Федеральное государственное бюджетное образовательное учреждение высшего образования "Московский авиационный институт (Национальный исследовательский университет)"

Институт №8 "Информационные технологии и прикладная математика"

Кафедра №806 "Вычислительная математика и программирование"

КУРСОВАЯ РАБОТА

По курсу "Практикум программирования"

Задание №6.

"Обработка последовательной файловой структуры на языке Си"

Выполнил: студент группы М8О-104Б-22
Тесля Данила Сергеевич
Руководитель:
Потенко Максим Алексеевич
Дата: 04.10.2023
Оценка:.....

Москва - 2023 г.

Содержание

1 Постановка задачи

Разработать последовательную структуру данных для представления простейшей базы данных на файлах СП Си. Составить программу генерации внешнего нетекстового файла заданной структуры, содержащего набор записей. Распечатать содержимое файла в виде таблицы и выполнить над ним заданное действие

Вариант №39.

Информация о пассажирах аэропорта : фамилия, инициалы, количество вещей, общий вес вещей, пункт назначения, время вылета, наличие пересадок, сведения о детях. Выяснить, имеется ли пассажир, багаж которого превышает багаж каждого из остальных пассажиров и по весу и по числу вещей

2 Ход решения

2.1 Алгоритм

Перед выполнением работы необходимо изучить материалы по разработке последовательных структур данных для представления простейшей базы данных на файлах на языке Си, изучить основные методы работы с файловыми потоками в программах на языке Си, а также изучить основы модульного программирования на Си и автоматизацию сборки программ модульной структуры на Си с использованием утилиты make. Для выполнения этого задания курсового проекта будет необходимо создать две подпрограммы – одна из которых будет отвечать за генерации внешнего нетекстового файла заданной структуры, содержащего набор записей, а вторая – главная программа, за обработку этих структур данных, и выполнения задания по варианту. Создадим заголовочный файл (.h) – общий для обеих подпрограмм, в котором определим структуру одного элемента базы данных – пассажира самолета. Создадим первую подпрограмму из трех файлов, один из которых будет главным файлом подпрограммы, генерирующей нетекстовый входной файл из текстового, в нем будет функция main, во втором файле будет реализация всех необходимых функций подпрограммы – функций для считывания текстового файла и записи данных в нетекстовый – бинарный файл, и вывода информации из файлов на экран, и третий – заголовочный файл с определениями этих функций. Создадим тестовый текстовый файл с заготовленными данными –их будет считывать первая подпрограмма, и бинарный файл в который эта подпрограмма будет осуществлять запись данных. Далее создадим основную программу из трех файлов – один из них – главный файл программы, второй – файл с

реализациями функций, необходимых для выполнения самого задания согласно варианту – считывание одной структуры данных из стандартного потока ввода, добавление структуры данных в бинарный файл, добавление структуры данных в текстовый файл, удаление структуры данных из бинарного файла, удаление структуры данных из текстового файла, функция поиска пассажира по условию варианта, удаление базы данных. Также дополнительно создадим файл в котором будут команды – указания для корректной сборки и компиляции обеих подпрограмм, запускаться данный файл будет с помощью утилиты make. Когда все эти функции будут реализованы – перейдем главному файлу главной программы – в функции main сделаем контекстное меню, где пользователь сможет произвести действия с базой данных по выбору - добавление структуры данных в бинарный файл, добавление структуры данных в текстовый файл, удаление структуры данных из бинарного файла, удаление структуры данных из текстового файла, функция поиска пассажира по условию варианта, вывод данных из текстового файла на экран, вывод данных из бинарного файла на экран, сделать это можно например конструкцией switch case в цикле while, и считывать выбор пользователя, пока он не захочет завершить работу с базой данных.

2.2 Решение

Для выполнения этого задания курсового проекта будет необходимо создать две подпрограммы – одна из которых будет отвечать за генерации внешнего нетекстового файла заданной структуры, содержащего набор записей, а вторая – главная программа, за обработку этих структур данных, и выполнения задания по варианту. Создадим текстовый файл Makefile, содержащий набор инструкций о том, как компилировать и линковать исходный код программы, в нем отдельно пропишем цель generate для сборки и линковки подпрограммы для генерации бинарного файла, у подпрограммы будет один выходной файл – generate.out. Создадим тестовый текстовый файл database.txt с заготовленными данными –их будет считывать первая подпрограмма, и бинарный файл database.bin в который эта подпрограмма будет осуществлять запись данных. Создадим заголовочный файл passenger.h – общий для обеих подпрограмм, в котором определим структуру одного элемента базы данных Passenger, – пассажира самолета, полями структуры будут char surname[LEN-STR];char initials[LEN-STR];int luggage;int weight;char destination[LEN-STR];Datetime time (сама по себе является структурой, с полями, содержащими дату и время вылета);int transfer; int children;, отвечающие соответственно за хранение данных о каждом пассажире - фамилия, инициалы, количество вещей, общий вес вещей, пункт назначения, время и дата вылета, наличие пересадок, сведения

о детях соответственно. Создадим первую подпрограмму из трех файлов, один из которых будет главным файлом подпрограммы – generate.c, генерирующей нетекстовый входной файл из текстового, в нем будет функция main, во втором файле inout.c - реализация всех необходимых функций подпрограммы – чтения данных и текстового файла – passenger-read-txt – функция получает указатель на структуру passenger, и файловый поток – читает из текстового файла одну строку с данными – заполняя ими все поля одной структуры, если удалось прочитать и записать корректно все данные, то возвращает true, если нет, то false, запись данных в текстовый файл passenger-write-txt - функция получает указатель на структуру passenger, и файловый поток – записывает в текстовый файл одну строку с данными – из одной структуры passenger, чтение данных из бинарного файла passenger-read-bin - функция получает указатель на структуру passenger, и файловый поток – читает из бинарного файла одну строку с данными – заполняя ими все поля одной структуры, если удалось прочитать и записать корректно все данные, то возвращает true, если нет, то false, запись данных в бинарный файл passenger-write-bin - - функция получает указатель на структуру passenger, и файловый поток – записывает в бинарный файл одну строку с данными – из одной структуры passenger вывода данных в виде таблицы из текстового passenger-print-from-txt и бинарного файлов passenger-print-from-bin соответственно – выводят содержимое текстового и бинарного файлов соотвественно, внутри функций в цикле с условием окончания - неправильного чтения – построчно читаются данные, вызовом соответствующих функций, и выводятся в виде таблицы в стандартный поток вывода. И третий – заголовочный файл с определениями этих функций. В третьем файле inout.h – будут прототипы всех функций, реализованных в inout.c. Перейдем к главному файлу программы – в функции main необходимо проверить, были ли переданы два дополнительных аргумента командной строки — текстовый и бинарный файл, и если не были, или были переданы неправильно, то выдать сообщение об ошибке, затем необходимо создать файловые потоки для двух файлов, открыв их для чтения и дозаписи – с модификатором ‘a’ или ‘a+’ – input и output – затем считать в цикле построчно данные из текстового файла и записать в бинарный. Бинарный файл сгенерирован, приступаем к выполнению основной программы.

Далее создадим основную программу из трех файлов – один из них – kp6main.c главный файл программы, второй – search.c файл с реализациями функций, необходимых для выполнения самого задания согласно варианту – считывание одной структуры данных из стандартного потока ввода – passenger-get-from-user – функция получает указатель на структуру passenger, считывает данные, введенные пользовтелем и записывает в структуру, возвращает указатель на эту структуру. добавление структуры

данных в бинарный файл passenger-add-to-bin – записывает в конец бинарного файла блок введенной пользователем информации из структуры, заполненной в результате вызова функции passenger-get-from-user, добавление структуры данных в текстовый файл passenger-add-to-txt - записывает в конец текстового файла блок введенной пользователем информации из структуры, заполненной в результате вызова функции passenger-get-from-user, удаление структуры данных из бинарного файла - passenger-delete-from-bin – получает аргументы – файловый поток и фамилию человека, которого нужно удалить из списка, внутри функции создается временный файловый поток, в который перезаписывается вся база данных из переданного основного файлового потока, кроме человека, которого нужно удалить из базы данных, если такой был найден в файле, затем основной поток закрывается иудаляется, а имя временного файлового потока заменяется на имя основного – переданного в функцию, таким образом – есть новый файл – в котором никогда не было удаленного пассажира – а на самом деле структура данных о нем просто не была перезаписана – т.е. удалена, удаление структуры данных из текстового файла - passenger-delete-from-txt - аналогично, функция поиска пассажира по условию варианта - search-max - получает указатель на массив структур данных типа Passenger из функции main (см. про функцию main), и размер этого массива структур, внутри происходит итерационный процесс в цикле for – проход по всем элементам массива и поиска индексов максимальных элементов по полям weight и luggage, и если по этим двум полям индексы совпадут – пассажир с наибольшим весом и количеством вещей найден – функция возвращает индекс этой структуры в массиве структур, созданном функции main, если они не совпадают – то такого пассажира в списке нет, и функция возвращает -1 – идентификатор ошибки. удаление базы данных - pass-free - очистка данных из под массива структур – по алгоритму очистки памяти для массивов. Создадим третий – заголовочный файл – search.h – в него запишем определения применяемых функций. Перейдем к главному файлу основной программы – в функции main необходимо проверить, были ли переданы два дополнительных аргумента командной строки – текстовый и бинарный файл, и если не были, или были переданы неправильно, то выдать сообщение об ошибке, затем необходимо создать файловые потоки для двух файлов открыв их для чтения и дозаписи – с модификатором ‘a’ или ‘a+’ – filetxt и filebin, а затем сделаем контекстное меню, где пользователь сможет произвести действия с базой данных по выбору - добавление структуры данных в бинарный файл, добавление структуры данных в текстовый файл, удаление структуры данных из бинарного файла, удаление структуры данных из текстового файла, функция поиска пассажира по условию варианта, вывод данных из текстового файла на экран, вывод данных из бинарного файла на экран, сделать это можно например

конструкцией switch case в цикле while, и считывать выбор пользователя, пока он не захочет завершить работу с таблицей. В случае выбора пункта – поиск пассажира с наибольшим весом и наибольшим количеством вещей – в функции main создается и инициализируется массив структур блоков данных о пассажирах – pass – в нее в цикле считываются блоки данных из бинарного файла – и таким образом база данных занесена в программу, далее указатель на эту базу данных передается в функцию поиска. После этого происходит очистка памяти из под базы данных в программе.

3 Код программы и тесты

https://github.com/tesla-2002/kp6

```
tesla-2002@tesla2002-HP-ProBook-430-G7:~/labs/kp6$ make generate
gcc -std=c99 -Wall -Werror generate.c inout.c search.c -o generate.out
tesla-2002@tesla2002-HP-ProBook-430-G7:~/labs/kp6$ valgrind --leak-check=full -s  ./generate.out ./database.txt ./database.bin
==4038== Memcheck, a memory error detector
==4038== Copyright (C) 2002-2022, and GNU GPL'd, by Julian Seward et al.
==4038== Using Valgrind-3.21.0 and LibVEX; rerun with -h for copyright info
==4038== Command: ./generate.out ./database.txt ./database.bin
==4038==
==4038==
==4038== HEAP SUMMARY:
==4038==     in use at exit: 0 bytes in 0 blocks
==4038==   total heap usage: 5 allocs, 5 frees, 9,264 bytes allocated
==4038==
==4038== All heap blocks were freed -- no leaks are possible
==4038==
==4038== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
tesla-2002@tesla2002-HP-ProBook-430-G7:~/labs/kp6$ valgrind --leak-check=full -s  ./kp6.out ./database.txt ./database.bin
==4064== Memcheck, a memory error detector
==4064== Copyright (C) 2002-2022, and GNU GPL'd, by Julian Seward et al.
==4064== Using Valgrind-3.21.0 and LibVEX; rerun with -h for copyright info
==4064== Command: ./kp6.out ./database.txt ./database.bin
==4064==
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
1
_____________________________________________________________________________________
|      Surname      |Luggage| Weight |       To       |     Date/time     |  Trans  |  Ch  |
_____________________________________________________________________________________
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
_____________________________________________________________________________________
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
2
```

```
|      Surname      |Luggage| Weight |       To       |     Date/time     | Trans  |  Ch  |
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
3
Surname:        Malishkin
Initials:       Y.E.
Amount of luggage:      3
Weight: 43
Destination:    Moscow
Date & time (day/month/year,hour:minutes):      13/06/2023,15:50
Amount of transfers:    2
Amount of children:     2
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
2
|      Surname      |Luggage| Weight |       To       |     Date/time     | Trans  |  Ch  |
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
Malishkin Y.E.          3       43      Moscow          13/6/2023,15:50         2       2
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
4
Surname:        Novikov
Initials:       A.A.
Amount of luggage:      2
Weight: 20
Destination:    Kiev
Date & time (day/month/year,hour:minutes):      13/06/2023,16:10
Amount of transfers:    0
Amount of children:     1
```

```
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
1
_____________________________________________________________________________________
|      Surname      |Luggage| Weight |       To       |     Date/time     |  Trans  |  Ch  |
_____________________________________________________________________________________
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
Novikov A.A.            2       20      Kiev            13/6/2023,16:10         0       1
_____________________________________________________________________________________
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
5
Введите фамилию пассажира, которого нужно удалить из списка:    Malishkin
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
2
_____________________________________________________________________________________
|      Surname      |Luggage| Weight |       To       |     Date/time     |  Trans  |  Ch  |
_____________________________________________________________________________________
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
_____________________________________________________________________________________
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
6
Введите фамилию пассажира, которого нужно удалить из списка:    Novikov
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
1
```

```
|      Surname      |Luggage| Weight |       To       |     Date/time     |  Trans  |  Ch  |
_____________________________________________________________________________________________
Ivanov V.S.             1       10      Moscow          15/6/2023,0:20          1       3
Petrov V.V.             1       13      Kiev            18/6/2023,14:15         2       2
Vladov F.G.             4       50      Lisbon          14/6/2023,8:45          2       5
Hetfield J.A.           3       40      Paris           13/6/2023,14:55         2       3
Sidorov K.D.            3       14      Antalya         16/6/2023,15:15         0       1
Malikov Y.E.            2       15      Moscow          13/6/2023,15:10         2       5
Gorkin M.B.             2       25      London          13/6/2023,14:44         1       0
Dolgov D.D.             2       35      Hanoi           13/6/2023,17:15         0       0
Teslya D.S.             3       55      Antalya         13/6/2023,14:50         0       0
Gorkin M.B.             5       56      Oslo            16/6/2023,15:45         0       1
_____________________________________________________________________________________________
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
7

Да, такой пассажир есть: Gorkin M.B.
Что Вы хотите сделать?
 [1]-Вывести содержимое бинарного файла
 [2]-Вывести содержимое текстового файла
 [3]-Добавить в текстовый файл
 [4]-Добавить в бинарный файл
 [5]-Удалить из текстового файла
 [6]-Удалить из бинарного файла
 [7]-Поиск пассажира по условию варианта
 [0]-Завершить
0
==4064==
==4064== HEAP SUMMARY:
==4064==     in use at exit: 0 bytes in 0 blocks
==4064==   total heap usage: 60 allocs, 60 frees, 65,784 bytes allocated
==4064==
==4064== All heap blocks were freed -- no leaks are possible
==4064==
```

4 Описание программы

4.1 Описание функций

passenger-read-txt	чтение одного блока данных из текстового файла
passenger-write-txt	запись одного блока данных в текстовый файл
passenger-read-bin	чтение одного блока данных из бинарного файла
passenger-write-bin	запись одного блока данных в бинарный файл
passenger-print-from-txt	печать данных из текстового файла
passenger-print-from-bin	печать данных из бинарного файла
passenger-get-from-user	чтение одного блока данных с stdin
passenger-add-to-bin	дозапись одного блока данных в бинарный файл
passenger-add-to-txt	дозапись одного блока данных в текстовый файл
passenger-delete-from-bin	удаление одного блока данных из бинарного файла
passenger-delete-from-txt	удаление одного блока данных из текстового файла
search-max	поиск пассажира с наибольшим весом и наибольшим количеством вещей
pass-free	удаление массива структур - базы данных
main	главная функция программы/подпрограммы

4.2 Описание ПО

ПК	HP ProBook 430 G7
ОС	Ubuntu 22.04.3 LTS
Интерпретатор команд	Bash
Редактор текстов	Sublime
Язык программирования	C
Местонахождение файла	/home/labs

5 Заключение

В ходе выполнения данной лабораторной работы я изучил и освоил на практике разработку последовательных структур данных для представления простейшей базы данных на файлах на языке Си, изучил основные методы работы с файловыми потоками в программах на языке Си. Я получил навыки программной генерации нетекстовых файлов, написав подпрограмму, генерирующую бинарный файл из текстового. Освоил основыне методы работы с текстовыми и бинарными файлами - считывание, запись, дозапись, удаление из файлов данных, переименование файлов, удаление файлов. Научился считывать блоки данных из файлов в массив структур блоков данных, и проводить операции с такими данными. Все полученные мной знания и навыки пригодятся в выполнении лабораторных и курсовых работ в будущем.