Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**Иркутский национальный исследовательский технический университет**

|  |
| --- |
| Институт информационных технологий и анализа данных |
| наименование института |

Допускаю к защите

|  |  |
| --- | --- |
| Руководитель |  |
|  | подпись |
|  | И.В. Игумнов |
|  | И.О. Фамилия |

Формирование бинарного файла

|  |
| --- |
| наименование темы |

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

|  |
| --- |
| Программирование на языке высокого уровня |

|  |
| --- |
| 1.041.00.00 ПЗ |
| обозначение документа |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент |  | ИСТб-21-1 |  |  |  | К.С. Казарян |
|  |  | шифр |  | подпись |  | И.О. Фамилия |
| Нормоконтроль |  |  |  |  |  | И. В. Игумнов |
|  |  |  |  | подпись |  | И.О. Фамилия |
| Курсовая работа защищена с оценкой | | | |  | | |

Иркутск 2022 г.

ЗАДАНИЕ

НА КУРСОВОЕ ПРОЕКТИРОВАНИЕ (КУРСОВУЮ РАБОТУ)

По курсу «Программирование на языке высокого уровня»

Студенту Казарян Константину Степановичу

Тема работы (проекта)Создание справочника по форматам графических файлов

Исходные данные: написать программу, рассчитанную на активное взаимодействие с пользователем посредством графического интерфейса.

Рекомендуемая литература:

1. Павловская Т. А. C/C++. Программирование на языке высокого уровня для магистров и бакалавров : учебник для студентов высших учебных заведений /Т.А. Павловская, 2011. - 460.
2. Трофимов, В. В. Алгоритмизация и программирование : учебник для академического бакалавриата / В. В. Трофимов, Т. А. Павловская ; под ред. В. В. Трофимова. — М. : Издательство Юрайт, 2019. — 137 с. (Серия: Бакалавр. Академический курс. Модуль.). — ISBN 978-5-534-07834-3.(<https://biblio-online.ru/book/algoritmizaciya-i-programmirovanie-423824>)
3. Кувшинов, Д. Р. Основы программирования : учеб. пособие для вузов / Д. Р. Кувшинов. — М. : Издательство Юрайт, 2019. — 104 с. — (Серия : Университеты России).
4. Хохлов Д.Г. Методы программирования на языке С: практикум Ч.2. «Бином. Лаборатория знаний». 2014. – 377 с. [Электронный ресурс]. – Режим доступа: http://e.lanbook.com/books/element.php ?pl1\_id=50555
5. Липман С., Лажойе Ж. Язык программирования С++. Полное руководство. СПб.: Невский диалект, М.: ДМК Пресс. 2006. – 1104с. [Электронный ресурс]. – Режим доступа: http://e.lanbook.com/view/book/ 1216/
6. Дейл Н., Уимз Ч., Хедингтон М. Программирование на C++. М.: ДМК Пресс. 2007.– 672с. [Электронный ресурс]. – Режим доступа: <http://e.lanbook.com/view/book/1219/>

Графическая часть на \_\_\_\_\_\_\_\_\_\_\_\_\_\_ листах.

Дата выдачи задания “\_\_\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Задание получил \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись И.О.Фамилия

Дата представления работы руководителю “\_\_\_\_” \_\_\_\_\_\_\_ 20\_\_г.

Руководитель курсовой работы \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись И.О.Фамилия

**Сод****ержание**

[Введение 4](#_Toc111811398)

[Задание 5](#_Toc111811399)

[Постановка задачи 6](#_Toc111811400)

[Таблица спецификаций глобальных переменных 7](#_Toc111811401)

[Схема иерархии модулей 7](#_Toc111811402)

[Описание функций в содержательном аспекте 10](#_Toc111811403)

[Алгоритмы 12](#_Toc111811404)

[Таблица тестов 14](#_Toc111811405)

[Листинг 14](#_Toc111811406)

[Результаты тестирования 23](#_Toc111811407)

[Проект руководства пользователя 27](#_Toc111811408)

[1. Общие сведения о программе 27](#_Toc111811409)

[2. Установка, запуск 27](#_Toc111811410)

[3. Основные возможности программы 28](#_Toc111811411)

[Заключение 31](#_Toc111811412)

[Список использованных источников 32](#_Toc111811413)

# Введение

Людям, изучающим определенные предметы, например компьютерную графику, или же людям, работающим с компьютерной графикой, необходимо знать какие существуют графические форматы файлов. На данный момент существует множество Графических форматов. Человек, который только начал работать с графическими файлами попросту не запомнит множество их характеристик, таких как расширение, тип, сжатие, цветовые возможности, какими программами поддерживается данный формат. Но ведь все эти характеристики очень важны для правильного сохранения графической информации. Поэтому, я считаю, что справочник, содержащий краткую информацию о графических форматах будет очень полезен выше перечисленным группам людей.

# Задание

Формулировка: Справочник по форматам графических файлов содержит сведения: название формата, расширение файла, тип (растровый, векторный), какими программами поддерживается, цветовые возможности, сжатие. Программа должна осуществлять поиск информации по расширению файла, определять прикладную программу, обрабатывающую наибольшее количество графических форматов. Формировать список файлов различного типа, отсортированный по расширению и подчитывать файлов какого формата больше: растрового или векторного.

# Постановка задачи

Необходимо спроектировать и полностью реализовать программу «Справочник абитуриента». Требования к программе:

1. Формирование и редактирование файла типа “.txt” (необходимо реализовать: запись введенных данных в файл, редактирование данных, возможность загрузки данных из файла).
2. Использование отдельных диалоговых окон для ввода и проверки корректности введенных пользователем данных.
3. Вывод исходных данных (просмотр всех сведений) и реализация всех указанных в задании действий по сортировке и выборке данных.
4. Использование главного, а также контекстного меню.
5. Корректная обработка диалога работы с файлами (это значит, что программа не должна пользователю открывать несуществующие файлы, спрашивать пользователя о перезаписи данных в уже существующем файле и т.п.).
6. Результаты расчетов и поиска должны быть оформлены в соответствующем виде, например таблица представлений с необходимым перечнем колонок и строк.
7. Наличие в главном меню раздела «О программе», в котором указать версию программного продукта, дату внесения последних изменений в программе и координаты разработчика.
8. Наличие в главном меню раздела «Руководство пользователя, содержание которого загружается из текстового файла.
9. Наличие в главном меню раздела «Задание», содержащего постановку задачи.
10. Проверка сохранения сведений в файле при выходе из программы.
11. Сортировка выводимой информации по любому из полей записи (критерий сортировки выбирается пользователем).

Программа должна иметь дружественный интерфейс и быть понятной пользователю.

# Таблица спецификаций глобальных переменных

Основные глобальные переменные, объявленные в модуле Main, представлены в таблице 1.

Таблица 1 – Глобальные переменные модуля Main.

|  |  |  |
| --- | --- | --- |
| Наименование | Назначение | Тип Данных |
| isShown | Изменение функции кнопки «О программе» | bool |
| FileType | Ввод в поле «тип» | String^ |
| FileRash | Ввод в поле «Расширение файла» | String^ |
| FileName | Ввод в поле «Название формата» | String^ |
| FileProg | Ввод в поле «Поддерживаемые программы» | String^ |
| FileColor | Ввод в поле «Цветные возможности» | String^ |
| FileShat | Ввод в поле «Сжатие» | String^ |
| specToFind | Расширение для поиска | String^ |

# Схема иерархии модулей

Иерархия модулей представлена на рисунке 1 [1].



Рисунок 1 – Схема иерархии модулей

**Схема иерархии функций модуля Main**

Иерархия функций главного модуля Main представлена на рисунке 2.

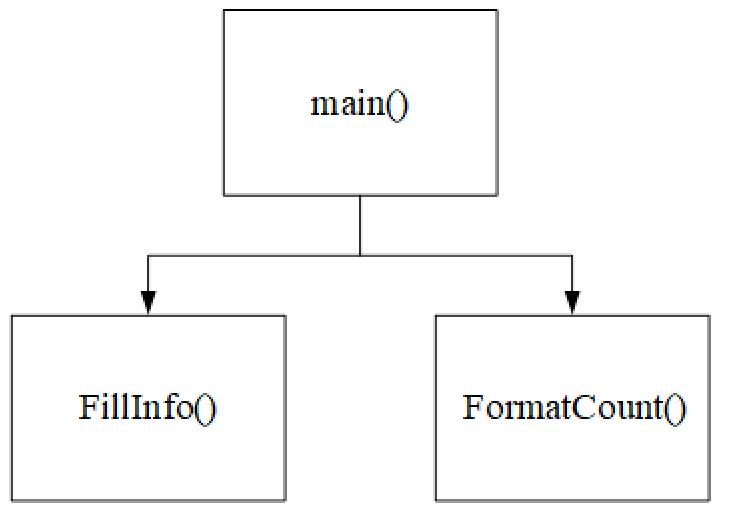


Рисунок 2 – Схема иерархии функций модуля Main

Спецификация функций модулей

Список всех функций модуля Main, их назначение, тип и входные параметры представлены в таблице 1.

Таблица 1 – Функции модуля Main

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Назначение | Входные параметры | Тип |
| FillInfo | Вывод результата поиска в таблицу | specToFind | void |
| FormatCount | Определение файлов какого формата больше | - | void |

Список всех функций модуля MyForm, их назначение, тип и входные параметры представлены в таблице 2.

Таблица 2 – Функции модуля MyForm

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Назначение | Входные параметры | Тип |
| btnBookOpen\_Click | Открыть справочник | Object^ Sender, EventArgs^ e | System::Void |
| btnTask\_Click | Открывает задание | Object^ Sender, EventArgs^ e | System::Void |
| btnAbout\_Click | Открывает сведения о программе | Object^ Sender, EventArgs^ e | System::Void |
| btnHowTo\_Click | Открывает руководство | Object^ Sender, EventArgs^ e | System::Void |
| btnExit\_Click | Закрывает программу | Object^ Sender, EventArgs^ e | System::Void |
| btnToMenu\_Click | Переход к главному меню | Object^ Sender, EventArgs^ e | System::Void |
| btnToAdd\_Click | Переход к добавлению записи | Object^ Sender, EventArgs^ e | System::Void |
| btnSearch\_Click | Поиск специальностей | Object^ Sender, EventArgs^ e | System::Void |
| btnAdd\_Click | Добавление записи в файл | Object^ Sender, EventArgs^ e | System::Void |
| IsChecked | Проверка на заполнение полей при добавлении | Object^ Sender, EventArgs^ e | System::Void |
| btnToBook\_Click | Переход к справочнику | Object^ Sender, EventArgs^ e | System::Void |

# Описание функций в содержательном аспекте

**Main:**

**Функция *FillInfo:***

Используется для вывода информации в таблицу. Функция осуществляет заполнение полей таблицы.

**Функция *FormatCount:***

Отвечает за определение файлов какого формата больше. Функция подсчитывает количество записей с тем или иным типом.

**MyForm:**

**Функция *btnBookOpen\_Click:***

Переход к справочнику. Функция вызывается нажатием на кнопку и открывает панель справочника

**Функция *btnTask\_Click:***

Функция перехода к заданию. Вызывается нажатием на кнопку и осуществляет открытие панели с заданием.

**Функция *btnAbout\_Click:***

Используется для отображения информации о программе. При нажатии на кнопку происходит отображение информации, при повторном нажатии – скрытие.

**Функция *btnHowTo\_Click:***

Производит переход к руководству пользователя. Нажатием на кнопку открывает раздел с руководством.

**Функция *btnExit\_Click:***

При нажатии на кнопку осуществляет выход из программы.

**Функция *btnToMenu\_Click:***

Используется для перехода к главному меню. Находится в соответствующих разделах и при нажатии производит переход к главному меню.

**Функция *btnToAdd\_Click:***

Используется для перехода к разделу с внесением данных. При нажатии производит переход к панели с добавлением специальности.

**Функция *btnSearch\_Click:***

Функция для поиска заданной специальности. При нажатии на кнопку происходит переход к таблице найденных специальностей, в которой отображены данные по каждому совпадению.

**Функция *btnAdd\_Click:***

Функция добавления записи в файл. Нажатие возможно только при условии, что все поля заполнены. При нажатии вносит данные в файл.

**Функция *IsChecked:***

Используется для проверки полей. Если все поля заполнены, то открывает доступ к нажатию на кнопку добавления

**Функция *btnToBook\_Click:***

При нажатии на кнопку осуществляет переход к справочнику из соответствующих разделов.

# Алгоритмы

**Main:**

**Функция *FillInfo:***

1. Чтение файлов “Data1.txt”.
2. Пока не дошли до последней строки файла:
   1. Запись данных в таблицу.
3. Закрываем файл “Data1.txt”.
4. Чтение файлов “Data2.txt”.
5. Пока не дошли до последней строки файла:
   1. Запись данных в таблицу.
6. Закрываем файл “Data2.txt”.

**Функция SpecCount*:***

1. Чтение файла “Data1.txt”.
2. Подсчет количества записей.
3. Закрываем файл “Data1.txt”.
4. Чтение файла “Data2.txt”.
5. Подсчет количества записей.
6. Закрываем файл “Data2.txt”.
7. Сравнение количества записей в файлах.
8. Передача ответа в переменную FormatName.

**MyForm:**

**Функция *btnBookOpen\_Click:***

1. Переход на панель pnlBook.

**Функция *btnTask\_Click:***

1. Переход на панель pnlTask.

**Функция *btnAbout\_Click:***

1. Переход на панель pnlAbout.

**Функция *btnHowTo\_Click:***

1. Переход на панель pnlHowTo.

**Функция *btnExit\_Click:***

1. Выход из программы функцией Exit().

**Функция *btnToMenu\_Click:***

1. Переход на панель pnlMenu.

**Функция *btnToAdd\_Click:***

1. Переход на панель pnlAdd.

**Функция *btnSearch\_Click:***

1. Передаем в функцию FillInfo значение введенного расширения.
2. Переход на панель pnlShowInfo.

**Функция *btnAdd\_Click:***

1. Проверка, что поля tbFormName, tbRushFile, tbProgramsSup, tbColor, tbSzhat заполнены, иначе выводим сообщение об ошибке.
2. Если выбран тип файла “растровый”, то производим запись данных в файл “Data1.txt”, иначе запись в файл “Data2.txt”.
3. Очищаем поля для дальнейшего ввода.

**Функция *IsChecked:***

1. Если в одном из полей TypeVect, TypeRustr, есть хотя бы одно значение True, то открываем доступ к кнопке btnAdd.

**Функция *btnToBook\_Click:***

1. Переход на панель pnlBook.

# Таблица тестов

Проверка программы на корректную работу, попытки выявить неожиданные ошибки и ответ программы на эти действия представлен в таблице 7.

Таблица 7 – Таблица тестов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Что проверяем | Входные данных | Ожидание | Результат |
| 1 | Открытие справочника | Нажатие кнопки “Открыть справочник” | Программа перейдет к меню справочника и покажет университет с наибольшим количеством специальностей | Открытие справочника |
| 2 | Поиск файлов с расширением “расширение” | “расширение” | Информация о форматах файлов с данным расширением | Программа выдала список файлов с введенным расширением |
| 3 | Открытие вкладки о программе | Нажатие кнопки “О программе” | Программа перейдет к вкладке информации о программе | Программа перешла к вкладке |
| 4 | Открытие вкладки задание | Нажатие кнопки “Задание” | Программа перейдет к вкладке задание | Программа перешла к вкладке |
| 5 | Открытие вкладки руководство | Нажатие кнопки “Руководство” | Программа перейдет к вкладке руководства | Программа перешла к вкладке |

# Листинг

**Модуль Main**

|  |
| --- |
| #include "MyForm.h"  using namespace System;  using namespace System::Windows::Forms;  [STAThread]  void main(array<String^>^ args) {  Application::EnableVisualStyles();  Application::SetCompatibleTextRenderingDefault(false);  Kks::MyForm form;  Application::Run(% form);  } |

**Модуль MyForm**

|  |
| --- |
| #pragma endregion  bool isShown = false;  String^ ChosenType;  String^ specToFind;  void FillInfo(String^ chosenSpec) {// поиск по параметру  StreamReader^ sr = gcnew StreamReader("data1.bin");  String^ lineReader;  String^ FileRash;  String^ FileName;  String^ FileProg;  String^ FileColor;  String^ FileShat;  String^ FileType;  lineReader = sr->ReadLine();  while (lineReader) {  if (lineReader == chosenSpec) {  FileRash += lineReader + "\r\n";  FileName += sr->ReadLine() + "\r\n";  FileProg += sr->ReadLine() + "\r\n";  FileColor += sr->ReadLine() + "\r\n";  FileShat += sr->ReadLine() + "\r\n";  FileType += sr->ReadLine() + "\r\n";  }  else {  sr->ReadLine();  sr->ReadLine();  sr->ReadLine();  sr->ReadLine();  sr->ReadLine();  }  lineReader = sr->ReadLine();  }  sr->Close();  StreamReader^ sr2 = gcnew StreamReader("data2.bin");  lineReader = sr2->ReadLine();  while (lineReader) {  if (lineReader == chosenSpec) {  FileRash += lineReader + "\r\n";  FileName += sr2->ReadLine() + "\r\n";  FileProg += sr2->ReadLine() + "\r\n";  FileColor += sr2->ReadLine() + "\r\n";  FileShat += sr2->ReadLine() + "\r\n";  FileType += sr2->ReadLine() + "\r\n";  }  else {  sr2->ReadLine();  sr2->ReadLine();  sr2->ReadLine();  sr2->ReadLine();  sr2->ReadLine();  }  lineReader = sr2->ReadLine();  }  sr2->Close();  tbName->Text = FileRash;  tbRash->Text = FileName;  tbProg->Text = FileProg;  tbCol->Text = FileColor;  tbShat->Text = FileShat;  tbType->Text = FileType;  }  void FormatCount() { //переписать полностью функцию (иначе не работает)  String^ lineReader;  int rust = 0;  int vect = 0;  StreamReader^ srData = gcnew StreamReader("data1.bin");  lineReader = srData->ReadLine();  while (lineReader) {  srData->ReadLine();  srData->ReadLine();  srData->ReadLine();  srData->ReadLine();  srData->ReadLine();  lineReader = srData->ReadLine();  rust += 1;  }  srData->Close();  StreamReader^ srtrash = gcnew StreamReader("data2.bin");  lineReader = srtrash->ReadLine();  while (lineReader) {  srtrash->ReadLine();  srtrash->ReadLine();  srtrash->ReadLine();  srtrash->ReadLine();  srtrash->ReadLine();  lineReader = srtrash->ReadLine();  vect += 1;  }  srtrash->Close();  if (vect > rust) {  FormatName->Text = "векторный";  }  else if (vect < rust) {  FormatName->Text = "растровый";  }  else {  FormatName->Text = "Равное кол-во";  }    }  private: System::Void btnBookOpen\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlBook->Show();  FormatCount();  }  private: System::Void btnTask\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlTask->Show();  String^ lineReader;  StreamReader^ srtask = gcnew StreamReader("Task.txt");  lineReader = srtask->ReadLine();  tbTask->Text = lineReader;  srtask->Close();  }  private: System::Void btnAbout\_Click(System::Object^ sender, System::EventArgs^ e) {  if (isShown) {  lblinfo1->Hide();  lblinfo2->Hide();  isShown = false;  btnAbout->Text = "О программе";  }  else {  lblinfo1->Show();  lblinfo2->Show();  isShown = true;  btnAbout->Text = "О программе (скрыть)";  }  }  private: System::Void btnExit\_Click(System::Object^ sender, System::EventArgs^ e) {  Application::Exit();  }  private: System::Void btnTaskMenu\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlTask->Hide();  }  private: System::Void HowToMenu\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlHowTo->Hide();  }  private: System::Void btnHowTo\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlHowTo->Show();  String^ lineReader;  StreamReader^ srhow = gcnew StreamReader("HowTo.txt");  lineReader = srhow->ReadLine();  tbHowTo->Text = lineReader;  srhow->Close();  }  private: System::Void btnBookMenu\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlBook->Hide();  }  private: System::Void btnToAdd\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlBook->Hide();  pnlAdd->Show();  btnAdd->Enabled = false;  }  private: System::Void btnSearch\_Click(System::Object^ sender, System::EventArgs^ e) {  specToFind = textBox2->Text;  FillInfo(specToFind);  pnlBook->Hide();  pnlShowResult->Show();  }  private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlBook->Hide();  }  private: System::Void btnAdd\_Click(System::Object^ sender, System::EventArgs^ e) {  if (!tbFormName->TextLength) {  MessageBox::Show("Название формата не введено", "Error", MessageBoxButtons::OK, MessageBoxIcon::Error);  }  else {  if (!tbRushFile->TextLength) {  MessageBox::Show("Расширение файла не введено", "Error", MessageBoxButtons::OK, MessageBoxIcon::Error);  }  else {  if (!tbProgramsSup->TextLength) {  MessageBox::Show("Поддерживаемые программы не введены", "Error", MessageBoxButtons::OK, MessageBoxIcon::Error);  }  else {  if (!tbColor->TextLength) {  MessageBox::Show("Цветовые возможности не введены", "Error", MessageBoxButtons::OK, MessageBoxIcon::Error);  }  else {  if (!tbSzhat->TextLength) {  MessageBox::Show("Сжатие не указано", "Error", MessageBoxButtons::OK, MessageBoxIcon::Error);  }  else { // добавить сортировку по расширению  if (ChosenType == "Растровый") {    StreamWriter^ sw = File::AppendText("data1.bin");  sw->WriteLine(tbRushFile->Text);  sw->WriteLine(tbFormName->Text);  sw->WriteLine(tbProgramsSup->Text);  sw->WriteLine(tbColor->Text);  sw->WriteLine(tbSzhat->Text);  sw->WriteLine(ChosenType);  sw->Flush();  sw->Close();  }  else {  StreamWriter^ sw = File::AppendText("data2.bin");  sw->WriteLine(tbRushFile->Text);  sw->WriteLine(tbFormName->Text);  sw->WriteLine(tbProgramsSup->Text);  sw->WriteLine(tbColor->Text);  sw->WriteLine(tbSzhat->Text);  sw->WriteLine(ChosenType);  sw->Flush();  sw->Close();  }  }  }  }  }  }  tbFormName->Clear();  tbRushFile->Clear();  tbProgramsSup->Clear();  tbColor->Clear();  tbSzhat->Clear();  typeRastr->Checked = false;  typeVect->Checked = false;  FormatCount();  }  private: System::Void btnAddMenu\_Click(System::Object^ sender, System::EventArgs^ e) {  pnlAdd->Hide();  pnlBook->Show();  }  private: System::Void typeRastr\_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {  if (typeRastr->Checked) {  ChosenType = "Растровый";  typeVect->Enabled = false;  btnAdd->Enabled = true;    }  else {  btnAdd->Enabled = false;  typeVect->Enabled = true;  }  }  private: System::Void typeVect\_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {  if (typeVect->Checked) {  ChosenType = "Векторный";  typeRastr->Enabled = false;  btnAdd->Enabled = true;    }  else {  btnAdd->Enabled = false;  typeRastr->Enabled = true;  }  }  private: System::Void btnResultsToBook\_Click(System::Object^ sender, System::EventArgs^ e) {  tbRash->Text = "";  tbName->Text = "";  tbCol->Text = "";  tbProg->Text = "";  tbShat->Text = "";  tbType->Text = "";  pnlShowResult->Hide();  pnlBook->Show();  } |

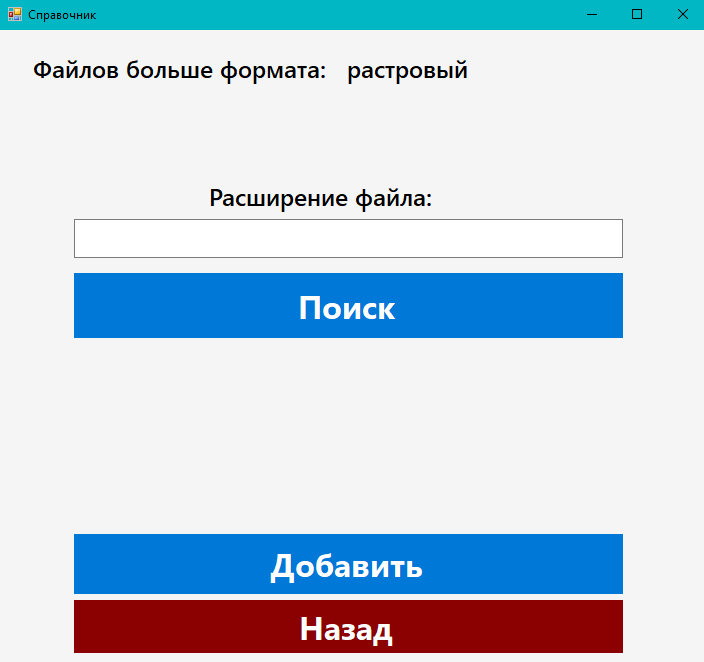
# 

# Результаты тестирования

**Тест 1**

Результат работы программы в ответ на тест 1 представлен на рисунке 3.

Рисунок 3 – Результат Теста 1



**Тест 2**

Результат работы программы в ответ на тест 2 представлен на рисунке 4.

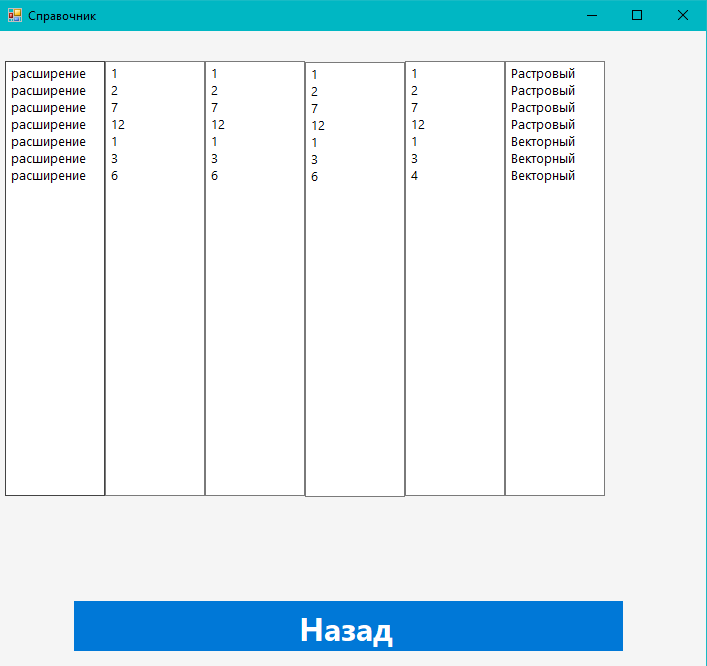


Рисунок 4 – Результат Теста 2

**Тест 3**

Результат работы программы в ответ на тест 3 представлен на рисунке 5.

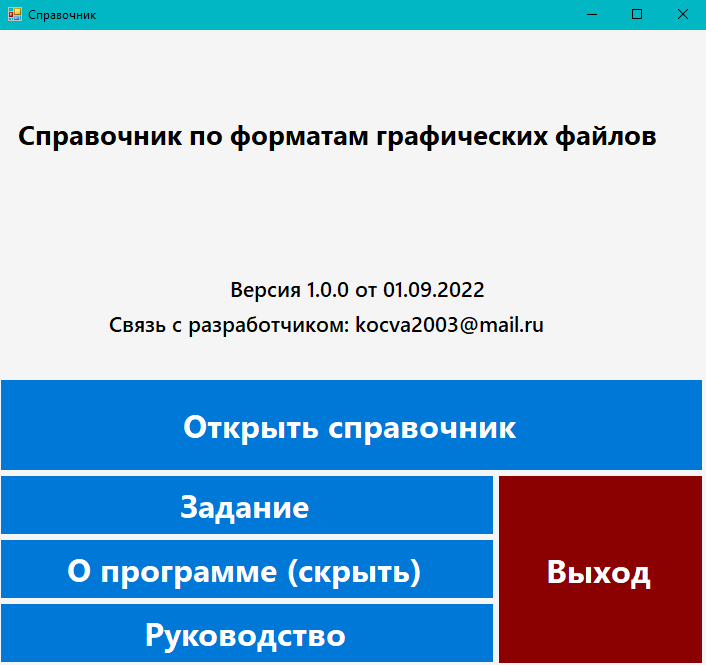


Рисунок 5 – Результат Теста 3

**Тест 4**

Результат работы программы в ответ на тест 4 представлен на рисунке 6.

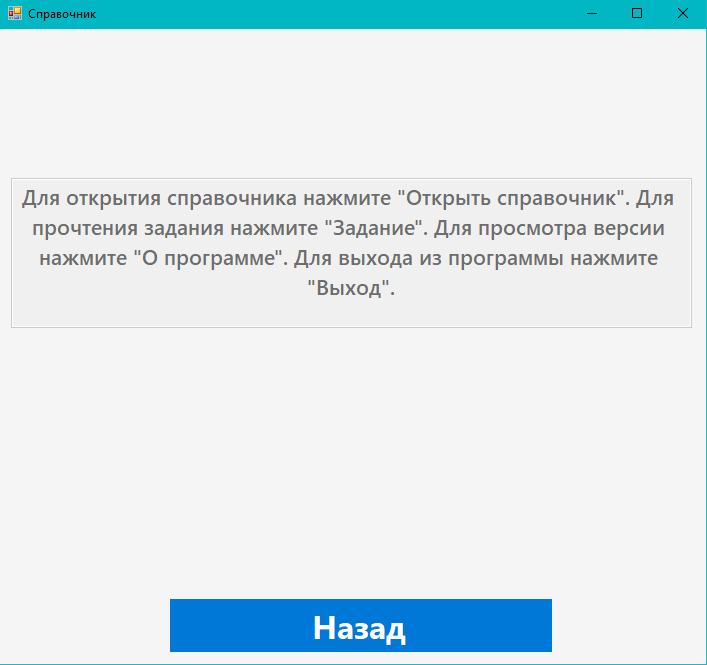


Рисунок 6 – Результат Теста 4

**Тест 5**

Результат работы программы в ответ на тест 5 представлен на рисунке 7.

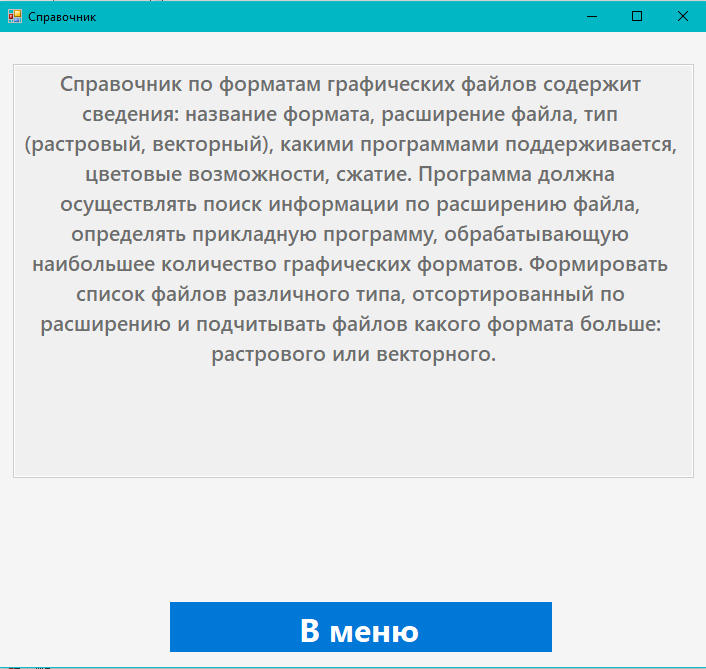


Рисунок 7 – Результат Теста 5

**Результаты работы программы**

**Результат 1**

Результат поиска специальности.

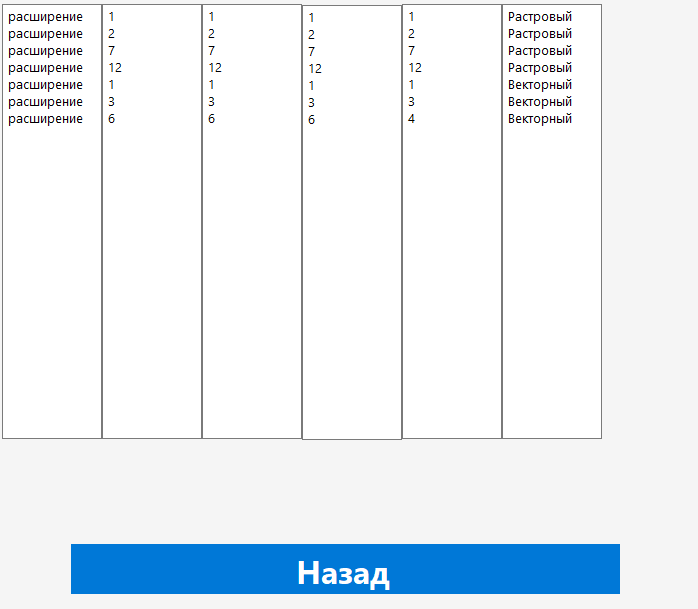


Рисунок 8 – Результат работы программы 1

**Результат 2**

Результат определения файлов какого формата больше.

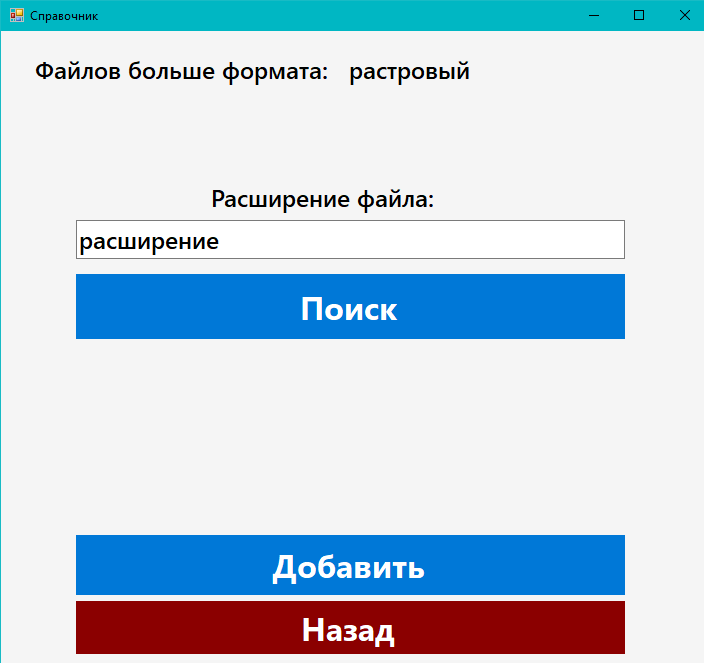


Рисунок 9 – Результат работы программы 2

# Проект руководства пользователя

# 1. Общие сведения о программе

Программа для компьютера под управлением операционной системы Windows представляет собой справочник по форматам графических файлов.

# 2. Установка, запуск

Для установки и запуска ПО необходимо загрузить репозиторий «https://github.com/Kocva/kurs» с GitHub и запустить исполняемый файл “kks.exe”. После успешного запуска приложения Вы попадете в главное меню, где вам будет предложен набор функций (см. рисунок 11).

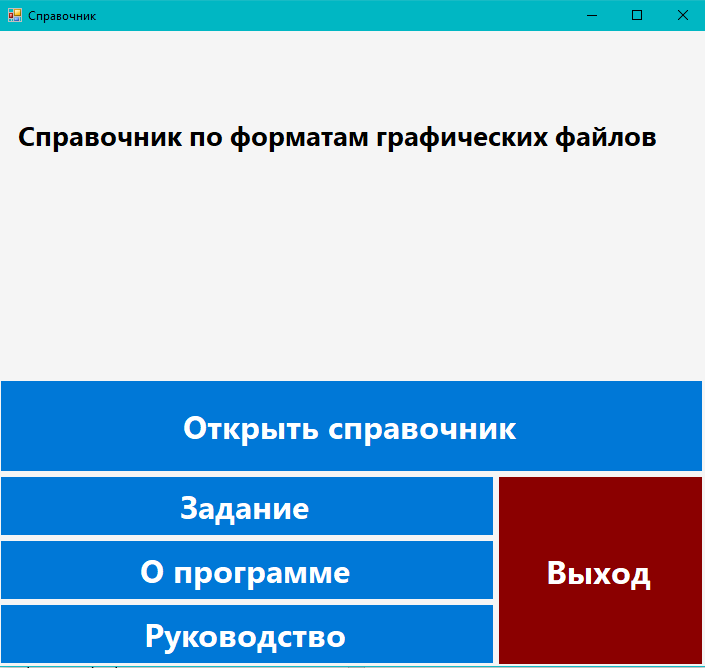
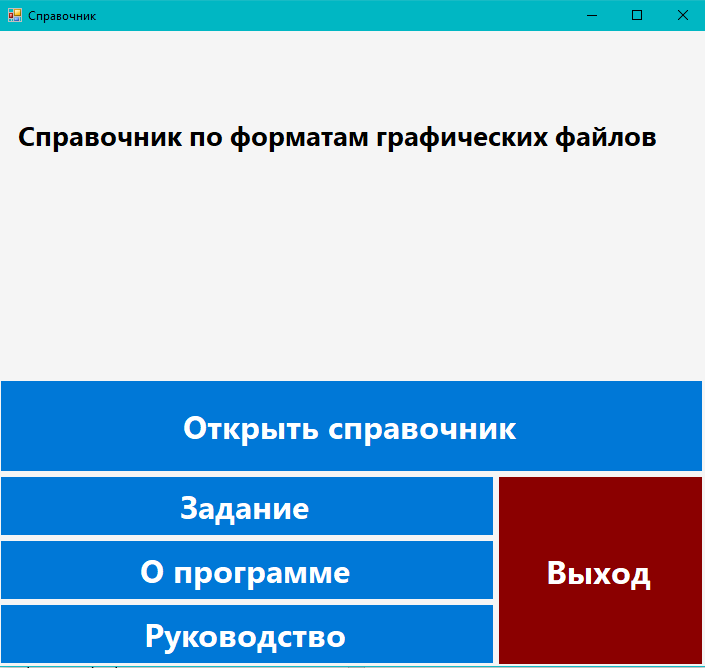


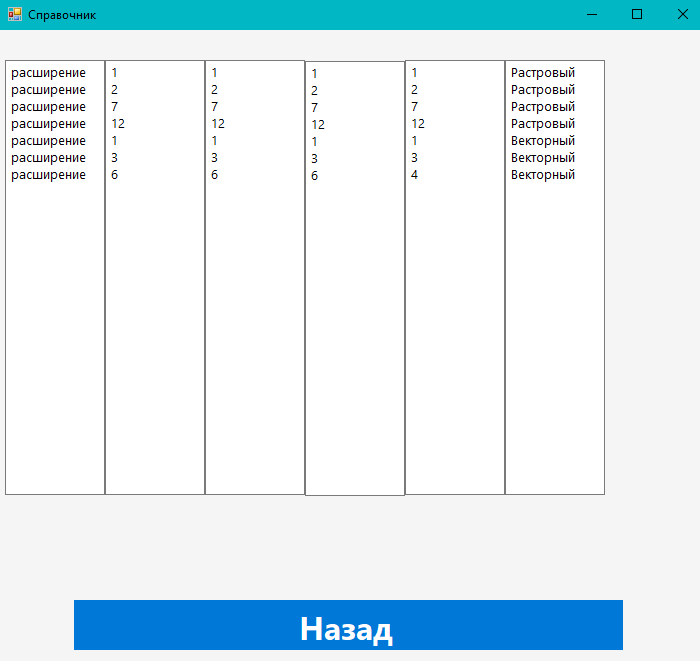
Рисунок 11 – Главное меню

# 3. Основные возможности программы

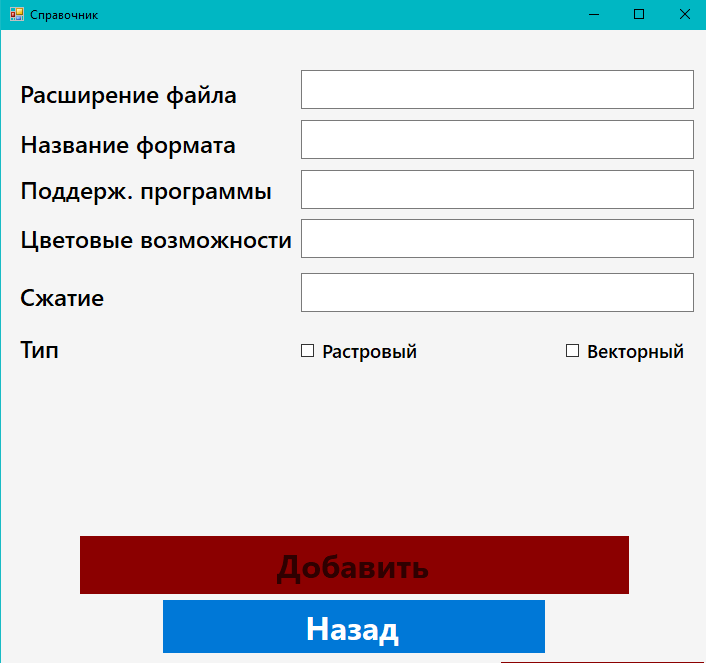
Для работы со справочником нажмите на “Открыть справочник”. Здесь вы увидите файлов какого формата больше, сможете найти нужный формат файла или же добавить новый формат.

  
Рисунок 12 – Меню справочника

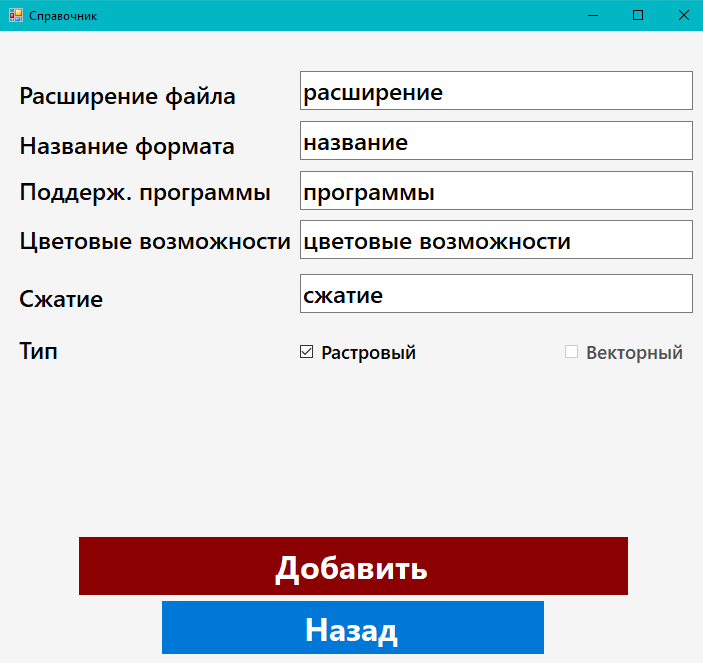
1. При нажатии на кнопку “Назад” вы вернетесь в главное меню.
2. При введении данных в поле “Расширение” и дальнейшем нажатии на поиск представится отсортированный по типу список файлов с выбранным расширением.

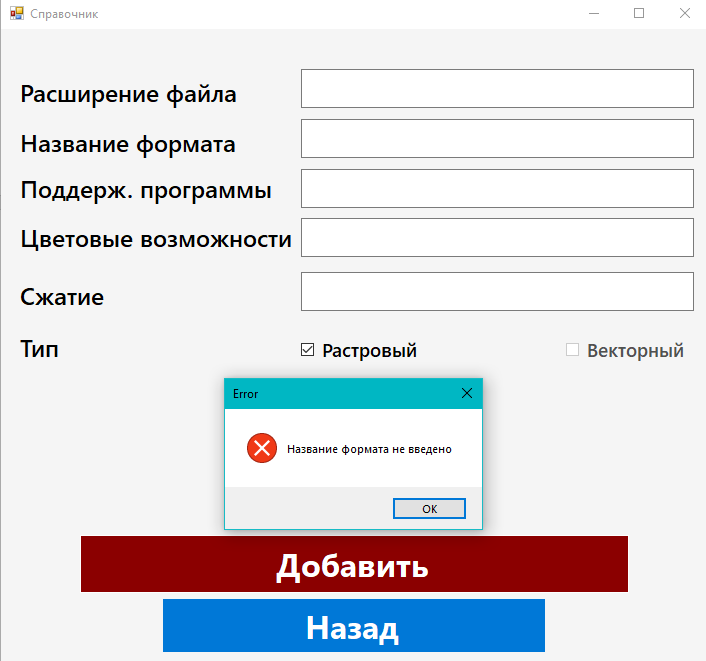
  
  
Рисунок 13 – Результат поиска

1. Здесь при нажатии на кнопку “Назад” вы перейдете в меню справочника.
2. При нажатии на кнопку добавить вы перейдете к добавлению специальности.

  
Рисунок 14 – Окно добавления специальности

1. После ввода всех данных у вас будет возможность добавления формата

  
Рисунок 15 – Экран отображения измененной таблицы

  
Рисунок 16 – Программа предупредит вас о некорректности введенного значения

# Заключение

В итоге я получил полностью функционирующую программу, которая может справится с данными ей задачами такими как: Ввод информации, обработка данных, работа с бинарными файлами, хранение информации и многими другими. В качестве среды разработки была выбрана программа Visual Studio. При выполнении работы изучены и закреплены на практике следующие навыки:

* разработки структур данных:
* работы с типизированными файлами (создание, открытие, вывод и корректировка);
* использования диалоговых окон для создания дружественного интерфейса (контроль входных данных, наличие комментариев и др.);
* проектирования и создания полностью законченных приложений.

Цели, поставленные на выполнение курсовой работы, были достигнуты.

# Список использованных источников

1. Иванов В. Б. Прикладное программирование на C/C++: с нуля до мультимедийных и сетевых приложений. М.: «СОЛОН-Пресс». 2008. – 240 с. [Электронный ресурс]. – Режим доступа: http://e.lanbook.com/books/ element.php&pl1\_id=13740
2. Хохлов Д.Г. Методы программирования на языке С: практикум Ч.1. «Бином. Лаборатория знаний». 2014. – 336 с. [Электронный ресурс]. – Режим доступа: http://e.lanbook.com/books/element.php ?pl1\_id=50555
3. Хохлов Д.Г. Методы программирования на языке С: практикум Ч.2. «Бином. Лаборатория знаний». 2014. – 377 с. [Электронный ресурс]. – Режим доступа: http://e.lanbook.com/books/element.php ?pl1\_id=50555