



## Компьютерная академия «Шаг»

Кафедра разработки программного обеспечения

**Дисциплина:** Программирование на языке «Си».

**Оборудование:** ПК, Microsoft Visual Studio, конспект.

**Ориентировочное время выполнения работы:** 3 – 4 недели.

**Примечание:** *Выполненную экзаменационную работу сдавать в виде исходных файлов с типами <.cpp><.h>, помещая их в папку с названием группы, Ф.И.О. Пример – “gr10c111\_Сидоров\_И\_И\_ЭкзаменПрактика”.*

### Практическое экзаменационное задание

Предлагается, разработать и написать консольное приложение “Библиотека”. Программа должна содержать базу данных, по книгам и пользователям, а также возможности обработки хранимой информации.

Программа должна хранить следующую информацию по книгам и читателям. Для организации хранения данных рекомендуется использовать составной тип данных struct.

Информация для хранения «Книга»	Информация для хранения «Читатель»
Дата издания. Название. Автор. Жанр. Стоимость. Рейтинг. ID номер книги.	Фамилия. Имя. Отчество. Дата рождения. Номер паспорта. ID номер читателя. Количество прочитанных книг. Список книг находящихся на руках у читателя.

В приложение должны быть заложены следующие функциональности по работе с книгами и пользователями:

Обработка данных «Книга»	Обработка данных «Читатель»
Добавление. Удаление. Редактирование полное и частичное. Поиск и сортировка по автору, названию, жанру, популярности. Выдача книги. Возврат книги. Вывод информации на экран о самых популярных книгах в своем жанре. Вывод информации о книгах находящихся на руках у читателей. При возврате книги читателем, учитывать, что если есть просроченные дни, то выводить на экран сумму начисленной пени.	Добавление. Удаление. Редактирование полное и частичное. Поиск и сортировка по ФИО, количеству прочитанных книг, ID, номеру паспорта. Вывод информации на экран о самых активных читателях. Вывод информации о читателях с просроченной датой возврата книги, обязательно выводить при этом количество просроченных дней и начисленной пени.

Замечания к разработке:

( Обязательно к выполнению )

Программа должна содержать, команды по обработке информации в виде пунктов меню. Само меню не должно быть представлено как один список, а иметь в себе подпункты, обработка команд меню осуществлять только строкой. Программа должна хранить информацию только в динамической памяти. Названия переменных, массивов, составных типов, указателей должны четко выражать свое назначение по характеру хранимой там информации, а название функций и макросов отображать их назначение. Каждая функция, переменная, макрос, массив должен быть прокомментирован (коротко), с какой целью используется в коде. Для полустационарных групп, часть информации должна заноситься в память сразу в момент старта программы (инициализация переменных, массивов, структур ) минимум 20 книг, 10 читателей. Для стационарных групп, хранение информации организовать в виде сохранения в файл.

( Необязательно к выполнению )

Программа может быть представлена в цветовой схеме. Выводимая часть информации реализовывать в виде таблиц. Добавление своих личных идей, не меняющих общего и внутреннего представления программы.

```

// Примеры функций, структур.
#include <windows.h>
#include <stdio.h>
//функция установки цветов заливки и текста в консоле принимает два аргумента, Bg – цвет
//фона от 0 до 15 и Tex -> цвет текста от 0 до 15
void SetColorBgTex(int Bg = 0, int Tex = 15) {
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    Bg = Bg << 4 | Tex;
    SetConsoleTextAttribute(hConsole, Bg);
}
//функция позиционирования курсора перед выводом информации, принимает два аргумента x –
//позиционирование по X ;), Y – позиционирование по Y, напоминая 0 координата расположена
//по умолчанию в левом верхнем углу монитора
void SetConsCurPos(int x, int y) {
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    COORD coord = { x, y };
    SetConsoleCursorPosition(hConsole, coord);
}
//функция установки размера окна и буфера консоли. Принимает два аргумента XSize – размер
//окна по X-оси, YSize – размер по Y-оси. Предупреждение если размер окна превысит
//разрешение экрана монитора, то оно будет попросту сворачиваться на панели задач.
void SetConsWinSize(int XSize = 20, int YSize = 10) {
    HWND hWnd = GetForegroundWindow();
    HANDLE wHnd = GetStdHandle(STD_OUTPUT_HANDLE);
    SetWindowPos(hWnd, NULL, 0, 0, 0, 0, NULL);
    SMALL_RECT windowSize;
    windowSize.Bottom = YSize;
    windowSize.Left = 0;
    windowSize.Right = XSize;
    windowSize.Top = 0;
    SetConsoleWindowInfo(wHnd, TRUE, &windowSize);
    COORD bufferSize = { XSize + 1, YSize + 1 };
    SetConsoleScreenBufferSize(wHnd, bufferSize);
}
//Работа с датой
//структура для хранения даты
SYSTEMTIME TimeDate;
//Поля структуры SYSTEMTIME, все поля созданы на базе типа WORD -> 16-bit unsigned
//integer. typedef unsigned short WORD;
//TimeDate.wDay
//TimeDate.wDayOfWeek
//TimeDate.wHour
//TimeDate.wMilliseconds
//TimeDate.wMinute
//TimeDate.wMonth
//TimeDate.wSecond
//TimeDate.wYear
//строковой массив для хранения даты в виде строки перед выводом на экран,
//Обратите внимание на ТИП !
WCHAR stringDateTime[50];
//Функция позволяющая сформировать дату в строку
//GetDateFormat(LOCALE_USER_DEFAULT, LOCALE_USE_CP_ACP, &TimeDate, NULL, stringDateTime,
sizeof(stringDateTime));
//Дополнительную информацию можно взять из MSDN
//По типам ms-help://MS.MSDNQTR.v90.ru/winprog/winprog/windows_data_types.htm
//По структуре ms-help://MS.MSDNQTR.v90.ru/dv_vclib/html/9aaef4d6-de79-4fa1-8158-
86b245ef5bfff.htm
//По функции GetDateFormat ms-help://MS.MSDNQTR.v90.ru/intl/nls_5w6s.htm

```