МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт компьютерных технологий и информационной безопасности

Кафедра информационно-аналитических систем безопасности

имени Л.С. Берштейна

**Отчет по лабораторной работе №5**

по дисциплине  
Основы алгоритмизации и программирования

На тему  
Работа со структурами

Вариант №4

Выполнил:

студент гр. КТбо1-8

Шалыгина В.А.

«\_\_\_» 2017 г.

Проверил:

Профессор ИКТИБ

кафедры ИАСБ

Беляков С.Л

«\_\_\_» 2017 г.

Таганрог – 2017

Оглавление

[Цель работы 3](#_Toc499380090)

[Вариант задания 4](#_Toc499380091)

[Алгоритм работы 5](#_Toc499380092)

[Примеры входных и выходных данных 6](#_Toc499380093)

[Заключение 7](#_Toc499380094)

[Код программы 8](#_Toc499380095)

# Цель работы

Цель данной лабораторной работы состоит в том, чтобы узнать какие бывают конструкции циклов, массивов , как записываются бесконечные циклы и для чего они нужны, что такое структуры, а также научиться работать с ними.

# Вариант задания

Написать программу, которая записывает с клавиатуры в файл структуру согласно выданному варианту задания. В качестве разделителя полей структуры использовать символ табуляции. «Футбольная команда»: название, город, количество сыгранных игр, количество очков (проигрыши, выигрыши, ничьи), количество игроков, фамилия тренера.

# Алгоритм работы

1. Составляем структуру FootballTeam и Point .
2. Подключаем языковые настройки.
3. Объявляем массив на основе двух структур.
4. Заполняем массив по средствам считывания строк, которые именуем под различные, нужные нам, переменные.
5. Высчитываем количество очков ( перем. team[i].points).
6. Подключаем файл вывода output.txt, устанавливаем связь файла и переменой.
7. Считываем параметр сортировки.
8. Сортируем и выводим в файл.
9. Считываем параметры поиска.
10. Проводим поиск и выводим результаты.
11. Если поиск ничего не обнаружил выводим” Нет удовлетворяющих поиску критериев”.
12. Закрываем все открытые файлы.

# Примеры входных и выходных данных

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 3  Первая  Первая  5  5  5  5  Первая  Вторая  Вторая  7  7  7  7  Второй  Третий  Третий  3  3  3  3  Третий  1  2  14 | Вывод в сортировке по количеству игроков  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Название команды вторая  Город команды вторая  Количество сыгранных игр команды 7  Количество очков команды 14  Количество игроков команды 7  Фамилия тренера команды второй  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Название команды первая  Город команды первая  Количество сыгранных игр команды 5  Количество очков команды 10  Количество игроков команды 5  Фамилия тренера команды первая  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Название команды третий  Город команды третий  Количество сыгранных игр команды 3  Количество очков команды 6  Количество игроков команды 3  Фамилия тренера команды третий  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Поиск выполнен по очкам  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Название команды которую нужно найти вторая  Количество игроков команды 14  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Нет удовлетворяющих поиску критериев  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Заключение

В данной работе студенты познакомились со средой создания сложных символьных структур, научились обрабатывать строки. Были изучены различные методы ввода и вывода информации, вызова функций, сортировки и поиска.

# Код программы

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <windows.h>

#include <string.h>

struct Point {

int numberOfWins;

int numberOfLosses;

int numberOfDraws;

};

struct FootballTeam {

char name[255];

char city[255];

int numberOfGamesPlayed;

struct Point {

int numberOfWins;

int numberOfLosses;

int numberOfDraws;

};

int amountPlayers;

char surnameOftheTrainer[255];

int points;

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

FootballTeam team[100];

Point points[100];

int amountTeam;

printf("Введите сколько будет команд ");

scanf("%d", &amountTeam);

printf("Введите даные о командах\n");

for (int i = 1; i <= amountTeam; i++)

{

printf("Ведите название команды: ");

scanf("%s", &team[i].name);

printf("Ведите город команды: ");

scanf("%s", &team[i].city);

printf("Ведите количесство сыграных игр: ");

scanf("%d", &team[i].numberOfGamesPlayed);

printf("Ведите количесство выйгрышей: ");

scanf("%d", &points[i].numberOfWins);

printf("Ведите количесство пройгрышей: ");

scanf("%d", &points[i].numberOfLosses);

printf("Ведите количесство ничьих: ");

scanf("%d", &points[i].numberOfDraws);

printf("Ведите количесство игроков: ");

scanf("%d", &team[i].amountPlayers);

printf("Ведите фамилию тренера: ");

scanf("%s", &team[i].surnameOftheTrainer);

printf("Система подсчета отчков: 2(ввыйгрыши)-пройгрыши+ниьчи (выйгрш 2 очка, пройгрыш -1, ничья 1) \n");

team[i].points = 2 \* points[i].numberOfWins - points[i].numberOfLosses + points[i].numberOfDraws;

printf("Количество очков:%d \n", team[i].points);

}

FILE \*f = fopen("tablOut.txt", "w");

int numderOfOut;

printf("Введите по какому признаку сортировать команды:\n");

printf(" 1-количество игроков, 2-количетво очков, ->");

scanf("%d", &numderOfOut);

switch (numderOfOut)

{

case 1: //сортировка по количеству игроков

{

for (int i = 1; i < amountTeam; i++)

{

for (int j = amountTeam - 1; j >= i; j--)

if (team[j].amountPlayers < team[j + 1].amountPlayers)

{

FootballTeam buf;

buf = team[j];

team[j] = team[j + 1];

team[j + 1] = buf;

}

}

fprintf(f, "Вывод в сортировке по количеству игроков \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

for (int i = 1; i <= amountTeam; i++)

{

fprintf(f, "Название команды %s\n", team[i].name);

fprintf(f, "Город команды %s\n", team[i].city);

fprintf(f, "Количество сыграных игр команды %d\n", team[i].numberOfGamesPlayed);

fprintf(f, "Количество очков команды %d\n", team[i].points);

fprintf(f, "Количество игроков команды %d\n", team[i].amountPlayers);

fprintf(f, "Фамилия тренера команды %s\n", team[i].surnameOftheTrainer);

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

break;

}

case 2: //сортировка по количеству очков

{

for (int i = 1; i < amountTeam; i++)

{

for (int j = amountTeam - 1; j >= i; j--)

if (team[j].points < team[j + 1].points)

{

FootballTeam buf;

buf = team[j];

team[j] = team[j + 1];

team[j + 1] = buf;

}

}

fprintf(f, "Вывод в сортировке по количеству очков \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

for (int i = 1; i <= amountTeam; i++)

{

fprintf(f, "Название команды %s\n", team[i].name);

fprintf(f, "Город команды %s\n", team[i].city);

fprintf(f, "Количество сыграных игр команды %d\n", team[i].numberOfGamesPlayed);

fprintf(f, "Количество очков команды %d\n", team[i].points);

fprintf(f, "Количество игроков команды %d\n", team[i].amountPlayers);

fprintf(f, "Фамилия тренера команды %s\n", team[i].surnameOftheTrainer);

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

break;

}

default:

{

fprintf(f, "Вывод в исходной сортировке \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

for (int i = 1; i <= amountTeam; i++)

{

fprintf(f, "Название команды %s\n", team[i].name);

fprintf(f, "Город команды %s\n", team[i].city);

fprintf(f, "Количество сыграных игр команды %d\n", team[i].numberOfGamesPlayed);

fprintf(f, "Количество очков команды %d\n", team[i].points);

fprintf(f, "Количество игроков команды %d\n", team[i].amountPlayers);

fprintf(f, "Фамилия тренера команды %s\n", team[i].surnameOftheTrainer);

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

}

}

int numderOfSearch,searchSign;

printf("Введите по какому признаку найти команду:\n");

printf(" 1-количество игроков, 2-количетво очков ->");

scanf("%d", &numderOfSearch);

printf(" Сколько надо искать ->");

scanf("%d", &searchSign);

switch (numderOfSearch)

{

case 1: //поиск по количеству игроков

{

fprintf(f, "Поиск выполнен по игрокам \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

int test=0;

for (int i = 1; i <= amountTeam; i++)

{

if (team[i].amountPlayers == searchSign)

{

fprintf(f, "Название команды которую нужно найти %s\n", team[i].name);

fprintf(f, "Количество игроков команды %d\n", team[i].amountPlayers);

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

else

{

test++;

}

}

if (test != 0 )

{

fprintf(f, "Нет удовлетворяющих поиску критериев \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

break;

}

case 2: //поиск по количеству очков

{

fprintf(f, "Поиск выполнен по очкам \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

int test = 0;

for (int i = 1; i <= amountTeam; i++)

{

if (team[i].points == searchSign)

{

fprintf(f, "Название команды которую нужно найти %s\n", team[i].name);

fprintf(f, "Количество игроков команды %d\n", team[i].points);

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

else

{

test++;

}

}

if (test != 0)

{

fprintf(f, "Нет удовлетворяющих поиску критериев \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

}

break;

}

default:

{

fprintf(f, "Нет поиска по таким критериям \n");

fprintf(f, "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \n");

break;

}

}

fclose(f);

return 0;

}