**ПРИЛОЖЕНИЕ А**

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

# КАФЕДРА ИНТЕЛЛЕКТУАЛЬНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ОБРАБОТКА МАССИВОВ СТРУКТУРИРОВАННЫХ ДАННЫХ

**«**МИРОВЫЕ РЕКОРДЫ В БЕГЕ 100 МЕТРОВ**»**

**ТЕКСТ ПРОГРАММЫ**

КР.ПО7.200149 – 01 12 00

## Листов 10

|  |  |
| --- | --- |
| Руководитель | О.Ю. Войцехович |
| Выполнил | А.Е. Комиссаров |
| Консультант |  |
| по ЕСПД | О.Ю. Войцехович |
|  |  |

2021

**СОДЕРЖАНИЕ**

Source.cpp – основной файл в котором содержится меню.

Kurs\_functions.h, kurs\_functions.cpp – библиотека в которой содержатся все пользовательские функции используемые программой.

**Source.cpp**

#include "kurs\_functions.h"

int main() {

setlocale(LC\_ALL, "Russian"); SetConsoleOutputCP(1251); SetConsoleCP(1251);

cout << "Вариант номер шесть. Массив данных – «Мировые рекорды в беге на 100 метров»." << endl;

int N, truefalse, act, exec = 1;

cout << "\nВведите N (количество записей): ";

N = getValue\_ArraySize();

records\* main\_array = new records[N];

inputall(main\_array, N); outputall(main\_array, N);

while (exec > 0) {

cout << "\nПродолжить выполнение программы? (1 - да, 0 - нет): ";

truefalse = getValue\_int();

if (truefalse == 1){

print\_menu();

act = getValue\_int();

if (act == 0 || act == 1 || act == 2 || act == 3 || act == 4 || act == 5 || act == 6 || act == 7 || act == 8 || act == 9) {

if (act == 0) { exec = 0; }

if (act == 1) { openfile(main\_array, N); }

if (act == 2) { addelement(main\_array, N); }

if (act == 3) { outputall(main\_array, N); }

if (act == 4) { createfile(main\_array, N); }

if (act == 5) { editline(main\_array, N); }

if (act == 6) { deleteelement(main\_array, N); }

if (act == 7) { sortall(main\_array, N); }

if (act == 8) { addbefore(main\_array, N); }

if (act == 9) { addafter(main\_array, N); }

}else {err\_msg1(); outputall(main\_array, N);}

}else {if (truefalse == 0) { exec = 0; } else {cout << "Введено неверное значение (число должно быть от 0 до 1), попробуйте ещё раз." << endl;}}

}

delete[] main\_array; return 0;

}

**kurs\_functions.h**

#pragma once

#include <iostream>

#include <iomanip>

#include <string.h>

#include <string>

#include <stdio.h>

#include <fstream>

#include <algorithm>

#include <windows.h>

using namespace std;

struct records { float result; char runner[22]; int year; float age; union { int pol; }runner\_info; };

enum poltype { male = 1, female = 2 };

float getValue\_float();

int getValue\_int();

int getValue\_ArraySize();

int getValue\_pol();

void print\_menu();

void err\_msg1();

string getpoltype(records\* M, int& I);

void outputall(records\* M, int& N);

void inputall(records\* M, int& N);

void createfile(records\* M, int& N);

void openfile(records\* M, int& N);

void addelement(records\* M, int& N);

void deleteelement(records\* M, int& N);

void editline(records\* M, int& N);

void sortall(records\* M, int& N);

void addbefore(records\* M, int& N);

void addafter(records\* M, int& N);

**kurs\_functions.cpp**

#include "kurs\_functions.h"

string getpoltype(records\* M, int& I) {

string shtring;

if(M[I].runner\_info.pol == male){return shtring = "Мужской";}

if(M[I].runner\_info.pol == female){return shtring = "Женский";}

if (M[I].runner\_info.pol != male && M[I].runner\_info.pol != female) {

return shtring = "ERROR01";}

}

float getValue\_float() {

float a; string msg = "значение должно быть числом";

while (true){ cin >> a; if (cin.fail()){

cin.clear();

cin.ignore(32767, '\n');

cout << "Неправильный ввод значения (" << msg << "), попробуйте ещё раз.\n";

}

else{cin.ignore(32767, '\n');return a;}

}

}

int getValue\_int() {

int a; string msg = "значение должно быть числом";

while (true){ cin >> a;

if (cin.fail()){

cin.clear();

cin.ignore(32767, '\n');

cout << "Неправильный ввод значения (" << msg << "), попробуйте ещё раз.\n";

}

else{cin.ignore(32767, '\n');return a;}

}

}

int getValue\_pol() {

int a; string msg = "значение должно быть числом";

while (true) {

while (true) {

cin >> a; if ( cin.fail() ){

cin.clear();

cin.ignore(32767, '\n');

cout << "Неправильный ввод значения (" << msg << "), попробуйте ещё раз.\n";

}

else{cin.ignore(32767, '\n'); break;}

}

if (a == 1 || a == 2) { return a; }

else { cout << "Неправильный ввод значения (число долно быть равно 1, либо 2), попробуйте ещё раз.\n"; }

}

}

int getValue\_ArraySize() {

int a; string msg = "значение должно быть числом";

while (true) {

while (true) {

cin >> a; if (cin.fail()) {

cin.clear();

cin.ignore(32767, '\n');

cout << "Неправильный ввод значения (" << msg << "), попробуйте ещё раз.\n";

}

else { cin.ignore(32767, '\n'); break; }

}

if (a != 0) { return a; }

else { cout << "Неправильный ввод значения (число долно быть больше нуля), попробуйте ещё раз.\n"; }

}

}

void print\_menu(){

cout << "\nВыберите действие: " << endl;

cout << " 0. Выход из программы" << endl;

cout << " 1. Ввод информации из текстового файла в массив указателей на записи" << endl;

cout << " 2. Добавление новых элементов в конец массива" << endl;

cout << " 3. Просмотр всех элементов массива" << endl;

cout << " 4. Вывод информации из массива в текстовый файл" << endl;

cout << " 5. Корректировка полей выбранного элемента" << endl;

cout << " 6. Удаление выбранного элемента" << endl;

cout << " 7. Сортировка массива по числовому полю" << endl;

cout << " 8. Вставка нового элемента ПЕРЕД выбранным элементом" << endl;

cout << " 9. Вставка нового элемента ПОСЛЕ выбранного элемента" << endl;

cout << "\nДействие (0-9): ";

}

void err\_msg1() {

system("cls"); cout << "\nВведено неверное значение (число должно быть от 0 до 9), попробуйте ещё раз.\n" << endl;

system("pause"); system("cls");

}

void outputall(records\* M, int& N) {

system("cls");

cout << "\n=x================================ Вывод всех рекордов. ===========================================x=" << endl;

for (int i = 0; i < N; i++) {

cout << "Рек. №" << left << setw(3) << i + 1 << " | " << "Рез-тат: " << left << setw(6) << M[i].result << " | " << "Автор: " << left << setw(21) << M[i].runner << " | " << "Год: " << left << setw(4) << M[i].year << " | " << "Возраст: " << left << setw(3) << M[i].age << " | " << "Пол: " << getpoltype(M, i) << endl;

}

cout << "=x=================================================================================================x=" << endl;

}

void inputall(records\* M, int& N) {

system("cls");

cout << "\n=x=============================== Ввод записей. ================================x=" << endl;

for (int i = 0; i < N; i++) {

cout << "\nЗапись №" << i + 1 << ": " << endl;

cout << "Введите результат забега (xx.xx): ";

M[i].result = getValue\_float();

cout << "Введите автора рекорда: ";

cin.getline(M[i].runner, 21);

cout << "Введите год установки рекорда (yyyy): ";

M[i].year = getValue\_int();

cout << "Введите возраст рекордсмена: ";

M[i].age = getValue\_float();

cout << "Введите пол рекордсмена (1 - мужской, 2 - женский): ";

M[i].runner\_info.pol = getValue\_pol();

}

cout << "=x==============================================================================x=" << endl;

}

void createfile(records\* M, int& N) { //output.txt

outputall(M, N); string path;

cout << "\nВведите путь и название файла для вывода (прим. D:\\output.txt): "; cin >> path;

ofstream out; out.open(path);

for (int i = 0; i < N; i++) {

out << "Рек. №" << left << setw(3) << i + 1 << " | " << "Рез-тат: " << left << setw(6) << M[i].result << " | " << "Автор: " << left << setw(21) << M[i].runner << " | " << "Год: " << left << setw(4) << M[i].year << " | " << "Возраст: " << left << setw(3) << M[i].age << " | " << "Пол: " << getpoltype(M, i) << endl;

} out.close();

cout << "Создание файла завершено." << endl; system("pause"); outputall(M, N);

}

void openfile(records\* M, int& N) { //input.txt

outputall(M, N); string path, replace\_line; int i2 = 0;

cout << "\nВведите путь и название файла для открытия (прим. D:\\input.txt): ";

cin >> path; ifstream in; in.open(path);

if (!in.is\_open()){

cout << "\n\n==(!)== Файл не найден, либо не может быть открыт! ==(!)==\n\nВозврат в главное меню...\n";

system("pause"); in.close(); return;

}

in >> i2; records\* mas2 = new records[i2];

for (int i = 0; i < i2; i++) {

in >> mas2[i].result;

in >> mas2[i].runner;

in >> mas2[i].year;

in >> mas2[i].age;

in >> mas2[i].runner\_info.pol;

}

in.close();

for (int i = 0; i < i2; i++) {

replace\_line = mas2[i].runner;

replace(replace\_line.begin(), replace\_line.end(), '\_', ' ');

strcpy\_s(mas2[i].runner, replace\_line.c\_str());

}

N = i2;

for (int i = 0; i < N; i++) {

M[i] = mas2[i];

}

cout << "Открытие файла завершено." << endl;

system("pause"); outputall(M, N);

delete[] mas2;

}

void addelement(records\* M, int& N) {

outputall(M, N); N = N + 1;

records\* M2 = new records[N];

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

M2[i] = M[i];

}

cout << "\nЗапись №" << N << ": " << endl;

cout << "Введите результат забега (xx.xx): ";

M2[N - 1].result = getValue\_float();

cout << "Введите автора рекорда: ";

cin.getline(M2[N - 1].runner, 21);

cout << "Введите год установки рекорда (yyyy): ";

M2[N - 1].year = getValue\_int();

cout << "Введите возраст рекордсмена: ";

M2[N - 1].age = getValue\_float();

cout << "Введите пол рекордсмена (1 - мужской, 2 - женский): ";

M2[N - 1].runner\_info.pol = getValue\_pol();

for (int i = 0; i < N; i++) { M[i] = M2[i]; }

cout << "\nДобавление завершено." << endl;

delete[] M2; outputall(M, N);

}

void deleteelement(records\* M, int& N) {

outputall(M, N); int del = 0;

cout << "\nВведите номер записи, которую хотите удалить: ";

del = getValue\_int(); del = del - 1;

if (del > (N - 1) || del < 0) {

cout << "\n\n==(!)== Введено неверное значение (выход за пределы записей). ==(!)==\n\nВозврат в главное меню...\n";

system("pause"); return;

}

records\* M2 = new records[N - 1];

for (int i = 0; i < N; i++) {

if (i < del) { M2[i] = M[i]; }

if (i > del) { M2[i - 1] = M[i]; }

}

N--; for (int i = 0; i < N; i++) { M[i] = M2[i]; } cout << "Запись удалена." << endl;

system("pause"); delete[] M2; outputall(M, N);

}

void editline(records\* M, int& N) {

outputall(M, N);

records\* M2 = new records[N];

for (int i = 0; i < N; i++) {

M2[i] = M[i];

}

int del; cout << "\nВведите номер записи, которую хотите отредактировать: ";

del = getValue\_int(); del = del - 1;

if (del > (N - 1) || del < 0) {

cout << "\n\n==(!)== Введено неверное значение (выход за пределы записей). ==(!)==\n\nВозврат в главное меню...\n";

system("pause"); delete[] M2; return;

}

cout << "\nЗапись №" << del + 1 << ": " << endl;

cout << "Введите результат забега (xx.xx): ";

M2[del].result = getValue\_float();

cout << "Введите автора рекорда: ";

cin.getline(M2[del].runner, 21);

cout << "Введите год установки рекорда (yyyy): ";

M2[del].year = getValue\_int();

cout << "Введите возраст рекордсмена: ";

M2[del].age = getValue\_float();

cout << "Введите пол рекордсмена (1 - мужской, 2 - женский): ";

M2[del].runner\_info.pol = getValue\_pol();

for (int i = 0; i < N; i++) { M[i] = M2[i]; }

cout << "\nРедактирование завершено." << endl;

delete[] M2; outputall(M, N);

}

void sortall(records\* M, int& N) {

records swap; float highest; int high, i;

for (i = 0; i < N - 1; i++) {

high = i;

for (int j = i + 1; j < N; j++) {

if (M[j].result < M[high].result)

high = j;

}

records swap = M[i]; M[i] = M[high]; M[high] = swap;

}

cout << "Сортировка завершена. " << endl; outputall(M, N);

}

void addbefore(records\* M, int& N) {

outputall(M, N); int add;

cout << "\nВведите номер записи, перед которой хотите добавить новый элемент: ";

add = getValue\_int(); add = add - 1;

if (add > (N - 1) || add < 0) {

cout << "\n\n==(!)== Введено неверное значение (выход за пределы записей). ==(!)==\n\nВозврат в главное меню...\n";

system("pause");

return;

}

records\* M2 = new records[N + 1];

for (int i = 0; i < N; i++) {

if (i < add) { M2[i] = M[i]; }

if (i >= add) { M2[i + 1] = M[i]; }

}

cout << "\nЗапись №" << add + 1 << ": " << endl;

cout << "Введите результат забега (xx.xx): ";

M2[add].result = getValue\_float();

cout << "Введите автора рекорда: ";

cin.getline(M2[add].runner, 21);

cout << "Введите год установки рекорда (yyyy): ";

M2[add].year = getValue\_int();

cout << "Введите возраст рекордсмена: ";

M2[add].age = getValue\_float();

cout << "Введите пол рекордсмена (1 - мужской, 2 - женский): ";

M2[add].runner\_info.pol = getValue\_pol();

N++; for (int i = 0; i < N; i++) { M[i] = M2[i]; } cout << "Запись добавлена." << endl; delete[] M2; outputall(M, N);

}

void addafter(records\* M, int& N) {

outputall(M, N); int add = 0;

cout << "\nВведите номер записи, после которой хотите добавить новый элемент: ";

add = getValue\_int(); add--;

if (add > (N - 1) || add < 0) {

cout << "\n\n==(!)== Введено неверное значение (выход за пределы записей). ==(!)==\n\nВозврат в главное меню...\n";

system("pause"); return;

}

records\* M2 = new records[N + 1];

for (int i = 0; i < N; i++) {

if (i <= add) { M2[i] = M[i]; }

if (i > add) { M2[i + 1] = M[i]; }

}

cout << "\nЗапись №" << add + 2 << ": " << endl;

cout << "Введите результат забега (xx.xx): ";

M2[add + 1].result = getValue\_float();

cout << "Введите автора рекорда: ";

cin.getline(M2[add + 1].runner, 21);

cout << "Введите год установки рекорда (yyyy): ";

M2[add + 1].year = getValue\_int();

cout << "Введите возраст рекордсмена: ";

M2[add + 1].age = getValue\_float();

cout << "Введите пол рекордсмена (1 - мужской, 2 - женский): ";

M2[add + 1].runner\_info.pol = getValue\_pol();

N++; for (int i = 0; i < N; i++) { M[i] = M2[i]; } cout << "Запись добавлена." << endl; delete[] M2; outputall(M, N);

}