МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ   
РОССИЙСКОЙ ФЕДЕРАЦИИ  
  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА С.П. КОРОЛЕВА(САМАРСКИЙ УНИВЕРСИТЕТ)  
  
Институт информатики и кибернетики  
Кафедра программных систем

Дисциплина  
**Объектно-ориентированное программирование**

**ОТЧЕТ**по лабораторной работе № 3  
 **Объектно-ориентированное программирование обработки строк с использованием функций стандартной библиотеки языка С++ в стиле языка С.**  
Вариант № 12

Студент: Мананников М.А.  
Группа: 6214-020302D  
  
Преподаватель: Баландин А.В.  
  
Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
  
Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Самара 2022

**Задание**

Реализовать информационную базу, состоящую из трёх файлов: FIO.DAT - содержит в виде символьных строк, оканчивающихся , список фамилий спортсменов; SPORT.DAT - содержит в виде символьных строк наименования видов спорта, которые не повторяются; LINK.IDX - содержит записи вида struct {unsigned long name\_idx; //индекс записи в файле FIO.DAT; unsigned long sport\_idx; //индекс записи в файле SPORT.DAT}, связывающие между собой фамилию и вид спорта. В качестве индексов используются значения позиций, с которых начинаются записи в соответствующих файлах. Записи в FIO.DAT и в LINK.IDX взаимно-однозначно соответствуют друг другу. Программа должна обеспечивать выполнение следующих функций: 1) включение новой фамилии спортсмена с указанием вида спорта, 2) добавление нового вида спорта, 3) по заданной фамилии спортсмена вывод на экран вида спорта, которым тот занимается.

**Объектно-ориентированный анализ (ООА)**

**Информационная модель предметной области**

Информационный объект Файл записей:

* Свойства:
* Имена файлов;
* Набор записей.
* Процедуры:
* Конструктор без параметров, пополняющий базу данных;
* Добавление спортсмена;
* Добавление вида спорта;
* Поиск вида спорта, которым занимается спортсмен с введённой фамилией.

**Объектно-ориентированное проектирование (ООD)**

* Свойства:
* Имена файлов - символьная строка;
* Набор записей. – набор строк;
* Процедуры:
* Конструктор без параметров, пополняющий базу данных – конструктор без параметров;
* Добавление спортсмена – процедура;
* Добавление вида спорта– процедура;
* Поиск вида спорта, которым занимается спортсмен с введённой фамилией– процедура;

**Структура алгоритма программы**

1. Создание и заполнение файлов FIO.dat, SPORT.dat и LINK.idx ;
2. Выполнение алгоритма добавления сотрудника;
3. Выполнение алгоритма добавления вида спорта;
4. Выполнение алгоритма поиска вида спорта по фамилии;
5. Завершение программы.

**Код программы**

Program.cpp

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#pragma once

#include <string.h>//библиотека для работы со си-строками

#include <stdio.h>// стандартная библиотека ввода\вывода

#include <stdlib.h>// поиск, сортировка и тд

#include <Windows.h>// использовать функионал ОС

class DataBase

{

FILE\* FIO\_DAT;// файл с фамилиями

FILE\* SPORT\_DAT;//файл с видами спорта

FILE\* LINK\_IDX;

int numbsp;// количество спортсменов

struct Index

{

unsigned long name\_idx; //индекс записи в файле FIO.DAT;

unsigned long sport\_idx;//индекс записи в файле SPORT.DAT},

};

Index sp;// создаем структуру

public: DataBase()//конструктор без параметров, пополняющий базу данных

{

FIO\_DAT = fopen("Fio.dat", "w+t");// открываем файл для записи фамилий

SPORT\_DAT = fopen("Sport.dat", "w+t");// открываем файл для записи видов спорта

LINK\_IDX = fopen("Link.idx", "w+b");// открываем файл для записи индексов

puts("Введите количество спортсменов:");

scanf("%d", &numbsp);

for (int i = 0;i < numbsp;i++)// цикл записи одного спортсмена

{

int y = 0;

char\* surname = (char\*)malloc(40);// переменная для записи фамилии

puts("Введите фамилию спортсмена");

scanf("%s", surname);//считываем фамилию

strcat(surname, "\n");// добавляем пропуск

if (i == 0)

{

sp.name\_idx = ftell(FIO\_DAT);// запись индекса фамилии в поле структуры

}

else

{

sp.name\_idx = ftell(FIO\_DAT) - 2 \* i;

}

fwrite(surname, sizeof(char), strlen(surname), FIO\_DAT);// запись фамилии в файл

puts("Введите вид спорта ");

char\* sport = (char\*)malloc(40);

scanf("%s", sport);

strcat(sport, "\n");

if (ftell(SPORT\_DAT) == 0)// если файл пуст, то записываем вид спорта

{

sp.sport\_idx = ftell(SPORT\_DAT);//запись индекса спорта в поле структуры

fwrite(sport, sizeof(char), strlen(sport), SPORT\_DAT);// запись спорта в файл

}

else// иначе проверяем на повторность

{

Index\* indmas = (Index\*)malloc(numbsp);// массив индексов

fseek(LINK\_IDX, 0, SEEK\_SET);

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

{

fread((char\*)&indmas[j].name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данные из файла в массив

fread((char\*)&indmas[j].sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данные из файла в массив

}

fseek(SPORT\_DAT, 0, SEEK\_END);

int k = ftell(SPORT\_DAT);

fseek(SPORT\_DAT, 0, SEEK\_SET);

for (int j = 0;j < i - 1;j++)

{

char\* p = (char\*)malloc(20);

fread(p, (indmas[j + 1].sport\_idx - indmas[j].sport\_idx), 1, SPORT\_DAT);

p[indmas[j + 1].sport\_idx - indmas[j].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[j].sport\_idx;// запись индекса спорта

y++;

}

}

char\* p = (char\*)malloc(20);

fread(p, (k - indmas[i - 1].sport\_idx), 1, SPORT\_DAT);

p[k - indmas[i - 1].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[i - 1].sport\_idx;// запись индекса спорта

y++;

}

if (y == 0)// если не нашли похожих

{

fseek(SPORT\_DAT, 0, SEEK\_END);

sp.sport\_idx = ftell(SPORT\_DAT);// запись индекса спорта

fwrite(sport, sizeof(char), strlen(sport), SPORT\_DAT);//запись спорта в файл

}

}

fwrite((char\*)&sp.name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//запись индекса имени

fwrite((char\*)&sp.sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//запись индекса спорта

}

}

void FindSp()// процедура поиска спортсмена

{

Index\* indmas = (Index\*)malloc(numbsp);// массив индексов

fclose(LINK\_IDX);

LINK\_IDX = fopen("Link.idx", "rb");

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

{

fread((char\*)&indmas[i].name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данных из файла в массив

fread((char\*)&indmas[i].sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данных из файла в массив

}

char\* surname = (char\*)malloc(40);

puts("Введите фамилию спортсмена");

scanf("%s", surname);

fseek(FIO\_DAT, 0, SEEK\_END);

int k = ftell(FIO\_DAT);

fclose(FIO\_DAT);

FIO\_DAT = fopen("Fio.dat", "r+t");//открытие файла на чтение

for (int j = 0;j < numbsp - 1;j++)

{

fseek(FIO\_DAT, indmas[j].name\_idx, SEEK\_SET);

char\* p = (char\*)malloc(40);

fread(p, (indmas[j + 1].name\_idx - indmas[j].name\_idx), 1, FIO\_DAT);

p[indmas[j + 1].name\_idx - indmas[j].name\_idx] = '\0';

if (strcmp(p, surname) == 0)// сравнение слова и введенной фамилии

{

fseek(SPORT\_DAT, indmas[j].sport\_idx, 0);// поиск по индексу вида спорта

char\* k = (char\*)malloc(20);

fread(k, (indmas[j + 1].sport\_idx - indmas[j].sport\_idx), 1, SPORT\_DAT);

k[indmas[j + 1].sport\_idx - indmas[j].sport\_idx - 1] = '\0';

puts(k);//вывод спорта

}

free(p);//освобождение памяти

}

fseek(FIO\_DAT, 0, SEEK\_END);

unsigned long m = ftell(FIO\_DAT);

fseek(FIO\_DAT, indmas[numbsp - 1].name\_idx, SEEK\_SET);

char\* p = (char\*)malloc(20);

fread(p, (m - indmas[numbsp - 1].name\_idx), 1, FIO\_DAT);

p[m - indmas[numbsp - 1].name\_idx] = '\0';

if (strcmp(p, surname) == 0)// сравнение слова и введенного спорта

{

fseek(SPORT\_DAT, 0, SEEK\_END);

unsigned long t = ftell(SPORT\_DAT);

fseek(SPORT\_DAT, indmas[numbsp - 1].sport\_idx, SEEK\_SET);

char\* p = (char\*)malloc(20);

fread(p, (t - indmas[numbsp - 1].sport\_idx), 1, SPORT\_DAT);

p[t - indmas[numbsp - 1].sport\_idx] = '\0';

puts(p);

}

}

void AddSport()// процедура добавления спорта

{

Index sp;

int y = 0;

Index\* indmas = (Index\*)malloc(numbsp);// массив индексов

fseek(LINK\_IDX, 0, SEEK\_SET);

for (int j = 0;j < numbsp;j++)

{

fread((char\*)&indmas[j].name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данне из файла в массив

fread((char\*)&indmas[j].sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данне из файла в массив

}

puts("Введите вид спорта ");

char\* sport = (char\*)malloc(40);

scanf("%s", sport);

fseek(SPORT\_DAT, 0, SEEK\_END);

int k = ftell(SPORT\_DAT);

fseek(SPORT\_DAT, 0, SEEK\_SET);

for (int j = 0;j < numbsp - 1;j++)

{

char\* p = (char\*)malloc(20);

fread(p, (indmas[j + 1].sport\_idx - indmas[j].sport\_idx), 1, SPORT\_DAT);

p[indmas[j + 1].sport\_idx - indmas[j].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[j].sport\_idx;// запись индекса спорта

y++;

}

}

char\* p = (char\*)malloc(20);

fread(p, (k - indmas[numbsp - 1].sport\_idx), 1, SPORT\_DAT);

p[k - indmas[numbsp - 1].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[numbsp - 1].sport\_idx;// запись индекса спорта

y++;

}

if (y == 0)// если не нашли похожих

{

fseek(SPORT\_DAT, 0, SEEK\_END);

sp.sport\_idx = ftell(SPORT\_DAT);// запись индекса спорта

fwrite(sport, sizeof(char), strlen(sport), SPORT\_DAT);//запись спорта в файл

}

}

public: void AddFio()//процедура добавления спортсмена

{

Index sp;

int y = 0;

char\* surname = (char\*)malloc(40);// переменная для записи фамилии

puts("Введите фамилию спортсмена");

scanf("%s", surname);//считываем фамилию

strcat(surname, "\n");// добавляем пропуск

sp.name\_idx = ftell(FIO\_DAT) - 2 \* numbsp;// запись индекса фамилии в поле структуры

fwrite(surname, sizeof(char), strlen(surname), FIO\_DAT);// запись фамилии в файл

puts("Введите вид спорта ");

char\* sport = (char\*)malloc(40);

scanf("%s", sport);

if (ftell(SPORT\_DAT) == 0)// если файл пуст, то записываем вид спорта

{

sp.sport\_idx = ftell(SPORT\_DAT);//запись индекса спорта в поле структуры

fwrite(sport, sizeof(char), strlen(sport), SPORT\_DAT);// запись спорта в файл

}

else// иначе проверяем на повторность

{

Index\* indmas = (Index\*)malloc(numbsp);// массив индексов

fseek(LINK\_IDX, 0, SEEK\_SET);

for (int j = 0;j < numbsp;j++)

{

fread((char\*)&indmas[j].name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данне из файла в массив

fread((char\*)&indmas[j].sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//считываем данне из файла в массив

}

fseek(SPORT\_DAT, 0, SEEK\_END);

int k = ftell(SPORT\_DAT);

fseek(SPORT\_DAT, 0, SEEK\_SET);

for (int j = 0;j < numbsp - 1;j++)

{

char\* p = (char\*)malloc(20);

fread(p, (indmas[j + 1].sport\_idx - indmas[j].sport\_idx), 1, SPORT\_DAT);

p[indmas[j + 1].sport\_idx - indmas[j].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[j].sport\_idx;// запись индекса спорта

y++;

}

}

char\* p = (char\*)malloc(20);

fread(p, (k - indmas[numbsp - 1].sport\_idx), 1, SPORT\_DAT);

p[k - indmas[numbsp - 1].sport\_idx] = '\0';

if (strcmp(p, sport) == 0)// сравнение слова и введенного спорта

{

sp.sport\_idx = indmas[numbsp - 1].sport\_idx;// запись индекса спорта

y++;

}

if (y == 0)// если не нашли похожих

{

fseek(SPORT\_DAT, 0, SEEK\_END);

sp.sport\_idx = ftell(SPORT\_DAT);// запись индекса спорта

fwrite(sport, sizeof(char), strlen(sport), SPORT\_DAT);//запись спорта в файл

}

}

fwrite((char\*)&sp.name\_idx, sizeof(unsigned long), 1, LINK\_IDX);//запись индекса имени

fwrite((char\*)&sp.sport\_idx, sizeof(unsigned long), 1, LINK\_IDX);//запись индекса спорта

numbsp++;

}

public:~DataBase()//деструктор

{

fclose(FIO\_DAT);

fclose(LINK\_IDX);

fclose(SPORT\_DAT);

}

};

int main()

{

system("chcp 1251");

system("cls");

printf("Лабораторная работа 3 (12 вариант)\nПодготовил: Мананников Максим, студент группы 6103\n");

DataBase check;//создание объекта

printf("Добавление спортсмена\n");

check.AddFio();//процедура добавления спортсмена

printf("Добавление спорта\n");

check.AddSport();//процедура добавления спорта

printf("Узнать вид спорта, которым занимается выбранный спортсмен\n");

check.FindSp();//процедура поиска спортсмена

system("pause");

return 0;

}

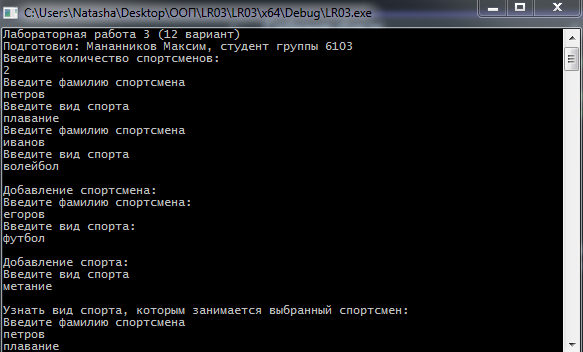


Рисунок 1 **–** Пример выполнения программы.

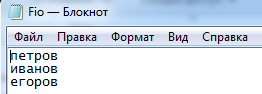


Рисунок 2 – файл FIO.DAT

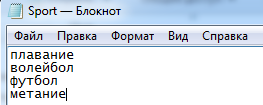


Рисунок 3 – файл SPORT.DAT