

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра Вычислительной техники**

**ОТЧЕТ**  
**по лабораторной работе № 7**  
**по дисциплине «Программирование»**  
**ТЕМА: УКАЗАТЕЛИ НА СТРУКТУРЫ И ФУНКЦИИ.**

Студент гр. 3312

\_\_\_\_\_

Шарапов И. Д.

Преподаватель

\_\_\_\_\_

Аббас С. А.

Санкт-Петербург

2024

## Содержание

Цель работы .....	3
Задание (Вариант 14) .....	3
Постановка задачи и описание решения.....	3
Описание переменных .....	5
Структура вызова функций .....	6
Схема алгоритма.....	7
Текст программы.....	10
Контрольные примеры.....	14
Содержимое файлов.....	16
Примеры выполнения программы.....	17
Выводы .....	20

## **Цель работы**

Целью работы является изучение указателей на структуры и функции в языке Си, а также работа с динамическими массивами структур.

## **Задание (Вариант 14)**

Для выбранной предметной области создать динамический массив структур, содержащих характеристики объектов предметной области.

Обязательный набор полей:

- динамический массив символов, включая пробелы (name)
- произвольный динамический массив символов
- числовые поля типов `int` и `float` (не менее двух полей каждого типа)
- поле с числовым массивом.

Написать программу, обеспечивающую начальное формирование массива структур при чтении из файла (текст с разделителями — CSV) с последующим возможным дополнением элементов массива при вводе с клавиатуры. Следует использовать указатели на структуры и указатели на функции обработки массива в соответствии с вариантом задания.

Во всех случаях, когда при поиске записей результат отсутствует, следует вывести сообщение.

Выбор записей, в которых значение любого символьного поля (выбор из меню) содержит указанную подстроку без учёта регистра, сортировка результата по возрастанию значений любого из элементов поля с числовым массивом (выбор признака сортировки — из меню).

## **Постановка задачи и описание решения**

Для получения данных будем запрашивать у пользователя имя файла до тех пор, пока не сможем успешно открыть его. После открытия файла, мы спросим у пользователя количество строк, которые он хочет прочитать из файла.

Далее, мы обработаем данные из файла. Для этого, мы будем построчно считывать информацию об атлете. Каждую считанную строку мы будем

обрабатывать с помощью функции *fill\_struct()*, которая выделит память для новой сущности структуры атлета и заполнит её. После считывания всех данных из файла, мы закроем его и выведем сообщение об успешной обработке данных.

Затем, мы выведем пользователю список доступных команд. Мы будем считывать команды до тех пор, пока пользователь не введёт строку *"!end"*.

Если пользователь введёт строку *"!print"*, мы выведем таблицу атлетов с помощью функции *pprint()*.

Если пользователь введёт строку *"!find"*, мы спросим у него, по какому полю он хочет искать атлетов (по имени или университету). Если пользователь введёт число, отличное от 1 и 2, мы выведем сообщение *"Invalid command!"*. В противном случае, мы вызовем функцию *find\_users()* с соответствующим параметром. Эта функция спросит у пользователя строку, приведёт её к нижнему регистру. Затем, она переберёт все элементы массива структур и проверит, является ли введённая строка подстрокой соответствующего поля данного атлета, приведённого к нижнему регистру. Все подходящие атлеты будут выведены в виде таблицы. Если ни один атлет не соответствует критериям, мы выведем сообщение *"No matches found!"*.

Если пользователь введёт строку *"!sort"*, мы спросим у него, по какому полю он хочет сортировать пользователей. Если пользователь введёт число, отличное от 1, 2, 3, 4, мы выведем сообщение *"Invalid command!"*. В противном случае, мы вызовем функцию *sort\_users()* с соответствующим параметром. Эта функция выполнит сортировку пузырьком, сравнивая выбранный пользователем критерий. После сортировки мы выведем сообщение об успешном окончании сортировки и выведем таблицу атлетов с помощью функции *pprint()*.

Если пользователь введёт строку *"!add"*, мы выведем ему формат ввода нового пользователя. Мы выделим дополнительную память в динамическом массиве структур, прочитаем информацию о новом атлете, обработаем ее с

помощью функции *fill\_struct()* и добавим в массив. При успешном добавлении пользователя мы выведем сообщение *"The data has been successfully added!"*.

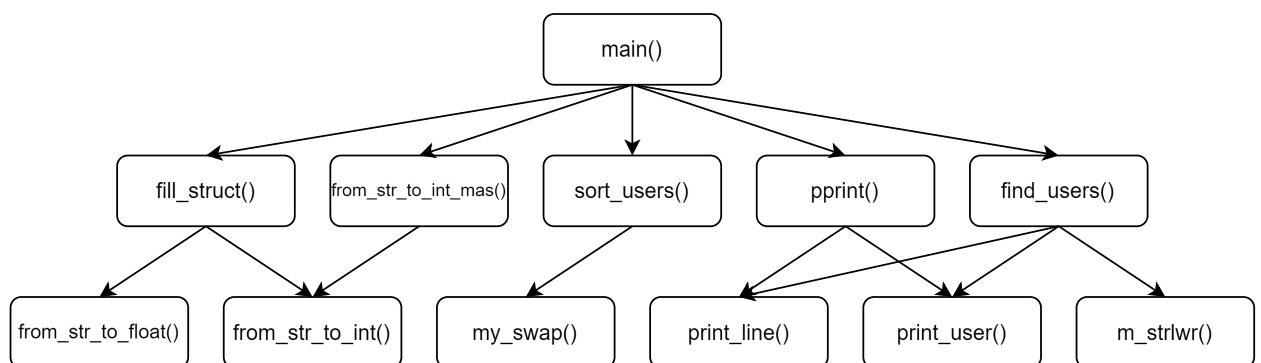
Если пользователь введёт какую-то, не описанную выше строку, мы выведем сообщение *"Unknown command!"*.

### Описание переменных

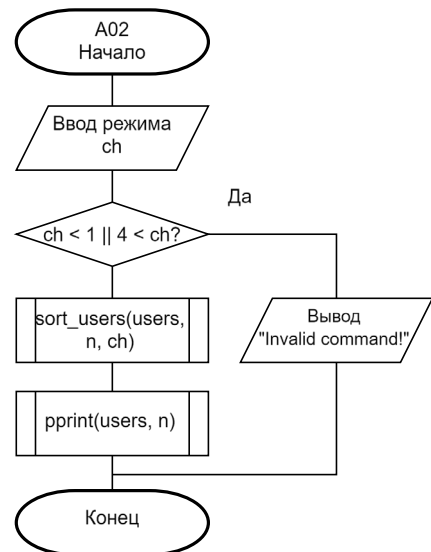
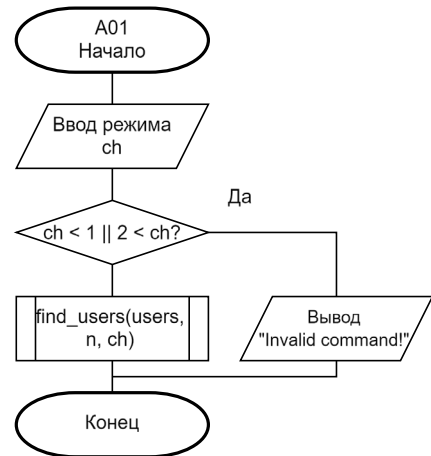
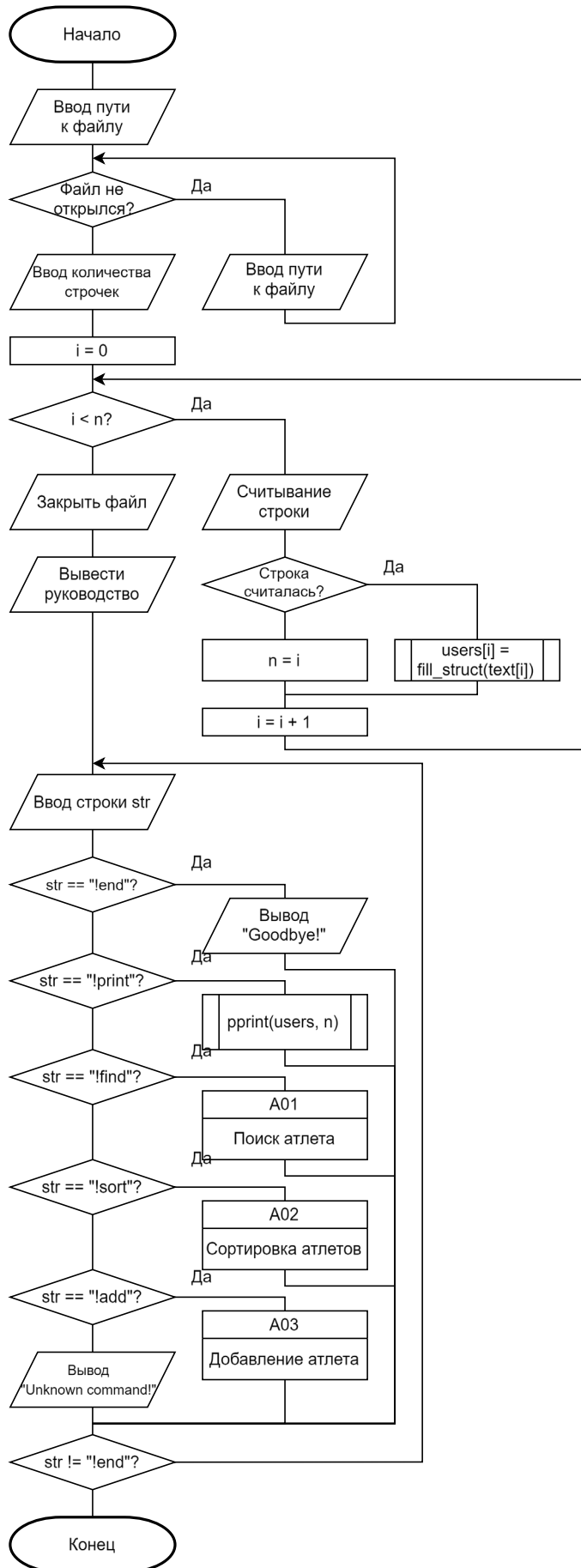
№	Имя переменной	Тип	Назначение
Функция <i>int from str to int(char *x)</i>			
1	x	char*	Строка, которую будем переводить в число
2	ans	int	Число, которое получим после обработки
Функция <i>int from str to float(char *x)</i>			
1	x	char*	Строка, которую будем переводить в число
2	ans	int	Число, которое получим после обработки
3	a	int	На что домножаем целую часть
4	b	int	На что делим дробную часть
Функция <i>void from str to int mas(char *x, int *mas)</i>			
1	x	char*	Строка, которую будем переводить в массив
2	mas	int*	Массив, который будем заполнять
3	ind	int	Индекс массива
4	j	int	Индекс в строке
Функция <i>Athlete *fill_struct(char *str)</i>			
1	str	char*	Строка в формате CSV
2	user	Athlete*	Сущность структуры Athlete
3	word	char*	Отдельное поле, разделённое ;
4	ind	int	Индекс поля, в котором мы находимся
5	tt	int	Счётчик для перебора строки
6	pole	char*[]	Массив указателей на начало полей
Функция <i>void print user(Athlete *user)</i>			
1	user	Athlete*	Атлет, которого выводим
Функция <i>void pprint(Athlete **users, int n)</i>			
1	users	Athlete**	Динамический массив атлетов
2	n	int	Количество атлетов
Функция <i>void my_swap(Athlete *a, Athlete *b)</i>			
1	a	Athlete*	Первый атлет
2	b	Athlete*	Второй атлет
3	c	Athlete*	Вспомогательный элемент для обмена
Функция <i>void sort users(Athlete **users, int n, int param)</i>			
1	users	Athlete**	Динамический массив атлетов
2	n	int	Количество атлетов
3	param	int	Флаг, показывающий поле для сортировки

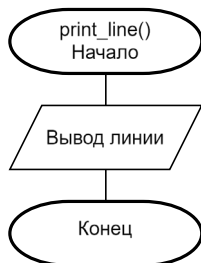
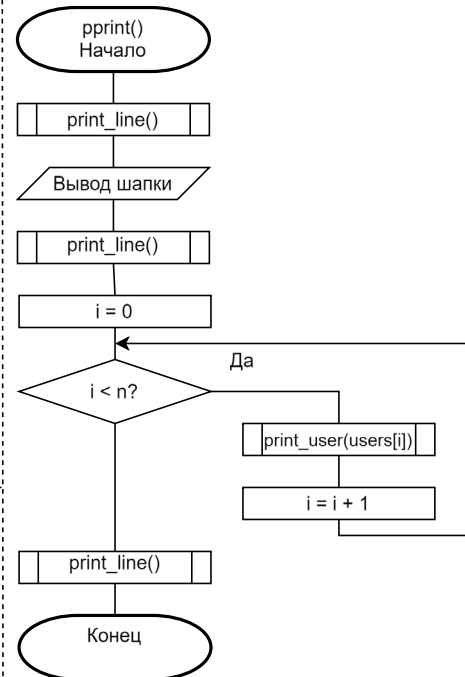
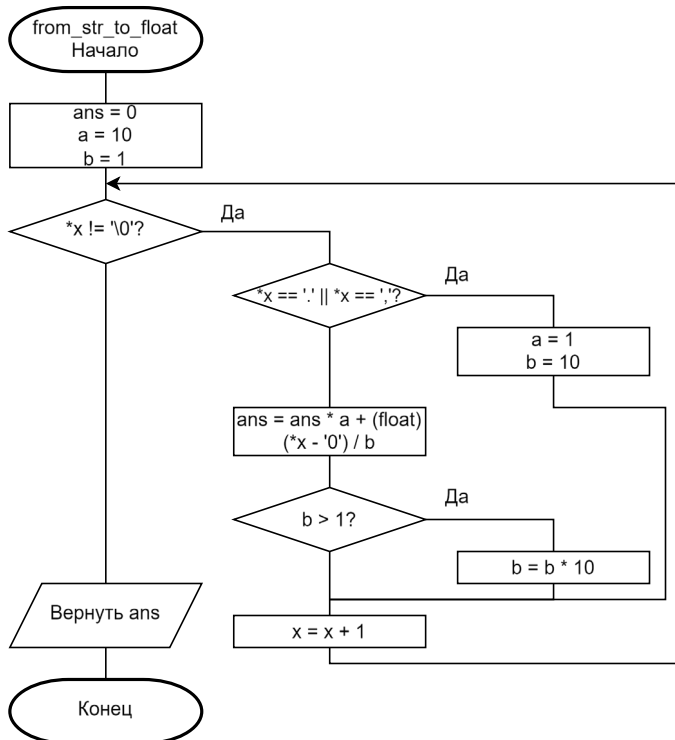
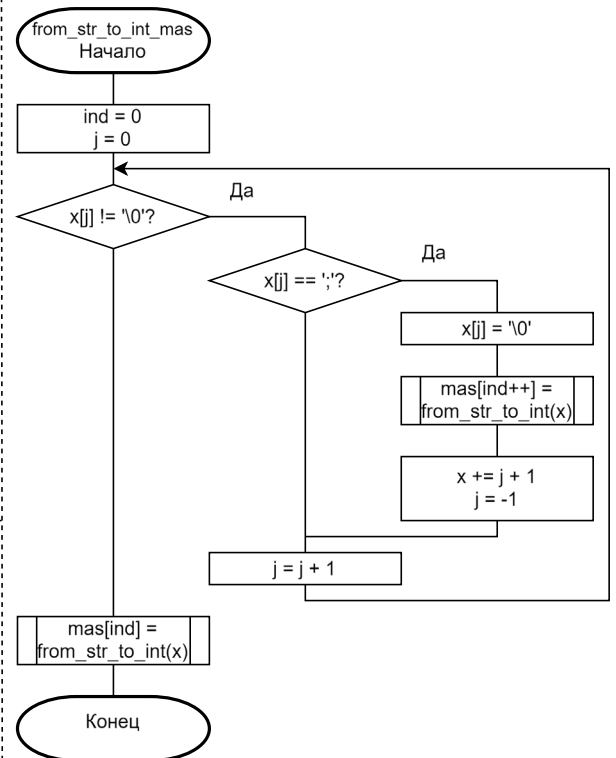
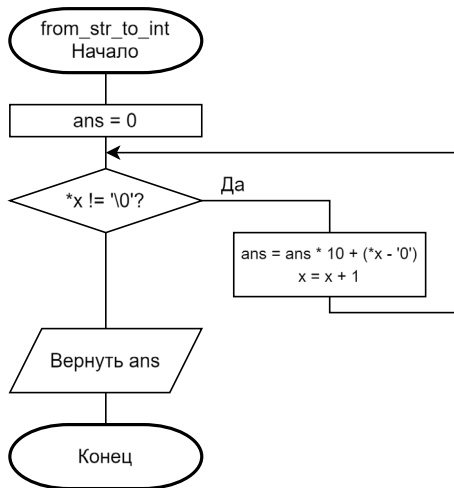
Функция <i>char *m_strlwr(const char *str)</i>			
1	str	char*	Исходная строка
2	len	size_t	Длина строки
3	new_str	char*	Строка, приведённая к нижнему регистру
Функция <i>void find_users(Athlete **users, int n, int param)</i>			
1	users	Athlete**	Динамический массив атлетов
2	n	int	Количество атлетов
3	param	int	Флаг, показывающий поле для поиска
4	x	char[]	Подстрока, которую ищем
5	str	char*	Строковое поле, по которому ищем
6	fl	int	Флаг – найдены пользователи или нет
Функция <i>int main()</i>			
1	filename	char[]	Имя файла
2	f	FILE*	Файл для считывания
3	n	int	Количество пользователей
4	ch	int	Выбор режима для сортировки и поиска
5	users	Athlete**	Динамический массив атлетов
6	str	char[]	Строка команд
7	text	char[][]	Массив всех строк о пользователях

### Структура вызова функций

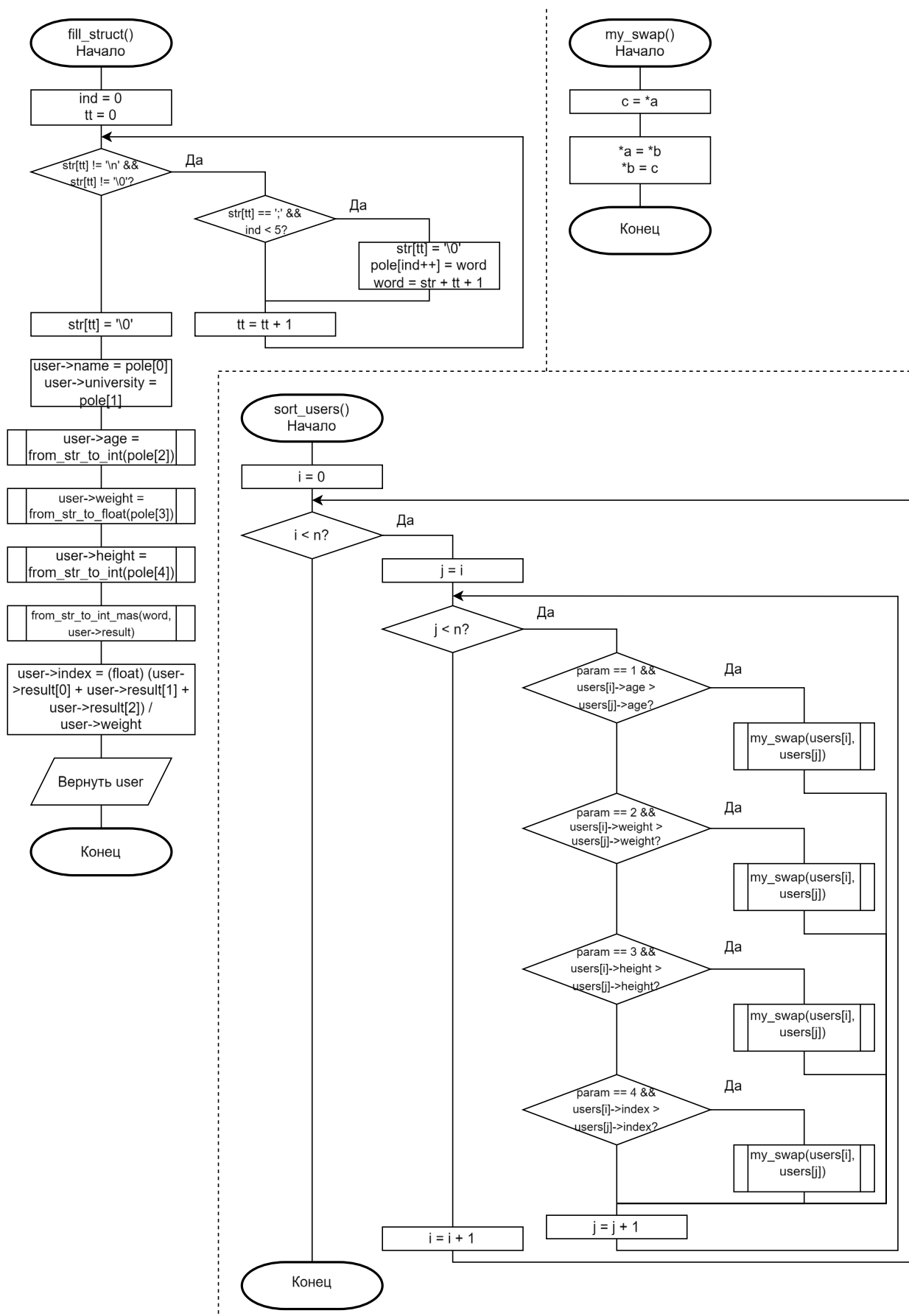


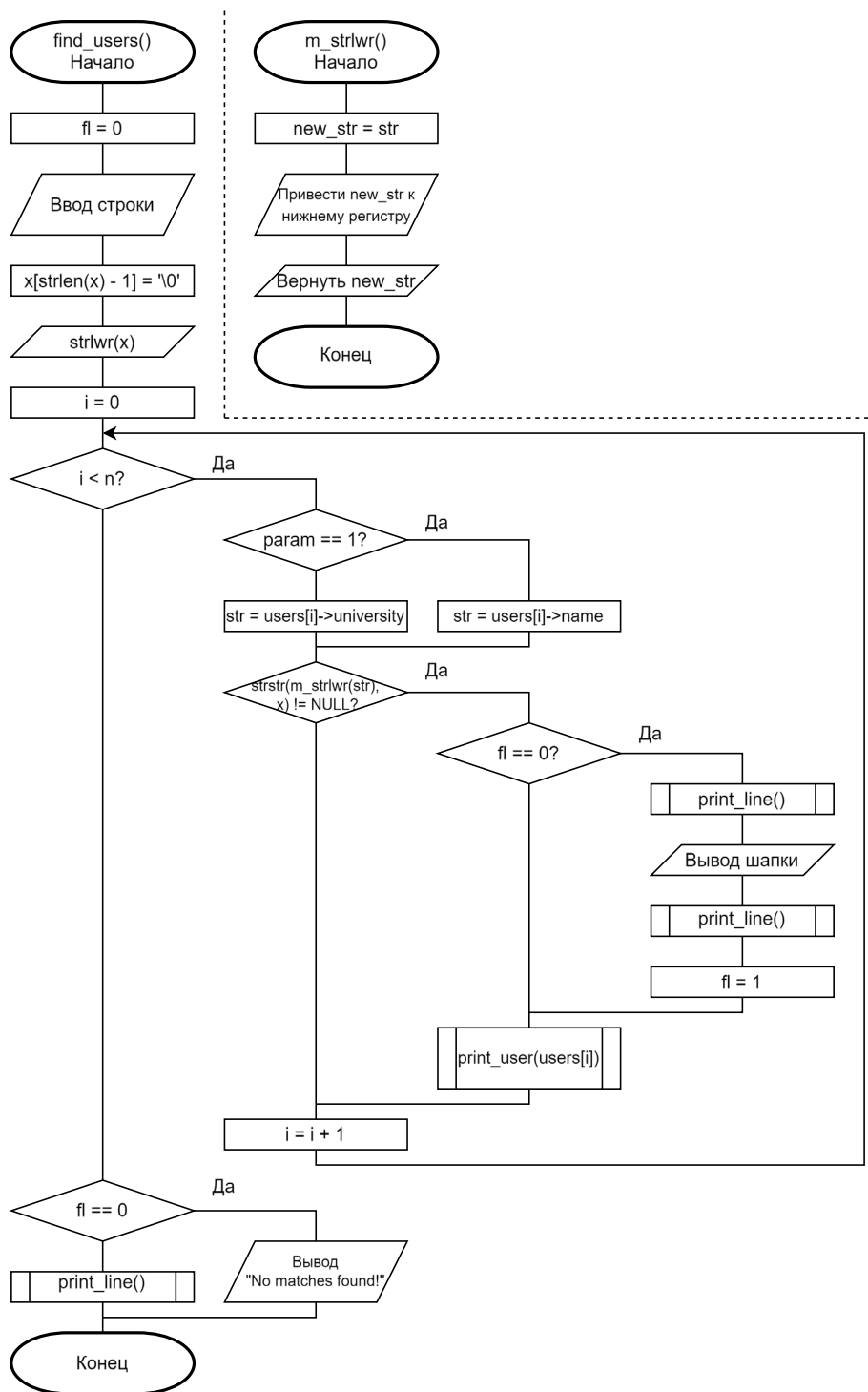
## Схема алгоритма











## Текст программы

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define MAX USERS 100

struct Athlete {
    char *name;          /* имя */
    char *university;    /* университет */
    int age;              /* возраст */
    float weight;         /* вес */
    int height;           /* рост */
    int result[3];        /* результаты */
    float index;          /* индекс результата к массе */
};
  
```

```

typedef struct Athlete Athlete;

int from str to int(char *x);
float from str to float(char *x);
void from str to int mas(char *x, int *mas);
Athlete *fill_struct(char *str);
void print_line();
void print_user(Athlete *user);
void pprint(Athlete **users, int n);
void my_swap(Athlete *a, Athlete *b);
void sort_users(Athlete **users, int n, int param);
char *m_strlwr(const char *str);
void find_users(Athlete **users, int n, int param);

int main() {
    char filename[128];
    FILE *f;
    int n, ch;
    Athlete **users;
    char str[128];

    printf("Please enter the file name:\n");
    scanf("%s", filename);
    f = fopen(filename, "r");
    while (f == NULL) {
        printf("Something went wrong!\n"
            "Perhaps such a file does not exist.\n"
            "Please enter the file name again:\n");
        scanf("%s", filename);
        f = fopen(filename, "r");
    }

    printf("Please enter count of lines in file (max %i):\n", MAX_USERS);
    scanf("%i", &n);
    getchar();

    char text[MAX_USERS][1024];
    users = (Athlete **) malloc(n * sizeof(Athlete *));
    if (users != NULL) {
        for (int i = 0; i < n; ++i) {
            if (fgets(text[i], sizeof(text[i]), f)) {
                users[i] = fill_struct(text[i]);
            } else {
                n = i;
            }
        }
        fclose(f);
        printf("The file has successfully been processed!\n");
    } else {
        printf("Memory error!");
    }

    printf("To display the data, enter the command \\"!print\\"\\n"
        "To find users, enter the command \\"!find\\"\\n"
        "To sort the data, enter the command \\"!sort\\"\\n"
        "To add new data, enter the command \\"!add\\"\\n"
        "To end the program, enter the command \\"!end\\"\\n");

    do {
        scanf("%s", str);
        if (!strcmp(str, "!end")) {
            printf("Goodbye!\n");
        } else if (!strcmp(str, "!print")) {
            pprint(users, n);
        } else if (!strcmp(str, "!find")) {
            printf("Select a field to find by:\n"
                "1 = name\n"
                "2 = university\n"
                "Enter only one number!\n");
            scanf("%i", &ch);
            if (ch < 1 || 2 < ch) {
                printf("Invalid command!\n");
            } else {
                find_users(users, n, ch);
            }
        } else if (!strcmp(str, "!sort")) {
            printf("Select a field to sort by:\n"
                "1 = age\n"

```

```

        "2 = weight\n"
        "3 = height\n"
        "4 = index\n"
        "Enter only one number!\n");
scanf("%i", &ch);
if (ch < 1 || 4 < ch) {
    printf("Invalid command!\n");
} else {
    sort users(users, n, ch);
    printf("The data has been successfully sorted!\n");
    pprint(users, n);
}
} else if (!strcmp(str, "!add")) {
    printf("Enter data of the athlete in format:\n"
        "name;university;age;weight;height;result1,result2,result3\n");
    ++n;
    users = (Athlete **) realloc(users, (n) * sizeof(Athlete *));
    if (users != NULL) {
        getchar();
        fgets(text[n - 1], sizeof(text[n - 1]), stdin);
        users[n - 1] = fill_struct(text[n - 1]);
        printf("The data has been successfully added!\n");
    } else {
        printf("Something went wrong!");
    }
} else {
    printf("Unknown command!\n");
}
} while (strcmp(str, "!end"));

free(users);
return 0;
}

int from_str_to_int(char *x) {
    int ans = 0;

    while (*x != '\0') {
        ans = ans * 10 + (*x - '0');
        ++x;
    }
    return ans;
}

float from_str_to_float(char *x) {
    float ans = 0, a = 10, b = 1;

    while (*x != '\0') {
        if (*x == '.' || *x == ',') {
            a = 1;
            b = 10;
        } else {
            ans = ans * a + (float) (*x - '0') / b;
            if (b > 1) b *= 10;
        }
        ++x;
    }
    return ans;
}

void from_str_to_int_mas(char *x, int *mas) {
    int ind = 0, j = 0;

    while (x[j] != '\0') {
        if (x[j] == ';') {
            x[j] = '\0';
            mas[ind++] = from_str_to_int(x);
            x += j + 1;
            j = -1;
        }
        ++j;
    }
    mas[ind] = from_str_to_int(x);
}

Athlete *fill_struct(char *str) {
    Athlete *user;
    char *word = str;
    int ind = 0, tt;
    char *pole[5];

    user = (Athlete *) malloc(sizeof(Athlete));
    if (user != NULL) {
        for (tt = 0; str[tt] != '\n' && str[tt] != '\0'; ++tt) {
            if (str[tt] == ';' && ind < 5) {
                str[tt] = '\0';
                pole[ind++] = word;
                word = str + tt + 1;
            }
        }
    }
}

```

```

        str[tt] = '\0';
        user->name = pole[0];
        user->university = pole[1];
        user->age = from str to int(pole[2]);
        user->weight = from str to float(pole[3]);
        user->height = from str to int(pole[4]);
        from str to int mas(word, user->result);
        user->index =
            (float) (user->result[0] + user->result[1] + user->result[2]) /
            user->weight;
    }
    return user;
}

void print_line() {
    printf("+");
    for (int i = 0; i < 22; printf("-"), ++i);
    printf("+-----+
        "+-----+\n");
}

void print_user(Athlete *user) {
    printf("| %-20s | %-10s | %-3i | %0.1f ", user->name, user->university,
        user->age, user->weight);
    if (user->weight < 100) printf(" ");
    printf("| %-6i | %-4i | %-4i | %-4i ", user->height, user->result[0],
        user->result[1], user->result[2]);
    printf("| %0.3f |\n", user->index);
}

void pprint(Athlete **users, int n) {
    print_line();
    printf("| Name | University | Age | Weight | Height "
        "| Res1 | Res2 | Res3 | Index |\n");
    print_line();
    for (int i = 0; i < n; ++i) {
        print_user(users[i]);
    }
    print_line();
}

void my_swap(Athlete *a, Athlete *b) {
    Athlete c = *a;

    *a = *b;
    *b = c;
}

void sort_users(Athlete **users, int n, int param) {
    for (int i = 0; i < n; ++i) {
        for (int j = i; j < n; ++j) {
            if (param == 1 && users[i]->age > users[j]->age) {
                my_swap(users[i], users[j]);
            } else if (param == 2 && users[i]->weight > users[j]->weight) {
                my_swap(users[i], users[j]);
            } else if (param == 3 && users[i]->height > users[j]->height) {
                my_swap(users[i], users[j]);
            } else if (param == 4 && users[i]->index > users[j]->index) {
                my_swap(users[i], users[j]);
            }
        }
    }
}

char *m_strlwr(const char *str) {
    size_t len = strlen(str);
    char *new_str = malloc(len + 1);

    if (new_str != NULL) {
        strcpy(new_str, str);
        strlwr(new_str);
    }
    return new_str;
}

void find_users(Athlete **users, int n, int param) {
    char x[128];
    char *str;
    int fl = 0;

    printf("Enter the search string:\n");
    getchar();
    fgets(x, sizeof(x), stdin);
    x[strlen(x) - 1] = '\0';
    strlwr(x); /* приводит x к нижнему регистру */
    for (int i = 0; i < n; ++i) {
        if (param == 1) str = users[i]->name;
        else str = users[i]->university;
        if (strstr(m_strlwr(str), x) != NULL) {
            if (fl == 0) {

```

```

        print line();
        printf("| Name | University | Age | Weight |
               | Height | Res1 | Res2 | Res3 | Index |\n");
        print line();
        fl = 1;
    }
    print user(users[i]);
}
}
if (fl == 0) {
    printf("No matches found!\n");
} else {
    print line();
}
}

```

## Контрольные примеры

№	Исходные данные	Результаты
1	input1.csv 10 !print !sort 2 !end	Please enter the file name: Please enter count of lines in file (max 100): The file has successfully been processed! To display the data, enter the command "!print" To find users, enter the command "!find" To sort the data, enter the command "!sort" To add new data, enter the command "!add" To end the program, enter the command "!end"  +-----+-----+-----+-----+-----+-----+-----+-----+-----+   Name   University   Age   Weight   Height   Res1   Res2   Res3   Index   +-----+-----+-----+-----+-----+-----+-----+-----+-----+   Ivanov I.I.   MSU   25   70.5   175   120   200   230   7.801     Petrov P.P.   SPbSU   22   65.2   180   140   180   210   8.129     Sidorov S.S.   MIPT   18   55.8   165   90   150   180   7.527     Kuznetsov K.K.   HSE   20   75.1   185   200   220   240   8.788     Smirnov A.A.   MGU   27   90.3   190   180   220   250   7.198     Fedorov F.F.   NRU HSE   19   68.7   170   110   170   200   6.987     Volkov V.V.   BSU   21   60.0   160   80   0   0   1.333     Mikhailov M.M.   TSU   24   72.4   178   160   190   220   7.873     Novikov N.N.   ITMO   26   80.6   195   210   240   250   8.685     Morozov M.I.   RANEPA   23   73.8   183   170   200   230   8.130   +-----+-----+-----+-----+-----+-----+-----+-----+-----+ Select a field to sort by: 1 = age 2 = weight 3 = height 4 = index Enter only one number! The data has been successfully sorted!  +-----+-----+-----+-----+-----+-----+-----+-----+-----+   Name   University   Age   Weight   Height   Res1   Res2   Res3   Index   +-----+-----+-----+-----+-----+-----+-----+-----+-----+   Sidorov S.S.   MIPT   18   55.8   165   90   150   180   7.527     Volkov V.V.   BSU   21   60.0   160   80   0   0   1.333     Petrov P.P.   SPbSU   22   65.2   180   140   180   210   8.129     Fedorov F.F.   NRU HSE   19   68.7   170   110   170   200   6.987     Ivanov I.I.   MSU   25   70.5   175   120   200   230   7.801     Mikhailov M.M.   TSU   24   72.4   178   160   190   220   7.873     Morozov M.I.   RANEPA   23   73.8   183   170   200   230   8.130     Kuznetsov K.K.   HSE   20   75.1   185   200   220   240   8.788     Novikov N.N.   ITMO   26   80.6   195   210   240   250   8.685     Smirnov A.A.   MGU   27   90.3   190   180   220   250   7.198   +-----+-----+-----+-----+-----+-----+-----+-----+-----+ Goodbye!

2	input2.csv 10 !find 2 m !end	<div>Please enter the file name: Please enter count of lines in file (max 100): The file has successfully been processed! To display the data, enter the command "!print" To find users, enter the command "!find" To sort the data, enter the command "!sort" To add new data, enter the command "!add" To end the program, enter the command "!end" Select a field to find by: 1 = name 2 = university Enter only one number! Enter the search string:</div> <table><tr><th>Name</th><th>University</th><th>Age</th><th>Weight</th><th>Height</th><th>Res1</th><th>Res2</th><th>Res3</th><th>Index</th></tr><tr><td>Orlov O.D.</td><td>MIIT</td><td>17</td><td>57.5</td><td>163</td><td>100</td><td>160</td><td>0</td><td>4.522</td></tr><tr><td>Sergeev S.I.</td><td>TomSU</td><td>22</td><td>67.2</td><td>168</td><td>0</td><td>0</td><td>0</td><td>0.000</td></tr></table> <div>Goodbye!</div>	Name	University	Age	Weight	Height	Res1	Res2	Res3	Index	Orlov O.D.	MIIT	17	57.5	163	100	160	0	4.522	Sergeev S.I.	TomSU	22	67.2	168	0	0	0	0.000																																																																																																																																																																																				
Name	University	Age	Weight	Height	Res1	Res2	Res3	Index																																																																																																																																																																																																									
Orlov O.D.	MIIT	17	57.5	163	100	160	0	4.522																																																																																																																																																																																																									
Sergeev S.I.	TomSU	22	67.2	168	0	0	0	0.000																																																																																																																																																																																																									
3	input3.csv 10 !print !add Klimov K.K.;SPbPU;21 ;64.5;170;130;180;210 !print !end	<div>Please enter the file name: Please enter count of lines in file (max 100): The file has successfully been processed! To display the data, enter the command "!print" To find users, enter the command "!find" To sort the data, enter the command "!sort" To add new data, enter the command "!add" To end the program, enter the command "!end"</div> <table><tr><th>Name</th><th>University</th><th>Age</th><th>Weight</th><th>Height</th><th>Res1</th><th>Res2</th><th>Res3</th><th>Index</th></tr><tr><td>Pavlov P.A.</td><td>MSU</td><td>24</td><td>70.2</td><td>177</td><td>160</td><td>200</td><td>230</td><td>8.405</td></tr><tr><td>Kiselev K.A.</td><td>SPbSU</td><td>20</td><td>65.8</td><td>175</td><td>140</td><td>180</td><td>210</td><td>8.055</td></tr><tr><td>Bogdanov B.B.</td><td>MIPT</td><td>18</td><td>55.5</td><td>162</td><td>90</td><td>150</td><td>180</td><td>7.568</td></tr><tr><td>Danilov D.D.</td><td>HSE</td><td>22</td><td>75.3</td><td>183</td><td>200</td><td>220</td><td>240</td><td>8.765</td></tr><tr><td>Nesterov N.N.</td><td>MGU</td><td>27</td><td>90.8</td><td>188</td><td>180</td><td>220</td><td>250</td><td>7.159</td></tr><tr><td>Romanov R.I.</td><td>NRU HSE</td><td>19</td><td>68.2</td><td>168</td><td>110</td><td>170</td><td>200</td><td>7.038</td></tr><tr><td>Zaitsev Z.I.</td><td>BSU</td><td>21</td><td>60.3</td><td>163</td><td>80</td><td>130</td><td>160</td><td>6.136</td></tr><tr><td>Konovalov K.M.</td><td>TSU</td><td>25</td><td>72.6</td><td>179</td><td>120</td><td>190</td><td>220</td><td>7.300</td></tr><tr><td>Frolov F.P.</td><td>ITMO</td><td>26</td><td>101.2</td><td>192</td><td>210</td><td>240</td><td>250</td><td>6.917</td></tr><tr><td>Belov B.V.</td><td>UralSU</td><td>28</td><td>85.8</td><td>190</td><td>220</td><td>240</td><td>250</td><td>8.275</td></tr></table> <div>Enter data of the athlete in format: name;university;age;weight;height;result1,result2,result3 The data has been successfully added!</div> <table><tr><th>Name</th><th>University</th><th>Age</th><th>Weight</th><th>Height</th><th>Res1</th><th>Res2</th><th>Res3</th><th>Index</th></tr><tr><td>Pavlov P.A.</td><td>MSU</td><td>24</td><td>70.2</td><td>177</td><td>160</td><td>200</td><td>230</td><td>8.405</td></tr><tr><td>Kiselev K.A.</td><td>SPbSU</td><td>20</td><td>65.8</td><td>175</td><td>140</td><td>180</td><td>210</td><td>8.055</td></tr><tr><td>Bogdanov B.B.</td><td>MIPT</td><td>18</td><td>55.5</td><td>162</td><td>90</td><td>150</td><td>180</td><td>7.568</td></tr><tr><td>Danilov D.D.</td><td>HSE</td><td>22</td><td>75.3</td><td>183</td><td>200</td><td>220</td><td>240</td><td>8.765</td></tr><tr><td>Nesterov N.N.</td><td>MGU</td><td>27</td><td>90.8</td><td>188</td><td>180</td><td>220</td><td>250</td><td>7.159</td></tr><tr><td>Romanov R.I.</td><td>NRU HSE</td><td>19</td><td>68.2</td><td>168</td><td>110</td><td>170</td><td>200</td><td>7.038</td></tr><tr><td>Zaitsev Z.I.</td><td>BSU</td><td>21</td><td>60.3</td><td>163</td><td>80</td><td>130</td><td>160</td><td>6.136</td></tr><tr><td>Konovalov K.M.</td><td>TSU</td><td>25</td><td>72.6</td><td>179</td><td>120</td><td>190</td><td>220</td><td>7.300</td></tr><tr><td>Frolov F.P.</td><td>ITMO</td><td>26</td><td>101.2</td><td>192</td><td>210</td><td>240</td><td>250</td><td>6.917</td></tr><tr><td>Belov B.V.</td><td>UralSU</td><td>28</td><td>85.8</td><td>190</td><td>220</td><td>240</td><td>250</td><td>8.275</td></tr><tr><td>Klimov K.K.</td><td>SPbPU</td><td>21</td><td>64.5</td><td>170</td><td>130</td><td>180</td><td>210</td><td>8.062</td></tr></table> <div>Goodbye!</div>	Name	University	Age	Weight	Height	Res1	Res2	Res3	Index	Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405	Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055	Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568	Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765	Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159	Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038	Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136	Konovalov K.M.	TSU	25	72.6	179	120	190	220	7.300	Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917	Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275	Name	University	Age	Weight	Height	Res1	Res2	Res3	Index	Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405	Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055	Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568	Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765	Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159	Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038	Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136	Konovalov K.M.	TSU	25	72.6	179	120	190	220	7.300	Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917	Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275	Klimov K.K.	SPbPU	21	64.5	170	130	180	210	8.062
Name	University	Age	Weight	Height	Res1	Res2	Res3	Index																																																																																																																																																																																																									
Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405																																																																																																																																																																																																									
Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055																																																																																																																																																																																																									
Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568																																																																																																																																																																																																									
Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765																																																																																																																																																																																																									
Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159																																																																																																																																																																																																									
Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038																																																																																																																																																																																																									
Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136																																																																																																																																																																																																									
Konovalov K.M.	TSU	25	72.6	179	120	190	220	7.300																																																																																																																																																																																																									
Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917																																																																																																																																																																																																									
Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275																																																																																																																																																																																																									
Name	University	Age	Weight	Height	Res1	Res2	Res3	Index																																																																																																																																																																																																									
Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405																																																																																																																																																																																																									
Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055																																																																																																																																																																																																									
Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568																																																																																																																																																																																																									
Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765																																																																																																																																																																																																									
Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159																																																																																																																																																																																																									
Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038																																																																																																																																																																																																									
Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136																																																																																																																																																																																																									
Konovalov K.M.	TSU	25	72.6	179	120	190	220	7.300																																																																																																																																																																																																									
Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917																																																																																																																																																																																																									
Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275																																																																																																																																																																																																									
Klimov K.K.	SPbPU	21	64.5	170	130	180	210	8.062																																																																																																																																																																																																									

## Содержимое файлов

input1.csv

1

Ivanov I.I.;MSU;25;70.5;175;120;200;230

2

Petrov P.P.;SPbSU;22;65.2;180;140;180;210

3

Sidorov S.S.;MIPT;18;55.8;165;90;150;180

4

Kuznetsov K.K.;HSE;20;75.1;185;200;220;240

5

Smirnov A.A.;MGU;27;90.3;190;180;220;250

6

Fedorov F.F.;NRU HSE;19;68.7;170;110;170;200

7

Volkov V.V.;BSU;21;60.0;160;80;0;0

8

Mikhailov M.M.;TSU;24;72.4;178;160;190;220

9

Novikov N.N.;ITMO;26;80.6;195;210;240;250

10

Morozov M.I.;RANEPA;23;73.8;183;170;200;230

input2.csv

1

Kozlov K.V.;UralSU;28;85.2;188;220;240;250

2

Orlov O.D.;MIIT;17;57.5;163;100;160;0

3

Nikitin N.P.;SFU;30;95.0;200;230;250;250

4

Kovalev K.N.;KPI;16;50.6;155;70;120;150

5

Ilyin I.S.;SPbPU;20;63.9;172;130;180;210

6

Sergeev S.I.;TomSU;22;67.2;168;0;0;0

7

Stepanov S.V.;BSTU;18;58.4;166;120;170;200

8

Gusev G.A.;KAI;26;76.8;182;200;220;240

9

Popov P.V.;NSU;25;71.7;176;180;210;230

10

Vasiliev V.S.;TSU;19;66.0;170;110;160;190

input3.csv

1

Pavlov P.A.;MSU;24;70.2;177;160;200;230

2

Kiselev K.A.;SPbSU;20;65.8;175;140;180;210

3

Bogdanov B.B.;MIPT;18;55.5;162;90;150;180

4

Danilov D.D.;HSE;22;75.3;183;200;220;240

5

Nesterov N.N.;MGU;27;90.8;188;180;220;250

6

Romanov R.I.;NRU HSE;19;68.2;168;110;170;200

7

Zaitsev Z.I.;BSU;21;60.3;163;80;130;160

8

Konovalov K.M.;TSU;25;72.6;179;120;190;220

9

Frolov F.P.;ITMO;26;101.2;192;210;240;250

10

Belov B.V.;UralSU;28;85.8;190;220;240;250



## Примеры выполнения программы

Please enter the file name:

*input1.csv*

Please enter count of lines in file (max 100):

*10*

The file has successfully been processed!

To display the data, enter the command "!print"

To find users, enter the command "!find"

To sort the data, enter the command "!sort"

To add new data, enter the command "!add"

To end the program, enter the command "!end"

*!print*

Name	University	Age	Weight	Height	Res1	Res2	Res3	Index
Ivanov I.I.	MSU	25	70.5	175	120	200	230	7.801
Petrov P.P.	SPbSU	22	65.2	180	140	180	210	8.129
Sidorov S.S.	MIPT	18	55.8	165	90	150	180	7.527
Kuznetsov K.K.	HSE	20	75.1	185	200	220	240	8.788
Smirnov A.A.	MGU	27	90.3	190	180	220	250	7.198
Fedorov F.F.	NRU HSE	19	68.7	170	110	170	200	6.987
Volkov V.V.	BSU	21	60.0	160	80	0	0	1.333
Mikhailov M.M.	TSU	24	72.4	178	160	190	220	7.873
Novikov N.N.	ITMO	26	80.6	195	210	240	250	8.685
Morozov M.I.	RANEPA	23	73.8	183	170	200	230	8.130

*!sort*

Select a field to sort by:

1 = age

2 = weight

3 = height

4 = index

Enter only one number!

*2*

The data has been successfully sorted!

Name	University	Age	Weight	Height	Res1	Res2	Res3	Index
Sidorov S.S.	MIPT	18	55.8	165	90	150	180	7.527
Volkov V.V.	BSU	21	60.0	160	80	0	0	1.333
Petrov P.P.	SPbSU	22	65.2	180	140	180	210	8.129
Fedorov F.F.	NRU HSE	19	68.7	170	110	170	200	6.987
Ivanov I.I.	MSU	25	70.5	175	120	200	230	7.801
Mikhailov M.M.	TSU	24	72.4	178	160	190	220	7.873
Morozov M.I.	RANEPA	23	73.8	183	170	200	230	8.130
Kuznetsov K.K.	HSE	20	75.1	185	200	220	240	8.788
Novikov N.N.	ITMO	26	80.6	195	210	240	250	8.685
Smirnov A.A.	MGU	27	90.3	190	180	220	250	7.198

*!end*

Goodbye!

Please enter the file name:

*input2.csv*

Please enter count of lines in file (max 100):

*10*

The file has successfully been processed!

To display the data, enter the command "!print"

To find users, enter the command "!find"

To sort the data, enter the command "!sort"

To add new data, enter the command "!add"

To end the program, enter the command "!end"

*!find*

Select a field to find by:

1 = name

2 = university

Enter only one number!

*2*

Enter the search string:

*m*

+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
Name	University	Age	Weight	Height	Res1	Res2	Res3	Index	
+-----+-----+-----+-----+-----+-----+-----+-----+-----+									
Orlov O.D.	MIIT	17	57.5	163	100	160	0	4.522	
Sergeev S.I.	TomSU	22	67.2	168	0	0	0	0.000	
+-----+-----+-----+-----+-----+-----+-----+-----+-----+									

*!end*

Goodbye!

Please enter the file name:

*input3.csv*

Please enter count of lines in file (max 100):

*10*

The file has successfully been processed!

To display the data, enter the command "!print"

To find users, enter the command "!find"

To sort the data, enter the command "!sort"

To add new data, enter the command "!add"

To end the program, enter the command "!end"

*!print*

Name	University	Age	Weight	Height	Res1	Res2	Res3	Index
Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405
Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055
Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568
Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765
Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159
Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038
Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136
Kononov K.M.	TSU	25	72.6	179	120	190	220	7.300
Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917
Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275

*!add*

Enter data of the athlete in format:

name;university;age;weight;height;result1,result2,result3

*Klimov K.K.;SPbPU;21;64.5;170;130;180;210*

The data has been successfully added!

*!print*

Name	University	Age	Weight	Height	Res1	Res2	Res3	Index
Pavlov P.A.	MSU	24	70.2	177	160	200	230	8.405
Kiselev K.A.	SPbSU	20	65.8	175	140	180	210	8.055
Bogdanov B.B.	MIPT	18	55.5	162	90	150	180	7.568
Danilov D.D.	HSE	22	75.3	183	200	220	240	8.765
Nesterov N.N.	MGU	27	90.8	188	180	220	250	7.159
Romanov R.I.	NRU HSE	19	68.2	168	110	170	200	7.038
Zaitsev Z.I.	BSU	21	60.3	163	80	130	160	6.136
Kononov K.M.	TSU	25	72.6	179	120	190	220	7.300
Frolov F.P.	ITMO	26	101.2	192	210	240	250	6.917
Belov B.V.	UralSU	28	85.8	190	220	240	250	8.275
Klimov K.K.	SPbPU	21	64.5	170	130	180	210	8.062

*!end*

Goodbye!

## **Выводы**

В результате выполнения работы изучены особенности работы с указателями на структуры и функции в языке Си. А также получены практические навыки в работе с динамическими массивами структур.