Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

“Рязанский государственный радиотехнический университет

имени В. Ф. Уткина”

**Пояснительная записка**

**к курсовой работе по дисциплине**

**«Алгоритмические языки и программирование»**

**на тему**

**«Информационная система»**

Выполнил: студент гр. 043

Проверил: доцент

Антипов О.В.

Рязань, 2022

Содержание

[Задание 3](#_Toc103095120)

[Введение 4](#_Toc103095121)

[Анализ задания и математическая постановка задачи 5](#_Toc103095122)

[Разработка схемы алгоритма и ее описание 6](#_Toc103095123)

[Основной модуль 8](#_Toc103095124)

[Модуль TUI.pas 11](#_Toc103095125)

[Модуль DB.pas 21](#_Toc103095126)

[Модуль StringHelper.pas 28](#_Toc103095127)

[Разработка программы и ее описание 29](#_Toc103095128)

[Инструкция по пользованию разработанной программой 30](#_Toc103095129)

[Минимальные системные требования 30](#_Toc103095130)

[Список файлов 30](#_Toc103095131)

[Результаты отладки 38](#_Toc103095132)

[Решение контрольных примеров и проверка правильности функционирования программы 39](#_Toc103095133)

[Календарь выполнения работ 40](#_Toc103095134)

[Заключение 41](#_Toc103095135)

[Приложение: код программы 42](#_Toc103095136)

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

“Рязанский государственный радиотехнический университет

имени В. Ф. Уткина”

# Задание

на курсовую работу по дисциплине

«Алгоритмические языки и программирование»

студенту

Разработать программу создания-редактирования-удаления таблиц произвольной длины с произвольным числом столбцов. Для каждого столбца задаётся тип данных – текстовый или числовой. В программе должны быть предусмотрены функции добавления и удаления строк и столбцов в созданных таблицах, поиск информации по заданному ключу, а также сортировка строк по заданному столбцу по возрастанию или убыванию. Программа должна адекватно пролистывать страницу, отображая несколько столбцов и строк на одном экране, а также отображая информацию с какого по какой столбец и строка отображаются в данный момент. В программе должно быть создано не менее 4 таблиц, с количеством строк не менее 10, и различным числом столбцов от 4 до 10.

Срок представления работы к защите:

Дата выдачи задания:

Преподаватель: Антипов О.В.

Введение

Данная курсовая работа является заключительным этапом в изучении предмета «Алгоритмические языки и программирование». Программный продукт разрабатывается в программной среде Pascal ABC.

Целью проектирования данной работы является создание именованной совокупности данных, отражающей состояние объектов и их отношений в рассматриваемой предметной области, т.е. файловой (информационной) системы, а также системы управления спроектированной структурой.

Анализ задания и математическая постановка задачи

При запуске программы пользователь видит форму для ввода логина, а затем и пароля для доступа к базе данных с правами чтения и/или редактирования.

При неудачной авторизации программа завершает свою работу.

После успешной авторизации пользователь видит главное меню, в котором доступны следующие действия с таблицами базы данных:

* Просмотреть таблицу;
* Создать таблицу;
* Отредактировать таблицу;
* Удалить таблицу.

При выборе «Отредактировать» доступные следующие действия:

* Добавить столбец (текстовый или числовой);
* Добавить строку;
* Удалить столбец;
* Удалить строку;
* Задать значение ячейки таблицы;
* Поиск значения в таблице;
* Сортировка строк таблицы по заданному столбцу (по возрастанию или убыванию).

Разработка схемы алгоритма и ее описание

По результатам анализа задания можно составить упрощенную схему алгоритма (рис. 1-3). Детализируем алгоритм (рис. 4-40).



Рисунок 1 – Упрощенная схема алгоритма



Рисунок 2 – Главное меню



Рисунок 3 – Редактирование таблицы

## Основной модуль



Рисунок 4 – Основная функция программы



Рисунок 5 – Главное меню



Рисунок 6 - Показ таблицы



Рисунок 7 – Создание таблицы



Рисунок 8 – Редактирование таблицы



Рисунок 9 – Удаление таблицы



Рисунок 10 – Добавление столбца



Рисунок 11 – Удаление столбца



Рисунок 12 – Добавление строки



Рисунок 13 – Удаление строки



Рисунок 14 – Изменение ячейки



Рисунок 15 – Поиск значения в таблице



Рисунок 16 – Сортировка таблицы



Рисунок 17 – Подтверждение действия



Рисунок 16 – Выбор индекса в пределах массива



Рисунок 16 – Выбор индекса в пределах массива



Рисунок 16 – Ожидание ввода кода возврата



Рисунок 16 – Вывод таблицы



Рисунок 16 – Конвертирование в тип Table



Рисунок 16 – Конвертирование в тип FileTable



Рисунок 16 – Чтение БД пользователей



Рисунок 16 – Удаление таблицы из массива таблиц



Рисунок 16 – Удаление столбца



Рисунок 16 – Удаление строки



Рисунок 16 – Удаление строки



Рисунок 16 – Чтение таблиц из БД



Рисунок 16 – Запись таблиц в БД



Рисунок 16 – Отображение названий таблиц



Рисунок 16 – Сортировка по возрастанию



Рисунок 16 – Сортировка по убыванию

Разработка программы и ее описание

Первым этапом была разработана система для работы с текстовой базой данных, методы считывания и записи на жесткий диск данных.

Затем были реализованы универсальные вспомогательные функции для работы с типизированным файлом, добавления, удаления элемента в динамический массив.

Следующим шагом стал вывод содержимого таблиц, меню с использованием псевдографики.

Большая часть функций возвращает коды ошибок в или значение успешного/неудачного выполнения.

Инструкция по пользованию разработанной программой

Данный программный продукт был реализован на языке программирования Паскаль с использованием среды PascalABC.NET и разработан для машин на базе ОС Microsoft Windows.

## Минимальные системные требования

* ОС: Microsoft Windows 7
* Процессор: 1ГГц
* ОЗУ: 512 МБ
* Microsoft .NET framework 4.0

## Список файлов

Исходники:

* kurs.pas

Готовый продукт:

* kurs.exe
* user.data
* database.dat

Основное управление в программе происходит через выбор нужного действия с клавиатуры.

При запуске программы пользователь увидит окно авторизации.

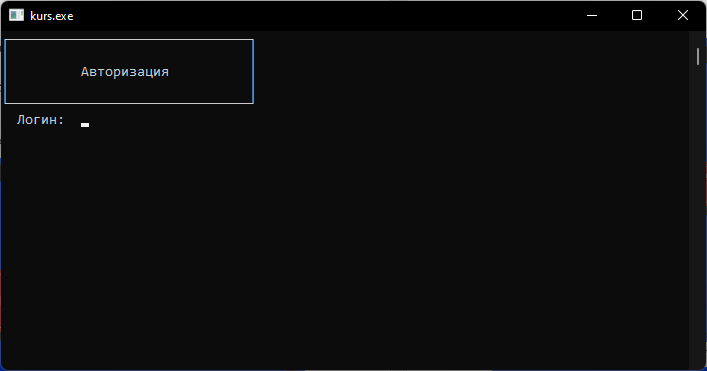


Рисунок 41 - Окно ввода логина

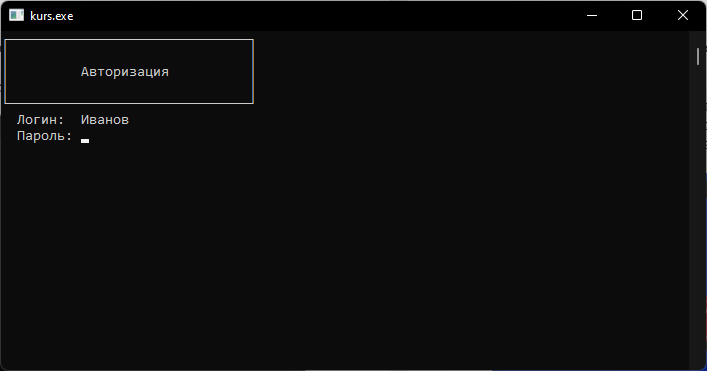


Рисунок 42 - Окно ввода пароля

В случае успешной авторизации откроется главное меню. Если пользователь имеет права для записи, об этом дополнительно появится уведомление.

В случае ввода неправильных учетных данных у пользователя есть в общем счете 5 попыток на вход в систему. После 5х неудачных попыток приложение завершит свою работу принудительно.

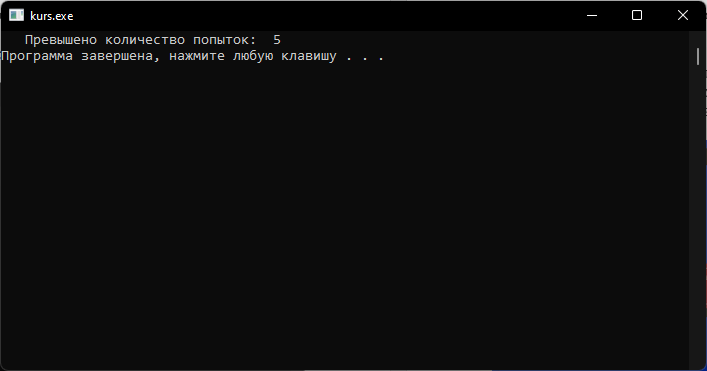


Рисунок 44 - Пример ошибки при авторизации

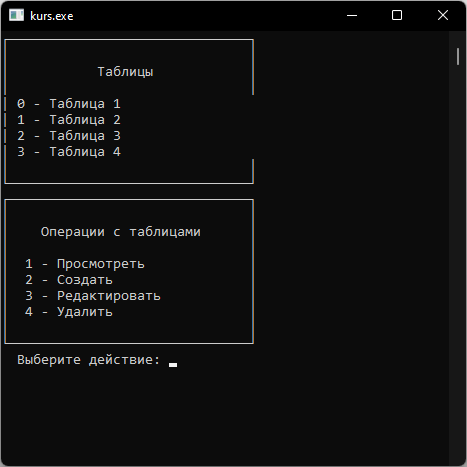


Рисунок 45 - Главное меню (решим администратора)

Из главного меню пользователь может перейти на меню просмотра, создания, редактирования и удаления таблицы.

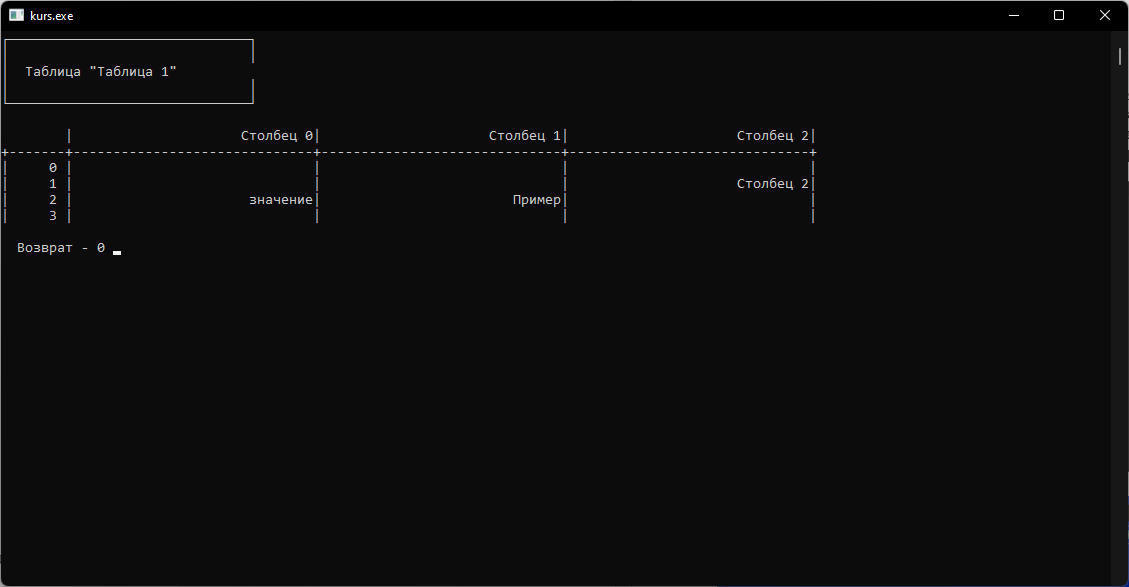


Рисунок 46 - Просмотр таблицы

При выборе действия «Редактировать» откроется меню с возможными действиями над таблицей, среди которых добавление/удаление столбцов, строк, изменение ячейки, поиск значений, сортировка.

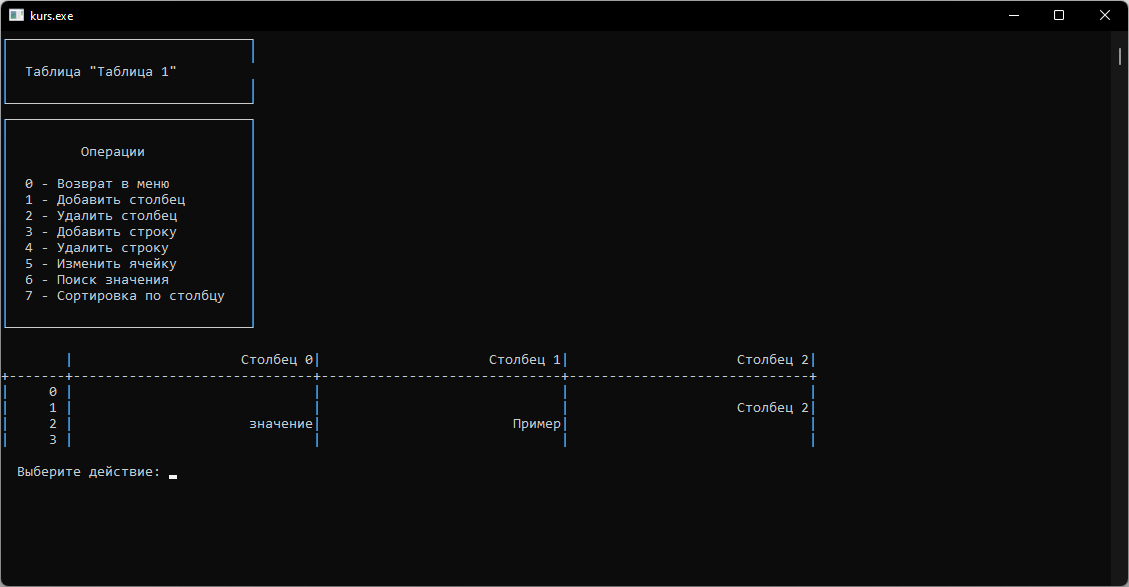


Рисунок 47 - Действия над базой данных

Результаты отладки

Была встречена ошибка в сегменте:

**function** toolsGrid(

**var** tools: tools\_t;

page: uint64\_t;

perPage: uint64\_t;

maxPage: uint64\_t;

selected: uint64\_t;

title: string

): int64\_t;

TUI.pas(67) : Неизвестное имя 'tools\_t'

После анализа было выяснено, что не был подключен модуль DB, в котором определен тип tools\_t.

Была встречена ошибка в сегменте:

**var** color: uint8\_t;

**if** (active) **then begin**

color := colorPopupActive;

**end else**

color := colorPopupInactive;

writeX('[ ' + text + ' ]', color);

TUI.pas(182) : Неизвестное имя 'colorPopupInactive'

В ходе анализа было выяснено, что константа colorPopupInactive не была определена.

Была встречена ошибка в сегменте

**var** nextElement: int64\_t = first + (selected **mod** perPage);

Было выяснено, что при неверном аргументе selected возможен выход за пределы массива элементов.

Сегмент был заменен на

**var** nextElement: int64\_t = min(first + (selected **mod** perPage), tools.length - 1);

Решение контрольных примеров и проверка правильности функционирования программы

Для проверки функциональности программы попробуем отсортировать таблицу и записать изменения.

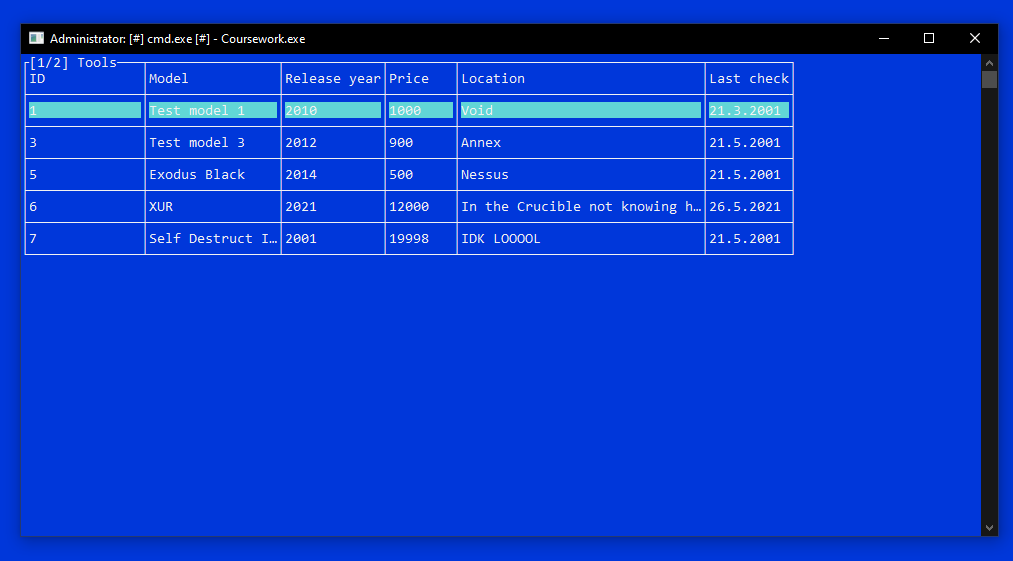


Рисунок 53 - Оригинальная БД

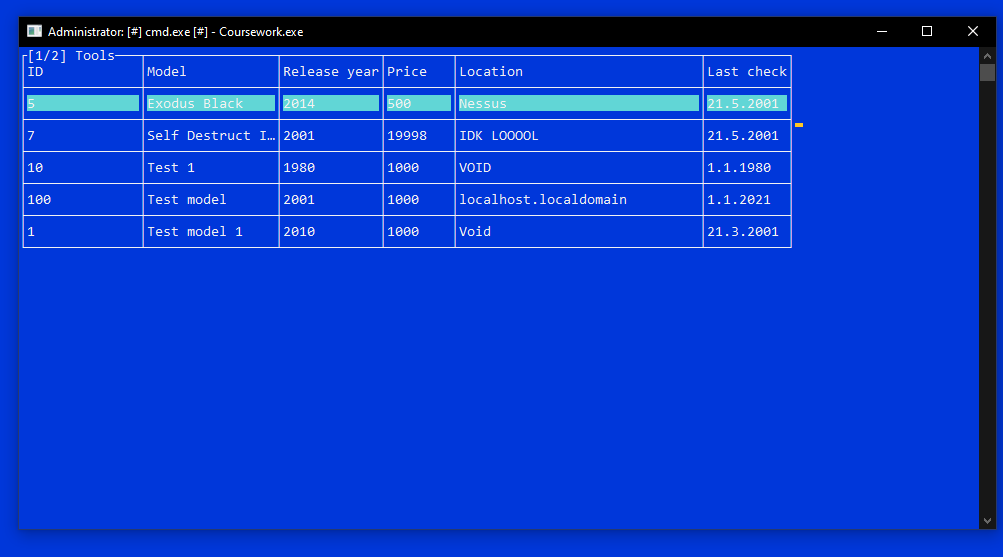


Рисунок 54 - Отсортированная по модели

Календарь выполнения работ

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Этапы решения задачи | Ориентировочная трудоёмкость, ч | Реально затраченное время, ч |
| 1 | Анализ предметной области и формализация задачи | 5 | 4 |
| 2 | Разработка структуры организации данных | 2 | 1 |
| 3 | Составление алгоритма | 4 | 8 |
| 4 | Разработка схемы интерфейса | 1 | 1 |
| 5 | Написание программы | 3 | 4 |
| 6 | Устранение синтаксических ошибок | 1 | 0,5 |
| 7 | Подготовка тестов |  | 0 |
| 8 | Тестирование и отладка | 5 | 10 |
| 9 | Подготовка документации | 10 | 5 |
| 10 | Внесение правок | 4 | 6 |
| Итого | | 35 | 39.5 |

Заключение

По результатам выполнения курсовой работы разработана программа создания-редактирования-удаления таблиц произвольной длины с произвольным числом столбцов. Пользователи имеют разграничение в правах. Все необходимые данные хранятся во внешних файлах. Также присутствует проверка на вводимые пользователем данные.

Таким образом задача курсовой работы выполнена.

Приложение: код программы