

Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

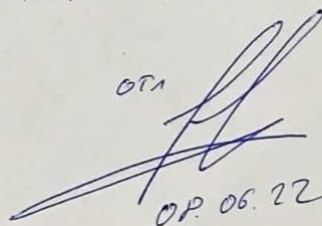
К курсовой работе
по курсу: «Программирование»
на тему: «База данных по студентам»

Выполнил студент группы 21ВВЗ:

Морозов К.В.

Принял:

к.т.н., доцент, зам. Декана ФИТЭ Голотенков Н.О.

от

09.06.22

Пенза, 2022

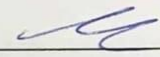
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ

 М.А. Митрохин

«___» _____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Программирование

Студенту Морозову Константину Вадимовичу Группа 21 В В 3

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

Исходные данные (технические требования) на проектирование

Наименование программы: База данных по студентам
Функции программы: Добавление записей; Удаление записей;
Редактирование записей; Поиск и сортировка записей; Фильтрация
записей по определенному критерию; Создание новой базы данных;
Сохранение текущей базы данных; Загрузка сохраненных баз 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.

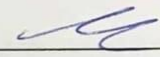
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ

 М.А. Митрохин

«___» _____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Программирование

Студенту Морозову Константину Вадимовичу Группа 21 ВВЗ

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

Исходные данные (технические требования) на проектирование

Наименование программы: База данных по студентам
Функции программы: Добавление записей; Удаление записей;
Редактирование записей; Поиск и сортировка записей; Фильтрация
записей по определенному критерию; Создание новой базы данных;
Сохранение текущей базы данных; Загрузка сохраненных баз 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.

Содержание

Введение	5
1.Постановка задачи	6
2.Выбор решения.....	7
3.Описание разработки программы	10
4.Отладка и тестирование.....	11
5.Описание программы	12
6.Руководство пользователя	24
Заключение.....	30
Список используемых материалов	31
Приложение А Листинг программы	32
Файл Counts.h	32
Файл Main.cpp	32
Приложение В Результаты работы программы	64

Введение

На сегодняшний день информация – ценный источник знаний, систематизации которой, позволяет оптимизировать работу как человека, так и общества в целом. Компьютерные технологии во многом упрощают нашу жизнь в самых разнообразных сферах нашей деятельности. И именно компьютерные технологии позволяют грамотно взаимодействовать с информацией, обрабатывать её и хранить на компактных носителях.

База данных представляет собой упорядоченную, структурированную совокупность данных, подобранных таким образом, что эти данные могут легко и быстро обрабатываться на ЭВМ.

История баз данных в узком смысле рассматривает базы данных в традиционном (современном) понимании. Эта история начинается с 1955 года, когда появилось программируемое оборудование обработки записей. Программное обеспечение этого времени поддерживало модель обработки записей на основе файлов. Для хранения данных использовались перфокарты.

Оперативные сетевые базы данных появились в середине 1960-х. Операции над оперативными базами данных обрабатывались в интерактивном режиме с помощью терминалов. Простые индексно-последовательные организации записей быстро развились к более мощной модели записей, ориентированной на наборы.

В наше время базы данных являются неотъемлемой частью любой организации, любого предприятия. Динамические сайты используют базу данных чтобы хранить данные пользователя, как и стриминговые сервисы с подписочной моделью. Банки хранят в огромных базах данных данные пользователей.

Данная программа является информационно-поисковой системой и позволяет работать с простейшей базой данных по студентам, позволяющей любому человеку найти информацию о нём, о его успеваемости, какие предметы он изучает.

1.Постановка задачи

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

Многомодульность программы. Программа должна быть поделена на логические модули. Это упростит поиск ошибок при отладке и тестировании программы, а также позволит легко расширять функционал программы.

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 достаточную информацию о ней.

2.Выбор решения

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

1. Добавление записи. Пользователь вводит данные, необходимые для нового элемента базы данных в порядке, указанном в программе. Все данные нужно вводить без пробелов. Программа считает количество записей, которых максимально может быть 30. Если количество превысит 30, то выйдет ошибка, что Невозможно добавить новый элемент.
2. Вывести записи. На экране будут отображаться все данные.
3. Удаление записи. Пользователь вводит ID записи, которую необходимо удалить. Программа находит данную запись и заменяет её ID на -1 для «отмены» отображения в функциях вывода. Поля при этом не заменяются.
4. Изменение записи. Пользователь вводит ID записи, которую необходимо изменить. Программа находит запись с данным ID и предлагает пользователю ввести новые данные для экземпляра (включая собственное ID. Пользователь вводит новые данные, и программа сохраняет новый экземпляр в базе данных.
5. Сортировка, поиск и фильтр записей. Пользователь выбирает один пункт из нового меню:
 - 5.1. Сортировка записей по возрастанию по ФИО и по ID. Пользователь выбирает один пункт из нового меню:
 - 5.1.1. Сортировка записей по возрастанию по ID. Программа сортирует базу данных по возрастанию по столбцу «ID».
 - 5.1.2. Сортировка записей по возрастанию по Фамилии. Программа сортирует базу данных по возрастанию по столбцу «Фамилия».
 - 5.1.3. Сортировка записей по возрастанию по Имени. Программа сортирует базу данных по возрастанию по столбцу «Имя».
 - 5.1.4. Сортировка записей по возрастанию по Отчеству. Программа сортирует базу данных по возрастанию по столбцу «Отчество».
 - 5.1.5. Отмена сортировки. Возвращает обратно главное меню

- 5.2. Сортировка записей по убыванию по ФИО и по ID. Пользователь выбирает один пункт из нового меню:
- 5.2.1. Сортировка записей по убыванию по ID. Программа сортирует базу данных по убыванию по столбцу «ID».
 - 5.2.2. Сортировка записей по убыванию по Фамилии. Программа сортирует базу данных по убыванию по столбцу «Фамилия».
 - 5.2.3. Сортировка записей по убыванию по Имени. Программа сортирует базу данных по убыванию по столбцу «Имя».
 - 5.2.4. Сортировка записей по убыванию по Отчеству. Программа сортирует базу данных по убыванию по столбцу «Отчество».
- 5.3. Фильтр записей по оценкам. Пользователь выписывает оценку для фильтрации. Затем пользователь выбирает предмет, по которому также будет проводится фильтрация.
- 5.4. Поиск записей по ФИО и по ID. Пользователь выбирает один пункт из нового меню:
- 5.4.1. Поиск записей по ID. Пользователь вводит ID записи. Если элемент с данным ID существует, то программа выводит его, иначе выведет пустую базу данных.
 - 5.4.2. Поиск записей по Фамилии. Пользователь вводит фамилию ученика. Если элемент с данным фамилией существует, то программа выводит его, иначе выведет пустую базу данных.
 - 5.4.3. Поиск записей по Имени. Пользователь вводит фамилию ученика. Если элемент с данным фамилией существует, то программа выводит его, иначе выведет пустую базу данных.
 - 5.4.4. Поиск записей по Отчеству. Пользователь вводит фамилию ученика. Если элемент с данным фамилией существует, то программа выводит его, иначе выведет пустую базу данных.
- 5.5. Обратно в главное меню. Возвращает в главное меню.
6. Загрузить БД в файл. Пользователь вводит название файла, в который необходимо записать данные. Если данный файл уже существует, то данные

в нем перезаписываются, в противном случае создается новый файл, в который записывается база данных.

7. Загрузить БД из файла. Пользователь вводит название файла, из которого необходимо считать данные. Если данный файл не существует или не соответствует формату, то программа выводит ошибку, иначе считывает базу данных из файла.

3.Описание разработки программы

Разработанная программа состоит из нескольких функций модуля, которые реализовывались в следующем порядке:

- 1) main()– основной файл программы. Меню программы.
- 2) consts.h – библиотека с константами.
- 3) Add()– добавление нового элемента в список.
- 4) Vivod() – вывод элементов в списке.
- 5) Del() – удаление элемента из списка.
- 6) Save() – запись базы данных в файл.
- 7) Load() – чтение базы данных из файла.
- 8) Change() – изменение элемента в списке.
- 9) Sort_Srch_Fltr() – меню для сортировки, поиска, фильтрации.
- 9) Sortup() – сортировка списка по возрастанию.
- 10) Sortdown – сортировка списка по убыванию.
- 11) Filter() – фильтрация по оценкам предметов.
- 12) Search() – поиск по ФИО и ID.

По порядку реализации модулей можно понять, что использовался нисходящий подход реализации программы. Этот метод основан на том, что сначала проектируются основные компоненты программы, а затем уже дорабатываются ее мелкие детали. Это позволяет упростить ее тестирование, так как каждая функция (модуль) тестируется сразу же после его реализации, что помогает в создании более структурированных программ.

4.Отладка и тестирование

В качестве среды разработки была выбрана программа Microsoft Visual Studio 2022. 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, с размерами самих массивов и т. д.

5.Описание программы

Приложение Студент Курсач.exe является основным модулем программы. При запуске программы происходит вывод заставки-приветствия, 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 таблице ниже.

Таблица 1 – Главная функция

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1, Enter	Выбран пункт «Добавить запись»	Запускается диалог добавления записи и добавление записи в БД
2, Enter	Выбран пункт «Вывести записи»	Выводит все записи БД
3, Enter	Выбран пункт «Удалить запись»	Запускается диалог удаление записи и удаление записи в БД
4, Enter	Выбран пункт «Изменить запись»	Запускается диалог изменение записи и изменение записи в БД
5, Enter	Выбран пункт «Сортировка, фильтр, поиск по записям»	Открывается новое меню см. Таблица 2
6, Enter	Выбран пункт «Загрузить Базу данных в файл»	Запускается диалог и запись существующей БД в файл.
7, Enter	Выбран пункт «Загрузить Базу данных из файла»	Запускается диалог и загрузка БД из файла

0, Enter	Выбран пункт «Выход»	Выполняется закрытие программы
----------	-------------------------	-----------------------------------

Таблица 2 - Сортировка, Фильтрация и Поиск по записям

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1, Enter	Выбран пункт «Сортировка по возрастанию по ФИО и ID»	Открывается новое меню см. Таблица 3
2, Enter	Выбран пункт «Сортировка по возрастанию по ФИО и ID»	Открывается новое меню см. Таблица 4
3, Enter	Выбран пункт «Фильтрация таблицы по оценкам предметов»	Открывается новое меню, в котором выбирается оценка для фильтрации. Затем открывается новое меню, см. Таблица 5
4, Enter	Выбран пункт «Поиск по ФИО и ID»	Открывается новое меню см. Таблица 6
5, Enter	Выбран пункт «Обратно в главное меню»	Выполняется возвращение в главное меню

Таблица 3 - Сортировка по возрастанию по ФИО и ID

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1, Enter	Выбран пункт «ID»	Программа выводит на экран отсортированную

		БД по возрастанию по столбцу ID
2, Enter	Выбран пункт «Фамилия»	Программа выводит на экран отсортированную БД по возрастанию по столбцу Фамилия
3, Enter	Выбран пункт «Имя»	Программа выводит на экран отсортированную БД по возрастанию по столбцу Имя
4, Enter	Выбран пункт «Отчество»	Программа выводит на экран отсортированную БД по возрастанию по столбцу Отчество
5, Enter	Выбран пункт «Отмена сортировки»	Выполняется возвращение в главное меню

Таблица 4 - Сортировка по убыванию по ФИО и ID

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1, Enter	Выбран пункт «ID»	Программа выводит на экран отсортированную БД по убыванию по столбцу ID
2, Enter	Выбран пункт «Фамилия»	Программа выводит на экран отсортированную БД по убыванию по столбцу Фамилия
3, Enter	Выбран пункт «Имя»	Программа выводит на экран отсортированную

		БД по убыванию по столбцу Имя
4, Enter	Выбран пункт «Отчество»	Программа выводит на экран отсортированную БД по убыванию по столбцу Отчество
5, Enter	Выбран пункт «Отмена сортировки»	Выполняется возвращение в главное меню

Таблица 5 - Фильтрация таблицы по оценкам предметов

Клавиши, вызывающие событие	Действие пользователя	Действие программы
0, Enter	Выбран пункт «Русский язык»	Программа выполняет фильтрацию по столбцу «Русский язык» и выводит на экран
1, Enter	Выбран пункт «Математика»	Программа выполняет фильтрацию по столбцу «Математика» и выводит на экран
2, Enter	Выбран пункт «Литература»	Программа выполняет фильтрацию по столбцу «Литература» и выводит на экран
3, Enter	Выбран пункт «География»	Программа выполняет фильтрацию по столбцу «География» и выводит на экран
4, Enter	Выбран пункт «Биология»	Программа выполняет фильтрацию по столбцу

		«Биология» и выводит на экран
5, Enter	Выбран пункт «Физика»	Программа выполняет фильтрацию по столбцу «Физика» и выводит на экран
6, Enter	Выбран пункт «Информатика»	Программа выполняет фильтрацию по столбцу «Информатика» и выводит на экран
7, Enter	Выбран пункт «Английский язык»	Программа выполняет фильтрацию по столбцу «Английский язык» и выводит на экран
8, Enter	Выбран пункт «История»	Программа выполняет фильтрацию по столбцу «История» и выводит на экран
9, Enter	Выбран пункт «Физическая культура»	Программа выполняет фильтрацию по столбцу «Физическая культура» и выводит на экран
S, Enter	Выбран пункт «Средняя оценка»	Программа выполняет фильтрацию по столбцу «Средняя оценка» и выводит на экран
E, Enter	Выбран пункт «Выход из фильтрации»	Выполняется возвращение в главное меню

Таблица 6 - Поиск по ФИО и ID

Клавиши, вызывающие событие	Действие пользователя	Действие программы
1, Enter	Выбран пункт «ID»	Программа выполняет поиск по столбцу «ID» и выводит на экран
2, Enter	Выбран пункт «Фамилия»	Программа выполняет поиск по столбцу «Фамилия» и выводит на экран
3, Enter	Выбран пункт «Имя»	Программа выполняет поиск по столбцу «Имя» и выводит на экран
4, Enter	Выбран пункт «Отчество»	Программа выполняет поиск по столбцу «Отчество» и выводит на экран
0, Enter	Выбран пункт «Отмена поиска»	Выполняется возвращение в главное меню

Далее на рисунке 1 представлена схема данных.

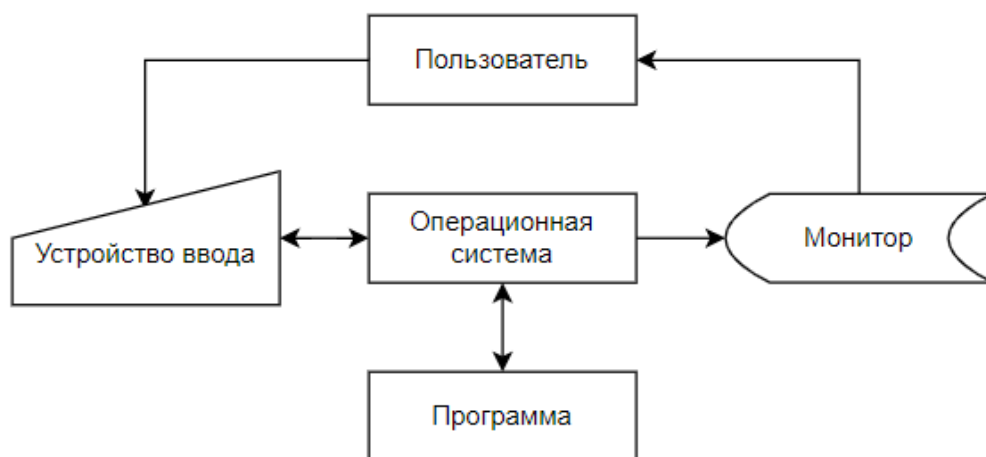


Рисунок - Схема данных

Схема программ:

Функция main.

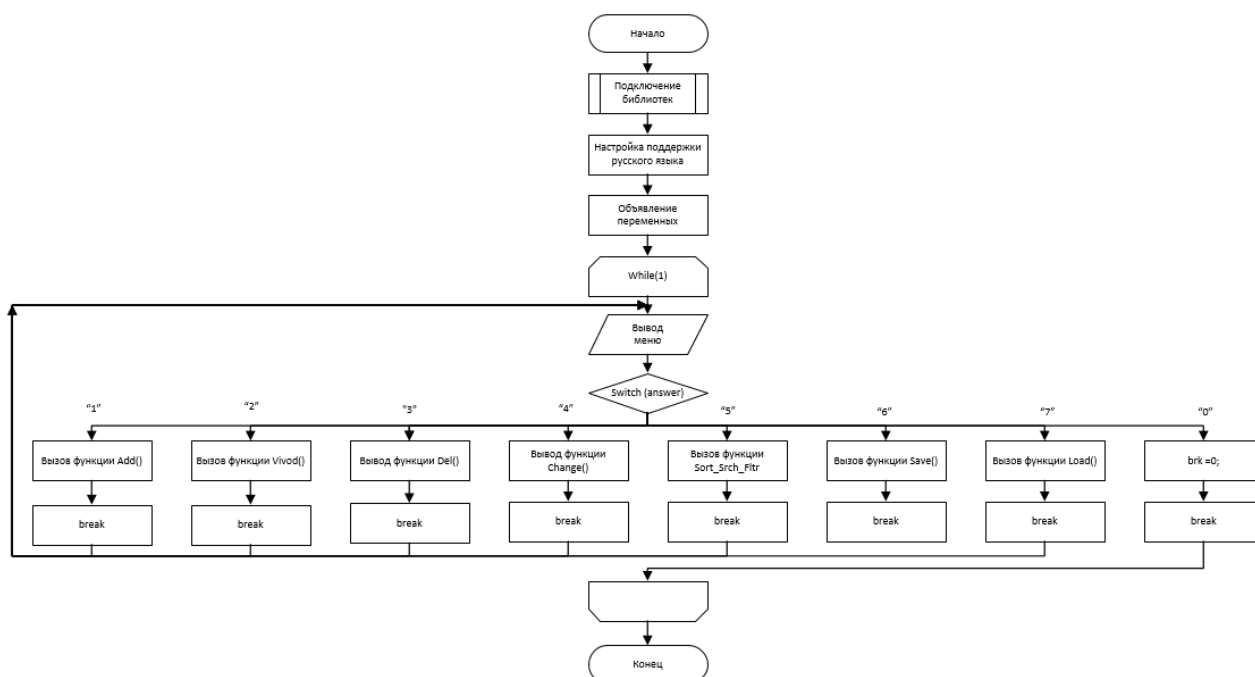


Рисунок - блок-схема функции main

Функция Add.

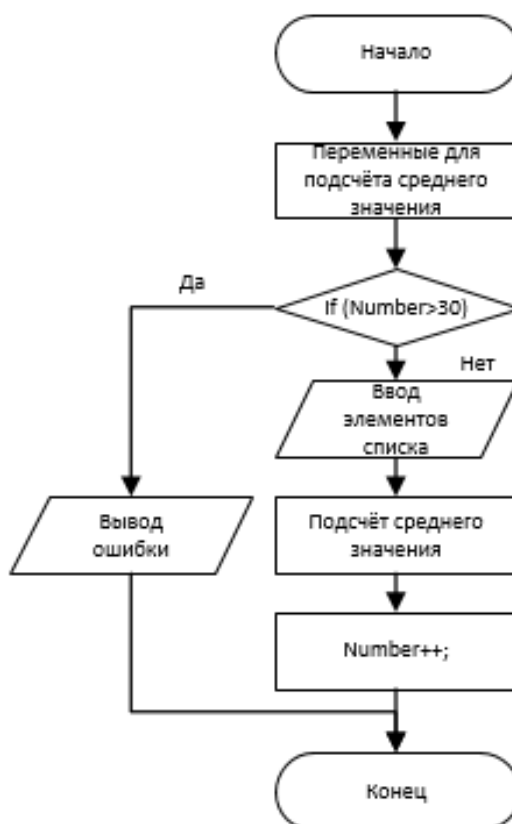


Рисунок - блок-схема функции Add

Функция Vivod.



Рисунок - Блок-схема функции Vivod

Функция Del.



Рисунок - блок-схема функции Del

Функция Save.

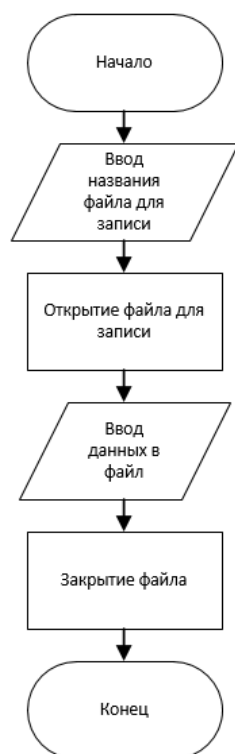


Рисунок - блок-схема функции Save

Функция Load.

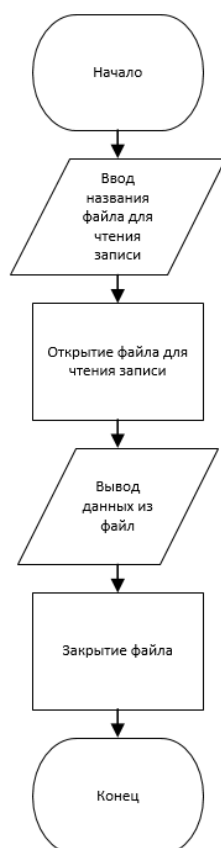


Рисунок - блок-схема функции Load

Функция Change.



Рисунок - блок-схема функции Change

Функция Sort_Srch_Fltr.

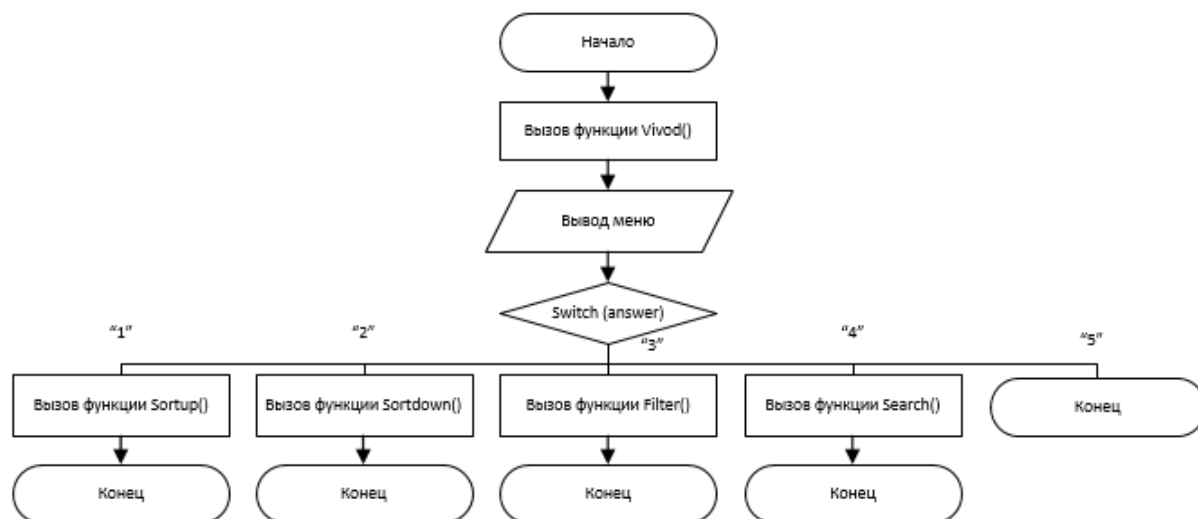


Рисунок - блок-схема функции Sort_Srch_Fltr

Функция Sortup.

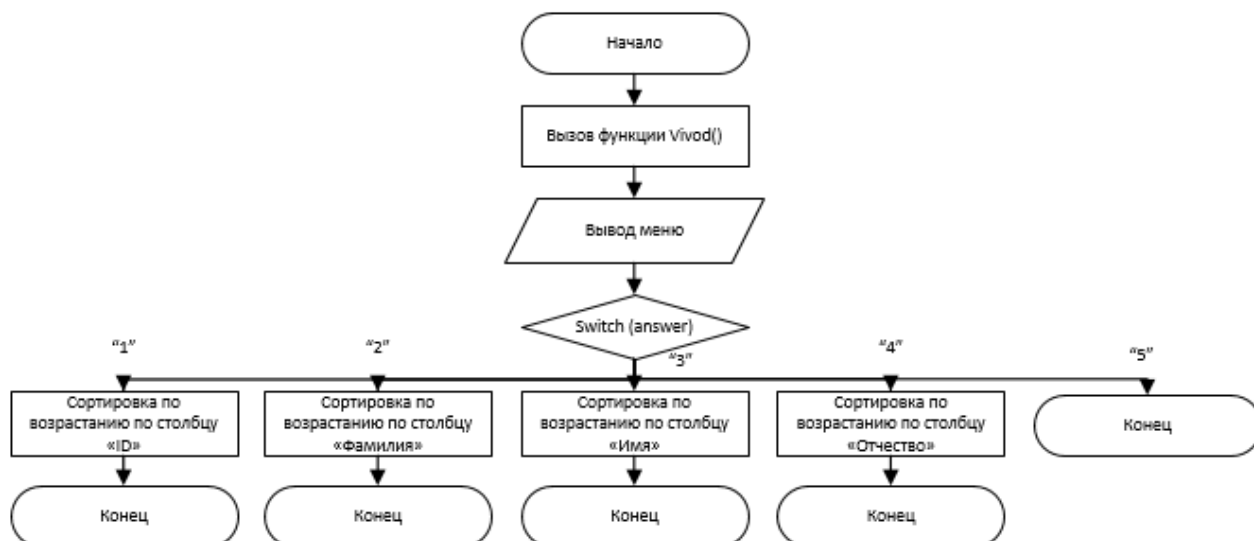


Рисунок - блок-схема функции Sortup

Функция Sortdown.

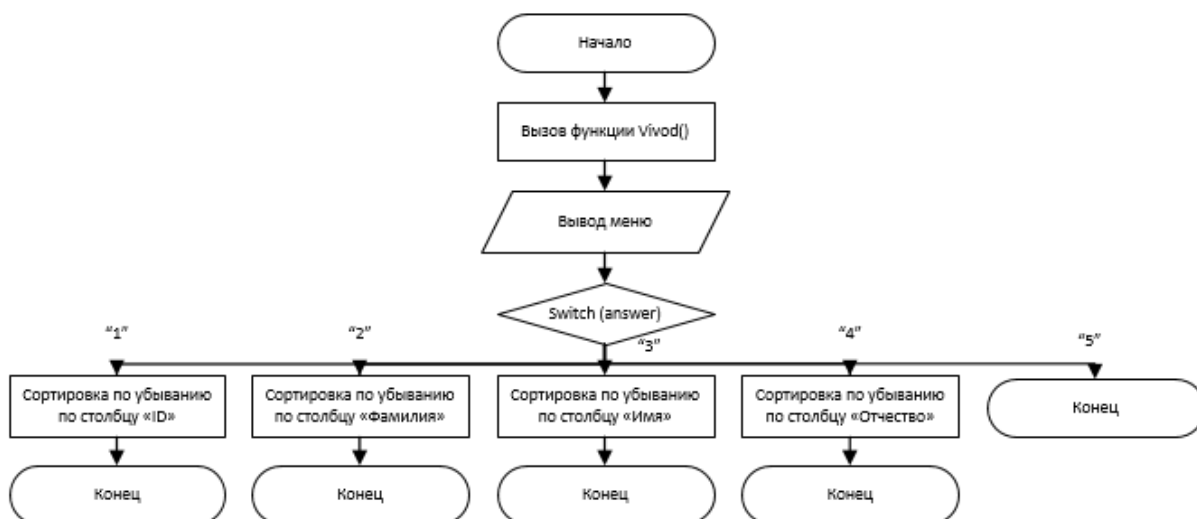


Рисунок - блок-схема функции Sortdown

Функция Filter.

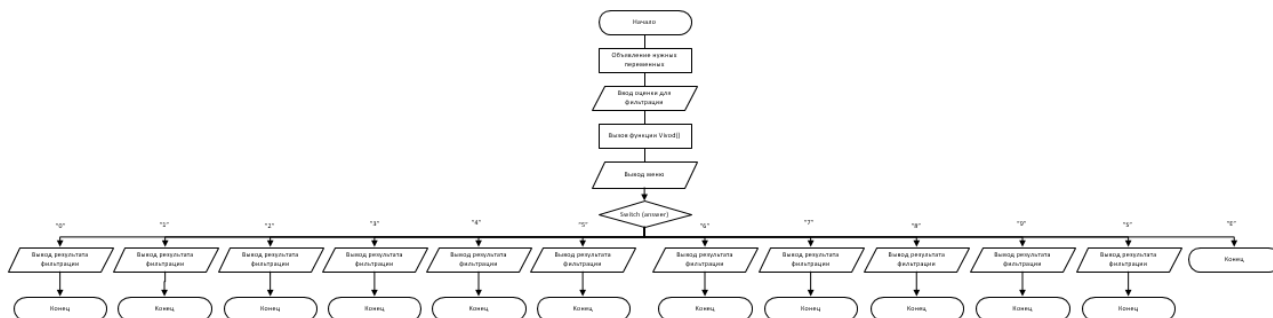


Рисунок - блок-схема функции Filter

Функция Search.

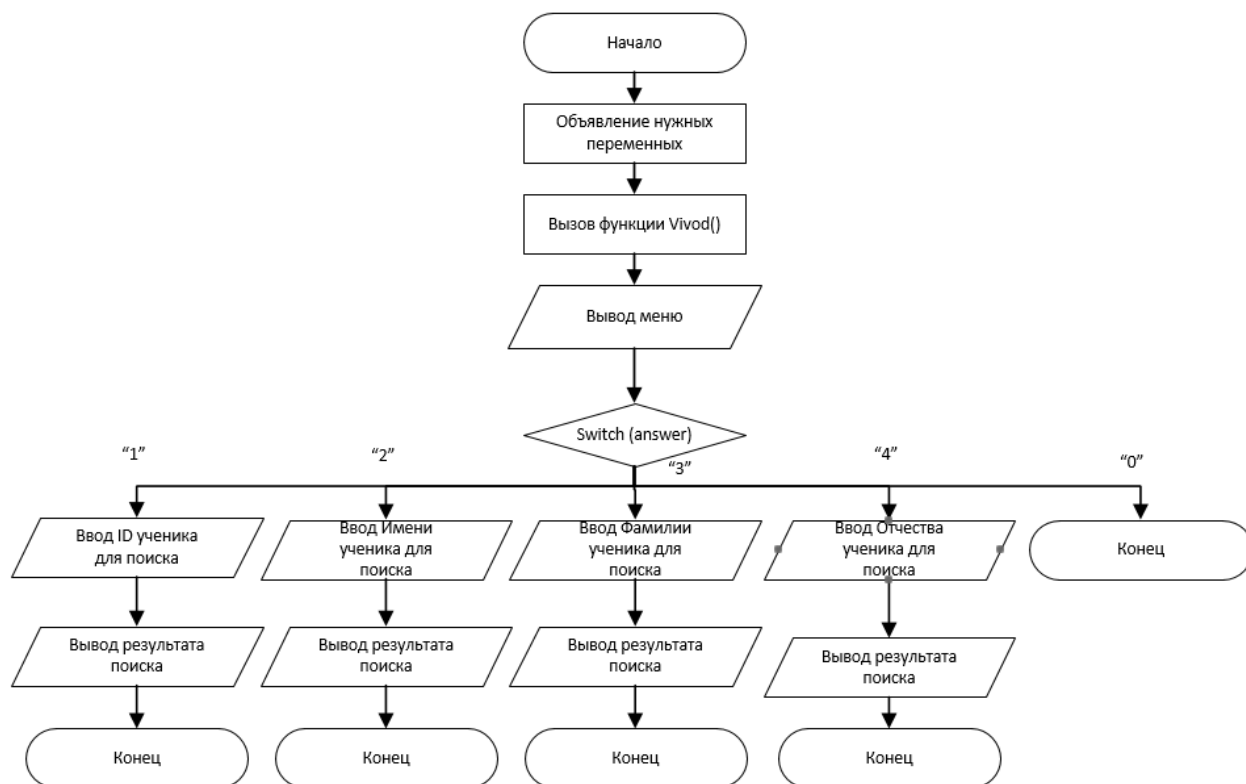


Рисунок - блок-схема функции Search

Ниже приведена иерархическая схема программы.

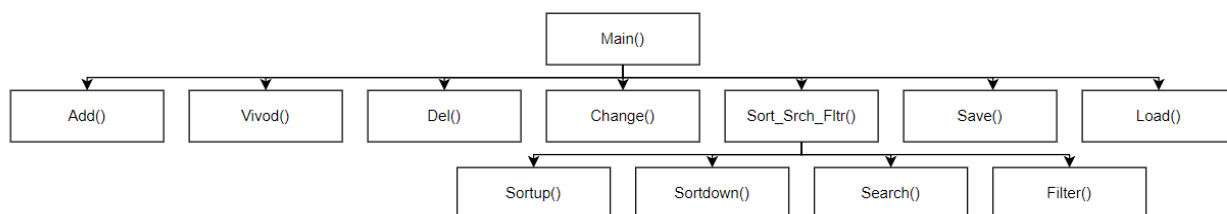


Рисунок - иерархическая схема программы

6.Руководство пользователя

Программа Студент Курсач.exe предназначена для хранения информации о студентах и их оценках. Программа имеет интуитивно понятный интерфейс и поддерживает такие операции как добавление новой записи в базу данных, удаление записи из базы данных, изменение записи в базе данных, поиск записи в базе данных, сортировка базы данных, сохранение существующей базы данных в файл, импорт базы данных из файла, сортированный вывод записей базы данных и фильтрация записей в базе данных.

Для открытия и вывода на экран существующего каталога необходимо выбрать пункт меню «Вывести записи».

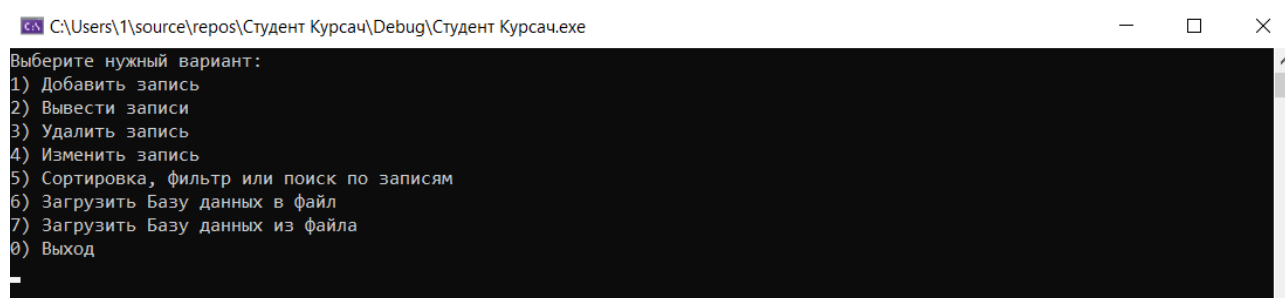
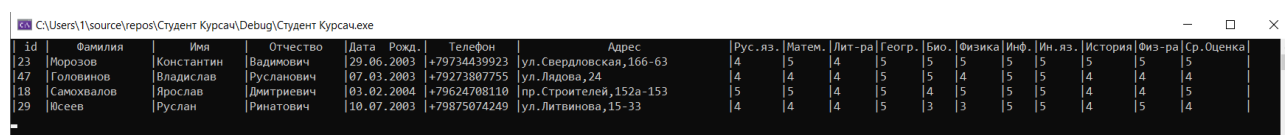


Рисунок - Главное меню



id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус. яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин. яз.	История	Физ-ра	Ср. Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
47	Головинов	Владислав	Русланович	07.03.2003	+79273807755	ул.Лядова,24	4	4	4	5	5	4	5	5	4	4	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - экран вывода

Для добавления записи в каталог необходимо выбрать пункт меню «Добавить запись». Далее необходимо заполнить форму структуры из семнадцати пунктов: ввести ID ученика, нажать клавишу Enter, ввести фамилию ученика, нажать клавишу Enter, ввести имя ученика, нажать клавишу Enter, ввести дату рождения ученика, нажать клавишу Enter, ввести телефон ученика, нажать клавишу Enter, ввести адрес ученика, нажать клавишу Enter, ввести оценку по русскому языку у ученика, нажать клавишу Enter, ввести оценку по математике у ученика, нажать клавишу Enter, ввести оценку по литературе у ученика, нажать клавишу Enter, ввести оценку по географии у ученика, нажать клавишу Enter, ввести оценку по биологии у ученика, нажать клавишу Enter, ввести оценку по физике у ученика, нажать клавишу Enter, ввести оценку по информатике у ученика, нажать клавишу Enter, ввести оценку по иностранному

языку у ученика, нажать клавишу Enter, ввести оценку по истории у ученика, нажать клавишу Enter, ввести оценку по физической культуре у ученика, нажать клавишу Enter.

Проверяем с помощью пункта «Вывести записи».



Рисунок - ввод элементов

ID	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
47	Головинов	Владислав	Русланович	07.03.2003	+79273807755	ул.Лядова,24	4	4	4	5	5	4	5	5	4	4	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4
15	Пронин	Илья	Геннадьевич	25.10.2003	+79354256796	пр.Строителей,138-27	4	3	3	5	4	4	5	5	4	5	4

Рисунок - полученный результат

Для удаления записи из каталога необходимо выбрать пункт меню «Удалить запись». Далее необходимо ввести ID удаляемой записи и нажать клавишу Enter.

Проверяем с помощью пункта «Вывести записи».

ID	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
47	Головинов	Владислав	Русланович	07.03.2003	+79273807755	ул.Лядова,24	4	4	4	5	5	4	5	5	4	4	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - экран удаления

ID	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
47	Головинов	Владислав	Русланович	07.03.2003	+79273807755	ул.Лядова,24	4	4	4	5	5	4	5	5	4	4	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - результат удаления

Для изменения записи необходимо выбрать пункт «Изменить запись» в меню. Далее необходимо ввести ID изменяемой записи и ввести новые данные для экземпляра из из семнадцати пунктов: ввести ID ученика, нажать клавишу Enter, ввести фамилию ученика, нажать клавишу Enter, ввести имя ученика, нажать клавишу Enter, ввести дату рождения ученика, нажать клавишу Enter, ввести телефон ученика, нажать клавишу Enter, ввести адрес ученика, нажать

клавишу Enter, ввести оценку по русскому языку у ученика, нажать клавишу Enter, ввести оценку по математике у ученика, нажать клавишу Enter, ввести оценку по литературе у ученика, нажать клавишу Enter, ввести оценку по географии у ученика, нажать клавишу Enter, ввести оценку по биологии у ученика, нажать клавишу Enter, ввести оценку по физике у ученика, нажать клавишу Enter, ввести оценку по информатике у ученика, нажать клавишу Enter, ввести оценку по иностранному языку у ученика, нажать клавишу Enter, ввести оценку по истории у ученика, нажать клавишу Enter, ввести оценку по физической культуре у ученика, нажать клавишу Enter.

Проверяем с помощью пункта «Вывести записи».

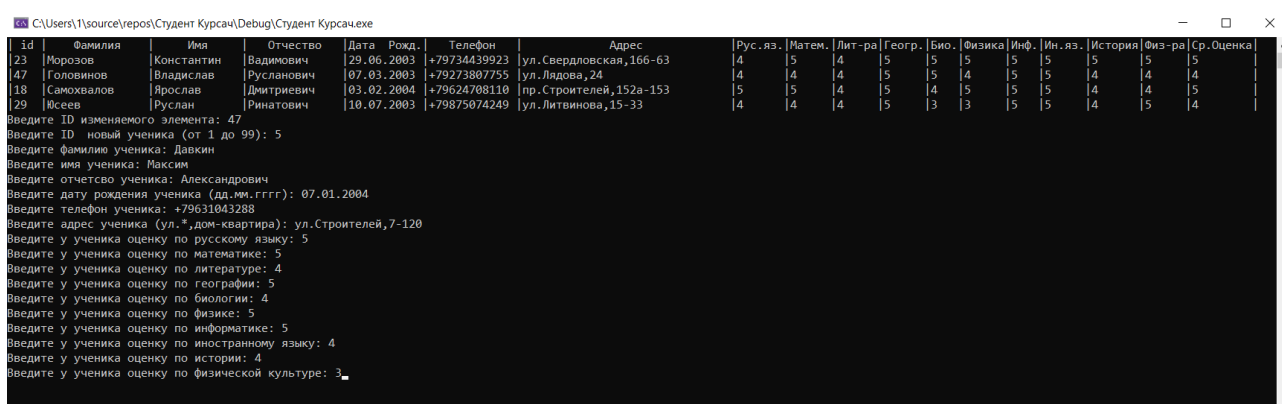


Рисунок - экран изменения

The screenshot shows the same Windows command prompt window as before, but the table of student data has been updated. The record for ID 47 has been replaced by a new record for ID 5.

id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Бiol.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - результат изменения

Для того чтобы сохранить текущую базу данных в файл необходимо выбрать пункт меню «Загрузить Базу данных в файл». Далее ввести имя файла, в который будет записана база данных. Если файл уже существует, то программа удалит информацию из файла и перезапишет данные БД.



Рисунок - экран сохранения

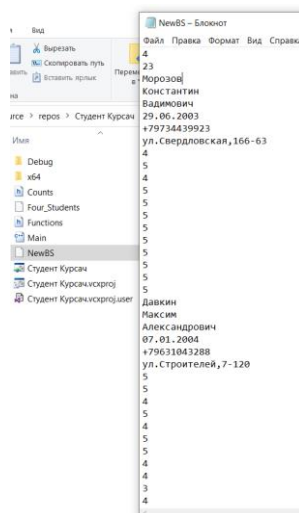


Рисунок - результат сохранения БД

Для того чтобы импортировать базу данных из файла необходимо выбрать пункт меню «Загрузить Базу данных из файла». Далее ввести имя файла, из которого произойдет чтение данных, нажать клавишу Enter. Текущая база данных будет заменена на новую, из файла. В случае ошибки программа выдаст сообщение «Невозможно открыть для чтения файла...».

Проверяем наличие данных с помощью пункта «Вывести записи».

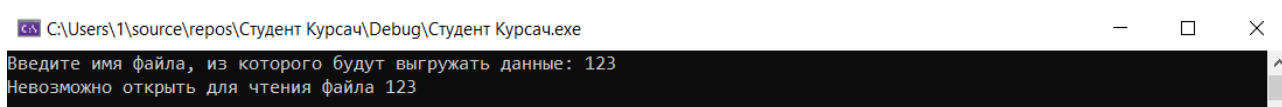


Рисунок - проверка на ошибку

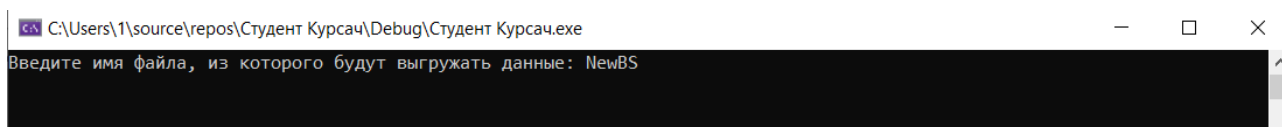


Рисунок - Ввод имени файла

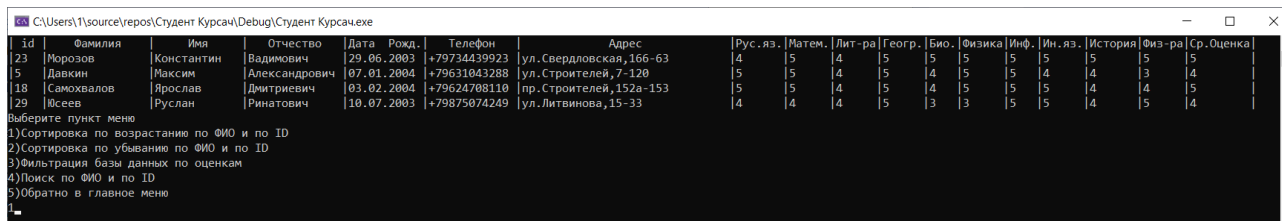
ID	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - результат загрузки БД из файла

Для того чтобы записи были выведены в отсортированном виде необходимо в главном меню выбрать пункт «Сортировка, фильтр или поиск по записям». В открывшемся меню нужно выбрать либо пункт «Сортировать записи по возрастанию по ФИО и по ID», либо пункт «Сортировать записи по убыванию по ФИО и по ID» в зависимости от того, как вы хотите отсортировать БД. Далее выбрать один из четырёх столбцов сортировки: «ID», «Фамилия», «Имя», «Отчество», нажать клавишу Enter.

Проверяем наличие данных с помощью пункта «Вывести записи».

Пример:

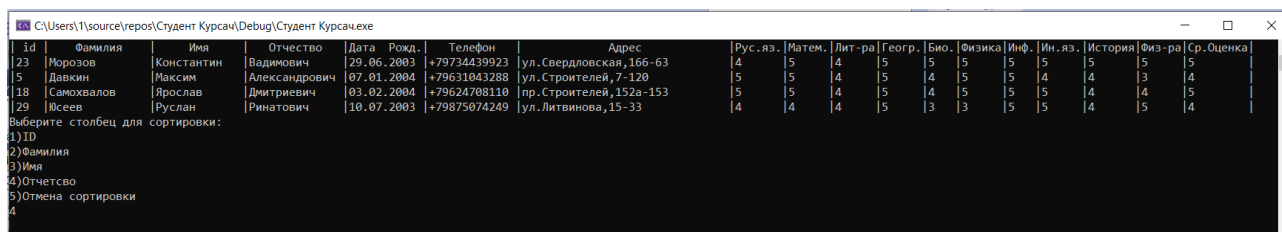


С:\Users\1\source\repos\Студент Курсач\Debug\Студент Курсач.exe

id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Выберите пункт меню:
1)Сортировка по возрастанию по ФИО и по ID
2)Сортировка по убыванию по ФИО и по ID
3)Фильтрация базы данных по оценкам
4)Поиск по ФИО и по ID
5)Обратно в главное меню

Рисунок - Выбор сортировки

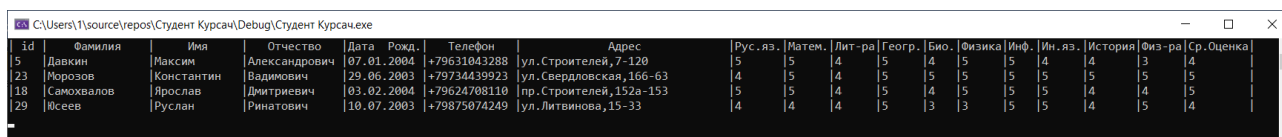


С:\Users\1\source\repos\Студент Курсач\Debug\Студент Курсач.exe

id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Выберите столбец для сортировки:
1) ID
2) Фамилия
3) Имя
4) Отчество
5) Отмена сортировки

Рисунок - Выбор столбца



С:\Users\1\source\repos\Студент Курсач\Debug\Студент Курсач.exe

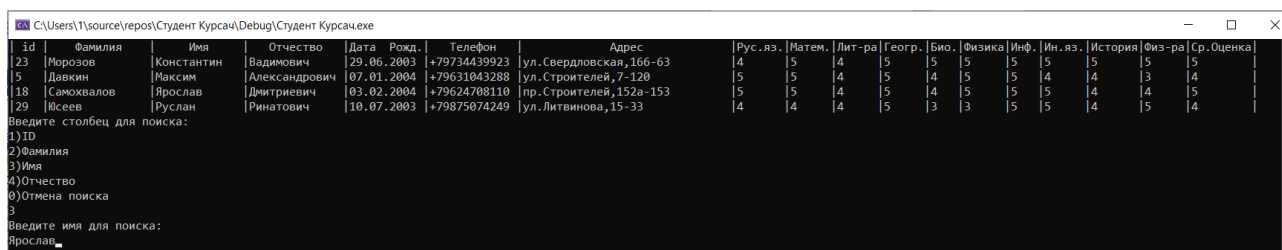
id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Рисунок - Результат сортировки

Для того чтобы найти определённую запись или группу записей необходимо выбрать пункт «Сортировка, фильтр или поиск по записям».

Если пользователю нужно найти определённую запись, выбираем пункт «Поиск по ФИО и по ID» Далее выбираем один из четырёх столбцов для поиска: «ID», «Фамилия», «Имя», «Отчество». Затем пользователю нужно ввести переменную для поиска по определённому столбцу, нажимаем клавишу Enter. Если всё было введено правильно, то экран выдаст результат поиска.

Пример:

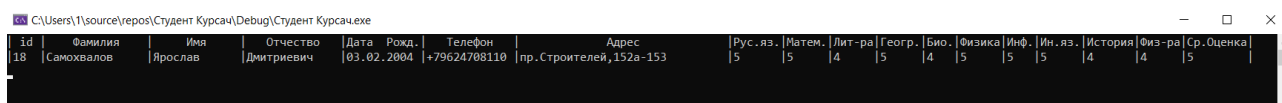


С:\Users\1\source\repos\Студент Курсач\Debug\Студент Курсач.exe

id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
23	Морозов	Константин	Вадимович	29.06.2003	+79734439923	ул.Свердловская,166-63	4	5	4	5	5	5	5	5	5	5	5
5	Давкин	Максим	Александрович	07.01.2004	+79631043288	ул.Строителей,7-120	5	5	4	5	4	5	5	4	4	3	4
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5
29	Юсеев	Руслан	Ринатович	10.07.2003	+79875074249	ул.Литвинова,15-33	4	4	4	5	3	3	5	5	4	5	4

Введите столбец для поиска:
1) ID
2) Фамилия
3) Имя
4) Отчество
0) Отмена поиска
3
Введите имя для поиска:
Ярослав

Рисунок - Экран поиска



С:\Users\1\source\repos\Студент Курсач\Debug\Студент Курсач.exe

id	Фамилия	Имя	Отчество	Дата Род.	Телефон	Адрес	Рус.яз.	Матем.	Лит-ра	Геогр.	Био.	Физика	Инф.	Ин.яз.	История	Физ-ра	Ср.Оценка
18	Самохвалов	Ярослав	Дмитриевич	03.02.2004	+79624708110	пр.Строителей,152а-153	5	5	4	5	4	5	5	5	4	4	5

Рисунок - результат поиска

Если пользователю нужно найти несколько записей, выбираем пункт «Фильтрация базы данных по оценкам». Далее пишем оценку и выбираем один из одиннадцати столбцов для фильтрации: Русский язык, Математика, Литература, География, Биология, Физика, Информатика, Английский(Иностранный) язык, История, Физическая культура и Средняя оценка. Если оценка совпадает с исходной записью, то на экране выйдет результат фильтрации.

Пример:

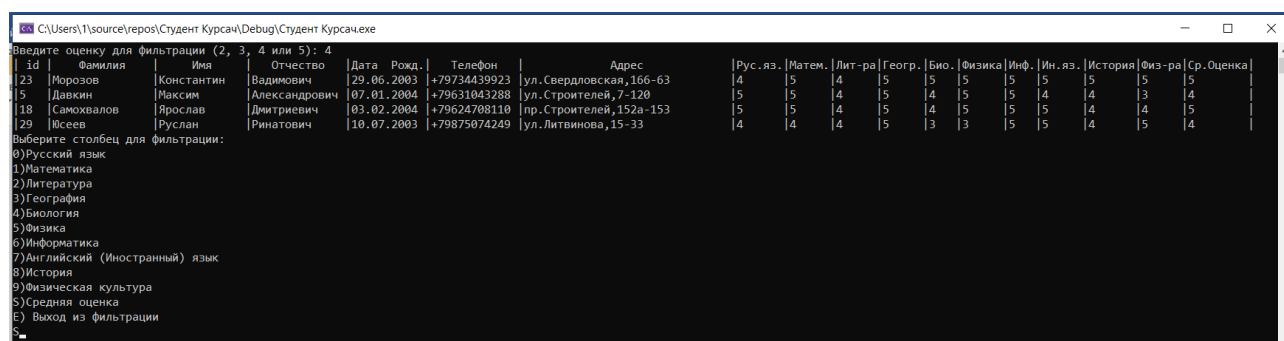


Рисунок - Экран Фильтрации

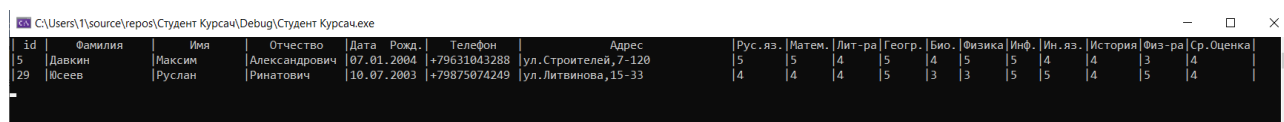


Рисунок - Результат фильтрации

Для выхода из программы необходимо выбрать пункт меню «Выход» и нажать клавишу Enter.

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 Visual Studio 2022 на языках Си и Assembler.

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

В дальнейшем программу можно улучшить, путём добавления графического интерфейса, что позволит улучшить навигацию по меню программы, путём добавления возможности использования мыши.

Список используемых материалов

1. Э. Таненбаум, Т. Остин. Архитектура компьютера (6-е издание). 2016 г.
2. Шилдт Герберт. Полный справочник по С. 2002 г.
3. MSDN.
4. Дэвид Гриффитс, Дон Гриффитс. Изучаем программирование на С. 2013 г.
5. Брайан Керниган. Деннис Ритчи. Язык программирования С. 2009 г.

Приложение А Листинг программы

Файл Counts.h

```
#pragma once
#include <iostream>
#include <sdkddkver.h>
#include <tchar.h>
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <math.h>
#include <string.h>
#include <locale.h>
#include <Windows.h>
#pragma warning(disable: 4996)
#define _CRT_SECURE_NO_WARNINGS
#define SetConsoleTitle
typedef struct TBL_Glv
{
    char id[3];
    char Fml[15];
    char Nm[13];
    char Ochv[15];
    char DoB[11];
    char tel[13];
    char address[31];
    char Russian[6];
    char Math[6];
    char Litra[6];
    char Geogr[6];
    char Bio[4];
    char Phis[6];
    char Inf[4];
    char Engl[6];
    char Story[7];
    char PE[6];
    char Sr_Znach[9];
}Glv;
Glv clv[30];
int Number;
bool brk = 0;
char answer;
int flti;
```

Файл Main.cpp

```
#include "Counts.h"
//Вывод данной БД
void Vivod(void)
{
    int i;
```

```

        printf("| id |      Фамилия      |      Имя      |      Отчество
|Дата  Рожд.|      Телефон      |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
                printf("|%-4s|%-15s|%-13s|%-14s|%-11s|%-13s|%-31s|%-7s|%-
6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-6s|%-9s|\n", clv[i].id,
clv[i].Fml, clv[i].Nm, clv[i].Ochv, clv[i].DoB, clv[i].tel,
clv[i].address, clv[i].Russian, clv[i].Math, clv[i].Litra,
clv[i].Geogr, clv[i].Bio, clv[i].Phis, clv[i].Inf, clv[i].Engl,
clv[i].Story, clv[i].PE, clv[i].Sr_Znach);
        }
        getchar();
        _getch();
}
//Добавление нового элемента в БД
void Add(void)
{
        if (Number > 30)
        {
                fprintf(stderr, "Невозможно добавить ещё один
элемент\n");
                return;
        }
        char Rs[1];
        char Mat[1];
        char Lit[1];
        char Geo[1];
        char Blog[1];
        char Ph[1];
        char Info[1];
        char Fl[1];
        char Str[1];
        char Phed[1];
        char average[1];

```

```

printf("Введите ID ученика (от 1 до 99): ");
scanf("%s", clv[Number].id);

printf("Введите фамилию ученика: ");
scanf("%s", clv[Number].Fml);

printf("Введите имя ученика: ");
scanf("%s", clv[Number].Nm);

printf("Введите отчетсво ученика: ");
scanf("%s", clv[Number].Ochv);

printf("Введите дату рождения ученика (дд.мм.гггг): ");
scanf("%s", clv[Number].DoB);

printf("Введите телефон ученика: ");
scanf("%s", clv[Number].tel);

printf("Введите адрес ученика (ул.*,дом-квартира): ");
scanf("%s", clv[Number].address);

printf("Введите у ученика оценку по русскому языку: ");
scanf("%s", clv[Number].Russian);
strncpy(Rs,clv[Number].Russian,1);

printf("Введите у ученика оценку по математике: ");
scanf("%s", clv[Number].Math);
strncpy(Mat,clv[Number].Math,1);

printf("Введите у ученика оценку по литературе: ");
scanf("%s", clv[Number].Litra);
strncpy(Lit, clv[Number].Litra, 1);

printf("Введите у ученика оценку по географии: ");
scanf("%s", clv[Number].Geogr);
strncpy(Geo, clv[Number].Geogr, 1);

```

```

printf("Введите у ученика оценку по биологии: ");
scanf("%s", clv[Number].Bio);
strncpy(Blog, clv[Number].Bio, 1);

printf("Введите у ученика оценку по физике: ");
scanf("%s", clv[Number].Phis);
strncpy(Ph, clv[Number].Phis, 1);

printf("Введите у ученика оценку по информатике: ");
scanf("%s", clv[Number].Inf);
strncpy(Info, clv[Number].Inf, 1);

printf("Введите у ученика оценку по иностранному языку: ");
scanf("%s", clv[Number].Engl);
strncpy(Fl, clv[Number].Engl, 1);

printf("Введите у ученика оценку по истории: ");
scanf("%s", clv[Number].Story);
strncpy(Str, clv[Number].Story, 1);

printf("Введите у ученика оценку по физической культуре: ");
scanf("%s", clv[Number].PE);
strncpy(Phed, clv[Number].PE, 1);
_asm
{
    xor eax, eax
    add al, Rs
    sub al, 30h;

    add al, Mat
    sub al, 30h;

    add al, Lit
    sub al, 30h;
}

```

```

        add al, Geo
        sub al, 30h;

        add al, Blog
        sub al, 30h;

        add al, Ph
        sub al, 30h;

        add al, Info
        sub al, 30h;

        add al, Fl
        sub al, 30h;

        add al, Str
        sub al, 30h;

        add al, Phed
        sub al, 30h;

        mov dl, 9
        div dl
        add al, 30h
        mov average, al
    }
    strncpy(clv[Number].Sr_Znach, average, 1);
    Number++;
}
//Удаление элемента БД по ID
void Del(void)
{
    int i = 0;
    Vivod();
    char DelID[3];
    printf("Введите ID ученика для удаления: ");

```

```

scanf("%s", DelID);
for (i == 0; i < Number; i++)
{
    if ((strncmp(clv[i].id, DelID, 2) == 0))
    {
        for (i; i < Number; i++)
            clv[i] = clv[i + 1];
        Number--;
        return;
    }
}
fprintf(stderr, "Элемент с номером %s не существует\n",
DelID);
_getch();
return;

    fprintf(stderr, "Элемент с номером %s не существует\n",
DelID);
    return;
}
//Сохранение данных в файл
void Save(void)
{
    char n_file[50];
    FILE* F;
    printf("Введите имя файла, в который будут записывать данные:
");
    scanf("%s", n_file);
    int i;
    if ((F = fopen(n_file, "w")) == 0)
    {
        fprintf(stderr, "\nНевозможно открыть для записи файл
%s\n", n_file);
        _getch();
        return;
    }
}

```

```

    fprintf(F, "%i\n", Number);
    for (i=0;i<Number;i++)

        fprintf(F,"%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n", clv[i].id, clv[i].Fml, clv[i].Nm,
        clv[i].Ochv, clv[i].DoB, clv[i].tel, clv[i].address,
        clv[i].Russian, clv[i].Math, clv[i].Litra, clv[i].Geogr,
        clv[i].Bio, clv[i].Phis, clv[i].Inf, clv[i].Engl, clv[i].Story,
        clv[i].PE, clv[i].Sr_Znach);

        fclose(F);
        getchar();
        return;
}
//Загрузка данных из файла
void Load(void)
{
    char n_file[50];
    FILE* F;
    printf("Введите имя файла, из которого будут выгружать
данные: ");
    scanf("%s", n_file);
    int i;
    if ((F = fopen(n_file, "r+")) == 0)
    {
        fprintf(stderr, "Невозможно открыть для чтения файла
%s\n", n_file);
        _getch();
        return;
    }
    fscanf(F, "%i\n", &Number);

    for (i = 0; i < Number; i++)
        fscanf(F,
"%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
        clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,

```

```

    clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
    clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
    clv[i].Sr_Znach);
    fclose(F);
    _getch();
    getchar();
    return;
}
//Изменение данных БД
void Change(void)
{
    int i;
    char ChangeID[3];
    Vivod();

    printf("Введите ID изменяемого элемента: ");
    scanf("%s", ChangeID);

    for (i = 0; i < Number; i++)
    {
        if ((strcmp(clv[i].id, ChangeID) == 0))
        {
            char Rs[1];
            char Mat[1];
            char Lit[1];
            char Geo[1];
            char Blog[1];
            char Ph[1];
            char Info[1];
            char Fl[1];
            char Str[1];
            char Phed[1];
            char average[1];
            printf("Введите ID  новый ученика (от 1 до 99): ");
            scanf("%s", clv[i].id);

```

```

printf("Введите фамилию ученика: ");
scanf("%s", clv[i].Fml);

printf("Введите имя ученика: ");
scanf("%s", clv[i].Nm);

printf("Введите отчетсво ученика: ");
scanf("%s", clv[i].Ochv);

printf("Введите дату рождения ученика (дд.мм.гггг):
");

scanf("%s", clv[i].DoB);

printf("Введите телефон ученика: ");
scanf("%s", clv[i].tel);

printf("Введите адрес ученика (ул.*,дом-квартира):
");

scanf("%s", clv[i].address);

printf("Введите у ученика оценку по русскому языку:
");

scanf("%s", clv[i].Russian);
strncpy(Rs, clv[i].Russian, 1);

printf("Введите у ученика оценку по математике: ");
scanf("%s", clv[i].Math);
strncpy(Mat, clv[i].Math, 1);

printf("Введите у ученика оценку по литературе: ");
scanf("%s", clv[i].Litra);
strncpy(Lit, clv[i].Litra, 1);

printf("Введите у ученика оценку по географии: ");
scanf("%s", clv[i].Geogr);
strncpy(Geo, clv[i].Geogr, 1);

```

```

printf("Введите у ученика оценку по биологии: ");
scanf("%s", clv[i].Bio);
strncpy(Blog, clv[i].Bio, 1);

printf("Введите у ученика оценку по физике: ");
scanf("%s", clv[i].Phis);
strncpy(Ph, clv[i].Phis, 1);

printf("Введите у ученика оценку по информатике: ");
scanf("%s", clv[i].Inf);
strncpy(Info, clv[i].Inf, 1);

printf("Введите у ученика оценку по иностранному
языку: ");

scanf("%s", clv[i].Engl);
strncpy(Fl, clv[i].Engl, 1);

printf("Введите у ученика оценку по истории: ");
scanf("%s", clv[i].Story);
strncpy(Str, clv[i].Story, 1);

printf("Введите у ученика оценку по физической
культуре: ");

scanf("%s", clv[i].PE);
strncpy(Phed, clv[i].PE, 1);
_asm
{
    xor eax, eax
    add al, Rs
    sub al, 30h;

    add al, Mat
        sub al, 30h;

    add al, Lit

```

```

        sub al, 30h;

        add al, Geo
        sub al, 30h;

        add al, Blog
        sub al, 30h;

        add al, Ph
        sub al, 30h;

        add al, Info
        sub al, 30h;

        add al, Fl
        sub al, 30h;

        add al, Str
        sub al, 30h;

        add al, Phed
        sub al, 30h;

        mov dl, 9
        div dl
        add al, 30h
        mov average, al
    }
    strncpy(clv[i].Sr_Znach, average, 1);
}
}
return;
}
//Сортировка по возрастанию
void Sortup(void)
{

```

```

Vivod();
int i, j;
Glv Temp;
while (1)
{
    printf("Выберите столбец для сортировки:\n");
    printf("1) ID\n");
    printf("2) Фамилия\n");
    printf("3) Имя\n");
    printf("4) Отчетсво\n");
    printf("5) Отмена сортировки\n");
    scanf_s("%c", &answer, 1);
    switch (answer)
    {
        case('1'):
        {
            for (j = Number - 1; j > 0; j--)
                for (i = 0; i < j; i++)
                    if (strcmp(clv[i].id, clv[i + 1].id)
> 0)
                        {
                            Temp = clv[i];
                            clv[i] = clv[i + 1];
                            clv[i + 1] = Temp;
                        }
            return;
        }
        case('2'):
        {
            for (j = Number - 1; j > 0; j--)
                for (i = 0; i < j; i++)
                    if (strcmp(clv[i].Fml, clv[i +
1].Fml) > 0)
                        {
                            Temp = clv[i];
                            clv[i] = clv[i + 1];

```

```

        clv[i + 1] = Temp;
    }
    return;
}
case('3'):
{
    for (j = Number - 1; j > 0; j--)
        for (i = 0; i < j; i++)
            if (strcmp(clv[i].Nm, clv[i + 1].Nm)
> 0)
            {
                Temp = clv[i];
                clv[i] = clv[i + 1];
                clv[i + 1] = Temp;
            }
    return;
}
case('4'):
{
    for (j = Number - 1; j > 0; j--)
        for (i = 0; i < j; i++)
            if (strcmp(clv[i].Ochv, clv[i +
1].Ochv) > 0)
            {
                Temp = clv[i];
                clv[i] = clv[i + 1];
                clv[i + 1] = Temp;
            }
    return;
}
case('5'):
{
    return;
}
}
system("cls");

```

```

        if (brk)
            break;
    }
}
//Сортировка по убыванию
void Sortdown(void)
{
    Vivod();
    int i, j;
    Glv Temp;
    while (1)
    {
        printf("Выберите столбец для сортировки:\n");
        printf("1) ID\n");
        printf("2) Фамилия\n");
        printf("3) Имя\n");
        printf("4) Отчетсво\n");
        printf("5) Отмена сортировки\n");
        scanf_s("%c", &answer, 1);
        switch (answer)
        {
            case('1'):
            {
                for (j = Number - 1; j > 0; j--)
                    for (i = 0; i < j; i++)
                        if (strcmp(clv[i].id, clv[i + 1].id)
< 0)
                        {
                            Temp = clv[i];
                            clv[i] = clv[i + 1];
                            clv[i + 1] = Temp;
                        }
                return;
            }
            case('2'):
            {

```

```

        for (j = Number - 1; j > 0; j--)
            for (i = 0; i < j; i++)
                if (strcmp(clv[i].Fml, clv[i +
1].Fml) < 0)

                    {

                        Temp = clv[i];
                        clv[i] = clv[i + 1];
                        clv[i + 1] = Temp;

                    }

            return;
        }
    case('3'):
    {
        for (j = Number - 1; j > 0; j--)
            for (i = 0; i < j; i++)
                if (strcmp(clv[i].Nm, clv[i + 1].Nm)
< 0)

                    {

                        Temp = clv[i];
                        clv[i] = clv[i + 1];
                        clv[i + 1] = Temp;

                    }

            return;
        }
    case('4'):
    {
        for (j = Number - 1; j > 0; j--)
            for (i = 0; i < j; i++)
                if (strcmp(clv[i].Ochv, clv[i +
1].Ochv) < 0)

                    {

                        Temp = clv[i];
                        clv[i] = clv[i + 1];
                        clv[i + 1] = Temp;

                    }

            return;
    }

```

```

        }
        case('5'):
        {
            return;
        }
    }
    system("cls");
    if (brk)
        break;
}
}
//Фильтрация данных по оценкам
void Filter(void)
{
    int i;
    int j;
    char Mark[1];
    char Mark2[]="2";
    char Mark3[]="3";
    char Mark4[]="4";
    char Mark5[]="5";
    printf("Введите оценку для фильтрации (2, 3, 4 или 5): ");
    scanf("%d", &j);
    if (j == 5)
        strncpy(Mark, Mark5, 1);
    if (j == 4)
        strncpy(Mark, Mark4, 1);
    if (j == 3)
        strncpy(Mark, Mark3, 1);
    if (j == 2)
        strncpy(Mark, Mark2, 1);
    while (1)
    {
        Vivod();
        printf("Выберите столбец для фильтрации:\n");
        printf("0) Русский язык\n");
    }
}

```

```

printf("1)Математика\n");
printf("2)Литература\n");
printf("3)География\n");
printf("4)Биология\n");
printf("5)Физика\n");
printf("6)Информатика\n");
printf("7)Английский (Иностранный) язык\n");
printf("8)История\n");
printf("9)Физическая культура\n");
printf("S)Средняя оценка\n");
printf("E) Выход из фильтрации\n");
scanf_s("%c", &answer, 1);
switch (answer)
{
    case('0'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество      |Дата  Рожд.|      Телефон      |      Адрес
| Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i=0; i < Number; i++)
        {
            if (strncmp(clv[i].Russian, Mark, 1) == 0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
    }
}

```

```

        return;
    }
    case('1'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");

        for (i = 0; i < Number; i++)
        {
            if (strncmp(clv[i].Math, Mark, 1) == 0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
        return;
    }
    case('2'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");

        for (i = 0; i < Number; i++)
        {
            if (strncmp(clv[i].Litra, Mark, 1) == 0)

```

```

        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);

        }

    }

    getchar();
    _getch();
    return;
}

case('3'):
{
    system("cls");
    printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");

    for (i = 0; i < Number; i++)
    {
        if (strncmp(clv[i].Geogr, Mark, 1) == 0)
        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);

        }

    }

    getchar();
    _getch();
}

```

```

        return;
    }
    case('4'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Bio, Mark, 1) == 0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
        return;
    }
    case('5'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Phis, Mark, 1) == 0)

```

```

        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
        }
    }
    getchar();
    _getch();
    return;
}
case('6'):
{
    system("cls");
    printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
    for (i = 0; i < Number; i++)
    {
        if (strncmp(clv[i].Inf, Mark, 1) == 0)
        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
        }
    }
    getchar();
    _getch();
}

```

```

        return;
    }
    case('7'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Engl, Mark, 1) == 0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
        return;
    }
    case('8'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Story, Mark, 1) == 0)

```

```

        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
        }
    }
    getchar();
    _getch();
    return;
}
case('9'):
{
    system("cls");
    printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
    for (i = 0; i < Number; i++)
    {
        if (strncmp(clv[i].PE, Mark, 1) == 0)
        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
        }
    }
    getchar();
    _getch();
}

```

```

        return;
    }
    case('S'):
    {
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Sr_Znach, Mark, 1) ==
0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
        return;
    }
    case('E'):
    {
        system("cls");
        return;
    }
    system("cls");
    if (brk)
        break;
}

```

```

    }
}
//Поиск по ФИО и по ID
void Search(void)
{
    int i;
    int j;
    char Srch[15];
    char SrchNm[13];
    char SrchID[3];
    while (1)
    {
        Vivod();
        printf("Введите столбец для поиска:\n");
        printf("1)ID\n");
        printf("2)Фамилия\n");
        printf("3)Имя\n");
        printf("4)Отчество\n");
        printf("0)Отмена поиска\n");
        scanf_s("%c", &answer, 1);
        switch (answer)
        {
            case('1'):
            {
                printf("Введите id для поиска:\n");
                scanf("%s", SrchID);
                system("cls");
                printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
| Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
                for (i = 0; i < Number; i++)
                {
                    if (strncmp(clv[i].id, SrchID, 2) == 0)
                    {

```

```

        printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);

    }

}

getchar();
_getch();
return;
}

case('2'):
{
    printf("Введите фамилию для поиска:\n");
    scanf("%s", Srch);
    system("cls");
    printf("| id |      Фамилия      |      Имя      |
Отчество      |Дата  Рожд.|      Телефон      |      Адрес
| Рус.яз. |Матем. |Лит-ра|Геогр. |Био. |Физика|Инф. |Ин.яз. |История|Физ-
ра|Ср.Оценка|\n");

    for (i = 0; i < Number; i++)
    {
        if (strcmp(clv[i].Fml, Srch) == 0)
        {
            printf("|%-4s|%-15s|%-13s|%-
14s|%-11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-
7s|%-6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);

        }

    }

    getchar();
}

```

```

        _getch();
        return;
    }
    case('3'):
    {
        printf("Введите имя для поиска:\n");
        scanf("%s", SrchNm);
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес
|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");
        for (i = 0; i < Number; i++)
        {
            if (strcmp(clv[i].Nm, SrchNm) == 0)
            {
                printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);
            }
        }
        getchar();
        _getch();
        return;
    }
    case('4'):
    {
        printf("Введите отчество для поиска:\n");
        scanf("%s", Srch);
        system("cls");
        printf("| id |      Фамилия      |      Имя      |
Отчество   |Дата  Рожд.|      Телефон   |      Адрес

```

```

|Рус.яз.|Матем.|Лит-ра|Геогр.|Био.|Физика|Инф.|Ин.яз.|История|Физ-
ра|Ср.Оценка|\n");

    for (i = 0; i < Number; i++)
    {
        if (strcmp(clv[i].Fml, Srch) == 0)
        {
            printf("|%-4s|%-15s|%-13s|%-14s|%-
11s|%-13s|%-31s|%-7s|%-6s|%-6s|%-6s|%-4s|%-6s|%-4s|%-6s|%-7s|%-
6s|%-9s|\n", clv[i].id, clv[i].Fml, clv[i].Nm, clv[i].Ochv,
clv[i].DoB, clv[i].tel, clv[i].address, clv[i].Russian,
clv[i].Math, clv[i].Litra, clv[i].Geogr, clv[i].Bio, clv[i].Phis,
clv[i].Inf, clv[i].Engl, clv[i].Story, clv[i].PE,
clv[i].Sr_Znach);

        }
    }
    getchar();
    _getch();
    return;
}

case('0'):
{
    return;
}

}

return;
}

//Отдельное меню для сортировки, поиска и фильтрации
void Sort_Srch_Fltr(void)
{
    Vivod();
    while (1)
    {
        printf("Выберите пункт меню\n");
        printf("1)Сортировка по возрастанию по ФИО и по ID\n");

```

```
printf("2)Сортировка по убыванию по ФИО и по ID\n");
printf("3)Фильтрация базы данных по оценкам\n");
printf("4)Поиск по ФИО и по ID\n");
printf("5)Обратно в главное меню\n");
scanf_s("%c", &answer, 1);
```

```
switch (answer)
{
    case('1'):
    {
        system("cls");
        Sortup();
        return;
    }
    case('2'):
    {
        system("cls");
        Sortdown();
        return;
    }
    case('3'):
    {
        system("cls");
        Filter();
        return;
    }
    case('4'):
    {
        system("cls");
        Search();
        return;
    }
    case('5'):
    {
        return;
    }
}
```

```

        }
        system("cls");
        if (brk)
            break;
    }
}
//Главное меню
void main(void)
{
    //Установка поддержки языка
    setlocale(LC_ALL, "Rus");
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    printf("Курсовая работа\nПо дисциплине
\n\"Программирование\"\n\nНа тему: \"База данных по
студентам\"\n\nВыполнил студент группы 21ВВ3: Мороозов
Константин\nПринял: зам. Декана ФИТЭ, доцент, к.т.н. Голотенков
Н.О.");
    _getch();
    system("cls");
    while (1)
    {
        //Вывод меню
        printf("Выберите нужный вариант:\n");
        printf("1) Добавить запись\n");
        printf("2) Вывести записи\n");
        printf("3) Удалить запись\n");
        printf("4) Изменить запись\n");
        printf("5) Сортировка, фильтр или поиск по записям\n");
        printf("6) Загрузить Базу данных в файл\n");
        printf("7) Загрузить Базу данных из файла\n");
        printf("0) Выход\n");
        scanf_s("%c", &answer, 1);
        switch (answer)
        {
            case ('1'):

```

```

{
    system("cls");
    Add();
    break;
}
case ('2'):
{
    system("cls");
    Vivod();
    break;
}
case ('3'):
{
    system("cls");
    Del();
    break;
}
case ('4'):
{
    system("cls");
    Change();
    break;
}
case ('5'):
{
    system("cls");
    Sort_Srch_Fltr();
    break;
}
case ('6'):
{
    system("cls");
    Save();
    break;
}
case ('7'):

```

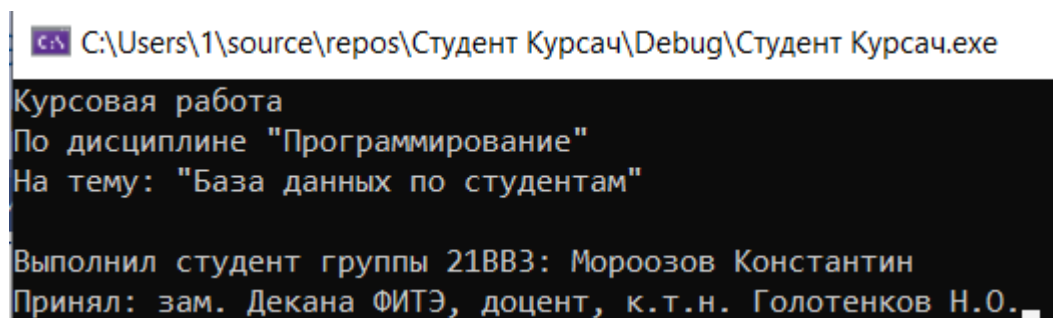
```

        {
            system("cls");
            Load();
            break;
        }
        case ('0'):
        {
            brk = 1;
            break;
        }
    }
    system("cls");
    if (brk)
        break;
}
_getch();
return;
}

```

Приложение В Результаты работы программы

Приложение В.1 – информационная заставка



Приложение С Результаты работы программы

Файл Four_Students

4

23

Морозов

Константин

Вадимович

29.06.2003

+79734439923

ул.Свердловская, 166-63

4

5

4

5

5

5

5

5

5

5

5

47

Головинов

Владислав

Русланович

07.03.2003

+79273807755

ул.Лядова, 24

4

4

4

5

5

4

5

5

4

4

4

18

Самохвалов

Ярослав

Дмитриевич

03.02.2004

+79624708110

пр.Строителей, 152а-153

5

5

4

5

4

5

5

5

4

4

5

29

Юсеев

Руслан

Ринатович

10.07.2003

+79875074249

ул. Литвинова, 15-33

4

4

4

5

3

3

5

5

4

5

4