

Отчет по Курсовой работе № 6

по курсу: 1 фундаментальная информатика

студент группы : М8О-105Б-21 Титеев Рамиль Маратович , № по списку: 23

Адреса www, e-mail, jabber, skype: derol.gym@gmail.com

Работа выполнена: "31 марта 2022г"

Преподаватель: каф. 806 В.К.Титов

Входной контроль знаний с оценкой: _____

Отчет сдан "___" _____ 20__ г., итоговая оценка _____

Подпись преподавателя _____

1. Тема: Обработка последовательной файловой структуры на языке Си.

2. Цель работы: Разработать последовательную структуру данных для представления простейшей базы данных на файлах в СП Си.

3. Задание(*вариант 23*): Найти абитуриентов-медалистов, получивших неудовлетворительную оценку по математике.

4. Оборудование(*лабораторное*):

ЭВМ _____, процессор _____, имя узла сети _____ с ОП _____ ГБ

НМД _____ ГБ. Терминал _____ адрес _____. Принтер _____

Другие устройства _____

Оборудование ПЭВМ студента, если использовалось:

Процессор Ryzen 4600 @ 6x 3.0 GHz , ОП 16384 МБ, НМД _____ ГБ. Монитор Встроенный

Другие устройства _____

5. Программное обеспечение(*лабораторное*):

Операционная система семейства UNIX, наименование _____ версия _____

Интерпретатор команд: _____ версия _____

Система программирования: _____ версия _____

Редактор текстов: _____ версия _____

Утилиты операционной системы: _____

Прикладные системы и программы: _____

Местонахождение и имена файлов и программ данных: _____

Программное обеспечение ЭВМ студента, если использовалось:

Операционная система семейства UNIX, наименование Ubuntu версия 20.04

Интерпретатор команд: bash версия _____

Система программирования: C версия _____

Редактор текстов: Emacs версия _____

Утилиты операционной системы: _____

Прикладные системы и программы: _____

Местонахождение и имена файлов и программ данных: /usr/bin , а также /bin

6. Идея, метод, алгоритм решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальное описание с пред- и постусловиями)

При запуске *dump.cpp* нужно ввести два названия файлов: входной файл, в котором находятся исходные данные, и выходной, в который будет произведена запись. Затем создается двоичный файл и в него производится запись.

После этого запускаем вторую программу *cols.cpp*. Для неё нужно ввести сначала ключ, который определяет, что требуется от программы, а затем два названия файлов: двоичный файл, который был создан предыдущей программой, и выходной *txt*-файл, в который будет произведена запись. Далее проверяем ключ: $-f$ значит, что нужно просто вывести все данные в виде таблицы и записать их в файл, $-t$ значит, что нужно вывести данные, удовлетворяющие заданному условию.

Для условия $-f$ все очевидно: просто проходимся по файлу и выводим все его данные.

Для $-t$ почти аналогично: считанные данные проверяем на выполнения условия, в зависимости от этого записываем его в файл и выводим в таблицу в терминале или пропускаем. После всех проверок в терминале выводится готовая таблица, а в файле записываются все данные, которые удовлетворяют условию.

7. Сценарий выполнения работы [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты, либо соображения по тестированию].

student.h:

```
#ifndef LABS_STUDENTS_H
#define LABS_STUDENTS_H
typedef struct{
    char surname[15];
    char initials[5];
    char is_entrant[5];
    char is_medal[5];
    int math;
    int inf;
    int fiz;
} student;
#endif
```

dump.cpp:

```
#include <stdio.h>
#include <string.h>
#include <errno.h>

#include "student.h"

void usage(){
    printf("Usage: programm input_filename output_filename\n");
}

int main(int argc, char *argv[]){
    if (argc != 3){
        usage();
        return 1;
    }
    student s;
    FILE *fi, *fo;
    fi = (fi=fopen(argv[1], "r"));
    if (!fi){
        perror("Can't open input file");
        return 2;
    }
```

```

    }
    fo = (fo=fopen(argv[2], "wb"));
    if (!fo){
        perror("Can`t open output file");
        return 2;
    }
    while(!feof(fi)){
        fscanf(fi, "%s %s %s %s %d %d\t%d\n", s.surname,
            s.initial, s.is_entrant, s.is_medal, &s.math,
            &s.inf, &s.fiz);
        fwrite(&s, sizeof(s), 1, fo);
    }

    return 0;
}

```

cols.cpp:

```

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <errno.h>
#include "student.h"

void usage(){
    printf("Usage: programm [-key] input_filename output_filename\n");
    printf("Keys: \n");
    printf("-f output all data \n");
    printf("-t complete the task \n");
}

int main(int argc, char *argv[]){
    if (argc != 4){
        usage();
        return 1;
    }
    student s;
    FILE *fi = (fi=fopen(argv[2], "r"));
    if (!fi){
        perror("Can`t open input file");
        return 2;
    }
    FILE *fo = (fo=fopen(argv[3], "wb"));
    if (!fo){
        perror("Can`t open output file");
        return 2;
    }
    if(!strcmp(argv[1], "-f")){
        printf("=====\n");
        printf("| Surname and initials |Is entrant|Is medal| Math\
| Inf | Fiz |\n");
        printf("=====\n");
        while(!feof(fi)){

```

```

        fread(&s, sizeof(s), 1, fi);
        fprintf(fo, "surname=%s initials=%s initials=%s is_medal=%s\
math=%d inf=%d fiz=%d\n", s.surname, s.initials, s.is_entrant, s.is_medal,
        s.math, s.inf, s.fiz);

        printf("|      %-10s %2s      |   %3s   |   %3s   |   %1d   |\
%1d   |   %1d   |\n", s.surname, s.initials, s.is_entrant,
        s.is_medal, s.math, s.inf, s.fiz);

        printf("-----\
-----\n");
    }
}
else{
    if (!strcmp(argv[1], "-t")){
        printf("=====\
=====\n");
        printf("| Surname and initials |Is entrant|Is medal| Math |\
Inf | Fiz |\n");
        printf("=====\
=====\n");
        while(!feof(fi)){
            fread(&s, sizeof(s), 1, fi);
            if (!strcmp(s.is_entrant, "yes")
                && !strcmp(s.is_medal, "yes")
                && (s.math < 3)){
                fprintf(fo, "surname=%s initials=%s initials=%s\
is_medal=%s math=%d inf=%d fiz=%d\n",
                s.surname, s.initials, s.is_entrant,
                s.is_medal, s.math, s.inf, s.fiz);
                printf("|      %-10s %2s      |   %3s   |   %3s   |\
%1d   |   %1d   |   %1d   |\n", s.surname, s.initials,
                s.is_entrant, s.is_medal, s.math, s.inf, s.fiz);
                printf("-----\
-----\n");
            }
        }
    }
    else{
        perror("Invalid key");
        return 1;
    }
}
}

```

Пункты 1-7 отчёта составляются **строго до** начала лабораторной работы.

Допущен к выполнению работы. Подпись преподавателя _____

8. Распечатка протокола (подклеить листинг окончательного варианта программы с текстовыми примерами, подписанный преподавателем).

```

(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat header.txt
*****
*                               *
*           Курсовая работа №6           *
*  Обработка последовательной файловой структуры на языке Си.  *
*           Выполнил студент гр. М80-105-Б           *
*           Титеев Рамиль Маратович           *
*                               *

```

```

*****
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat in1.txt
Ivanov II yes yes 2 3 3
Petrov PP yes no 4 5 5
Suvova SS no no 2 4 4
Smirova SS no no 4 4 4
Orlova OO yes yes 2 4 4
Babova AA no no 3 3 5
Gromova KK yes yes 4 4 5
Antonov AA yes no 5 5 5
Kovrov EE no no 4 5 5
Serova NN yes yes 2 5 4
Petrov PP yes no 4 5 5
Suvova SS yes no 5 4 4
Smirova SS yes no 2 4 4
Orlova OO no no 4 4 4
Babova AA yes no 4 4 5
Gromova KK yes yes 5 4 5
Antonov AA yes yes 3 5 5
Kovrov EE yes no 5 5 5
Serova NN yes yes 4 5 5
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat in2.txt
Ivanov II yes yes 5 5 5
Petrov PP yes yes 2 5 5
Suvova SS yes yes 2 4 4
Smirova SS no no 4 5 2
Orlova KK no yes 2 4 4
Kovrov AA yes no 3 3 5
Suvova SS yes yes 2 4 4
Smirova SS yes yes 2 2 4
Orlova OO no no 4 2 2
Babova AA yes no 4 4 5
Gromova OO no yes 5 4 5
Antonov AA yes yes 2 5 2
Babova EE no yes 5 5 5
Serova NN yes yes 4 5 5
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat in3.txt
Kovrov II no yes 2 3 3
Petrov PP no no 4 5 5
Suvova SS no no 2 4 4
Antonov NN no no 4 4 4
Orlova OO yes yes 2 4 4
Babova AA no no 3 3 5
Gromova KK yes yes 4 4 5
Gromova KK yes no 5 4 5
Smirova AA yes no 3 5 5
Ivanov EE yes no 5 5 5
Serova SS no yes 4 5 5
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ g++ dump.cpp
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out
Usage: programm input_filename output_filename
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out in1.txt out1.dat
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out in2.txt out2.dat
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out in3.txt out3.dat
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ls
a.out  cols.cpp  dump.cpp  header.txt  in1.txt  in2.txt  in3.txt  out1.dat  out2.dat  out3.dat  report
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ g++ cols.cpp
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out
Usage: programm [-key] input_filename output_filename
Keys:
-f output all data
-t complete the task
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -f out1.dat out1.txt

```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Ivanov II	yes	yes	2	3	3
Petrov PP	yes	no	4	5	5
Suvova SS	no	no	2	4	4
Smirova SS	no	no	4	4	4
Orlova OO	yes	yes	2	4	4
Babova AA	no	no	3	3	5
Gromova KK	yes	yes	4	4	5
Antonov AA	yes	no	5	5	5
Kovrov EE	no	no	4	5	5
Serova NN	yes	yes	2	5	4
Petrov PP	yes	no	4	5	5
Suvova SS	yes	no	5	4	4
Smirova SS	yes	no	2	4	4
Orlova OO	no	no	4	4	4
Babova AA	yes	no	4	4	5
Gromova KK	yes	yes	5	4	5
Antonov AA	yes	yes	3	5	5
Kovrov EE	yes	no	5	5	5
Serova NN	yes	yes	4	5	5
Serova NN	yes	yes	4	5	5

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -t out1.dat out1.txt
```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Ivanov II	yes	yes	2	3	3
Orlova OO	yes	yes	2	4	4
Serova NN	yes	yes	2	5	4

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat out1.txt
```

```
surname=Ivanov initials=II initials=yes is_medal=yes math=2 inf=3 fiz=3
```

```
surname=Orlova initials=OO initials=yes is_medal=yes math=2 inf=4 fiz=4
```

```
surname=Serova initials=NN initials=yes is_medal=yes math=2 inf=5 fiz=4
```

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -f out2.dat out2.txt
```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Ivanov II	yes	yes	5	5	5

Petrov	PP	yes	yes	2	5	5
Suvova	SS	yes	yes	2	4	4
Smirova	SS	no	no	4	5	2
Orlova	KK	no	yes	2	4	4
Kovrov	AA	yes	no	3	3	5
Suvova	SS	yes	yes	2	4	4
Smirova	SS	yes	yes	2	2	4
Orlova	OO	no	no	4	2	2
Babova	AA	yes	no	4	4	5
Gromova	OO	no	yes	5	4	5
Antonov	AA	yes	yes	2	5	2
Babova	EE	no	yes	5	5	5
Serova	NN	yes	yes	4	5	5
Serova	NN	yes	yes	4	5	5

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -t out2.dat out2.txt
```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Petrov PP	yes	yes	2	5	5
Suvova SS	yes	yes	2	4	4
Suvova SS	yes	yes	2	4	4
Smirova SS	yes	yes	2	2	4
Antonov AA	yes	yes	2	5	2

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat out2.txt
```

```
surname=Petrov initials=PP initials=yes is_medal=yes math=2 inf=5 fiz=5
surname=Suvova initials=SS initials=yes is_medal=yes math=2 inf=4 fiz=4
surname=Suvova initials=SS initials=yes is_medal=yes math=2 inf=4 fiz=4
surname=Smirova initials=SS initials=yes is_medal=yes math=2 inf=2 fiz=4
surname=Antonov initials=AA initials=yes is_medal=yes math=2 inf=5 fiz=2
```

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -f out3.dat out3.txt
```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Kovrov II	no	yes	2	3	3
Petrov PP	no	no	4	5	5
Suvova SS	no	no	2	4	4
Antonov NN	no	no	4	4	4
Orlova OO	yes	yes	2	4	4
Babova AA	no	no	3	3	5

Gromova	KK	yes	yes	4	4	5
Gromova	KK	yes	no	5	4	5
Smirova	AA	yes	no	3	5	5
Ivanov	EE	yes	no	5	5	5
Serova	SS	no	yes	4	5	5
Serova	SS	no	yes	4	5	5

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ ./a.out -t out3.dat out3.txt
```

Surname and initials	Is entrant	Is medal	Math	Inf	Fiz
Orlova 00	yes	yes	2	4	4

```
(base) ramil@ramil:~/projects/laboratory/2_semester/curs_6$ cat out3.txt
surname=Orlova initials=00 initials=yes is_medal=yes math=2 inf=4 fiz=4
```

9. Дневник отладки должен содержать дату и время сеансов отладки, и основные ошибки (ошибки в сценарии и программе, не стандартные операции) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

№	Лаб или дом.	Дата	Время	Событие	Действие по исправлению	Примечание

10. Замечание автора по существу работы _____

11. Выводы _____ Я научился работать с текстовыми файлами в языке Си.

Недочеты, допущенные при выполнении задания, могут быть устранены следующим образом _____

Подпись студента _____