**Лабораторная работа №16.**

**СОЗДАНИЕ И ОБРАБОТКА БИНАРНЫХ ФАЙЛОВ**

**Цель работы**: Приобрети навыков разработки алгоритмов обработки файлов.

Если в файле хранятся только числа или данные определённой структуры, то для хранения таких значений удобно пользоваться двоичными (бинарными) файлами.

В бинарных файлах *информация* считывается и записывается в виде блоков определённого размера, в них могут храниться данные любого вида и структуры.

Кроме того, бинарные  файлы относятся к структурам прямого доступа, это означает, что в любой момент времени можно обратиться к любому, а не только к текущему элементу бинарного файла.

Пример программы для работы с бинарными файлами

1. Создать файл lib.dat, содержащий 10 записей следующей структуры: автор книги, название книги, год издания, количество страниц.

#include<iostream>

#include<fstream>

#include<cstring>

#include<windows.h>

using namespace std;

struct Book // Объявляется структура

{

char author[40];

char title[80];

int year;

int pages;

};

int main()

{

setlocale(LC\_ALL, "rus");

SetConsoleCP(1251); // установка кодовой страницы win-cp 1251 в поток ввода

SetConsoleOutputCP(1251); // установка кодовой страницы win-cp 1251 в поток вывода

Book b[50], bk, bk1, bk2;

int i = 0, k = 0, n = 10;

// Заполнение массива структур данными с клавиатуры

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

{

cout<< "автор ";

cin.getline(b[i].author, 40);

cout<< "название книги: ";

cin.getline(b[i].title, 80);

cout<< "год издания: ";

cin>> b[i].year;

cout<< "количество страниц: ";

cin>> b[i].pages;

cin.get();

}

// запись структуры в файл

ofstream f2\_out("lib.dat", ios::binary);

f2\_out.write((char\*)&b, n\*sizeof(Book));

f2\_out.close();

system("pause");

return 0;

}

Для проверки корректности записи данных в файл рекомендуется организовать их чтение из файла и вывод данных на экран

//Чтение данных из файла 1 способ

ifstream f2\_in("lib.dat", ios::binary);

while (f2\_in.read((char\*)&bk, sizeof(Book)))

{

cout<<bk.author<< " " <<bk.title<< " " <<bk.year<< " " <<bk.pages<<endl;

}

f2\_in.close();

cout<<endl;

//Чтение данных из файла 2 способ

ifstream f2\_in("lib.dat", ios::binary);

f2\_in.seekg(0, ios\_base::end); // становимся в конец файла

cout<<"Размер файла (в байтах): "<< f2\_in.tellg()<<endl; //Получаем текущую позицию

int N = f2\_in.tellg() / sizeof(Book);

cout<<"Количество записей в базе данных"<< N <<endl;

f2\_in.seekg(0, ios\_base::beg); // становимся в начало файла

for (inti = 0; i<N; i++)

{

f2\_in.read((char\*)&bk, sizeof(Book));

cout<<bk.author<<" "<<bk.title<<" "<<bk.year<<" "<<bk.pages<<endl;

}

f2\_in.close();

1. Написать программу, выполняющую следующую обработку файла lib.dat: · поиск в файле данных о книгах определенного автора, фамилия которого вводится с клавиатуры; если таких книг нет, выдать соответствующее сообщение на дисплей.

#include<iostream>

#include<fstream>

#include<cstring>

#include<windows.h>

using namespace std;

struct Book // Объявляется структура

{

char author[40];

char title[80];

int year;

int pages;

};

int main()

{

setlocale(LC\_ALL, "rus");

SetConsoleCP(1251); // установка кодовой страницы win-cp 1251 в поток ввода

SetConsoleOutputCP(1251); // установка кодовой страницы win-cp 1251 в поток вывода

Book bk, bk1, bk2;

char avtor[40];

int i = 0, k = 0, n = 10;

// Поиск по автору

ifstream f2\_in("lib.dat", ios::binary);

cout<<"автор ";

cin.getline(avtor, 40);

for (inti = 0; i< N; i++)

{

f2\_in.read((char\*)&bk, sizeof(Book));

if(strstr(bk.author,avtor)!=NULL)

{

cout<<bk.author<<" "<<bk.title<<" "<<bk.year<<" "<<bk.pages<<endl;

k++;

}

}

f2\_in.close();

if(k==0) cout<<"Книги указанного автора не найдены "<<endl;

system("pause");

return 0;

}

Дополнительные приемы, которые могут пригодиться при работе с бинарными файлами

Считывание выборочной записи из файла

// считывание 4 записи

ifstream f2\_in("lib.dat", ios::binary);

intpos = 3 \* sizeof(Book);

f2\_in.seekg(pos);

f2\_in.read((char\*)&bk, sizeof(Book));

cout<<bk.author<< " " <<bk.title<< " " <<bk.year<< " " <<bk.pages<<endl;

// Считывание последней записи

pos= sizeof(Book);

f2\_in.seekg(-pos, ios::end);

f2\_in.read((char\*)&bk, sizeof(Book));

cout<<bk.author<< " " <<bk.title<< " " <<bk.year<< " " <<bk.pages<<endl;

// Считывание 3 с конца записи

pos = 3\*sizeof(Book);

f2\_in.seekg(-pos, ios::end);

f2\_in.read((char\*)&bk, sizeof(Book));

cout<<bk.author<< " " <<bk.title<< " " <<bk.year<< " " <<bk.pages<<endl;

f2\_in.close();

// Вывод информации из файла в обратном порядке

f2\_in.open("lib.dat", ios::binary);

f2\_in.seekg(0, ios::end); //расчет объема данных в файле в байтах

int s = f2\_in.tellg()/sizeof(Book);//Расчет количества записей

intpos;

for (inti = 1; i<= 5; i++)

{

pos = i \* sizeof(Book);

f2\_in.seekg(-pos, ios::end);

f2\_in.read((char\*)&bk, sizeof(Book));

cout<<bk.author<< " " <<bk.title<< " " <<bk.year<< " " <<bk.pages<<endl;

}

f2\_in.close();

// Редактирование 3 записи

fstream f("lib.dat", ios::binary| ios::in | ios::out);

intpos = 2 \* sizeof(Book);

f.seekp(pos);

bk = { "Булгаков М.А", "Собачье сердце", 2010, 500 };

f.write((char\*)&bk, sizeof(Book));

f.close();

// Перестановка первой и пятой записей

fstream f("lib.dat", ios::binary | ios::in | ios::out);

f.read((char\*)&bk1, sizeof(Book));

intpos = 4 \* sizeof(Book);

f.seekp(pos);

f.read((char\*)&bk2, sizeof(Book));

f.seekp(0);

f.write((char\*)&bk2, sizeof(Book));

f.seekp(pos);

f.write((char\*)&bk1, sizeof(Book));

f.close();

**Задание к лабораторной работе**

В соответствии с вариантом задания разработать две программы: программу создания и программу обработки бинарного файла.

**Варианты заданий:**

|  |  |
| --- | --- |
| Вариант | Условие задачи |
|  | 1.Создать файл Train.dat, содержащий 8 записей следующей структуры: название пункта назначения; номер поезда; время отправления.  2.Написать программу, выполняющую следующую обработку файла Train.dat: · поиск в файле и вывод на экран информации о поезде, номер которого введен с клавиатуры; · если таких поездов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Spravka.dat, содержащий 10 записей следующей структуры: название начального пункта маршрута; название конечного пункта маршрута; номер маршрута;  2.Написать программу, выполняющую следующую обработку файла Spravka.dat: · поиск в файле данных о маршруте, номер которого вводится с клавиатуры; если таких маршрутов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Train.dat, содержащий 8 записей следующей структуры: название пункта назначения; номер поезда; время отправления.  2.Написать программу, выполняющую следующую обработку файла Train.dat: · поиск в файле поездов, отправляющихся после введенного с клавиатуры времени; · если таких поездов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Spravka.dat, содержащий 10 записей следующей структуры: название начального пункта маршрута; название конечного пункта маршрута; номер маршрута;  2.Написать программу, выполняющую следующую обработку файла Spravka.dat: · поиск в файле данных о маршрутах, которые начинаются или заканчиваются в пункте, название которого вводится с клавиатуры; · если таких маршрутов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Wedomost.dat, содержащий 10 записей следующей структуры: фамилия и инициалы студента; номер группы; успеваемость по трем предметам; 2.Написать программу, выполняющую следующую обработку файла Wedomost.dat: · поиск в файле информации о студентах, имеющих хотя бы одну оценку 2; · если таких студентов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Spravka.dat, содержащий 10 записей следующей структуры: название пункта назначения; номер рейса; тип самолета.  2.Написать программу, выполняющую следующую обработку файла Spravka.dat: · поиск в файле номеров рейсов, вылетающих в пункт, название которого вводится с клавиатуры; · если таких рейсов нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл Wedomost.dat, содержащий 8 записей следующей структуры: фамилия и инициалы студента; номер группы; успеваемость по трем предметам; 2. Написать программу, выполняющую следующую обработку файла Wedomost.dat: · поиск в файле информации о студентах, имеющих только оценки 4 и 5; · если таких студентов нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл Wedomost.dat, содержащий 6 записей следующей структуры: фамилия и инициалы студента; номер группы; успеваемость по трем предметам; 2. Написать программу, выполняющую следующую обработку файла Wedomost.dat: · поиск в файле информации о студентах, имеющих средний балл меньше 4; · если таких студентов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл Spravka.dat, содержащий 10 записей следующей структуры: название пункта назначения; номер рейса; тип самолета. 2.Написать программу, выполняющую следующую обработку файла Spravka.dat: · поиск в файле номеров рейсов, обслуживаемых самолетом, тип которого вводится с клавиатуры; · если таких рейсов нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл Train.dat, содержащий записи следующей структуры: название пункта назначения; номер поезда; время отправления;  2. Написать программу, выполняющую следующую обработку файла Train.dat: · поиск в файле поездов, отправляющихся в пункт, название которого вводится с клавиатуры; · если таких поездов нет, выдать соответствующее сообщение на дисплей. |
|  | 1.Создать файл WORKER.dat, содержащий 6 записей следующей структуры: фамилия и инициалы; номер телефона; день рождения (массив из трех чисел);  2. Написать программу, выполняющую следующую обработку файла WORKER.dat: · поиск в файле информации о человеке, чья фамилия введена с клавиатуры; · если такого человека нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл STUDENT.dat , содержащий записи следующей структуры: ФИО студента; его экзаменационные оценки по трем дисциплинам. 2. Написать программу, которая выбирает из файла студентов, имеющих хотя бы одну задолженности. Вывести ФИО этих студентов и количество несданных экзаменов. |
|  | 1. Создать файл F1.dat, содержащий 8 записей следующей структуры: ФИО; номер телефона; день рождения (массив из трех чисел)  2. Написать программу, которая переписывает файл F1.dat в файл F2.dat таким образом, чтобы записи расположились а алфавитном порядке. |
|  | 1. Создать файл ZNAK.dat, содержащий записи следующей структуры: фамилия и инициалы; знак Зодиака; день рождения (массив из трех чисел);  2. Написать программу, выполняющую следующие действия: · поиск в файле ZNAK.dat информации о людях, родившихся под знаком, название которого введено с клавиатуры; · если таких людей нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл Work.dat, содержащий 6 записей следующей структуры: ФИО рабочих; их среднемесячный заработок.  2. Написать программу, выполняющую следующую обработку файла Work.dat: Вывести ФИО рабочих, имеющих наибольший заработок. |
|  | 1. Создать файл WORKER.dat, содержащий 5 записей следующей структуры: фамилия и инициалы; номер телефона; день рождения (массив из трех чисел); 2. Написать программу, выполняющую следующую обработку файла WORKER.dat: · поиск в файле информации о людях, родившихся в месяц, номер которого введен с клавиатуры; · если таких людей нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл ABONENT.dat, содержащий 5 записей следующей структуры: ФИО абонента; его номер телефонов.  2. Составить программу, которая по ФИО абонента, введенной с клавиатуры, выводит его номер телефона. · если такого абонента нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл AEROFLOT.dat, содержащий 6 записей следующей структуры: номер рейса самолета; количество свободных мест.  2. Составить программу, которая по вводимому с клавиатуры номеру рейса выводит из файла сведе­ния о количестве свободных мест на этот рейс. · если такого рейса нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл PRICE.dat, содержащий 5 записей следующей структуры: название детской игрушки; ее стоимость.  2. Написать программу, выполняющую следующую обработку файла PRICE.dat: Вывести наименование и стоимость самой дорогой игрушки. |
|  | 1. Создать файл PRICE.dat, содержащий 5 записей следующей структуры: название детской игрушки, ее стоимость, название магазина, в котором она продается.  2. Написать программу, выполняющую следующую обработку файла PRICE.dat: Вывести полную информацию об игрушке, название которой введено с клавиатуры Если игрушки такой нет в PRICE, выдать сообщение на дисплей. |
|  | 1. Создать файл PriceList.dat, содержащий записи следующей структуры: название товара; название магазина, в котором он продается; стоимость;  2. Написать программу, выполняющую следующую обработку файла PriceList.dat: · поиск в файле информации о товаре, название которого введено с клавиатуры; · если такого товара нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл PriceList.dat, содержащий записи следующей структуры: название товара; название магазина, в котором он продается; стоимость; 2. Написать программу, выполняющую следующую обработку файла PriceList.dat: · поиск в файле информации о товарах, имеющихся в магазине, название которого введено с клавиатуры; · если такого магазина нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл ORDER.dat, содержащий записи следующей структуры: расчетный счет плательщика; расчетный счет получателя; перечисляемая сумма. 2. Написать программу, выполняющую следующую обработку файла ORDER.dat: · поиск в файле информации о сумме и расчетном счете получателя, которому перечислил эту сумму плательщик, номер которого введен с клавиатуры; · если такого плательщика нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл ZNAK.dat, содержащий записи следующей структуры: фамилия, имя; знак Зодиака; день рождения (массив из трех чисел); 2. Написать программу, выполняющую следующую обработку файла ZNAK.dat: · поиск в файле информации о людях, родившихся в одно и тоже число, значение которого введено с клавиатуры; · если таких людей нет, выдать соответствующее сообщение на дисплей. |
|  | 1. Создать файл F1.dat, содержащий 8 записей следующей структуры: ФИО; номер телефона; день рождения (массив из трех чисел) 2. Написать программу, которая переписывает файл F1.dat в файл F2.dat таким образом, чтобы записи расположились в порядке убывания дат рождений. |