

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ ИМ. Р.Е. АЛЕКСЕЕВА

ИНСТИТУТ РАДИОЭЛЕКТРОНИКИ И ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ

Кафедра «Информационные радиосистемы»

### КУРСОВАЯ РАБОТА

*«Рассматривается информация о ресурсах библиотеки. Сведения о*

*печатном издании включают в себя: название, год выпуска, жанр.*

*Требуется найти все издания указанного жанра, вышедшие в указанный*

*период времени.»*

Выполнил:

студент группы **24-Рз И. И. Иванов**

Проверил:

доцент кафедры ИРС С. Б. Сидоров

Нижний Новгород

2025

## Содержание

Введение.....	3
1. Постановка задачи.....	4
2. Руководство пользователя.....	5
3 Руководство программиста.....	7
3.1 Структура программы.....	7
3.2 Структура данных.....	7
3.3 Алгоритм поиска книг в библиотеке.....	8
Заключение.....	9
Список литературы.....	10

## **Введение**

В работе рассматривается решение задачи разработки прикладной программы поиска всех изданий указанного жанра, вышедшие в указанный период времени.

Одним из способов преодоления сложности данной задачи является использование отдельной функции для выполнения обработки. Этот подход позволяет свести задачу к последовательному решению более простых задач. Его использование позволяет уменьшить затраты на отладку и повысить надёжность программы. Также существенным моментом является использование структурных типов данных для адекватного отображения сущностей проблемной области в область программной реализации.

В начале основной части отчёта приводится точная формализованная постановка задачи с указанием полного набора операций, выполнение которых должна обеспечивать прикладная программа.

В руководстве пользователя раскрывается назначение программы, её возможности и выполняемые операции. Подробно объясняются правила пользования программой и приводятся конкретные примеры диалога с пользователем.

В руководстве программиста рассматриваются вопросы внутренней организации программы, в том числе перечень функций и их взаимодействие. Кроме этого описываются используемые структуры данных и наиболее важные, и интересные с точки зрения их реализации алгоритмы.

В заключении делаются выводы о полноте решения поставленной задачи, а также приводится ряд соображений по возможным направлениям доработки полученной прикладной программы. Также приведена техническая информация, включающая листинги программы.

## **1. Постановка задачи**

Рассматривается модель информации о ресурсах библиотеки. Описание печатного издания представлено совокупностью свойств: название, год выпуска и жанр. В области программной реализации модель издания имеет вид структурного типа данных.

Рассматривается набор экземпляров печатных изданий с конечным количеством элементов. Требуется получить программную реализацию заданной обработки такого набора структурированных данных.

Все данные, необходимые для обработки, запрашиваются у пользователя: значения элементов, входящих в набор, и дополнительная информация, необходимая для выполнения конкретной обработки данных.

Обработка должна быть реализована отдельной функцией. При этом вся необходимая для выполнения обработки информация должна передаваться в функцию через список аргументов. Результат обработки набора структурированных данных должен передаваться из функции также через список аргументов.

Полученные результаты обработки должны быть выданы на монитор, то есть на стандартное устройство вывода.

Задан набор печатных изданий. Требуется найти все издания указанного жанра, вышедшие в указанный период времени.

## **2. Руководство пользователя**

Программа предназначена для поиска печатных изданий заданного жанра, вышедших в указанный период времени, в режиме диалога с пользователем.

Программа позволяет пользователю задать набор изданий и указать критерии поиска (жанр и временной интервал). Результаты обработки выводятся на экран монитора.

Стоит отметить, что программа является чувствительной к регистру, соответственно названия жанров для поиска необходимо указывать с точностью регистра.

Запуск программы осуществляется либо через командную строку с вводом имени исполняемого файла, полученного в результате компиляции, с последующим нажатием клавиши Enter, либо иным способом в зависимости от операционной системы.

Программа является интерактивным консольным приложением. Весь диалог с пользователем осуществляется в текстовом режиме.

Сначала пользователю предлагается указать количество изданий, для которых будет осуществлён ввод данных:

`Enter the number of publications:`

Затем последовательно для каждого издания запрашиваются следующие данные:

- название издания
- год выпуска
- жанр

После завершения ввода всех изданий программа предлагает указать критерии поиска:

- Enter genre to search:
- Enter start year:
- Enter end year:

После ввода критериев выполняется поиск соответствующих изданий. По завершении обработки выводится результат следующего вида:

Titles in genre Novel from 1900 to 2000:

- *Название 1*
- *Название 2*

Если подходящих изданий не найдено, выводится соответствующее сообщение.

По завершении выдачи результатов программа завершает свою работу.

### 3 Руководство программиста

#### 3.1 Структура программы

Прикладная программа разработана с использованием принципов императивного программирования. Она является совокупностью взаимодействующих функций. Структура программы представлена на рис. 1.

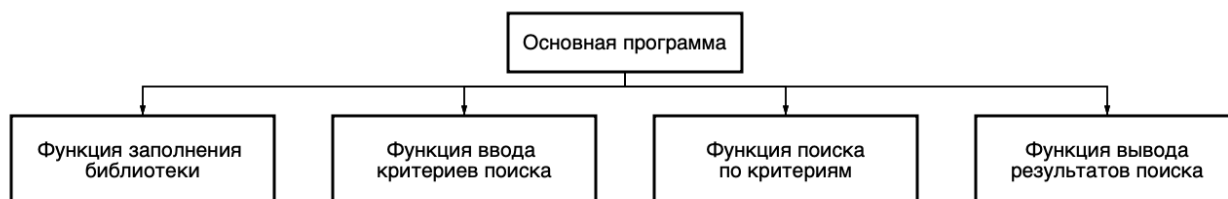


Рисунок 1. Структура программы

Программа состоит из 5 функций, назначение каждой из которых приведено ниже:

1. `main` – основная функция приложения
2. `do_fill` – функция заполнения библиотеки, путем запроса значений у пользователя
3. `do_input` – функция ввода пользователем критериев поиска
4. `do_search` – функция поиска по заданным критериям
5. `do_output` – функция вывода результатов поиска

#### 3.2 Структура данных

Для описания книги из библиотеки определяем структурный тип данных `Book`. Из постановки задачи явно следует, что каждая книга задается параметрами: название, год выпуска, жанр. Каждому из этих свойств соответствует свое поле в структурном типе данных. Таким образом получаем следующее определение структурного типа данных, для описания понятия “Книга”:

```
typedef struct {  
  
    char title[MAX_TITLE_LENGTH];  
  
    int year;  
  
    char genre[MAX_GENRE_LENGTH];  
  
} Book;
```

В данном случае, количеству символов в названии и жанре соответствуют значения указанных макросов, а именно: 100 и 50.

Для представления библиотеки используем массив из элементов типа **Book – library**. Максимально допустимое количество элементов в массиве ограничим макросом **MAX\_PUBLICATIONS**, со значением 100. Так же, для хранения подходящих под критерии книг, вводим второй массив из элементов типа **Book – matches**. По аналогии с массивом **library**, ограничим число элементов макросом **MAX\_PUBLICATIONS**, со значением 100

### 3.3 Алгоритм поиска книг в библиотеке

Определение наличия книг указанного жанра, за указанный период сводится к проверке, внутри цикла по количеству всех книг, логического выражения:

```
(жанр_текущей_книги == искомый_жанр) && (начальный_год_поиска <=  
год_текущей_книги <= конечный_год_поиска)
```

Если текущая книга удовлетворяет условию, значения логического выражения – ИСТИНА, то книга заносится в массив **matches**.



## **Заключение**

В данной работе задача разработки прикладной программы поиска книг по жанру, вышедшие в указанный период времени, была решена с использованием императивного программирования. На основании проведённой отладки и испытаний с помощью контрольных примеров можно сделать вывод, что полученная прикладная программа решает поставленную задачу правильно и в полном объёме.

### **Список литературы**

1. Керниган, Брайан У., Ритчи, Деннис М. Язык программирования С, 2-е издание. [пер. с англ.] / Б.У. Керниган, Д.М. Ритчи – М.: Вильямс, 2007.
2. Павловская, Т.А. С/С++. Программирование на языке высокого уровня: учебник для ВУЗов / Т.А. Павловская. – СПб.: Питер, 2009.
3. Орлов, С.А. Технологии разработки программного обеспечения. учеб. пособие. 2-е изд./ С.А. Орлов, – СПб.: Питер, 2003. – 480 с.: ил.
4. Борисенко, В.В. Основы программирования / В.В.Борисенко, – Интернет-университет информационных технологий – ИНТУИТ.ру, 328 стр. – 2005 г.
5. Шилдт, Г. Полный справочник по С: учеб. пособие / Г. Шилдт. – 4-е изд. – М.: Изд. дом "Вильямс", 2008.
6. Костюкова, Н.И. Язык Си и особенности работы с ним / Н.И. Костюкова, Н.А. Калинина – Интернет-университет информационных технологий – ИНТУИТ.ру, 208 стр. – 2006 г.

## Приложение А

### Основная программа

```
#include <stdio.h>
#include <string.h>

#define MAX_PUBLICATIONS 100
#define MAX_TITLE_LENGTH 100
#define MAX_GENRE_LENGTH 50

typedef struct {
    char title[MAX_TITLE_LENGTH];
    int year;
    char genre[MAX_GENRE_LENGTH];
} Book;

void do_fill(Book library[], int *total_publications) {
    printf("Enter the number of publications: ");
    scanf("%d", total_publications);
    getchar();
    for (int i = 0; i < *total_publications; i++) {
        printf("Enter title for publication %d: \n", i + 1);
        fgets(library[i].title, MAX_TITLE_LENGTH, stdin);
        library[i].title[strcspn(library[i].title, "\n")] = 0;

        printf("Enter year for publication %d: \n", i + 1);
        scanf("%d", &library[i].year);
        getchar();

        printf("Enter genre for publication %d: \n", i + 1);
        fgets(library[i].genre, MAX_GENRE_LENGTH, stdin);
```

```

        library[i].genre[strcspn(library[i].genre, "\n")] = 0;
    }
}

```

```

void do_input(char *search_genre, int *search_year_start, int
*search_year_end) {
    printf("Enter genre to search: \n");
    fgets(search_genre, MAX_GENRE_LENGTH, stdin);
    search_genre[strcspn(search_genre, "\n")] = 0;
    printf("Enter start year: \n");
    scanf("%d", search_year_start);
    printf("Enter end year: \n");
    scanf("%d", search_year_end);
}

```

```

int do_search(Book library[], int library_size, const char *genre,
int year_start, int year_end, Book matches[]) {
    int match_count = 0;
    for (int i = 0; i < library_size; i++) {
        if ((strcmp(library[i].genre, genre) == 0) &&
            (library[i].year >= year_start && library[i].year <=
            year_end)) {
            matches[match_count++] = library[i];
        }
    }
    return match_count;
}

```

```

void do_output(Book matches[], int match_count, const char *genre,
int year_start, int year_end) {

```

```

    printf("Titles in genre \"%s\" from %d to %d:\n", genre,
year_start, year_end);
    if (match_count == 0) {
        printf("No matching publications found.\n");
        return;
    }

    for (int i = 0; i < match_count; i++) {
        printf(" - %s (%d)\n", matches[i].title,
            matches[i].year);
    }
}

int main() {

    Book library[MAX_PUBLICATIONS];
    int total_publications;
    do_fill(library, &total_publications);

    char search_genre[MAX_GENRE_LENGTH];
    int search_year_start, search_year_end;
    do_input(search_genre, &search_year_start, &search_year_end);

    Book matches[MAX_PUBLICATIONS];
    int match_count = do_search(library, total_publications,
        search_genre, search_year_start, search_year_end, matches);
    do_output(matches, match_count, search_genre,
        search_year_start, search_year_end);

    return 0;
}

```