

## 10 РАБОТА С ФАЙЛАМИ, СОРТИРОВКА МАССИВОВ СТРУКТУР

**Цель работы** - получение практических навыков по написанию и отладке программ сортировки массивов структур.

### 10.1 Подготовка к работе

При подготовке к работе необходимо изучить особенности формирования массивов структур из файлов; изучить алгоритмы сортировки массивов.

### 10.2 Теоретические сведения

Для решения ряда задач необходимо использовать массивы структур, в которых информация отсортирована по какому-либо полю. После сортировки одинаковые значения полей располагаются подряд, потому можно подсчитать количество структур с определенным значением в этом поле, а для числовых полей найти сумму, среднее, максимальное или минимальное значение.

**Пример 10.1.** Создать структуру Книга: название, имя автора, цена, количество экземпляров книги в книжном магазине. Найти самую дорогую книгу. Создать новый файл с информацией о книгах, упорядоченной по полю «имя автора» (отсортировать массив структур). Подсчитать количество книг каждого автора. Найти автора, имеющего максимальное количество книг.

```
#include <fstream>
#include <iostream>
#include <iomanip>
#include <string.h>
#include <windows.h>
using namespace std;
struct book
{
    char title[100];
    char author[100];
    float price;
    int count;
};
// сортировка массива структур по полю «имя автора»
void sort(book arr[], int size)
{
    for (int i=size-1; i > 0; --i)
        for (int j=0; j < i; ++j)
            if(strcmp(arr[j].author,arr[j+1].author)>0)
            {
                book tmp = arr [j];
                arr [j] = arr [j+1];
                arr [j+1] = tmp;
            }
}
// подсчет количества книг каждого автора и поиск автора с максимальным количеством книг
void kol_books(book arr[], int size)
{
    char auth[100], max_auth[100];
    int i=0, max_c=0;
    cout<<"\n"<<setw(25)<<left<<"АВТОР"<<setw(25)<<left<<"КОЛИЧЕСТВО КНИГ"<<"\n\n";
    while ( i < size) // проход по массиву
    {
        strcpy(auth,arr[i].author);
        int c=0;
        while (strcmp(arr[i].author,auth)==0)
        {
            c+=arr[i].count; // подсчет количества книг автора auth
            ++i;
        }
    }
}
```

```

        cout<<setw(25)<<left<<auth<<setw(25)<<left<<c<<"\n";
        //поиск автора с максимальным количеством книг
        if (c>max_c)
        {
            max_c=c;
            strcpy(max_auth,auth);
        }
    }
    cout<<"\n"<<"АВТОР С МАКСИМАЛЬНЫМ КОЛИЧЕСТВОМ КНИГ ("<<max_c<<"
"<<max_auth<<"\n\n";
}

// поиск самой дорогой книги
void max_price(book arr[], int size)
{
    float max=arr[0].price;
    char max_title[100];
    strcpy(max_title,arr[0].title);
    char max_author[100];
    strcpy(max_author,arr[0].author);
    for (int j=0; j < size; ++j)
        if (max<arr[j].price)
        {
            max=arr[j].price;
            strcpy(max_title,arr[j].title);
            strcpy(max_author, arr[j].author);
        }
    cout<<"\nСАМАЯ ДОРОГАЯ КНИГА - "<<max_title<<" "<<max_author<<" ЕЕ ЦЕНА "<<max<<"\n";
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    ifstream fin("c:\\my_books.txt"); //исходный файл
    ofstream bookf; //файл для записи отсортированного массива
    bookf.open("c:\\BOOKINFO.txt");
    if(!fin.is_open())
    {
        cout<<"Файл не найден";
        return 1;
    }
    book b[10]; //массив из 10 структур book
    int N=0; //первое значение в файле - размер массива структур
    fin>>N;
    for(int i = 0; i < N; i++)
    {
        fin.ignore(); //пропустить символ '\n' (значения в файле разделены переводом строки)
        fin.getline(b[i].title,100); //прочитать строку с названием книги
        fin.getline(b[i].author,100); //прочитать строку с именем автора
        fin>>b[i].price; // прочесть строку с ценой книги
        fin>>b[i].count; // прочесть строку с числом экземпляров книги
    }
    cout<<"ИСХОДНЫЙ МАССИВ\n\n";
    cout<<setw(25)<<left<<"title"<<setw(10)<<left<<"author"<<"\t"<<"price"<<"\t"<<"count"<<"\n\n";
    //вывод массива на экран
    for(int i = 0; i < N; i++)
        cout<<setw(25)<<left<<b[i].title<<setw(10)<<left<<b[i].author<<"\t"<<b[i].price<<"\t"<<b[i].count<<"\n";
    sort(b, N);
    cout<<"\nМАССИВ ОТСОРТИРОВАННЫЙ ПО ПОЛЮ author\n\n";
    //вывод массива на экран и запись в файл
    for(int i = 0; i < N; i++)
    {
        cout<<setw(25)<<left<<b[i].title<<setw(10)<<left<<b[i].author<<"\t"<<b[i].price<<"\t"<<b[i].count<<"\n";
    }
}

```

```

bookf<<setw(25)<<left<<b[i].title<<setw(10)<<left<<b[i].author<<"t"<<b[i].price<<"t"<<b[i].count<<"n";
}
kol_books(b, N);
max_price(b, N);
fin.close();           //закрыть файл
bookf.close();
return 0;
}

```

### Пример 10.2. (С типом **string**)

Создать структуру Книга: название, имя автора, цена, количество экземпляров книги в книжном магазине. Найти самую дорогую книгу. Подсчитать количество книг каждого автора. Найти автора, имеющего максимальное количество книг.

```

#include <fstream>
#include <iostream>
#include <iomanip>
#include <string>
#include <windows.h>

```

```
using namespace std;
```

```

struct book
{
    string title;
    string author;
    float price;
    int count;
};
// сортировка массива структур по полю «автор»
void sort(book arr[], int size)
{
    for (int i = size - 1; i > 0; --i)
        for (int j = 0; j < i; ++j)
            if (arr[j].author > arr[j + 1].author)
            {
                book tmp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = tmp;
            }
}
// поиск самой дорогой книги
void max_price(book arr[], int size)
{
    float max = arr[0].price;
    string max_title = arr[0].title;
    string max_author = arr[0].author;
    for (int j = 0; j < size; ++j)
        if (max < arr[j].price)
        {
            max = arr[j].price;
            max_title = arr[j].title;
            max_author = arr[j].author;
        }
    cout << "\nСАМАЯ ДОРОГАЯ КНИГА - ";
    cout << max_title;
    cout << " " << max_author << " ЕЕ ЦЕНА " << max << "\n";
}

```

// подсчет количество книг каждого автора. Поиск автора, имеющего максимальное количество книг.

```

void kol_books(book arr[], int size)
{
    string auth, max_auth;
    int i = 0, max_c = 0;
    cout << "\n" << setw(25) << left << "АВТОР" << setw(25) << left << "КОЛИЧЕСТВО КНИГ" << "\n\n";
}

```

```

while (i < size)    // проход по массиву
{
    auth=arr[i].author;
    int c = 0;
    while (arr[i].author==auth)
    {
        c += arr[i].count; // подсчет количества книг автора auth
        ++i;
    }
    cout << setw(25) << left << auth << setw(25) << left << c << "\n";
    //поиск автора с максимальным количеством книг
    if (c > max_c)
    {
        max_c = c;
        max_auth=auth;
    }
}
cout << "\n" << "АВТОР С МАКСИМАЛЬНЫМ КОЛИЧЕСТВОМ КНИГ (" << max_c << ") " << max_auth
<< "\n\n";
}

int main()
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    ifstream fin("d:\\my_books.txt");    //исходный файл
    ofstream bookf;                      //файл для записи отсортированного массива
    bookf.open("c:\\BOOKINFO1.txt");
    if (!fin.is_open())
    {
        cout << "Файл не найден";
        return 1;
    }
    book b[10];    //массив из 10 структур book
    int N = 8;    //первое значение в файле - размер массива структур
    for (int i = 0; i < N; i++)
    {
        fin.ignore();    //пропустить символ '\n' (значения в файле разделены переводом строки)
        getline(fin, b[i].title, '\n'); //fin >> b[i].title;    //прочитать строку с названием книги
        getline(fin, b[i].author, '\n'); //fin >> b[i].author;    //прочитать строку с именем автора
        fin >> b[i].price;    // прочитайте строку с ценой книги
        fin >> b[i].count;    // прочитайте строку с числом экземпляров книги
    }
    cout << "ИСХОДНЫЙ МАССИВ\n\n";
    cout << setw(25) << left << "title" << setw(10) << left << "author" << "\t" << "price" << "\t" << "count" <<
    "\n\n";
    //вывод массива на экран
    for (int i = 0; i < N; i++)
        cout << setw(25) << left << b[i].title << setw(10) << left << b[i].author << "\t" << b[i].price << "\t" <<
    b[i].count << "\n";
    sort(b, N);
    cout << "\nМАССИВ ОТСОРТИРОВАННЫЙ ПО ПОЛЮ count\n\n";
    //вывод массива на экран и запись в файл
    for (int i = 0; i < N; i++)
    {
        cout << setw(25) << left << b[i].title << setw(10) << left << b[i].author << "\t" << b[i].price << "\t" <<
    b[i].count << "\n";
        bookf << b[i].title << '\n' << b[i].author << '\n' << b[i].price << '\n' << b[i].count << '\n';
    }
    kol_books(b, N);
    max_price(b, N);
    fin.close();    //закрывать файл
    bookf.close();
    return 0;
}

```

### 10.3 Варианты заданий

Заготовить файл (количество структур в файле должно быть не менее 10), а затем выполнить обработку этого файла в соответствии с заданием.

1. Создать файл с полями: шифр товара, поставщик, стоимость партии товара. Определить товар с самой большой стоимостью. Определить, на какую сумму получены товары от каждого поставщика.
2. Создать файл с полями: фамилия спортсмена, вид спорта, разряд. Вывести спортсменов - перворазрядников. Определить вид спорта, которым занимается наибольшее количество человек.
3. Создать файл с полями: название фильма, жанр, год выпуска. Вывести фильмы, выпущенные с 2002 по 2004 годы. Определить самый популярный жанр фильма.
4. Создать файл с полями: название команды, страна, количество очков. Подсчитать среднее количество очков всех команд. Определить суммарное количество очков, которое набрали команды каждой страны
5. Создать файл с полями: фамилия, количество книг, группа. Вывести список студентов, сдавших все книги. Определить группу, студенты которой взяли наибольшее количество книг.
6. Создать файл с полями: группа, предмет, количество неудовлетворительных оценок. Определить группу и предмет, по которому получено больше всего двоек. Создать новый файл, содержащий информацию о количестве двоек в каждой группе.
7. Создать файл с полями: номер отдела, фамилия сотрудника, наличие ученой степени. Вывести список сотрудников с ученой степенью. Определить отдел, в котором больше всего сотрудников.
8. Создать файл с полями: фамилия, сумма долга, номер ЖЭКа. Вывести список должников. Определить сумму задолженностей по каждому ЖЭКу.
9. Создать файл с полями: номер рейса, цена билета, тип самолета. Вывести рейсы, на которые цена билета минимальна. Определить количество самолетов каждого типа.
10. Создать файл с полями: группа, предмет, количество неудовлетворительных оценок. Вывести информацию о результатах экзаменов по заданному предмету. Определить группу, в которой получено наибольшее количество двоек.
11. Создать файл с полями: марка автомобиля, год выпуска, фамилия владельца. Вывести автомобили заданной марки. Определить владельцев, имеющих более одного автомобиля.
12. Создать файл с полями: фамилия, класс, буква, средний балл. Вывести список лучших учеников. Определить лучший класс по успеваемости.
13. Создать файл с полями: марка автомобиля, его стоимость, фамилия владельца. Подсчитать общую стоимость автомобилей. Определить самую популярную марку автомобиля.
14. Создать файл с полями: название ВУЗа, город, количество студентов. Определить среднее количество студентов в ВУЗах. Создать новый файл с информацией о количестве ВУЗов в каждом из городов.
15. Создать файл с полями: название альбома, название музыкального произведения, фамилия исполнителя, длительность произведения. Вывести список исполнителей указанного альбома. Определить исполнителя с наибольшим количеством записей.
16. Создать файл с полями: наименование товара, количество, наименование магазина, в который товар поставлен. Определить максимальный объем поставки в магазин. Подсчитать количество поставок в каждый магазин.
17. Создать файл с полями: фамилия спортсмена, год рождения, вид спорта. Вывести список самых молодых спортсменов. Подсчитать количество спортсменов каждого вида спорта.
18. Создать файл с полями: фамилия студента, дата рождения. Вывести всех студентов, родившихся летом. Подсчитать количество студентов, родившихся в каждом году.
19. Создать файл с полями: фамилия студента, группа, три оценки, полученные студентом в последнюю сессию. Вывести средние баллы каждого студента. Подсчитать количество студентов в каждой группе.
20. Создать файл с полями: фамилия сотрудника, номер цеха, должность. Вывести список сотрудников заданного цеха. Подсчитать количество инженеров в каждом цехе.
21. Создать файл с полями: название магазина, месяц, объем продаж за этот месяц. Подсчитать количество записей о заданном магазине. Определить месяц, в котором общий объем продаж был максимальным.
22. Создать файл с полями: название товара, производитель, цена. Определить самый дешевый товар. Подсчитать количество наименований товара каждого производителя.
23. Создать файл с полями: страна, город, количество жителей в городе. Вывести список городов указанной страны. Подсчитать общее количество жителей городов каждой страны.
24. Создать файл с полями: фамилия кандидата, возраст, партия. Вывести список кандидатов моложе 25 лет. Определить партию с наибольшим количеством кандидатов.

25. Создать файл с полями: фамилия студента, средний балл, место практики. Вывести список студентов, которые проходят практику на НПЗ. Подсчитать количество студентов- практикантов на каждом предприятии.

#### **10.4 Контрольные вопросы**

1. Что такое структура?
2. Какого типа могут быть поля структуры?
3. Правила обращения к полям структуры.
4. Какой заголовочный файл необходимо подключить для работы с файлами?
5. Какие методы используют для открытия и закрытия файлов?
6. Как проверить достижение конца файла?
7. В чем заключается сортировка массивов структур?
8. Что изменится в приведенном выше примере, если сортировка будет выполнена по убыванию, а не по возрастанию?
9. Как выполняется сравнение строк?