

Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра "Вычислительная техника"

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

по курсу "Программирование"

на тему "Разработка программы сложной структуры методом нисходящего
программирования."

Выполнил:

Студент группы 24BVB3

Виноградов Б.С.

15.05.2025

ВН.

Принял:

к.т.н., доцент Токарев А.Н.

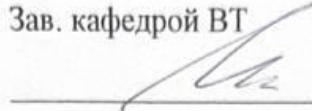
Пенза 2025

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ



М.А. Митрохин

«___» _____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Программирование

Студенту Виноградову Богдану Сергеевичу Группа 24ВВВЗ

Тема проекта: *Разработка программы сложной структуры методом нисходящего программирования.*

Исходные данные (технические требования) на проектирование

База данных "Автомобили":
Добавление, удаление, редактирование, поиск, сортировка записей;
Фильтрация записей по определенному критерию;
Создание новой базы данных;
Сохранение текущей базы данных;
Загрузка существующей базы данных.

Обязательные требования к программе:

1. Многомодульность.
2. Использование сложных типов данных – структур, списков, файлов.
3. Режим работы видеосистемы – текстовый / графический.
4. Устройство ввода информации – клавиатура, мышь.
5. Пользовательский интерфейс должен быть построен на основе меню и панели инструментов.
6. Наличие заставки.
7. Операционная система MS Windows.
8. Язык программирования – Си и Ассемблер.
9. Среда разработки ПО – Microsoft Visual Studio.

Объем работы по курсу

1. Расчетная часть

Разработка программы.

2. Графическая часть

Схема данных, схема ресурсов системы, схема работы системы,
иерархическая структура программы, схема взаимодействия программы.

3. Экспериментальная часть

Отладка программы.

Срок выполнения проекта по разделам

В соответствии с графиком.

Дата выдачи задания "13" 02. 2025

Дата защиты проекта "15" 05. 2025

Руководитель _____

Задание получил "13" февраля 2025 г.

Студент Виноградов Б.С. Вит

Содержание

Введение	5
Постановка задачи	6
Выбор решения	7
Описание разработки программы	9
Отладка и тестирование.....	10
Описание программы.....	11
Руководство пользователя	26
Заключение	40
Список используемых источников	41
Приложение А.....	Ошибка! Закладка не определена.
Приложение В.....	Ошибка! Закладка не определена.

Введение

В современном мире информация служит важнейшим ресурсом знаний, а её грамотная систематизация помогает повысить эффективность работы как отдельных людей, так и всего общества. Компьютерные технологии значительно облегчают нашу жизнь в самых разных сферах деятельности, а также позволяют корректно работать с информацией: анализировать, обрабатывать и хранить её на компактных носителях.

База данных — это организованный и структурированный набор данных, спроектированный таким образом, чтобы их можно было быстро и удобно обрабатывать с помощью вычислительной техники.

История баз данных в классическом понимании берёт начало в 1955 году, когда появилось программируемое оборудование для обработки записей. В то время программное обеспечение работало с файловыми системами, а данные хранились на перфокартах.

В середине 1960-х возникли оперативные сетевые базы данных, которые обрабатывались в интерактивном режиме через терминалы. Простые индексно-последовательные структуры данных быстро эволюционировали в более сложные модели, основанные на наборах записей.

Сегодня базы данных — неотъемлемый компонент любого предприятия или организации. Они используются в динамических веб-сайтах для хранения пользовательской информации, в стриминговых сервисах с подпиской, а также в банковской сфере, где обрабатываются огромные объёмы клиентских данных.

Постановка задачи

Необходимо разработать программу для работы с базой данных автомобилей. Программа должна иметь интуитивно понятный интерфейс для пользователя. Необходимо определить названия пунктов меню, построить алгоритмы работы с массивами структур для фильтрации, поиска, удаления и сортировки записей по определенным критериям, изучить функции для работы с файлами, чтобы сохранять базу данных в файл и загружать из файла.

Использование сложных типов данных – массивов, структур, файлов. Это необходимо для более простой и интуитивной обработки данных в коде программы.

Режим работы видеосистемы – текстовый/графический. Для начала необходимо определиться с типом интерфейса и с элементами управления, затем необходимо изучить способы их реализации.

Устройство ввода-вывода – клавиатура и мышь. Необходимо различать и идентифицировать действия, произведенные с их помощью, это облегчит использование программы.

Интерфейс должен быть построен на основе меню. Это необходимо для создания интуитивно понятного интерфейса для пользователя.

Заставка необходима для того, чтобы пользователь, запустивший программу, смог получить информацию о ней.

Выбор решения

При запуске программы открывается приветственный экран. Далее появляется приветственное меню, состоящее из трех пунктов:

1. создать новый файл. Пользователь вводит название файла, который хочет создать. Если файла не существует, пользователь может его создать;
2. открыть существующий файл. Пользователь вводит название файла, из которого необходимо считать данные. Если название файла не содержит расширения, автоматически добавляется расширение (.txt). Если название файла содержит запрещенные символы, программа выводит ошибку;
3. выход. Выход из программы.

После успешного создания файла или открытия созданного файла пользователь вводит данные, необходимые для нового элемента базы данных в порядке, указанном в программе. После заполнения базы данных, пользователь может завершить ввод по символу “END”.

Далее появляется меню редактирования, состоящее из шести пунктов:

1. удалить транспорт. Пользователь вводит название транспорта, которое необходимо удалить. Программа находит данную запись и заменяет ее на “ничего”;
2. поиск по названию. Пользователь вводит название транспорта, которое необходимо найти. Если элемент с данным названием существует, то программа выводит его, иначе выводится ошибка;
3. сортировать по названию. Происходит пузырьковая сортировка записей по названию в алфавитном порядке (A-Z);
4. фильтровать по параметрам. Пользователь выбирает один пункт из нового меню фильтрации:

– по типу транспорта. Фильтрация по критерию “Тип транспорта”. Пользователь вводит тип транспорта (0-5). Если соответствующая запись существует, команда выводит ее на экран в виде строки в базе данных. В противном случае, возникает ошибка и возврат к меню редактирования;

– по объему двигателя. Фильтрация по критерию “Объем двигателя”. Пользователь вводит нижний и верхний предел, по которому будет осуществлена фильтрация. Если соответствующая запись существует, команда выводит ее на экран в виде строки в базе данных. В противном случае, возникает ошибка и возврат к меню редактирования;

– по мощности. Фильтрация по критерию “Мощность”. Пользователь вводит нижний и верхний предел, по которому будет осуществлена фильтрация. Если соответствующая запись существует, команда выводит ее на экран в виде строки в базе данных. В противном случае, возникает ошибка и возврат к меню редактирования;

– сбросить фильтры и выйти. Возврат к меню редактирования.

5. добавить новый транспорт. Пользователь вводит данные, необходимые для нового элемента базы данных в порядке, указанном в программе. После заполнения базы данных, пользователь может завершить ввод по символу “END”;

6. сохранить и выйти. Программа сохраняет данные в файл, после чего пользователь может выйти в приветственное меню.

Навигация в приветственном меню осуществляется с помощью клавиатуры, вводом цифр от 1 до 3.

Навигация в меню редактирования осуществляется с помощью клавиатуры, вводом цифр от 1 до 6.

Навигация в меню фильтрации осуществляется с помощью клавиатуры, вводом цифр от 1 до 4.

Описание разработки программы

Разработанная программа состоит из нескольких модулей:

1. `main.cpp` – основной файл программы.
2. `functions.cpp` – файл реализации функции.
3. `functions.h`– заголовочный файл содержащий объявление функций.
4. `transport.h`– заголовочный файл с константами.

В разработанной программе использован нисходящий (top-down) подход к проектированию и реализации. Программа начинается с общего видения (управление списком транспорта), а затем дробится на подзадачи. Каждая функция решает одну конкретную задачу, что соответствует принципу декомпозиции.Верхнеуровневые функции вызывают более специализированные. Добавление новых функций не требует переписывания всей программы.Программа демонстрирует классический подход к работе со структурами в C, сочетая:

1. чистый код с модульной структурой;
2. удобный консольный интерфейс;
3. низкоуровневую оптимизацию (Assembly).

Отладка и тестирование

В качестве среды разработки была выбрана программа Microsoft Visual Studio 2019. Программа обладает всеми средствами необходимыми при разработке и отладке программы. Для отладки использовались несколько возможностей Visual Studio: точка останова, трассировка, анализ содержимого переменных. Тестирование проводилось во время разработки и также после завершения разработки. В ходе нее было выявлено огромное количество проблем, связанных с работой с файлами, работой с памятью, работой с полями структуры.

Описание программы

Приложение main.exe является основным модулем программы. При запуске программы происходит вывод заставки, после чего пользователю необходимо выбрать пункт меню для дальнейшего использования базы данных. Выход из меню осуществляется после выбора пользователем нужного пункта. Описание состояний программы выполнено в таблицах ниже.

Таблица 1 отображает стартовое меню.

Таблица 1. – Приветственное меню

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1. Enter	Выбран пункт “Создать новый файл”	Запускается диалог создания файла
2. Enter	Выбран пункт “Открыть существующий файл”	Запускается диалог открытия файла
3. Enter	Выбран пункт “Выход”	Завершение работы программы

Таблица 2 отображает меню редактирования.

Таблица 2. – Меню редактирования

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1. Enter	Выбран пункт “Удалить транспорт”	Запускается диалог удаления транспорта по названию
2. Enter	Выбран пункт “Поиск по названию”	Запускается диалог поиска транспорта по названию
3. Enter	Выбран пункт “Сортировать по названию”	Запускается диалог сортировки транспорта по названию в алфавитном порядке
4. Enter	Выбран пункт “Фильтровать по параметрам”	Запускается новое меню см. Таблица 3
5. Enter	Выбран пункт “Добавить новый транспорт”	Запускается диалог добавления нового транспорта
6. Enter	Выбран пункт “Сохранить и выйти”	Сохранение данных в файле. Запускается диалог завершения работы программы

Таблица 3 отображает меню фильтрации.

Таблица 3. – Меню фильтрации

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1. Enter	Выбран пункт “По типу транспорта”	Программа выводит записи с указанным типом транспорта
2. Enter	Выбран пункт “По объему двигателя”	Программа выводит записи в указанном диапазоне объема двигателя
3. Enter	Выбран пункт “По мощности”	Программа выводит записи в указанном диапазоне мощности
4. Enter	Выбран пункт “Сбросить фильтры и выйти”	Выход из меню фильтрации, переход в меню редактирования

На рисунке 1 показана схема функции main. Функция main является главной функцией программы, обеспечивающей ее работу.

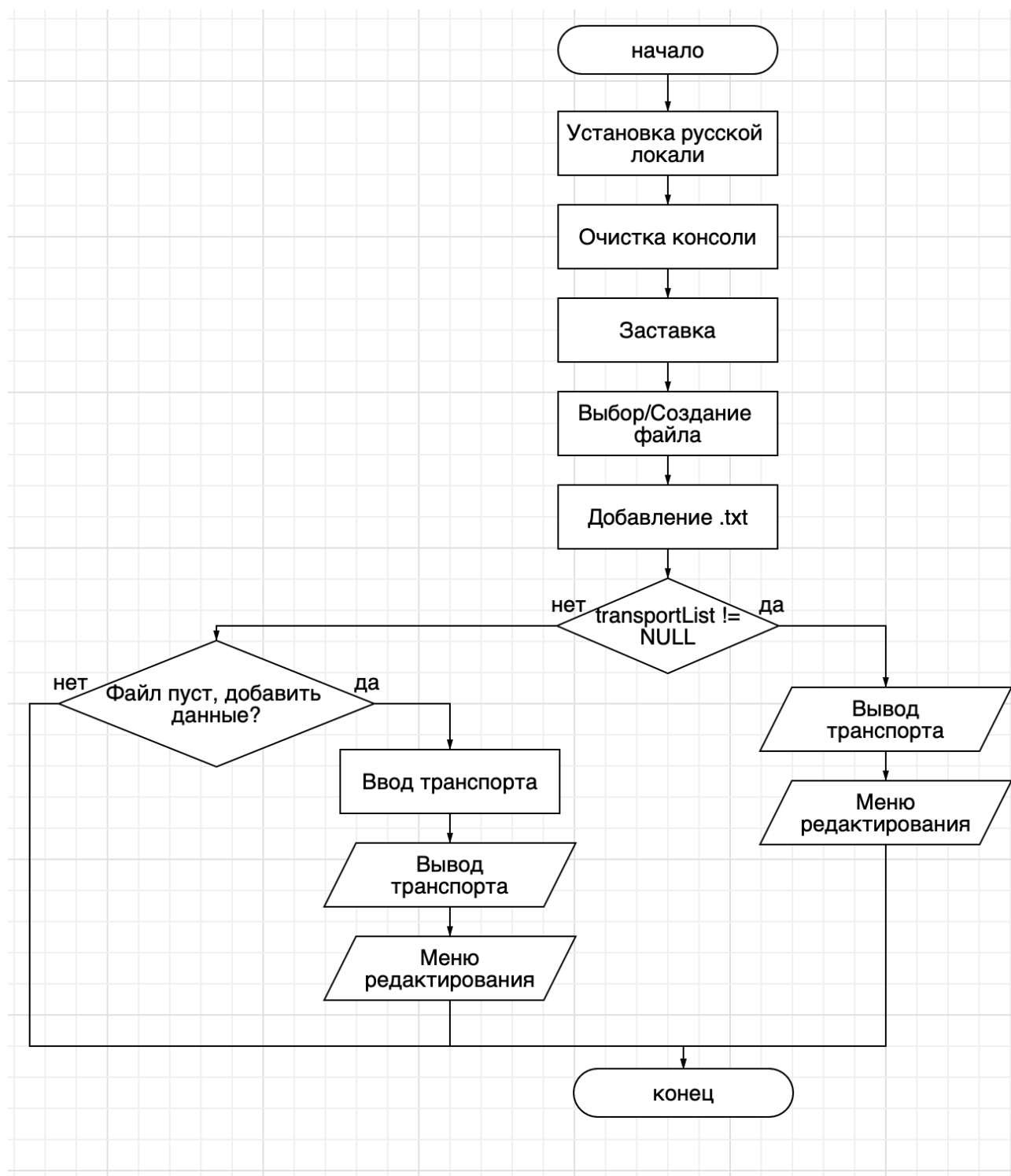


Рисунок 1. – Функция main

На рисунке 2 показана схема функции fileMenu. Для удобного чтения, схема разбита на 4 части (а, б, в, г). Функция fileMenu отражает работу приветственного меню, с помощью которого пользователь может создать новый файл, импортировать базу данных из файла или завершить работу программы.

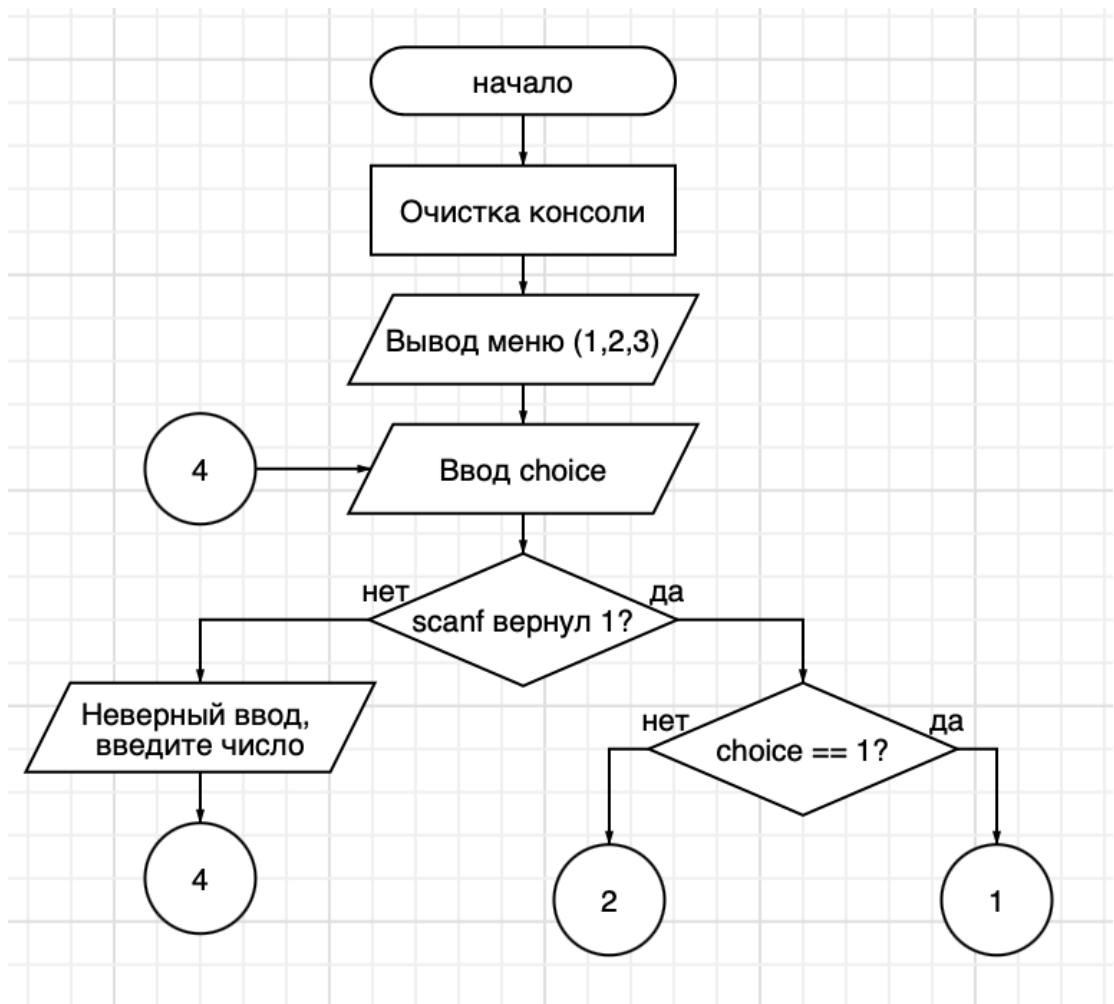


Рисунок 2(а). – Функция fileMenu

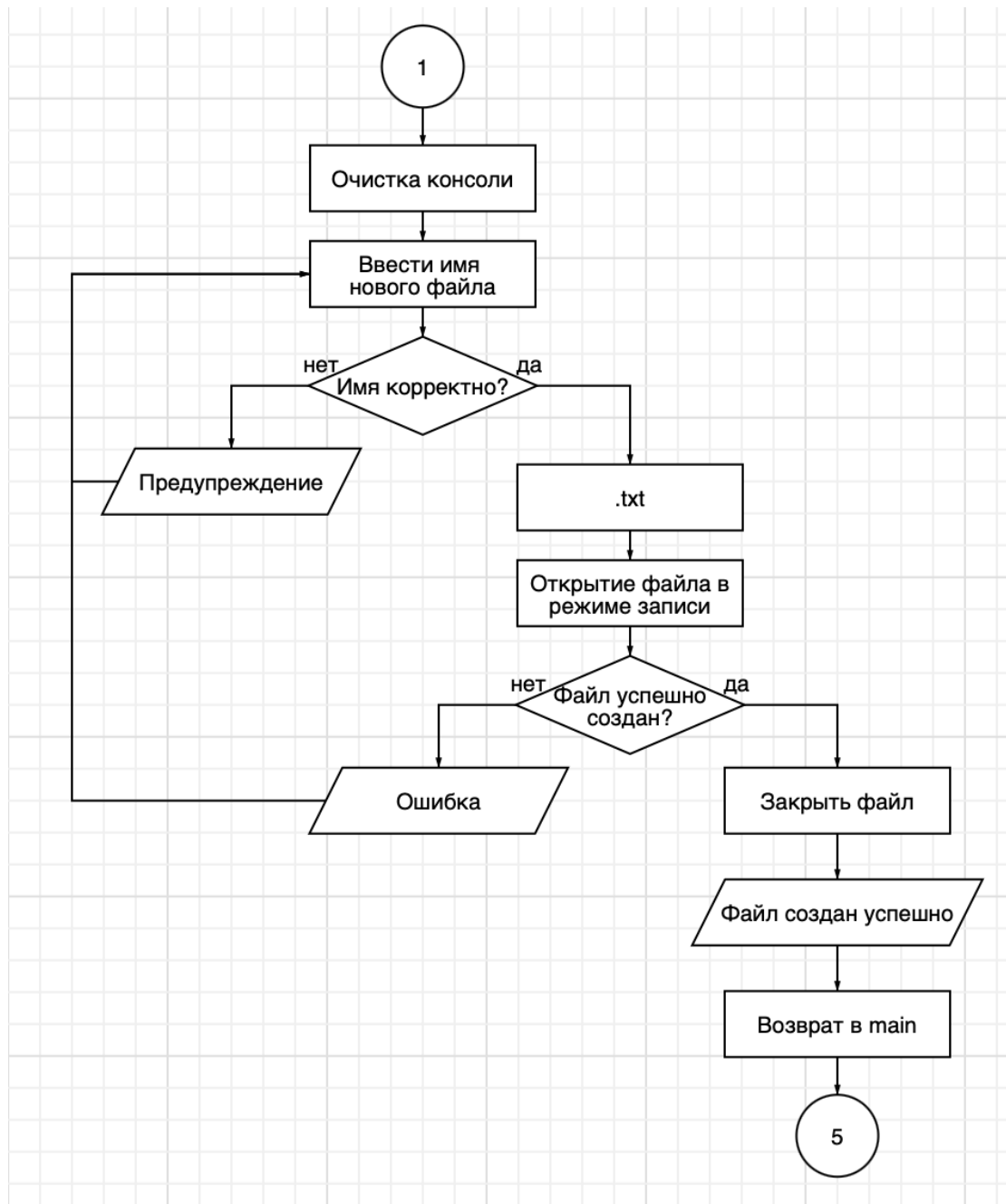


Рисунок 2(б). – Функция fileMenu

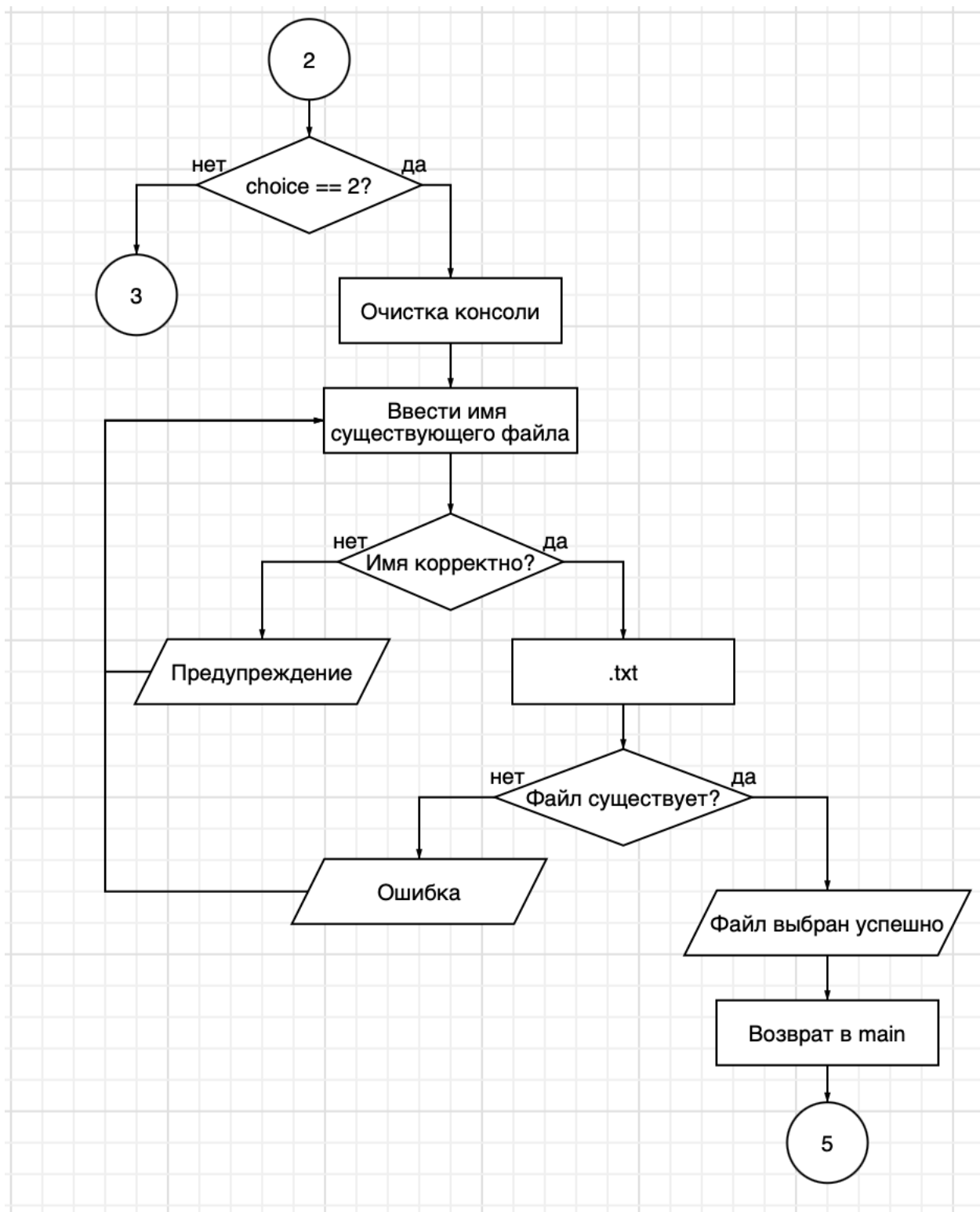


Рисунок 2(в). – Функция fileMenu

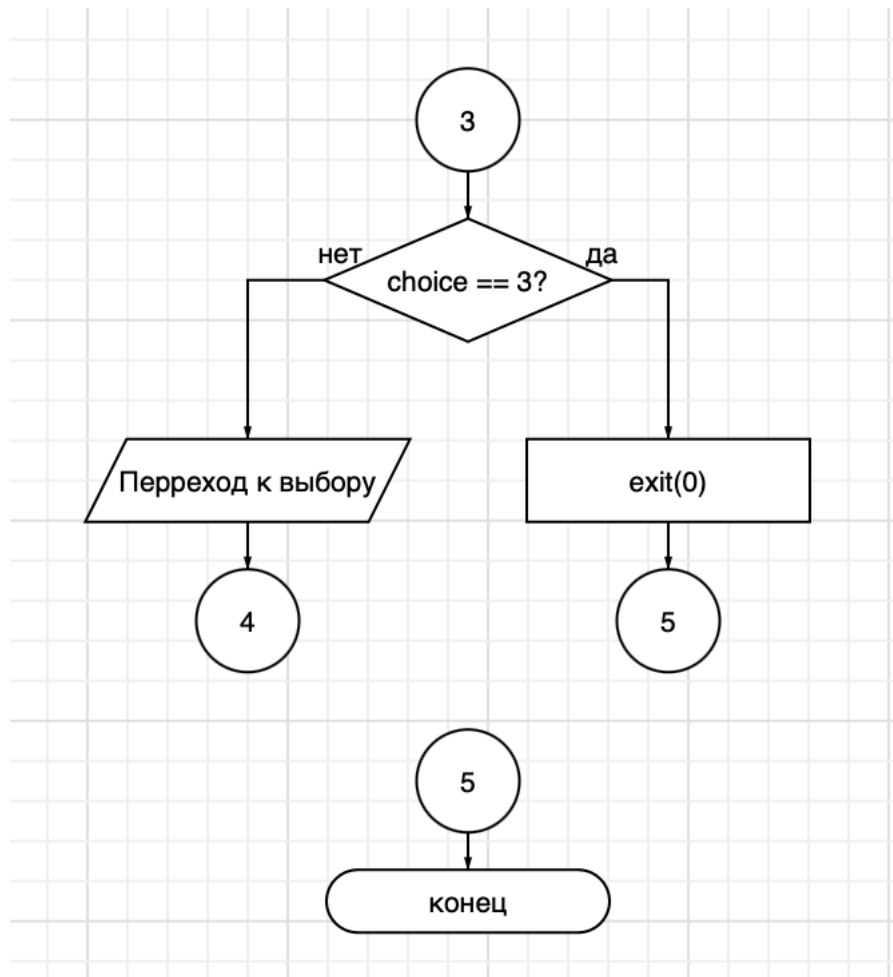


Рисунок 2(г). – Функция fileMenu

На рисунке 3 показана схема функции editMenu. Для удобного чтения, схема разбита на 3 части (а, б, в). Функция editMenu отражает работу меню редактирования, с помощью которого пользователь может редактировать базу данных, а также вернуться к приветственному меню.

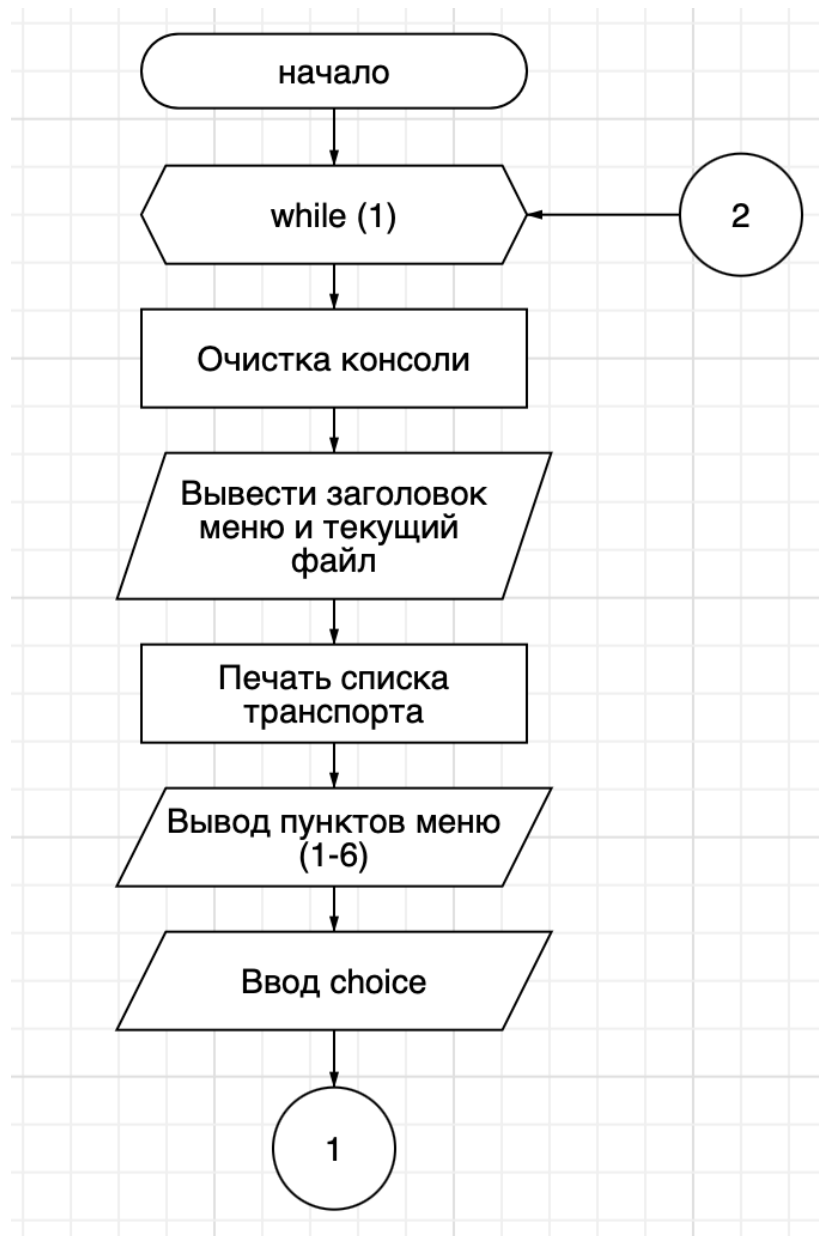


Рисунок 3(а). – Функция editMenu

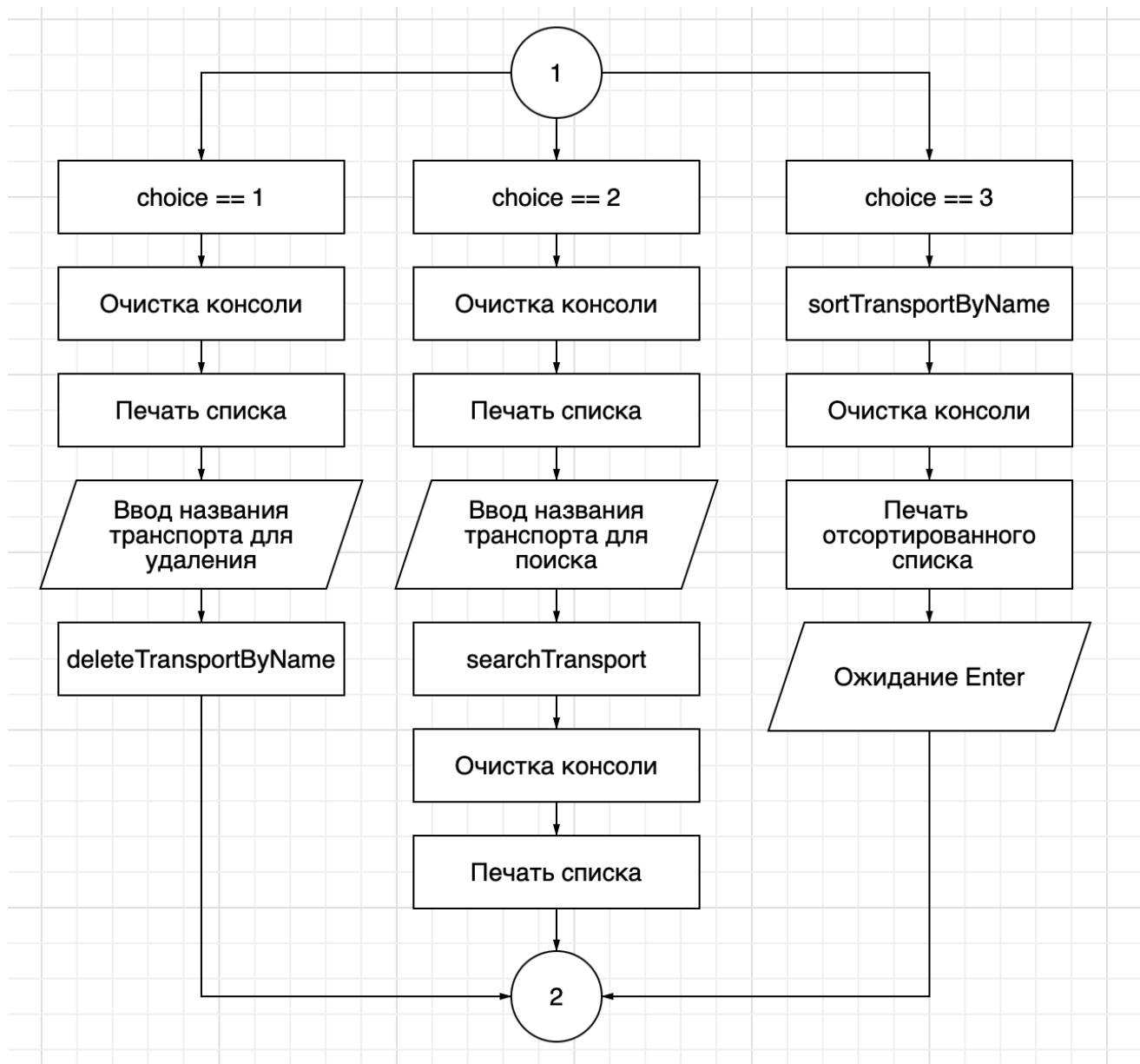


Рисунок 3(б). – Функция editMenu

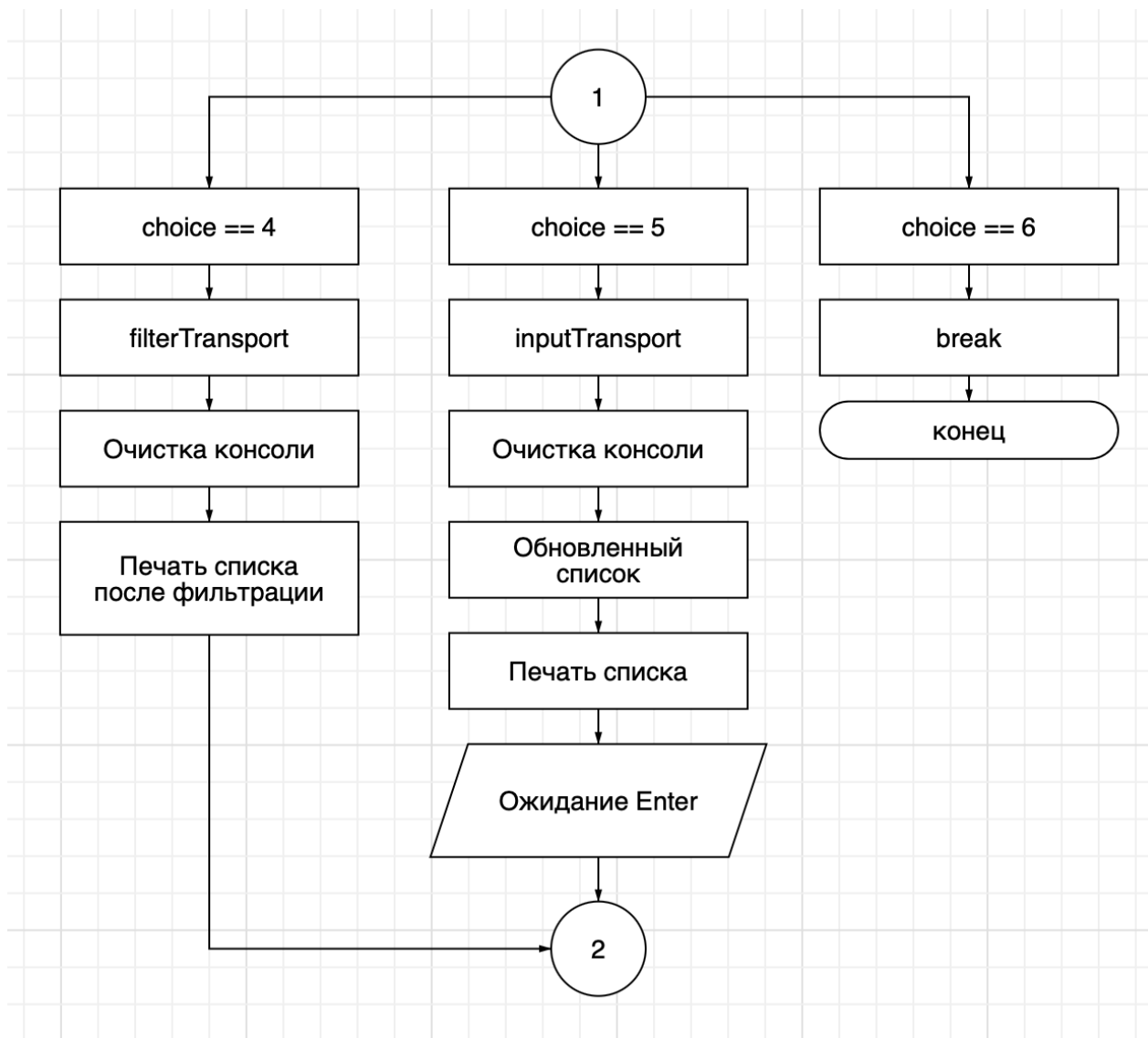


Рисунок 3(в). – Функция editMenu

На рисунке 4 показана схема функции inputTransport. Для удобного чтения, схема разбита на 2 части (а, б). С помощью функции inputTransport осуществляется добавление транспорта в базу данных. Пользователь завершает добавление новых элементов с помощью стоп-слова “END”.

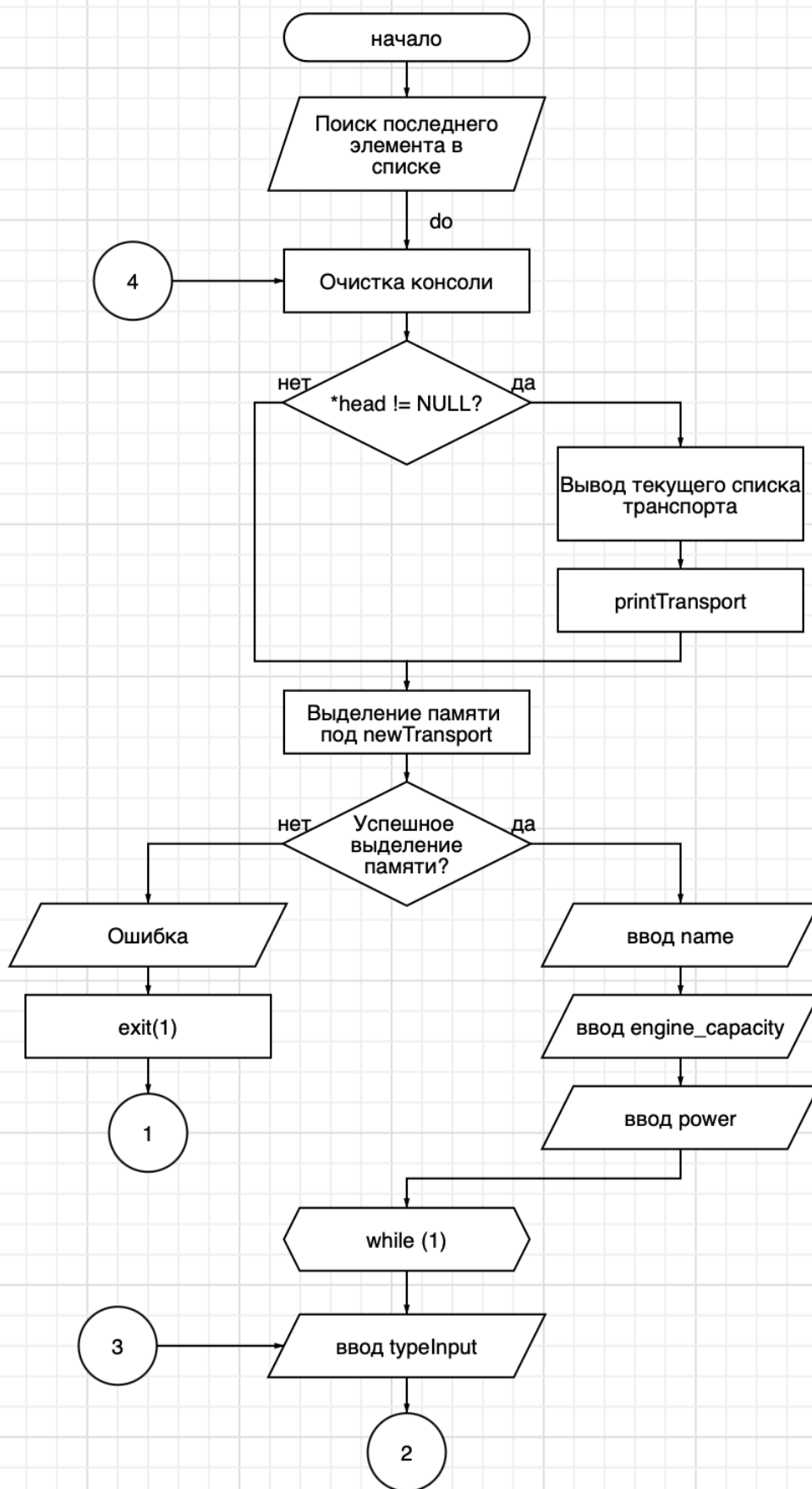


Рисунок 4(а). – Функция inputTransport

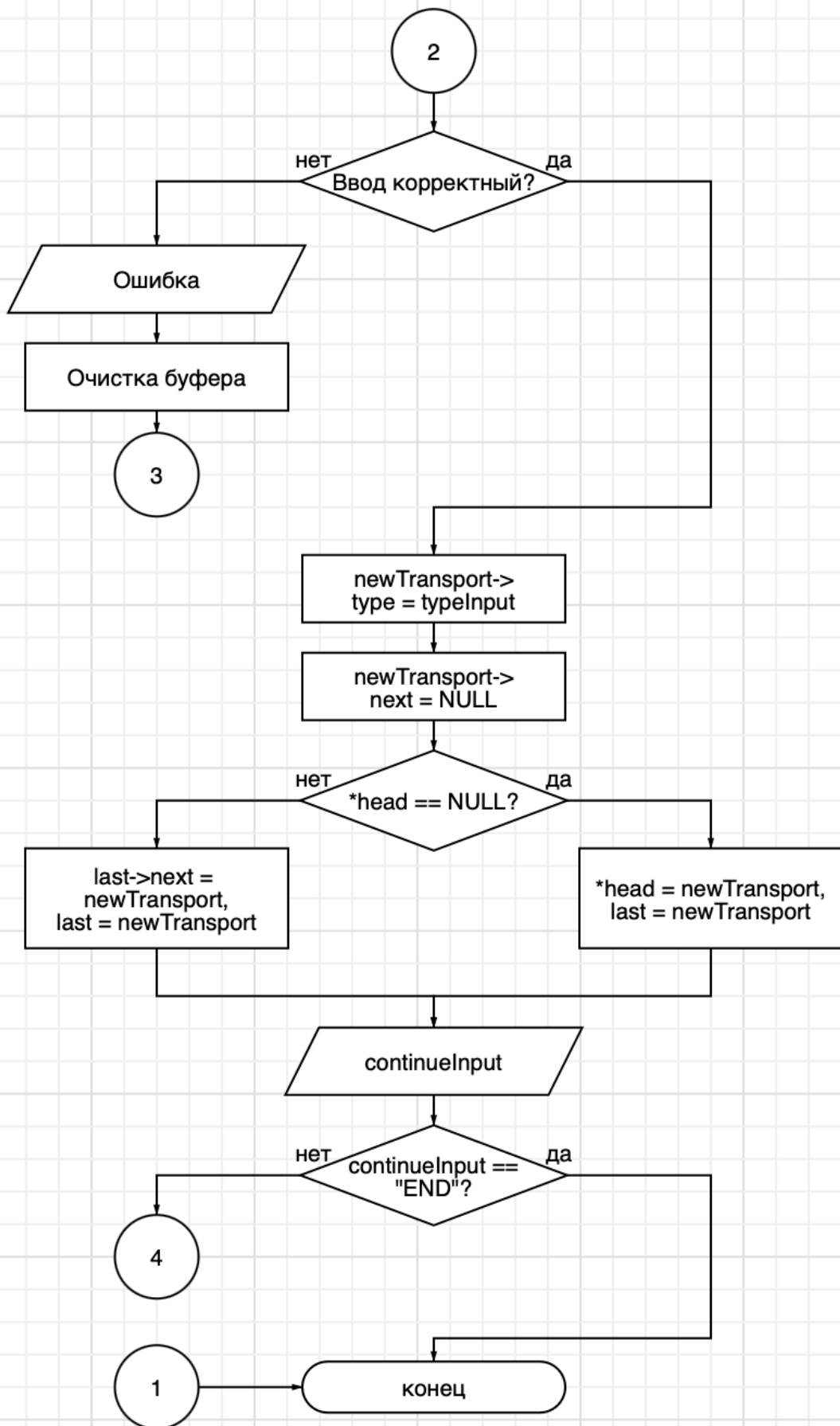


Рисунок 4(б). – Функция inputTransport

На рисунке 5 показана схема функции `sortTransportByName.C` с помощью функции `sortTransportByName` осуществляется пузырьковая сортировка транспорта по названию в алфавитном порядке.

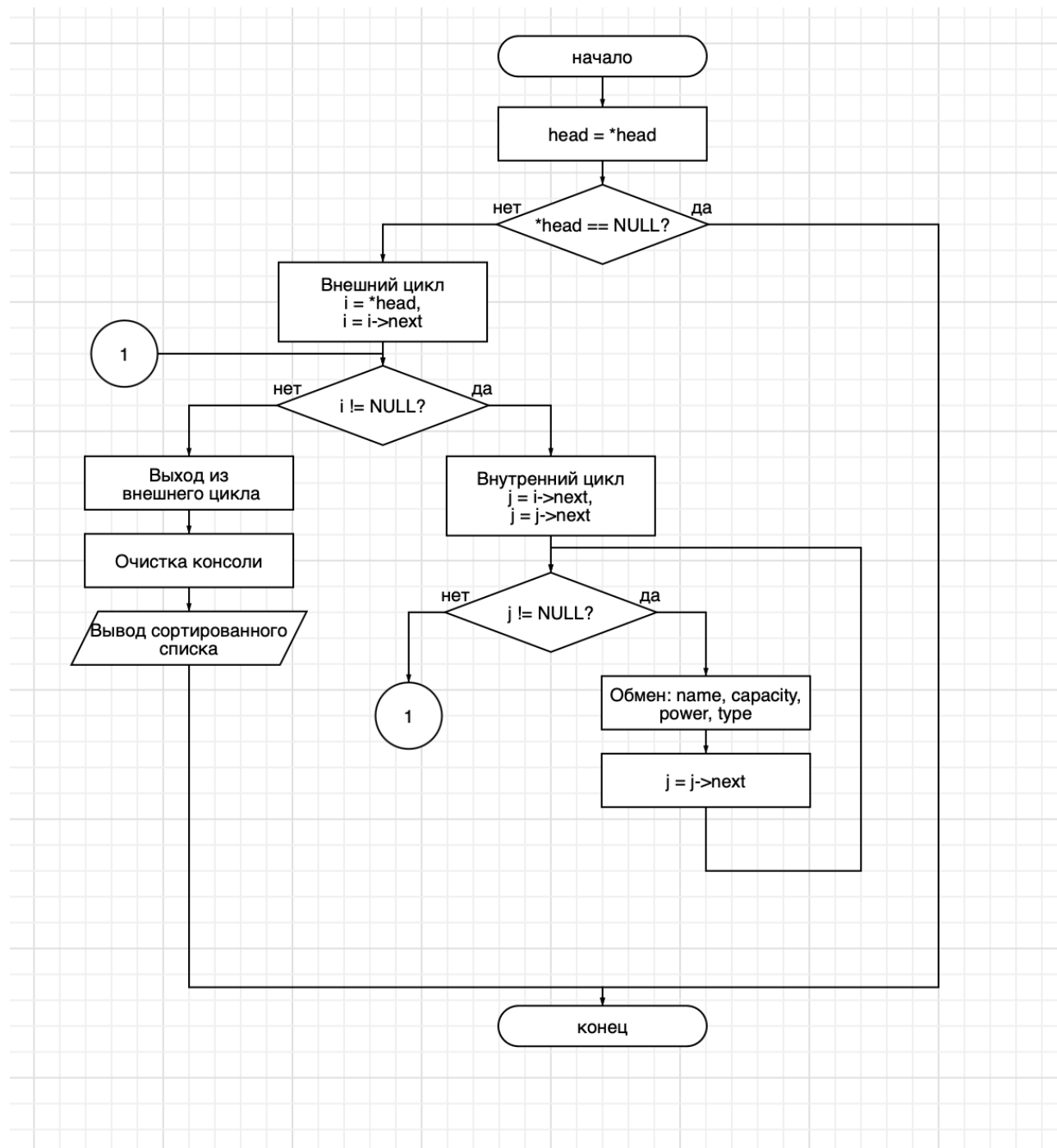


Рисунок 5. – Функция `sortTransportByName`

На рисунке 6 представлена схема данных.



Рисунок 6. – Схема данных

На рисунке 7 представлена схема взаимодействия программы.

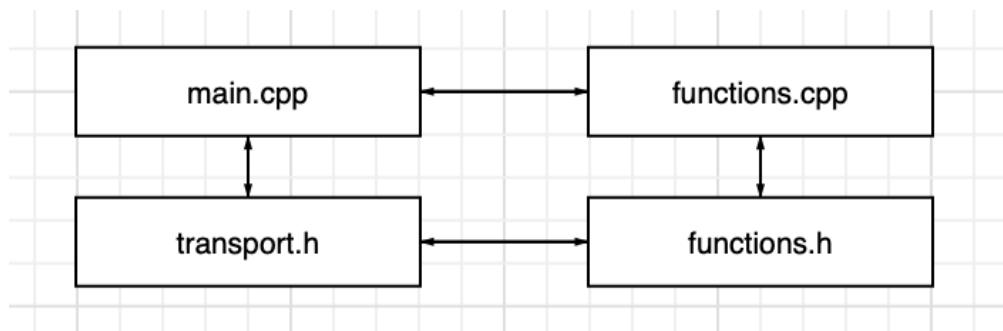


Рисунок 7. – Схема взаимодействия программы

На рисунке 8 представлена иерархическая схема программы.

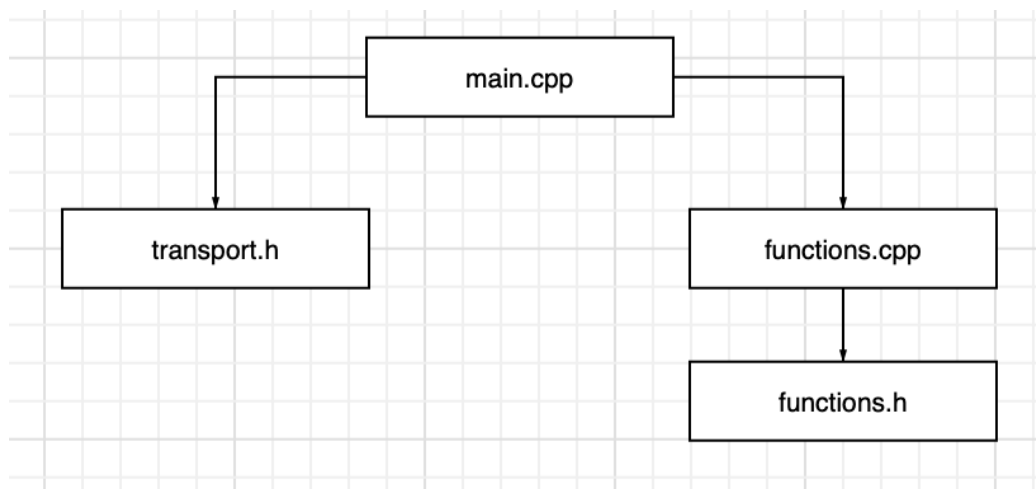


Рисунок 8. – Иерархическая схема программы

Руководство пользователя

Программа main.exe предназначена для хранения информации о видах транспортных средств и их характеристики. Программа имеет интуитивно понятный интерфейс и поддерживает такие операции как создание нового файла, импорт базы данных из файла, добавление новой записи в базу данных, удаление записи по названию, поиск записи по названию, фильтрация по определенным критериям, сортировка записей в алфавитном порядке по названию.

Для создания нового файла необходимо выбрать пункт приветственного меню “Создать новый файл”, ввести название нового файла, ввести “у”.

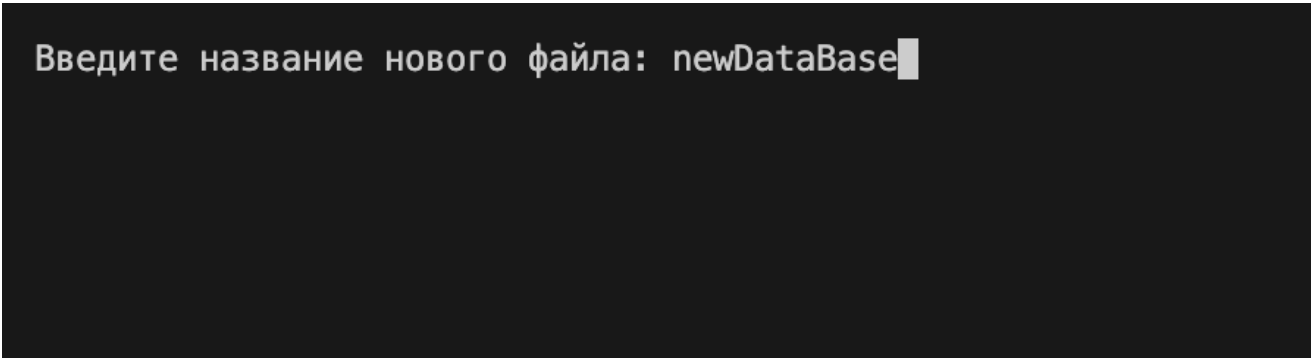
Введите “1” и нажмите “Enter”. Пример создания нового файла отображен на рисунке 9.



```
1. Создать новый файл
2. Открыть существующий файл
3. Выход
Ваш выбор: █
```

Рисунок 9. – Создание нового файла

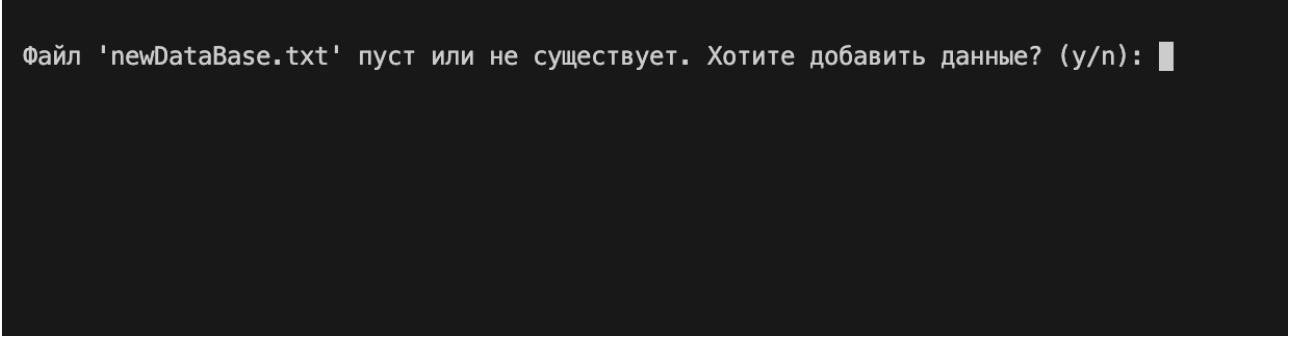
Введите название файла, который хотите создать. Расширение “.txt” будет создано автоматически, если вы его не укажете. Пример ввода данных для пункта 1 приветственного меню отображен на рисунке 10.



```
Введите название нового файла: newDataBase█
```

Рисунок 10. – Пример ввода данных для пункта 1 приветственного меню

Введите “у” или “н” в зависимости от ожидаемого результата. При выборе “у” вы должны ввести данные о транспорте, который хотите добавить. При выборе “н” вы сможете вернуться к стартовому меню, либо завершить работу программы по стоп-слову “END”. Пример диалога добавления данных при создании файла отображен на рисунке 11.



```
Файл 'newDataBase.txt' пуст или не существует. Хотите добавить данные? (y/n): █
```

Рисунок 11. – Диалог добавления данных при создании файла

Для импортирования базы данных из файла необходимо выбрать пункт приветственного меню “Открыть существующий файл”, ввести название существующего файла.

Введите “2” и нажмите “Enter”. Пример открытия существующего файла отображен на рисунке 12.



```
1. Создать новый файл
2. Открыть существующий файл
3. Выход
Ваш выбор: 2█
```

Рисунок 12. – Открытие существующего файла

Введите название существующего файла. Расширение “.txt” будет добавлено автоматически, если вы его не укажете. Пример ввода данных для пункта 2 приветственного меню отображен на рисунке 13.

```
Введите название существующего файла: newDataBase
```

Рисунок 13. – Пример ввода данных для пункта 2 приветственного меню

В случае успешного импортирования данных из файла программа выведет на экран вашу базу данных и меню редактирования. Пример успешного импортирования данных из файла отображен на рисунке 14.

№	Название	Объем двигателя	Мощность	Тип
1	Volvo	8.00	400	Грузовик
2	Suzuki	1.00	120	Мотоцикл
3	Lada	4.00	180	Авто
4	Tram	150.00	1200	Трамвай

1. Удалить транспорт
2. Поиск по названию
3. Сортировать по названию
4. Фильтровать по параметрам
5. Добавить новый транспорт
6. Сохранить и выйти

Ваш выбор:

Рисунок 14. – Пример успешного импортирования данных из файла

В случае ввода несуществующего файла программа выдаст ошибку и попросит вас ввести название существующего файла. Пример ввода несуществующего файла отображен на рисунке 15.

```
Введите название существующего файла: newnewDataBase
Файл 'newnewDataBase.txt' не существует. Попробуйте еще раз.
Введите название существующего файла: 
```

Рисунок 15. – Пример ввода несуществующего файла

Для завершения работы программы необходимо выбрать пункт приветственного меню “Выход”.

Введите “3” и нажмите “Enter”. Программа успешно завершит работу. Пример завершения работы программы отображен на рисунке 16.

```
1. Создать новый файл
2. Открыть существующий файл
3. Выход
Ваш выбор: 3
```

Рисунок 16. – Завершение работы программы

Для удаления записи из каталога необходимо выбрать пункт меню «Удалить транспорт». Далее необходимо ввести название удаляемой записи и нажать клавишу Enter. В случае если не возникли ошибки, удаление пройдет успешно. Если транспорта с данным названием нет, программа выдаст ошибку “Транспорт не найден”.

Введите “1” и нажмите “Enter”. Пример открытия пункта меню редактирования “Удалить транспорт” отображен на рисунке 17.

№	Название	Объем двигателя	Мощность	Тип
1	Volvo	8.00	400	Грузовик
2	Suzuki	1.00	120	Мотоцикл
3	Lada	4.00	180	Авто
4	Tram	150.00	1200	Трамвай

```
1. Удалить транспорт
2. Поиск по названию
3. Сортировать по названию
4. Фильтровать по параметрам
5. Добавить новый транспорт
6. Сохранить и выйти
Ваш выбор: 1
```

Рисунок 17. – Открытие пункта меню редактирования “Удалить транспорт”

Попытка ввода существующего названия. Пример ввода существующего названия для удаления транспорта отображен на рисунке 18.

№	Название	Объем двигателя	Мощность	Тип
1	Volvo	8.00	500	Грузовик
2	Suzuki	1.00	120	Мотоцикл
3	Lada	4.00	180	Авто
4	Tram	150.00	1200	Трамвай

Введите название для удаления: Volvo

Рисунок 18. – Пример ввода существующего названия

(удаление)

В случае правильного ввода программа удаляет все найденные строки с этим названием. При нажатии “Enter” вы вернетесь к меню редактирования. Пример вывода результата после успешного удаления отображен на рисунке 19.

Удалено 1 транспортных средств с названием 'Volvo'.
Обновленный список транспорта:

№	Название	Объем двигателя	Мощность	Тип
1	Suzuki	1.00	120	Мотоцикл
2	Lada	4.00	180	Авто
3	Tram	150.00	1200	Трамвай

Нажмите Enter чтобы продолжить...

Рисунок 19. – Результат после верного ввода (удаление)

Попытка ввода несуществующего названия. Пример ввода несуществующего названия для удаления транспорта отображен на рисунке 20.

№	Название	Объем двигателя	Мощность	Тип
1	Suzuki	1.00	120	Мотоцикл
2	Lada	4.00	180	Авто
3	Tram	150.00	1200	Трамвай

Введите название для удаления: Mazda

Рисунок 20. – Пример ввода несуществующего названия (удаление)

В случае ввода названия несуществующего транспорта программа выдаст ошибку, в результате ничего не будет удалено. При нажатии “Enter” вы вернетесь к меню редактирования. Пример вывода результата после неверного ввода для удаления отображен на рисунке 21.

Транспорт с названием 'Mazda' не найден.
Обновленный список транспорта:

№	Название	Объем двигателя	Мощность	Тип
1	Suzuki	1.00	120	Мотоцикл
2	Lada	4.00	180	Авто
3	Tram	150.00	1200	Трамвай

Нажмите Enter чтобы продолжить...

Рисунок 21. – Результат после неверного ввода (удаление)

Для поиска записи по названию необходимо выбрать пункт меню «Поиск по названию». Далее необходимо ввести название транспорта, данные о котором надо найти и нажать “Enter”. В случае отсутствия ошибок, вывод данных о транспорте пройдет успешно. Если транспорта с данным названием нет, программа выдаст ошибку “Транспорт не найден”.

Введите “2” и нажмите “Enter”. Пример открытия пункта меню редактирования “Поиск по названию” отображен на рисунке 22.

Меню редактирования (файл: newDataBase.txt):

№	Название	Объем двигателя	Мощность	Тип
1	Suzuki	1.00	120	Мотоцикл
2	Lada	4.00	180	Авто
3	Tram	150.00	1200	Трамвай

1. Удалить транспорт
2. Поиск по названию
3. Сортировать по названию
4. Фильтровать по параметрам
5. Добавить новый транспорт
6. Сохранить и выйти

Ваш выбор: 2

Рисунок 22. – Открытие пункта меню редактирования “Поиск по названию”

Попытка ввода существующего названия. Пример ввода существующего названия для поиска транспорта отображен на рисунке 23.

№	Название	Объем двигателя	Мощность	Тип
1	Suzuki	1.00	120	Мотоцикл
2	Lada	4.00	180	Авто
3	Tram	150.00	1200	Трамвай

Введите название для поиска: Lada

Рисунок 23. – Пример ввода существующего названия (поиск)

В случае правильного ввода программа ищет и выводит все найденные строки с этим названием. При нажатии “Enter” вы вернетесь к меню редактирования. Пример вывода результата после успешного поиска отображен на рисунке 24.

Результаты поиска по запросу 'Lada':

№	Название	Объем двигателя	Мощность	Тип
1	Lada	4.00	180	Авто

Нажмите Enter чтобы продолжить...

Рисунок 24. – Результат после верного ввода (поиск)

Попытка ввода несуществующего названия. Пример ввода несуществующего названия для поиска транспорта отображен на рисунке 25.

№	Название	Объем двигателя	Мощность	Тип
1	Lada	4.00	180	Авто
2	Suzuki	1.00	120	Мотоцикл
3	Tram	150.00	1200	Трамвай

Введите название для поиска: Mazda

Рисунок 25. – Пример ввода несуществующего названия (поиск)

В случае ввода названия несуществующего транспорта программа выдаст ошибку, в результате ничего не будет найдено. При нажатии “Enter” вы вернетесь к меню редактирования. Пример вывода результата после неверного ввода для поиска отображен на рисунке 26.

Результаты поиска по запросу 'Mazda':

№	Название	Объем двигателя	Мощность	Тип
---	----------	-----------------	----------	-----

Транспорт с названием, содержащим 'Mazda', не найден.

Нажмите Enter чтобы продолжить...

Рисунок 26. – Результат после неверного ввода (поиск)

Для сортировки записей необходимо выбрать пункт меню «Сортировать по названию». После сортировки, мы получаем отсортированную базу данных по названию в алфавитном порядке.

Введите “3” и нажмите “Enter”. Пример открытия пункта меню редактирования “Сортировать по названию” отображен на рисунке 27.

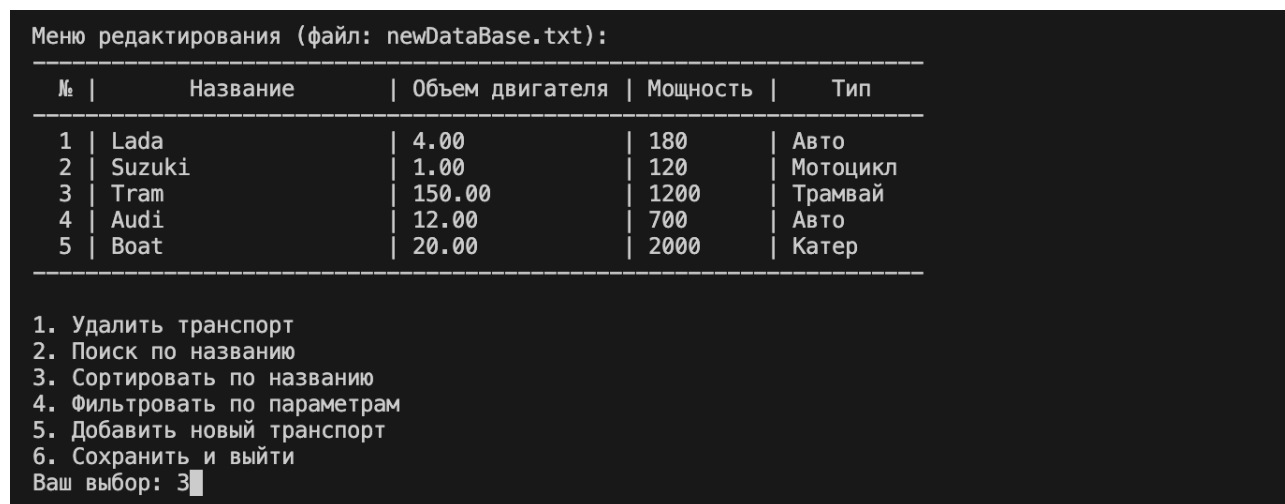


Рисунок 27. – Открытие пункта меню редактирования “Сортировать по названию”

После нажатия “Enter” программа выведет отсортированную базу данных по алфавиту. Пример результата сортировки отображен на рисунке 28.

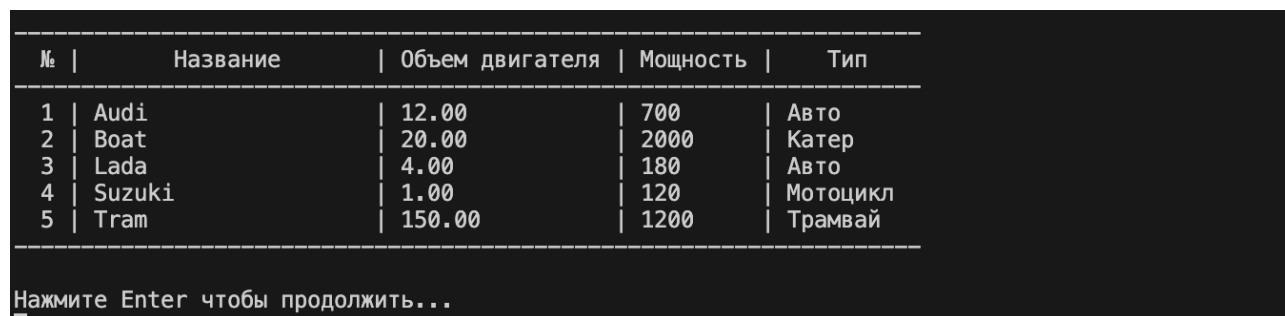


Рисунок 28. – Результат сортировки

Для фильтрации записи по параметрам необходимо выбрать пункт меню «Фильтровать по параметрам». Программа открывает меню фильтрации. При выборе параметра фильтрации «По типу транспорта», ему предстоит выбрать тип транспорта (0-5). При выборе параметров фильтрации «По объему двигателя» и «По мощности», пользователь должен ввести нижнюю и верхнюю границу фильтра. Также фильтры могут накладываться друг на друга. Если нет транспорта, удовлетворяющего данному фильтру, программа выдаст ошибку и вернется в меню редактирования. Если данные фильтрации введены некорректно, программа потребует корректный ввод данных.

Введите «4» и нажмите «Enter». Пример открытия пункта меню редактирования «Фильтровать по параметрам» отображен на рисунке 29.

Меню редактирования (файл: newDataBase.txt):

№	Название	Объем двигателя	Мощность	Тип
1	Audi	12.00	700	Авто
2	Boat	20.00	2000	Катер
3	Lada	4.00	180	Авто
4	Suzuki	1.00	120	Мотоцикл
5	Tram	150.00	1200	Трамвай

1. Удалить транспорт
2. Поиск по названию
3. Сортировать по названию
4. Фильтровать по параметрам
5. Добавить новый транспорт
6. Сохранить и выйти

Ваш выбор: 4

Рисунок 29. – Открытие пункта меню редактирования «Фильтровать по параметрам»

После нажатия «Enter» программа выведет текущую базу данных и меню фильтрации. Пример открытия меню фильтрации отображен на рисунке 30.

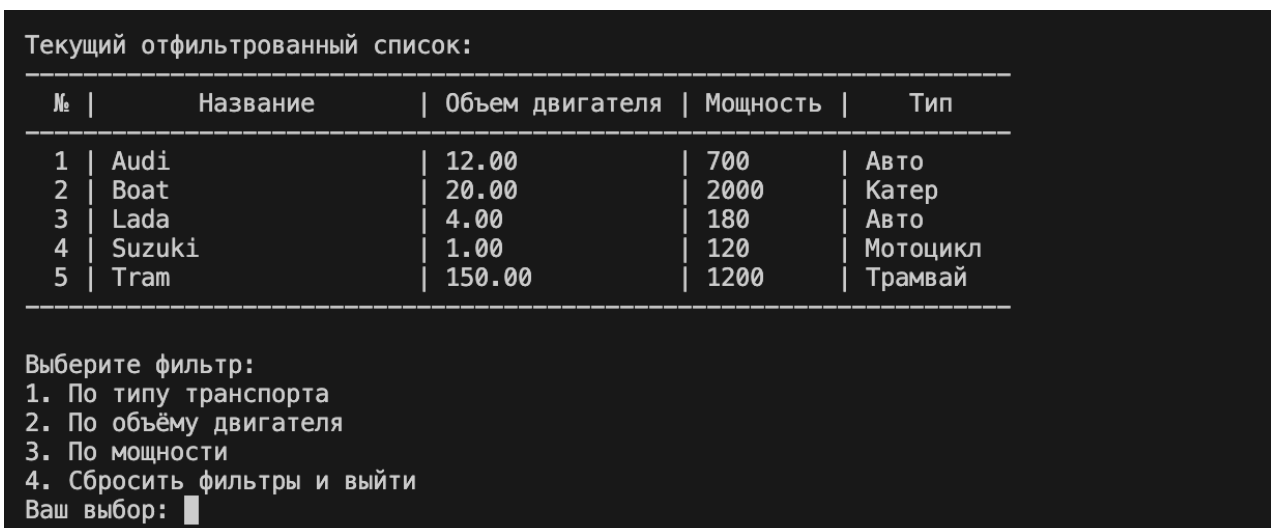


Рисунок 30. – Меню фильтрации

Попытка фильтрации по типу транспорта. Пример ввода параметра фильтрации “По типу транспорта” отображен на рисунке 31.

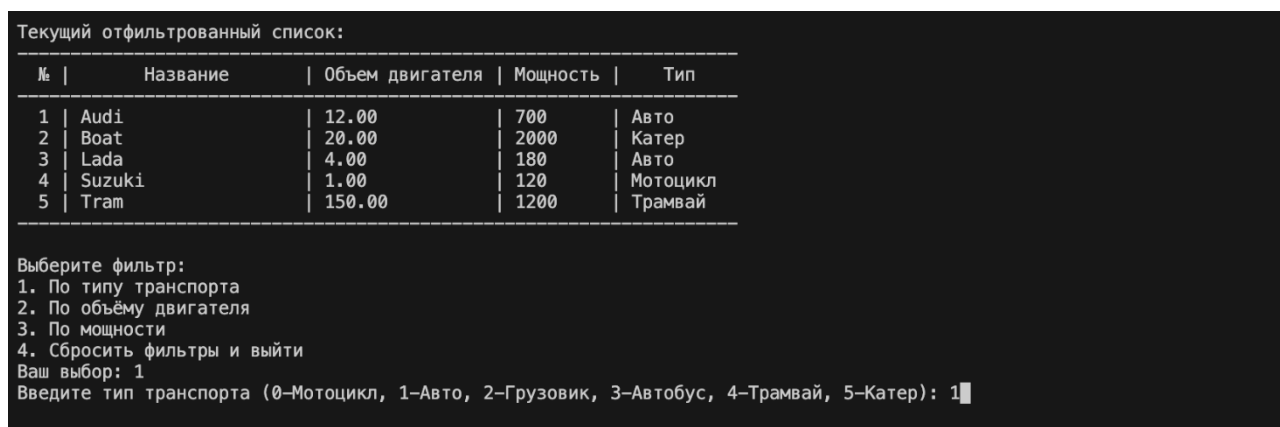


Рисунок 31. – Пример ввода параметров фильтрации (по типу транспорта)

При успешном вводе программа выводит на экран текущий отфильтрованный список и список фильтров. Попытка фильтрации по типу транспорта. Пример результата после применения параметра фильтрации “По типу транспорта” отображен на рисунке 32.

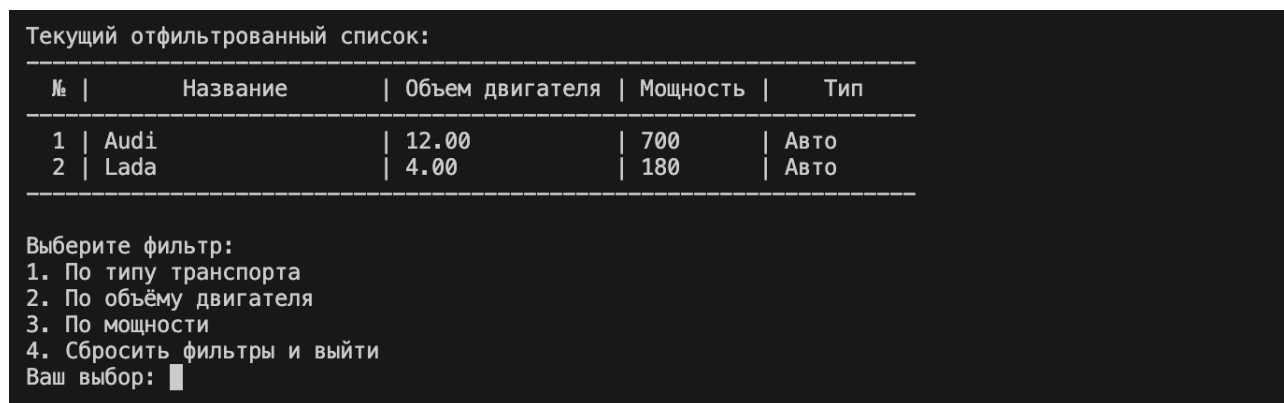


Рисунок 32. – Результат фильтрации (по типу транспорта)

Попытка фильтрации по объему двигателя (аналогично происходит фильтрация по мощности). Пример ввода параметра фильтрации “По объему двигателя” отображен на рисунке 33.

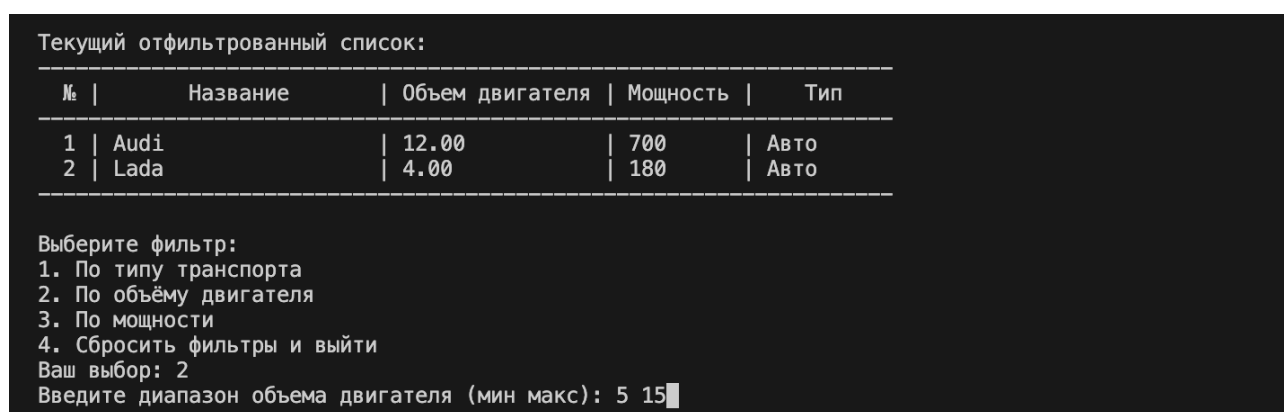


Рисунок 33. – Пример ввода параметров фильтрации (по объему двигателя)

При успешном вводе программа выводит на экран текущий отфильтрованный список и список фильтров. Фильтры в данном случае были наложены друг на друга. Пример результата после применения параметра фильтрации “По объему двигателя” отображен на рисунке 34.

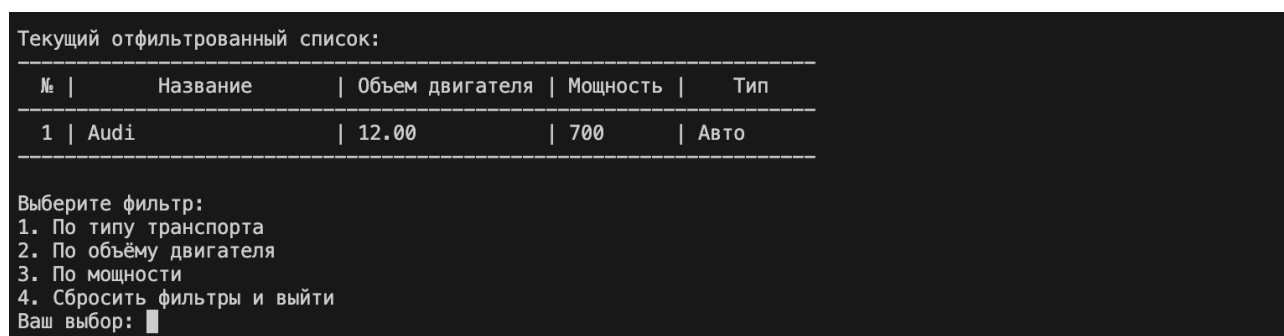


Рисунок 34. – Результат фильтрации (по объему двигателя)

Для сброса фильтров и выхода в меню редактирование пользователю необходимо ввести “4”. Пример выбора пункта “Сбросить фильтры и выйти” отображен на рисунке 35.

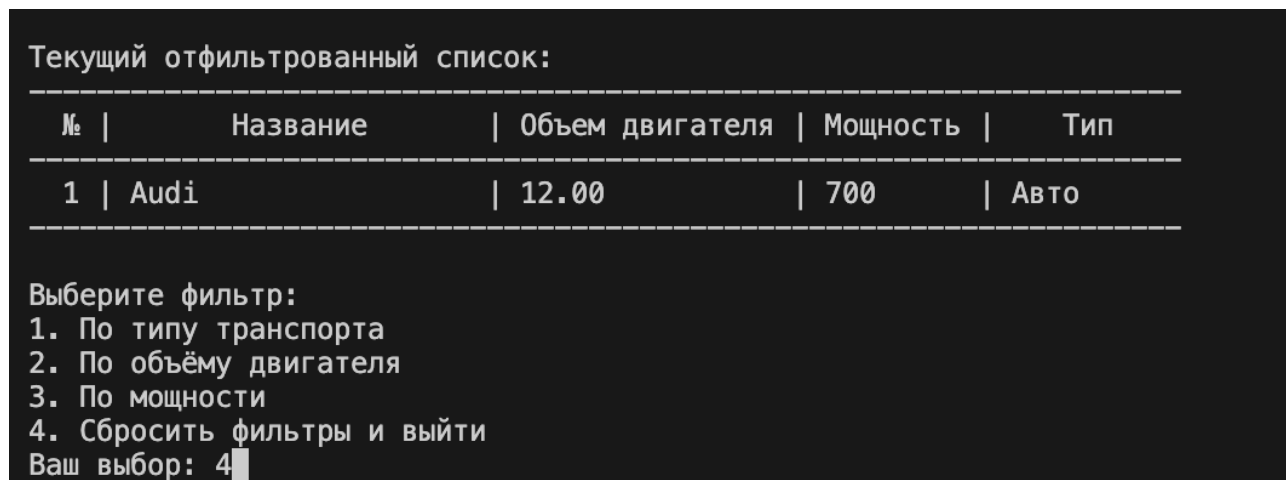


Рисунок 35. –Выбор пункта “Сбросить фильтры и выйти”

Нажмите “Enter”. Пример сброса фильтров и выхода в меню редактирования отображен на рисунке 36.

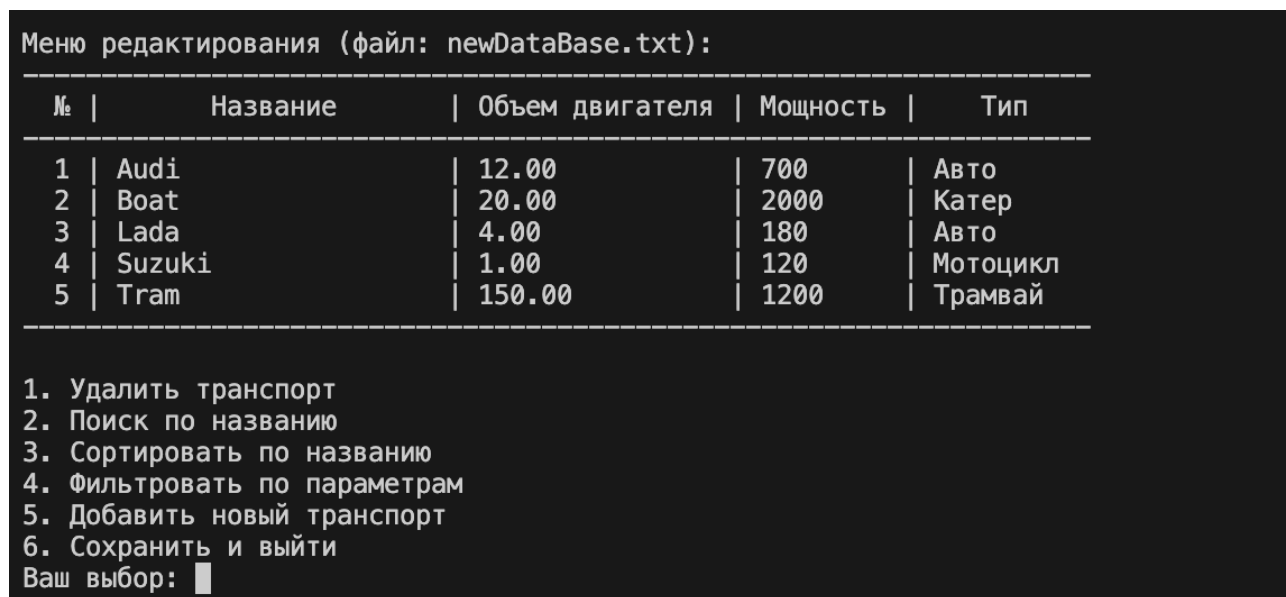


Рисунок 36. – Сброс фильтров и выход в меню редактирования

Для создания новой записи необходимо выбрать пункт меню «Добавить новый транспорт». После этого пользователь вводит данные для транспорта. В случае, если не возникли ошибки, добавление пройдет успешно, и программа выведет обновленный список транспорта. В ином случае, программа выведет ошибку и потребует корректный ввод. Завершение ввода осуществляется по специальному стоп-символу “END”.

Введите “5” и нажмите “Enter”. Пример открытия пункта меню редактирования “Добавить новый транспорт” отображен на рисунке 37.

Меню редактирования (файл: newDataBase.txt):

№	Название	Объем двигателя	Мощность	Тип
1	Audi	12.00	700	Авто
2	Boat	20.00	2000	Катер
3	Lada	4.00	180	Авто
4	Suzuki	1.00	120	Мотоцикл
5	Tram	150.00	1200	Трамвай

1. Удалить транспорт
2. Поиск по названию
3. Сортировать по названию
4. Фильтровать по параметрам
5. Добавить новый транспорт
6. Сохранить и выйти
Ваш выбор: 5

Рисунок 37. – Открытие пункта меню редактирования “Добавить новый транспорт”

Введите требуемые данные и нажмите “Enter”. Пример ввода данных для добавления транспорта отображен на рисунке 38.

Текущий список транспорта:

№	Название	Объем двигателя	Мощность	Тип
1	Audi	12.00	700	Авто
2	Boat	20.00	2000	Катер
3	Lada	4.00	180	Авто
4	Suzuki	1.00	120	Мотоцикл
5	Tram	150.00	1200	Трамвай

Введите название транспорта: Maserati
Объем двигателя (л): 4.7
Мощность (л.с.): 400
Тип транспорта (0–Мотоцикл, 1–Авто, 2–Грузовик, 3–Автобус, 4–Трамвай, 5–Катер): 1
Хотите добавить новый транспорт? (Введите 'END' для завершения):

Рисунок 38. – Ввод данных для добавления транспорта

Для сохранения данных и выхода из меню редактирования необходимо выбрать пункт меню “Сохранить и выйти”. Завершение работы программы осуществляется по специальному стоп-символу “END”, ввод любого другого символа возвращается нас к приветственному меню.

Введите “6” и нажмите “Enter”. Пример открытия пункта меню редактирования “Сохранить и выйти” отображен на рисунке 39.

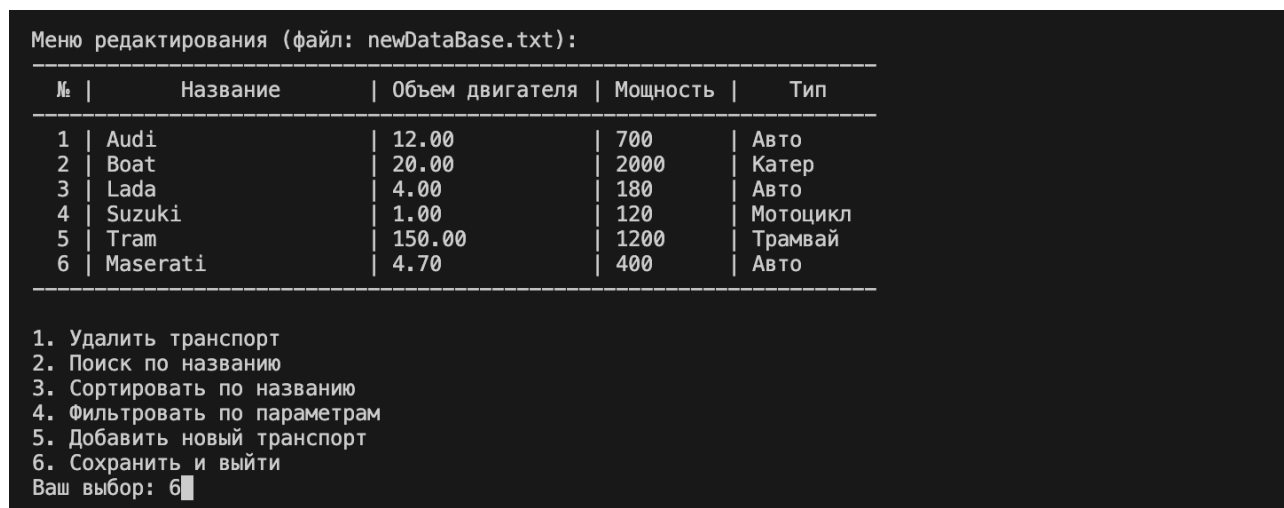


Рисунок 39. – Открытие пункта меню редактирования “Сохранить и выйти”

Завершение работы программы. Введите “END”. Пример завершения работы программы отображен на рисунке 40.

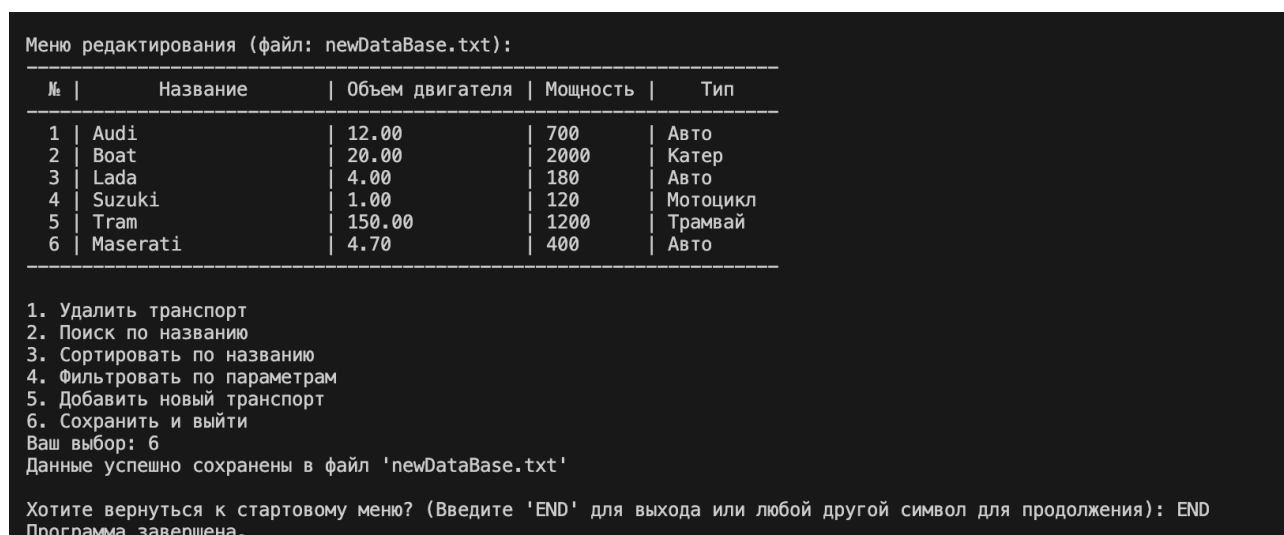


Рисунок 40. – Завершение работы программы

Заключение

При выполнении данной курсовой работы были получены навыки разработки многомодульных программ. Была освоена работа с файлами, изучены функции работы с консолью и оперативной памятью. Также были получены основные навыки отладки и тестирования программ и программирования в среде VisualStudio 2019 на языках Си/C++ и Assembler.

В рамках курсовой работы была написана программа для работы с базой данных автомобилей. Она имеет все необходимые функции для работы с базами данных.

В дальнейшем программу можно улучшить, путем добавления графического интерфейса, что позволит улучшить навигацию по меню программы.

Список используемых источников

1. Kernighan, B. W. The C Programming Language / B. W. Kernighan, D. M. Ritchie. – 2nd ed. – Upper Saddle River, NJ: Prentice Hall, 1988. – 274 p. – ISBN 0-13-110362-8.
2. Irvine, K. R. Assembly Language for Intel-Based Computers / K. R. Irvine. – 5th ed. – Upper Saddle River, NJ: Pearson Prentice Hall, 2007. – xxv, 722 p.: ill.; 25 cm. – ISBN 978-0-13-238310-3.
3. Prata, S. C++ Primer Plus / S. Prata. – 6th ed. – Upper Saddle River, NJ: Addison-Wesley, 2012. – 976 p. – ISBN 978-0-321-77640-2.
4. Тюгашев, А. А. Языки программирования: учеб. пособие / А. А. Тюгашев. – Санкт-Петербург: Изд-во Политехн. ун-та, 2018. – 235 с. – ISBN 978-5-7422-6827-4.
5. Аблязов, Р. М. Программирование на ассемблере на платформе x86-64 / Р. М. Аблязов. – Москва: ДМК Пресс, 2016. – 302 с. – ISBN 978-5-97060-364-2.