

std::cin >> list[i].public\_year;

}

while (list[i].request < 0) {

std::cout << "Введите корректное количество запросов за 3 года:" << std::endl;

std::cin >> list[i].request;

}

}

else if (flag == 'N') i = n;

else if (flag != 'N') {

std::cout << "Ошибка: Необходимо было ввести Y или N" << std::endl;

system("pause");

return -1;

}

}

if (((list[i].public\_year <= 0) || (list[i].public\_year > 2021)) && (i < n)) {

std::cout << "Год книги №" << i + 1 << " некорректен. Хотите изменить год?(Y-да/N-нет)" << std::endl;

char flag;

std::cin >> flag;

if (flag == 'Y') {

while ((list[i].public\_year <= 0) || (list[i].public\_year > 2021)) {

std::cout << "Введите новый год для книги №" << i + 1 << " повторно:" << std::endl;

std::cin >> list[i].public\_year;

}

}

else if (flag != 'N') {

std::cout << "Ошибка: Необходимо было ввести Y или N" << std::endl;

system("pause");

return -1;

}

}

if ((list[i].request < 0) && (i < n)) {

std::cout << "Количество запросов книги №" << i + 1 << " некорректны. Хотите изменить количество запросов?(Y-да/N-нет)" << std::endl;

char flag;

std::cin >> flag;

if (flag == 'Y') {

while (list[i].request < 0) {

std::cout << "Введите количество запросов для книги №" << i + 1 << " повторно:" << std::endl;

std::cin >> list[i].request;

}

}

else if (flag != 'N') {

std::cout << "Ошибка: Необходимо было ввести Y или N" << std::endl;

system("pause");

return -1;

}

}

}

file.close(); // Закрытие файла

}

else {

std::cout << "Ошибка: Необходимо было ввести F или H" << std::endl;

return -1;

}

// Вывод книг, удовлетворяющих условию

std::cout << std::setw(8) << "Книга №:" << std::setw(25) << "Название:" << std::setw(20) <<"Автор:" << std::setw(20) << "Год публикации:" << std::setw(25) << "Количество запросов:" << std::endl;

unsigned int booknum = 0;

for (int i = 0; i < n; i += 1) {

if ((list[i].public\_year < 1980) && (list[i].request == 0)) {

std::cout << std::setw(8) << i + 1 << std::setw(25) << list[i].name << std::setw(20) << list[i].author << std::setw(20) << list[i].public\_year << std::setw(25) << list[i].request << std::endl;

booknum = booknum + 1;

}

}

std::cout << "Количество книг: " << booknum << std::endl;

system("pause");

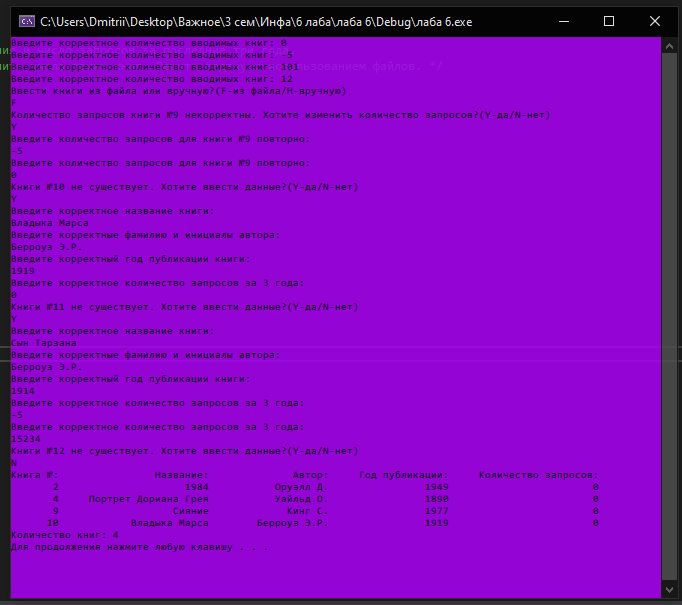
return 0;

}

**6.Контрольные тесты**

**Из файла с книгами:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер | Название | Автор | Год издания | Количество запросов за 3 года |
| 1 | 451° по Фаренгейту | Брэдбери Р. | 1953 | 1053486 |
| 2 | 1984 | Оруэлл Д. | 1949 | 0 |
| 3 | Мастер и Маргарита | Булгаков М.А. | 1966 | 15357 |
| 4 | Портрет Дориана Грея | Уайльд О. | 1890 | 0 |
| 5 | Над пропастью во ржи | Сэлинджер Дж.Д. | 1951 | 136546 |
| 6 | Вино из одуванчиков | Брэдбери Р. | 1957 | 789456 |
| 7 | Замыслил я побег… | Поляков Ю.А. | 1999 | 0 |
| 8 | Ацтек | Дженнингс Г. | 1980 | 0 |
| 9 | Сияние | Кинг С. | 1977 | -10 |



**Вручную:**

