**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. 3311 |  | Шарпинский Д. А. |
| Преподаватель |  | Хахаев И. А. |

Санкт-Петербург

2023

**Цель работы.**

Научиться работать с указателями на структуры и функциями

**Задание (вариант 4)**

В **работе № 7** нужно выбирать осмысленную предметную область с таким расчетом, чтобы у объектов можно было бы написать указанные в задании характеристики

Задание: выбор записей по значению любого символьного поля (выбор из меню), сортировка результата по убыванию значений последнего числового поля.

**Постановка задачи и описание решения**

Задача: Реализация программы для обработки данных о пользователях, хранящихся в формате CSV. Необходимо обеспечить чтение данных из файла, добавление новых данных с клавиатуры, сортировку по числовому полю и поиск по текстовому полю.

Выбранная мною предметная область – упрощенная модель пользователя в базе данных социальной сети.

Структура User включает в себя следующие поля:

|  |  |  |
| --- | --- | --- |
| Название поля | Тип | Описание |
| id | int | Уникальный идентификатор пользователя |
| fullName | char \* | Полное имя пользователя |
| age | int | Возраст пользователя |
| profession | char \* | Профессия пользователя |
| friendsRating | float | Рейтинг среди друзей |
| publicRating | float | Общественный рейтинг |
| friendsCount | int | Количество друзей |
| friendsId | int \* | Массив идентификаторов друзей |

Основные функции программы:

Чтение данных из файла CSV и их обработка.

Добавление новых пользователей через ввод с клавиатуры.

Сортировка пользователей по количеству друзей.

Поиск пользователей по имени или профессии.

Описание работы программы:

При запуске программа пытается открыть файл CSV. В случае успеха, данные из файла считываются и обрабатываются.

Пользователь может добавить новых пользователей, вводя их данные через консоль.

Программа предоставляет возможность сортировки массива пользователей по убыванию количества друзей.

Реализован поиск пользователей по заданному имени или профессии.

Особенности реализации:

Для хранения данных о пользователях использован динамический массив указателей на структуры User.

Использование динамического выделения памяти для строковых полей в структуре.

Функция qsort применяется для сортировки массива структур.

Обработка текстовых данных осуществляется через функции из string.h.

Управление памятью: при работе программы память для строковых полей структур, самих структур и массива структур выделяется динамически. Производится очистка памяти во избежание её утечек.

**Описание переменных**

**Функция main()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | users | User \*\* | Массив для хранения структур |
| 2 | slen | int | Длина строки |
| 3 | i | int | Индекс в цикле |
| 4 | n | int | Размер массива структур |
| 5 | count | int | Количество элементов в массиве структур |
| 6 | j | int | Количество элементов, соответствующих пользовательскому вводу |
| 7 | sep | char | Разделитель в .csv файле |
| 8 | ask | char | Выбор пользователя |
| 9 | temp | char \* | Временная строка |
| 10 | splitArray | char \*\* | Массив строк для заполнения структуры |
| 11 | file | FILE | Файл для чтения |

**Функция simpleSplit()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | str | char \* | Строка для разбиения |
| 2 | length | int | Длина строки |
| 3 | sep | char | Символ разделитель |

**Функция fillStruct()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | str | char \*\* | Массив строк – будущие поля структуры |

**Функция cmp()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | a | void \* | Элемент массива |
| 2 | b | void \* | Следующий элемент массива |

**Функция sortStructs()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | users | User \*\* | Массив структур |
| 2 | count | int | Количество элементов в массиве структур |

**Функция outStruct()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | user | User \* | Структура |

**Функция printAllUsers()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | users | User \*\* | Массив структур |
| 2 | count | int | Количество элементов в массиве структур |

**Функция trim()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | str | char \* | Строка для обработки |

**Функция addUser()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | usersPtr | User \*\*\* | Указатель на массив структур |
| 2 | count | int \* | Количество элементов в массиве структур |
| 3 | n | int \* | Размер массива структур |

**Функция startsWithIgnoreCase()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | str | char \* | Строка для сравнения |
| 2 | prefix | char \* | Префикс для сравнения |

**Функция clearStruct()**

|  |  |  |  |
| --- | --- | --- | --- |
| № | Имя переменной | Тип | Назначение |
| 1 | user | User \* | Структура для очистки |

**Контрольные примеры**

**Пример:**

Initial array:

ID Full Name Age Profession Friends Rating Public Rating Friends Count Friends IDs

1 John Doe 30 teacher 4.5 3.9 3 [2, 5, 7]

2 Jane Smith 25 engeneer 3.8 4.1 2 [1, 3]

3 Alice Johnson 28 driver 4.2 3.7 4 [1, 2, 6, 8]

4 Michael Brown 33 pilot 3.9 4.0 5 [3, 6, 9, 10, 2]

5 Emily Davis 27 dentist 4.1 3.8 3 [1, 2, 3]

6 David Wilson 35 actor 4.0 4.2 2 [5, 2]

7 Linda Martinez 32 actor 3.9 3.7 4 [4, 6, 5, 1]

8 Robert White 29 teacher 4.3 3.8 3 [1, 2, 3]

9 Sarah Taylor 31 teacher 4.0 4.1 5 [8, 5, 6, 3, 1]

10 James Anderson 34 pilot 4.2 3.9 2 [1, 2]

11 Davidios Morgan 20 teacher 2.0 1.0 0 []

Do you want to add another user? (y/n): y

Enter user ID: 12

Enter full name: newUser

Enter age: 12

Enter profession: pilot

Enter friends rating: 4.3

Enter public rating: 4.0

Enter friends count: 3

Enter friends IDs (example: 1,2,3,4): 1,4,6

New user successfully added!

Do you want to add another user? (y/n): n

Press ENTER to see all users sorted by number of friends

ID Full Name Age Profession Friends Rating Public Rating Friends Count Friends IDs

9 Sarah Taylor 31 teacher 4.0 4.1 5 [8, 5, 6, 3, 1]

4 Michael Brown 33 pilot 3.9 4.0 5 [3, 6, 9, 10, 2]

3 Alice Johnson 28 driver 4.2 3.7 4 [1, 2, 6, 8]

7 Linda Martinez 32 actor 3.9 3.7 4 [4, 6, 5, 1]

5 Emily Davis 27 dentist 4.1 3.8 3 [1, 2, 3]

8 Robert White 29 teacher 4.3 3.8 3 [1, 2, 3]

1 John Doe 30 teacher 4.5 3.9 3 [2, 5, 7]

12 newUser 12 pilot 4.3 4.0 3 [1, 4, 6]

10 James Anderson 34 pilot 4.2 3.9 2 [1, 2]

2 Jane Smith 25 engeneer 3.8 4.1 2 [1, 3]

6 David Wilson 35 actor 4.0 4.2 2 [5, 2]

11 Davidios Morgan 20 teacher 2.0 1.0 0 []

You can now sort users by either name or profession. Choose one option (1 or 2): 1

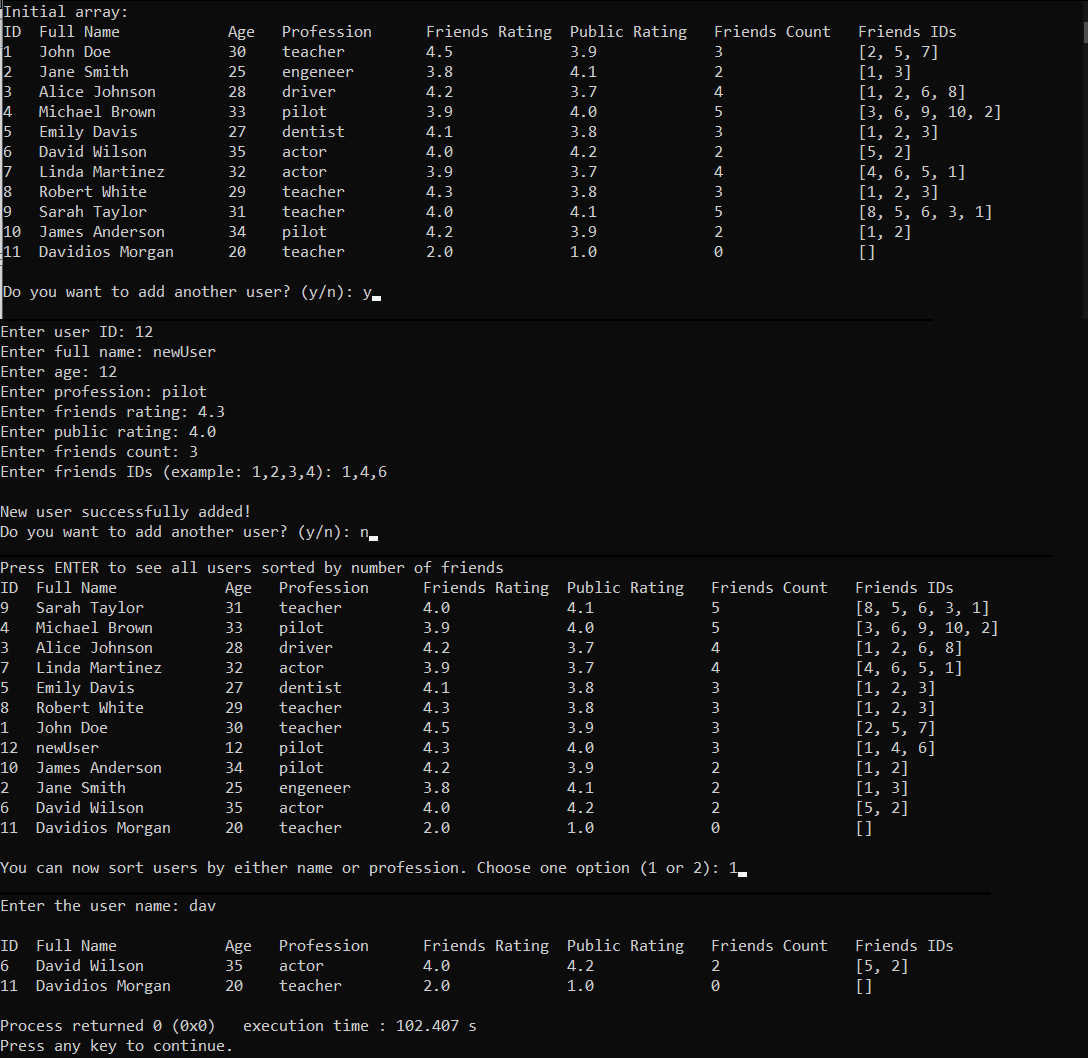
Enter the user name: dav

ID Full Name Age Profession Friends Rating Public Rating Friends Count Friends IDs

6 David Wilson 35 actor 4.0 4.2 2 [5, 2]

11 Davidios Morgan 20 teacher 2.0 1.0 0 []

**Примеры выполнения программы**

****

**Выводы.**

В результате выполнения лабораторной работы были получены навыки работы с указателями на структуры и функциями в C.