

Московский Авиационный Институт
(Национальный Исследовательский Университет)

**Кафедра вычислительной математики и
программирования**

Курсовая работа

По курсу «Языки и методы программирования»

II семестр

**Задание №6 «Обработка последовательной файловой
структуры на языке Си»**

Выполнил студент

1-го курса, 105-ой
группы

Махмудов О. С.

(подпись)

Научный руководитель

Доцент кафедры 806

Сластушенский Ю. В.

(подпись)

Работа защищена

«__»_____ 2019

Оценка_____

ОГЛАВЛЕНИЕ

ЦЕЛЬ.....	3
ПОСТАНОВКА ЗАДАЧИ.....	4
ПРОГРАММА.....	5
Функции	5
Алгоритм	5
Код программы	5
Тесты.....	8
ЗАКЛЮЧЕНИЕ	10

ЦЕЛЬ

Разработать последовательную структуру данных для представления простейшей базы данных на файлах в СП Си в соответствии с заданным вариантом. Составить программу генерации внешнего нетекстового файла заданной структуры, содержащего представительный набор записей (15-20). Распечатать содержимое сгенерированного файла в виде таблицы и выполнить над ним заданное действие для 2-3 значений параметров запроса `p` и распечатать результат. Действие по выборке данных из файла оформить в виде отдельной программы с параметрами запроса, вводимыми из стандартного входного текстового файла, или получаемыми из командной строки UNIX. Второй способ задания параметров обязателен для работ, оцениваемых на хорошо и отлично. Параметры задаются с помощью ключей `-f` (распечатка файла) или `-p <parameter>` (параметры конкретного варианта задания). Получение параметров из командной строки производится с помощью стандартных библиотечных функций `argc` и `argv`.

Структуры данных и константы, совместно используемые программами, следует вынести в отдельный заголовочный файл.

В процессе отладки и тестирования рекомендуется использовать команды обработки текстовых файлов ОС UNIX и переадресацию ввода-вывода. Сгенерированные и отформатированные тестовые данные необходимо заранее поместить в текстовые файлы и распечатывать при протоколировании. Рекомендуется подобрать реальные или правдоподобные тестовые данные. Число наборов тестовых данных должно быть не менее трёх. Имя файла с бинарными данными является обязательным параметром второй программы.

Отчёт должен содержать оценку пространственной и временной сложности использованного алгоритма. В состав отчета также рекомендуется включить графическую иллюстрацию структуры файла и запроса на выборку.

ПОСТАНОВКА ЗАДАЧИ

Сведения о вступительных экзаменах абитуриентов: фамилия, инициалы, пол, номер школы, наличие медали, оценки в баллах.

22. Найти абитуриентов-медалистов, не набравших проходной балл **p**.

Входной текстовый файл представляет собой список (пример) :

фамилия | инициалы | пол | № школы | медаль | экзамены(мат. инф. рус. яз.)

Gribov	MV	m	863	y	76	82	84
--------	----	---	-----	---	----	----	----

Нужно написать 2 программы на языке Си, которые могут:

1. Генерировать внешний нетекстовый файл заданной структуры по входному текстовому файлу с исходными данными.

Синтаксис вызова: компиляция `gcc person_dump.` и запуск файла с параметрами `./a.out input.txt output.bin`

22. Обработать внешний нетекстовый файл с помощью параметров запроса:

- a. Добавить в список новую запись (добавление в конец списка) (параметр – a).
- b. Удалить из списка запись (удаление по фамилии И.О.) (параметр –d).
- c. Вывести весь список (параметр –f).
- d. Вывести абитуриентов-медалистов, не набравших проходной балл **p** (параметр –p).

Синтаксис вызова: компиляция `gcc kursach6.c` и запуск файла с параметрами `./a.out output.bin [a|d|f|p]`

ПРОГРАММА

Функции

<code>void print_data(const char *file)</code>	Печатает данные из бинарного файла
<code>void task(const char *file, int par)</code>	Печатает имена и инициалы абитуриентов согласно заданию

Алгоритм

1. Запускаем `person_dump.c` с параметрами: названия файла с базой данных и бинарного файла
2. Запускаем основную программу с нужной функцией и названием бинарного файла
3. Программа сравнивает значение `argv` с параметрами, узнает написанный параметр и обращается к нужному пункту оператора выбора `switch`
4. Вывод базы данных происходит в виде таблицы
5. Вывод абитуриентов-медалистов, ненабравших проходной балл `p`: производится с помощью процедуры, в которой сравниваются пол и наличие медали абитуриента, а также параметр `p` сравнивается с суммой всех экзаменов.

Код программы

Содержимое библиотеки «`person.h`»

```
#ifndef __person_h__
```

```
#define __person_h__
```

```

typedef struct person {

char surname[15];

char np[3];

char gender[2];

char school[5];

char medal[2];

int mat;

int inf;

int rus;

} person;

#endif

```

Содержимое person_dump.c

```

#include "stdio.h"
#include "string.h"
#include "locale.h"
#include "stdbool.h"

#include "person.h"

int main(int argc, const char *argv[]) {
    if (argc != 3) {
        printf("Error\n");
        return 1;
    }
    person p;
    FILE *in = fopen(argv[1], "r");
    FILE *out = fopen(argv[2], "wb");
    while (!feof(in)) {
        fscanf(in, "%s %s %s %s %s %d %d %d\n", p.surname, p.np, p.gender, p.school,
p.medal, &p.mat, &p.inf, &p.rus);
        fwrite(&p, sizeof(person), 1, out);
    }
    fclose(in);
    fclose(out);
    return 0;
}

```

Содержимое kursach6.c

```
#include "stdio.h"
#include "stdlib.h"
#include "malloc.h"
#include "string.h"

#include "person.h"

void print_data(const char *file) {
    FILE *in = fopen(file, "rb");
    person p;
    while (fread(&p, sizeof(person), 1, in)) {
        printf("%s %s %s %s %s %d %d %d\n", p.surname, p.np, p.gender, p.school, p.medal,
p.mat, p.inf, p.rus);
    }
    fclose(in);
}

void task(const char *file, int par) {
    FILE *in = fopen(file, "rb");
    person p;
    int sum = 0;
    while (fread(&p, sizeof(person), 1, in)) {
        sum = p.mat + p.inf + p.rus;
        if (strcmp(p.gender, "m") == 0 && strcmp(p.medal, "y") == 0 && sum < par) {
            printf("%s %s\n", p.surname, p.np);
        }
    }
    fclose(in);
}

int main(int argc, const char *argv[]) {
    if (argc < 2) {
        printf("Enter a file name!\n");
        return 1;
    }
    if (argc == 2) {
        printf("Enter operation key!\n");
        return 2;
    }
    person p;
    FILE *in = fopen(argv[1], "rb+wb");
    int par = 0, x = 0;
    char surname[20], np[3];
    if (argc > 2) {
        if (*argv[2] == 'f') {
            print_data(argv[1]);
        }
        if (*argv[2] == 'p') {
            printf("Enter pass mark: ");
            scanf("%d", &par);
            task(argv[1], par);
        }
        if (*argv[2] == 'a') {
            fseek(in, 0, SEEK_END);
        }
    }
}
```

```

        printf("Enter data about applicant: ");
        scanf("%s %s %s %s %s %d %d %d", p.surname, p.np, p.gender, p.school,
p.medal, &p.mat, &p.inf, &p.rus);
        fwrite(&p, sizeof(person), 1, in);
    }
    if (*argv[2] == 'd') {
        printf("Enter surname and np applicant: ");
        scanf("%s %s", surname, np);
        FILE *out = fopen("temp.bin", "wb+");
        while (!feof(in)) {
            if (fread(&p, sizeof(p), 1, in) == 0)
                break;
            if (strcmp(p.surname, surname) == 0 && strcmp(p.np, np) == 0)
                x = 1;
            else
                fwrite(&p, sizeof(p), 1, out);
        }
        if (x == 0)
            printf("Not found\n");
        else {
            fclose(in);
            in = fopen(argv[1], "wb+");
            fseek(out, 0, SEEK_SET);
            while (!feof(out)) {
                if (fread(&p, sizeof(p), 1, out) == 0)
                    break;
                fwrite(&p, sizeof(p), 1, in);
            }
        }
        fclose(out);
    }
    }
    fclose(in);
    return 0;
}

```

Тесты

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$ gcc person_dump.c

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out input.txt output.bin

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$ cat input.txt

Bovshovskiy EI m 974 y 70 94 80

Arefieva OV w 1835 y 62 76 90

Klimov AS m 984 n 70 72 86

Sigacheva LV w 567 n 68 76 84

Kobyakova TL w 440 y 76 80 100

Novikov AI m 356 y 80 90 84

Kondakova EV w 672 n 62 76 88

Cay UA m 7896 n 62 70 78

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$ gcc kursach6.c

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin f

Bovshovskiy EI m 974 y 70 94 80

Arefieva OV w 1835 y 62 76 90

Klimov AS m 984 n 70 72 86

Sigacheva LV w 567 n 68 76 84

Kobyakova TL w 440 y 76 80 100

Novikov AI m 356 y 80 90 84

Kondakova EV w 672 n 62 76 88

Cay UA m 7896 n 62 70 78

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin p

Enter pass mark: 255

Bovshovskiy EI

Novikov AI

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin a

Enter data about applicant: Ivanov AA m 475 y 84 84 84

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin f

Bovshovskiy EI m 974 y 70 94 80

Arefieva OV w 1835 y 62 76 90

Klimov AS m 984 n 70 72 86

Sigacheva LV w 567 n 68 76 84

Kobyakova TL w 440 y 76 80 100

Novikov AI m 356 y 80 90 84

Kondakova EV w 672 n 62 76 88

Cay UA m 7896 n 62 70 78

Ivanov AA m 475 y 84 84 84

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin p

Enter pass mark: 255

Bovshovskiy EI

Novikov AI

Ivanov AA

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin d

Enter surname and np applicant: Novikov AI

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin f

Bovshovskiy EI m 974 y 70 94 80

Arefieva OV w 1835 y 62 76 90

Klimov AS m 984 n 70 72 86

Sigacheva LV w 567 n 68 76 84

Kobyakova TL w 440 y 76 80 100

Kondakova EV w 672 n 62 76 88

Cay UA m 7896 n 62 70 78

Ivanov AA m 475 y 84 84 84

Admin@LAPTOP-Q5U6S2UH:/mnt/c/Users/Admin/Desktop/Все для вуза\$./a.out output.bin p

Enter pass mark: 255

Bovshovskiy EI

Ivanov AA

ЗАКЛЮЧЕНИЕ

В процессе выполнения данной курсовой работы, я освоил навыки работы с последовательной файловой структурой, изучил много новых функций для обработки файлов.