Министерство образования Республики Беларусь

Учреждение образования

“Могилевский государственный университет имени А.А.Кулешова”

**ФАЙЛОВЫЕ МЕНЕДЖЕРЫ**

Курсовой проект

по дисциплине «Операционные системы и системное программирование»

студента 3 курса, специальности 1-40 01 01

«Программное обеспечение информационных технологий»

группы 5142201 дневной формы обучения

Никиты Дмитриевича Куриловича

Научный руководитель

Ассистент кафедры ПОИТ

[Марина](https://fme.msu.by/index.php/home/personalii/67-kozhurenko-natalya-vladimirovna) Васильевна Мироненко

Могилев

2022

**ОГЛАВЛЕНИЕ**

ВВЕДЕНИЕ…………………………………………………………………….3  
ГЛАВА 1 АНАЛИТИЧЕСКИЙ ОБЗОР ЛИТЕРАТУРЫ……………………4

1.1ОБЗОР ЛИТЕРАТУРЫ………………………………………………4

1.2 ОБЗОР АНАЛОГОВ………………………………………………….6

1.2.1 ПОНЯТИЕ О ФАЙЛОВОМ МЕНЕДЖЕРЕ…………………6

1.2.2 ПРОВОДНИК WINDOWS……………………………………7

1.2.3 TOTAL COMMANDER……………………………………….8

1.2.4 XYPLORER PRO……………………………………………...10  
ГЛАВА 2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ…....13

2.1 КЛАСС ГЛАВНОЙ ФОРМЫ………………………………………..13

2.2 МЕТОДЫ КЛАССА………………………………………………….14  
ГЛАВА 3 ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ……….19  
ЗАКЛЮЧЕНИЕ………………………………………………………………..30  
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ………………………….31

**ВВЕДЕНИЕ**

Ни одна операционная система на сегодняшний день не может обойтись без удобного файлового менеджера. Огромное количество нарастающих с каждым днем данных нуждается в грамотной структуризации и разделении. Не зря все современные операционные системы, как правило, включают в свой состав, в первую очередь, именно файловый менеджер, как неотъемлемую часть самой ОС и необходимый компонент для реализации всех возможностей по доступу к файловой системе. При этом такой доступ должен удовлетворять многим, зачастую противоположным условиям, к которым относятся: возможность поиска и отображения нужной информации, полнота операций над этими данными, гарантированное исключение ошибок при этих операциях, простота, понятность и так далее.

Наличие файлового менеджера в самих ОС не останавливает пользователей в поисках программы «для себя». В данной программе сделана попытка объединить только самые нужные для пользователя функции по работе с файловой системой в наглядном и простом виде.

Целью курсового проекта является создание файлового менеджера на языке C#.

Задачи:

1. Изучить материал, который необходим для выполнения работы
2. Разработать программу
3. Протестировать программу

Для выполнения поставленной задачи, необходимо современное оборудование, которое позволит комфортно работать. Программа разрабатывалась на операционной системе Windows 11, в среде разработки Visual Studio 2022.

Объект исследования: использование среды Visual Studio 2022, возможностей языка С# и его библиотек для решения поставленной задачи.

Предмет исследования: разработка файлового менеджера.

# **ГЛАВА 1**

# **АНАЛИТИЧЕСКИЙ ОБЗОР ЛИТЕРАТУРЫ**

* 1. **Обзор литературы**

**Мюллер Д. П. C# для чайников**

Даже если вы никогда не имели дела с программированием, эта книга поможет вам освоить язык C# и научиться писать на нем программы любой сложности. Для читателей, которые уже знакомы с каким-либо языком программирования, процесс изучения C# только упростится, но иметь опыт программирования для чтения книги совершенно необязательно. Из этой книги вы узнаете не только о типах, конструкциях и операторах языка C#, но и о ключевых концепциях объектно-ориентированного программирования, реализованных в этом языке, который в настоящее время представляет собой один из наиболее приспособленных для создания программ для Windows инструментов.

Если вы в начале большого пути в программирование, смелее покупайте эту книгу: она послужит вам отличным путеводителем, который облегчит ваши первые шаги на этом длинном, но очень увлекательном пути. Узнайте, как создать консольное приложение и что такое делегаты, события и интерфейсы.C# - мощный язык программирования, который стал любимым инструментом программистов, работающих с Visual Studio, и эта книга поможет вам быстро и безболезненно освоить новейшую его версию. Вы научитесь создавать приложения для Windows, использовать графику, потоки, контейнеры, базы данных и многое другое, узнаете, что такое .NET Framework, полиморфизм, наследование и обобщенное программирование.

**Джеффри Рихтер, CLR via C#**

Эта книга, выходящая в четвертом издании и уже ставшая классическим учебником по программированию, подробно описывает внутреннее устройство и функционирование общеязыковой исполняющей среды (CLR) Microsoft .NET Framework версии 4.5. Написанная признанным экспертом в области программирования Джеффри Рихтером, много лет являющимся консультантом команды разработчиков .NET Framework компании Microsoft, книга научит вас создавать по-настоящему надежные приложения любого вида, в том числе с использованием Microsoft Silverlight, ASP.NET, Windows Presentation Foundation и т.д. Четвертое издание полностью обновлено в соответствии со спецификацией платформы .NET Framework 4.5, а также среды Visual Studio 2012 и C# 5.0.

**Мэтью Мак-Дональд, Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0 для профессионалов.**

Книга "WPF: Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0 для профессионалов" представляет собой исчерпывающее авторитетное руководство по внутренней работе WPF. Благодаря серьезным примерам и практическим рекомендациям, вы изучите все, что необходимо знать для профессионального использования WPF. Книга начинается с построения прочного фундамента из элементарных концепций, подкрепленного существующими знаниями языка C#. Затем предлагается обсуждение сложных концепций с их демонстрацией на полезных примерах, которые подчеркивают получаемую экономию времени и затраченных усилий.

Платформа Windows Presentation Foundation (WPF) от Microsoft предоставляет инфраструктуру разработки, предназначенную для построения высококачественных пользовательских интерфейсов для операционной системы Windows. Она сочетает в себе насыщенный контент из широкого диапазона источников и позволяет получить неограниченный доступ ко всей вычислительной мощи компьютера, функционирующего под управлением Windows. В книге "WPF: Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0 для профессионалов" подробно рассматриваются следующие темы. Фундаментальные основы программирования для WPF, начиная с XAML и заканчивая элементами управления и потоком данных. Разработка реалистичных приложений, позволяющих увидеть навигацию, локализацию и развертывание в действии. Исследование расширенных элементов управления пользовательского интерфейса, которые предлагаются WPF. Изучение способов управления документами внутри WPF: компоновка текста, вывод на печать и упаковка документов. Использование графики и мультимедиа для совершенствования приложений. Книга рассчитана на разработчиков, которые впервые сталкиваются с WPF. Опыт программирования на C# и знание базовой архитектуры .NET поможет быстрее разобраться с примерами, но все необходимые концепции кратко объясняются с самого начала.

**Пахомов Б.И. C# для начинающих**

Книга является руководством для начинающих по разработке приложений на языке C#. Приведены общие сведения о языке C# и платформе .NET. Рассмотрены базовые типы данных, переменные, функции и массивы. Показана работа с датами и перечислениями. Описаны основные элементы и конструкции языка: классы, интерфейсы, сборки, манифесты, пространства имен, коллекции, обобщения, делегаты, события и др. Приведены сведения о процессах и потоках Windows, а также примеры организации работы в многопоточном режиме. Рассмотрено создание консольных приложений, приложений типа Windows Forms и приложений для работы с базами данных. В качестве среды разработки в книге использован бесплатный пакет SharpDevelop. Для начинающих программистов.

**Чарльз Петцольд, Программирование с использованием microsoft windows forms**

В этой книге подробно рассказывается о создании программ для Microsoft Windows с использованием языка C# и библиотеки классов Windows Forms, входящей в Microsoft .NET Framework 2.0. Вы научитесь создавать новые нестандартные и комбинировать существующие элементы управления, а также разрабатывать панели инструментов, меню и строки состояния, используя появившиеся в .NET Framework 2.0 новинки, узнаете о новом механизме динамического размещения элементов управления на форме и о привязке элементов управления к данным. Книга состоит из 7 глав и адресована программистам, стремящимся освоить новые возможности, связанные с выходом .NET Framework 2.0, а также разработчикам, имеющим значительный опыт программирования для Windows и желающим максимально быстро и эффективно освоить Windows Forms.

## Обзор аналогов

**1.2.1 Понятие о файловом менеджере**

Файловый менеджер - компьютерная программа, предоставляющая интерфейс пользователя для работы с файловой системой и файлами. Файловый менеджер позволяет выполнять наиболее частые операции над файлами - создание, открытие/проигрывание/просмотр, редактирование, перемещение, переименование, копирование, удаление, изменение атрибутов и свойств, поиск файлов и назначение прав. Помимо основных функций, многие файловые менеджеры включают ряд дополнительных возможностей, например, таких как работа с сетью (через FTP, NFS и т. п.), резервное копирование, управление принтерами и пр.

Файловые менеджеры обеспечивают более удобный и наглядный способ общения с ПК по сравнению с операционной системой (ОС). Одна из самых известных первых программных оболочек называлась Norton Commander. Ее разработал американский программист Питер Нортон. Файловый менеджер наглядно показывал на экране всю файловую структуру компьютера: диски, каталоги и файлы. С такой программой не надо было набирать сложные команды MS-DOS в командной строке. Файлы можно было копировать, перемещать, разыскивать, удалять, сортировать, изменять, запускать, пользуясь всего лишь несколькими клавишами.

Выделяют различные типы файловых менеджеров:

1) навигационные и пространственные - иногда поддерживается переключение между этими режимами;

2) двупанельные – в общем случае имеют две равноценных панели для списка файлов, дерева каталогов и т. п.

Таким образом, файловые менеджеры - специальные программы, предназначенные для облегчения общения пользователя с командами операционной системы. Это программы, запускаемые под управлением ОС, занимающие промежуточное положение между ОС и прикладными программами и служащие для интеграции прикладных пакетов.

Наиболее распространенным файловыми менеджерами в настоящее время в среде операционной системы Windows являются Проводник Windows, Total Commander и XYplorer Pro.

* + 1. **Проводник Windows**

Проводник Windows - это приложение, реализующее графический интерфейс доступа пользователя к файлам в операционной системе Microsoft Windows. В настоящее время фактически является основой графической оболочки пользователя Windows.

Иногда Проводником называют его часть, предназначенную для манипуляции файлами. Её можно вызвать двойным щелчком по иконке «Мой компьютер», запустить из «меню Пуск», воспользоваться клавиатурным сокращением Win + E .

Последующие версии Windows, и в некоторых случаях, версии Internet Explorer, добавляли новые возможности, убирали другие, постепенно формируя из простой системы навигации по файлам полнофункциональную задаче-ориентированную систему управления файлами.

Окно Проводника состоит из двух панелей. Левая панель показывает информационные ресурсы, представленные в виде иерархического дерева. Правая панель показывает содержимое текущей папки.

Процесс перемещения по папкам с целью открытия необходимой называют навигацией. Проводник является инструментом поиска – навигатором. Чтобы эффективно работать в среде Проводника, нужно знать приемы навигации в нем.

Чтобы открыть объект, находящийся внутри папки, надо выполнить одно из следующих действий:

* Выполнить двойной щелчок на значке объекта;
* Выделить объект щелчком мыши и дать команду Файл/Открыть;
* В контекстном меню объекта выбрать пункт Открыть.

Если объект – программа, то Windows запустит ее. Если объект – документ, то Windows запустит программу, с помощью которой он создавался, и откроет в ее окне документ.

Проводник позволяет не только просматривать существующие объекты, но и создавать новые.

**1.2.3 Total Commander**

Total Commander — файловый менеджер с закрытым исходным кодом, работающий на платформе Microsoft Windows.

К основным функциям относится:

* Операции с файлами и папками
* Работа с архивами (Внутренний упаковщик, Внутренние распаковщики для форматов ZIP, ARJ, LZH, gzip, tar, RAR и ACE)
* Разрезание и сборка файлов большого объёма
* Групповое переименование файлов и синхронизация каталогов
* Встроенный FTP-клиент, с поддержкой SSL/TLS
* Поддержка Drag&Drop при помощи мыши
* Расширенный поиск файлов, включая поиск текста в любых файлах, в том числе на нескольких дисках и FTP; поиск дубликатов файлов; поиск внутри архивов
* Встроенный просмотрщик файлов (lister) для просмотра файлов любого размера в шестнадцатеричном, двоичном или текстовом формате с использованием набора символов «DOS», «ANSI»/«Windows», Unicode, UTF-8 и других
* клавиатурные комбинации для просмотра, редактирования, копирования и удаления файлов (возможность полного управления без использования мыши)

Как и любой другой файловый менеджер, восходящий корнями к Norton Commander, TC представляет собой две панели, в каждой из которых можно отображать содержимое различных дисков и папок. Это удобно тем, что позволяет легко переносить и копировать файлы и целые папки из одного места в другое. Кроме того, становится удобно сравнивать содержимое различных папок между собой и синхронизировать содержимое каталогов. Кстати, TC может сравнивать содержимое отдельных файлов.

Практически любая операция, поддерживаемая Total Commander (перемещение, удаление, копирование, переименование и т. д.), может выполняться над отдельным файлом, группой файлов или целой папкой (включая ее подкаталоги). При этом процедуры выбора и выделения файлов, для последующего выполнения каких-либо действий над ними, очень просты. Пользователь может выделять файлы, как и в Проводнике Windows, при помощи мыши и клавиш Shift и Ctrl, а также с помощью клавиш Insert и вспомогательной цифровой клавиатуры, как это было реализовано в Norton Commander. Кроме того, доступно выделение с применением маски. Например, пользователь может указать маску \*.docx для того, чтобы в текущей папке были выделены все документы MS Word.

Total Commander обладает очень мощным инструментом поиска файлов. Можно искать файлы по маске (с применением регулярных выражений) в любых указанных папках и подкаталогах. Более того, файл-менеджер обнаружит отдельные файлы, соответствующие условиям поиска, даже внутри архивов всех поддерживаемых форматов. Можно искать файлы:

* содержащие (или не содержащие) указанный пользователем текст;
* созданные и изменявшиеся в указанный пользователем интервал времени;
* определенного размера и обладающие определенными атрибутами

Файловый менеджер облегчает работу с архивированными файлами. В него встроены распаковщики нескольких архивных форматов: ZIP, ARJ, LZH, RAR, TAR, GZ, CAB и ACE. Поэтому для извлечения файлов из архивов перечисленных форматов никакого дополнительного программного обеспечения не требуется.

Кроме того, в Total Commander встроены несколько упаковщиков, поэтому он позволяет создавать архивы форматов ZIP, TAR, GZ и TGZ без установки соответствующих программ-архиваторов. Если же пользователю потребуется создать ARJ, RAR, LHA, ACE или UC2-архив, ему достаточно указать файл-менеджеру местоположение соответствующих архиваторов, чтобы надлежащий архив был создан. Таким образом, при любых операциях с архивами пользователю не нужно покидать Total Commander.

Total Commander показывает объем файлов или их группы, а также место, которое эти файлы фактически занимают на диске.

В Total Commander встроена мощная утилита переименования групп файлов. Пользователь может вставить в имя и расширение каждого файла собственную текстовую строку, текущую дату и время, счетчик, чтобы пронумеровать файлы с одинаковыми именами и другую информацию. При этом доступны поиск и замена отдельных фрагментов имени и расширения. Кстати, кроме переименования файлов, Total Commander позволяет изменять и метки жестких дисков.

Интерфейс программы довольно удобен и предоставляет различные возможности настройки для пользователей разного уровня подготовки. В Интернете пользователь легко найдет огромное количество дополнительных модулей (плагинов), которые еще больше расширят список функций Total Commander.

* + 1. **XYplorer Pro**

XYplorer Pro — профессиональный файловый менеджер для операционной системы Windows, который можно использовать в качестве дополнения или замены проводника Windows.

XYplorer поставляется с множеством простых в использовании функций, которые делают жизнь намного проще, когда дело доходит до повседневных операций. Например, пользователи заметят выделение некоторых папок в файловом менеджере. Системные файлы, соединения или зашифрованные файлы имеют цветовую маркировку для улучшения идентификации. То же самое сделано для нескольких популярных типов файлов, таких как исполняемые файлы, веб-файлы или изображения. Опытные пользователи могут даже войти в параметры и создать свои собственные папки и группы программ, которые они хотят выделить в системе. Выделение типов файлов по умолчанию отключено и может быть активировано нажатием кнопки включения цветных фильтров в главном интерфейсе.

Когда дело доходит до интерфейса приложения, все находится на расстоянии одного клика от настройки. Доступны параметры для отображения информационной панели для отображения всего, что необходимо знать о файле или папке в открытом виде, каталога с параметрами фильтрации и закладок, нескольких режимов просмотра для отображения файлов и папок в различных форматах и ​​вкладок для отображения нескольких местоположений и информация рядом друг с другом.

На информационной панели отображается информация, скрытая во вложенных меню в проводнике Windows. Например, он отображает все свойства и атрибуты файла, предварительный просмотр и необработанный вид, а также информацию о версии. Свойства файла можно редактировать сразу же одним щелчком мыши.

Поиск файлов также отображается в информационной панели, к которой можно получить доступ напрямую, если панель отображается, или с помощью сочетания клавиш Ctrl-f. XYplorer поддерживает поиск по логическим и регулярным выражениям, фильтры (по размеру, дате, атрибуту содержимого файла) и опции для поиска только в определенных местах.

Результаты поиска отображаются практически мгновенно на одной из панелей файлового браузера. Файлы можно открывать, изменять, перемещать или удалять прямо на этой панели. Файловый менеджер поддерживает операции перетаскивания, например, из одной панели файлового браузера в другую, чтобы сделать операции копирования и перемещения более удобными.

Каталоги, которых может быть несколько, действуют как избранное и фильтры файлов. Эта мощная система позволяет пользователям выполнять базовые операции, такие как открытие определенной папки в системе, фильтрация папки для определенных типов файлов или открытие файлов с помощью выбранной программы.

Еще один способ быстрого поиска файлов и работы с ними — это функция тегов программы. Кнопки добавления тегов не отображаются на панели инструментов по умолчанию, но вы можете добавить их, щелкнув правой кнопкой мыши на панели инструментов и выбрав параметр настройки, или непосредственно из меню, не выполняя эти настройки.

По-прежнему стоит взглянуть на меню настройки панели инструментов, поскольку в меню есть десятки кнопок, которые по умолчанию не отображаются на панели инструментов. Это включает буквы дисков, вышеупомянутые кнопки тегов и даже кнопки, определяемые пользователем.

Опытные пользователи находят в своем распоряжении большое количество функций, которые еще больше улучшают рабочий процесс. Сочетания клавиш, уже доступные для многих основных функций программы, можно определить более или менее для любой функции программы. Например, ярлыки могут быть добавлены к пользовательским командам, которые работают точно так же, как функция каталога.

Для расширенных операций доступен язык сценариев. От сохранения прошлогодних файлов документации в zip-файл, преобразования символов в текстовых файлах или создания отформатированного HTML-документа, в котором перечислены все файлы каталога. Примеры приведены в файле справки программы и на домашней странице программы в Интернете.

Меню конфигурации программы содержит сотни параметров конфигурации, которые изменяют функциональность приложения. От редактирования цветовых фильтров и тегов поверх шрифтов и шаблонов до размеров миниатюр и поведения при запуске и выходе — все это можно редактировать одним щелчком мыши.

Список возможностей XYplorer на этом не заканчивается. В рамках обзора стоит упомянуть многоуровневые параметры отмены и повтора, операции с файлами в очереди, режим мини-дерева, в котором отображается только структура папок выбранной в данный момент папки, и режим плоского просмотра, в котором отображается все содержимое подпапок в файловый браузер напрямую.

**ГЛАВА 2**

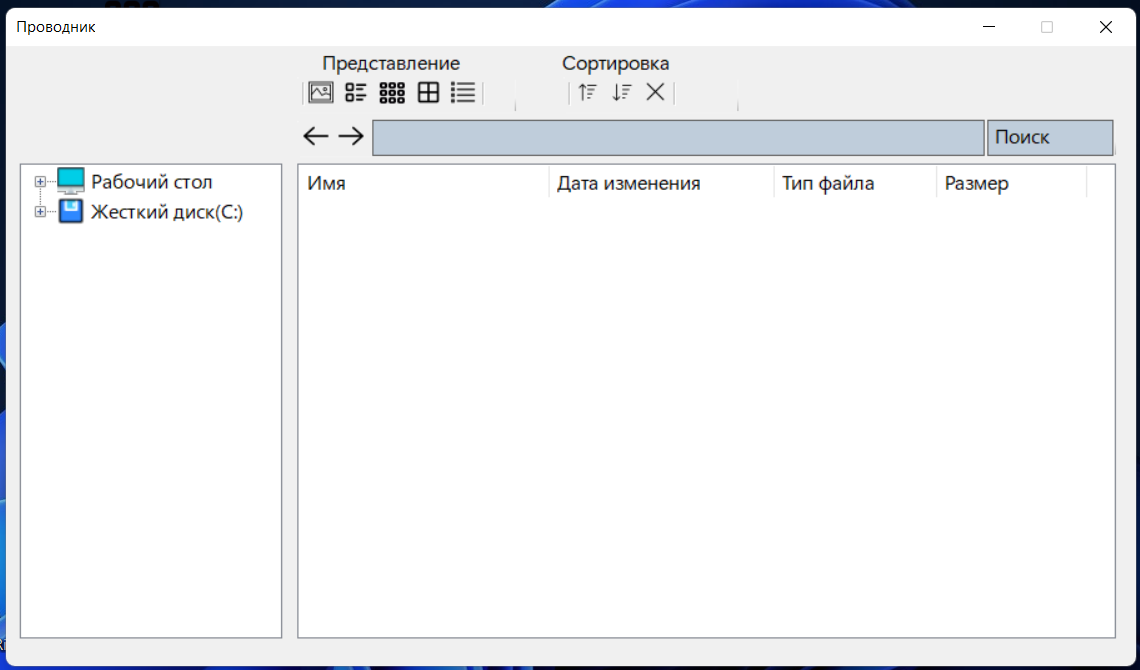
**ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ**

**2.1 Класс главной формы**

С помощью главной формы происходит взаимодействие пользователя с программой. рисунок 2.1.1

Рисунок 2.1.1

Главная форма приложения



При разработке формы были использованы следующие визуальные компоненты:

* ListView – для отображения файлового представления;
* Button – для выбора соответствующей операции для работы с файлами и папками;
* TextBox – для отображения текущих путей в каждом файловом представлении, а также для поиска файлов по их названию;
* ContextMenuStrip – для отображения контекстного меню в каждом из файловых представлений.
* ImageList – для хранения рисунков, которые используются для элементов.

**2.1.1 Методы класса**

Класс главной формы содержит в себе множество методов и обработчиков событий формы, которые используются в работе приложения.

Метод private void InitDisplay() обеспечивает отображение папок в TreeView (Листинг 1).

Листинг 1

Отображение папок в TreeView

treeView.Nodes.Clear();

DriveInfo[] driveInfo = DriveInfo.GetDrives();

TreeNode desktop = null;

desktop = treeView.Nodes.Add("Рабочий стол");

desktop.Tag = "C:\\Users\\kotfr\\Desktop";

desktop.ImageIndex = 3;

desktop.SelectedImageIndex = 3;

foreach (DriveInfo info in driveInfo)

{

TreeNode driveNode = null;

switch (info.DriveType)

{

case DriveType.Fixed:

driveNode = treeView.Nodes.Add("Жесткий диск(" + info.Name.Split('\\')[0] + ")");

driveNode.Tag = info.Name;

driveNode.ImageIndex = 1;

driveNode.SelectedImageIndex = 1;

break;

case DriveType.Removable:

driveNode = treeView.Nodes.Add("Внешний диск(" + info.Name.Split('\\')[0] + ")");

driveNode.Tag = info.Name;

driveNode.ImageIndex = 2;

driveNode.SelectedImageIndex = 2;

break;

}

}

foreach (TreeNode node in treeView.Nodes)

{

LoadChildNodes(node);

}

Метод private void CreateFolder(object sender, EventArgs e) используется для создания папок в ListView (Листинг 2).

Листинг 2

Создание папки

try

{

int num = 1;

string path = Path.Combine(FilePath, "новая папка");

string newFolderPath = path;

while (Directory.Exists(newFolderPath))

{

newFolderPath = path + "(" + num + ")";

num++;

}

Directory.CreateDirectory(newFolderPath);

ListViewItem item = ListFiles.Items.Add("новая папка" + (num == 1 ? "" : "(" + (num - 1) + ")"), 0);

item.Tag = newFolderPath;

}

catch (Exception f)

{

MessageBox.Show(f.Message, "ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

Метод private void CreateFileTxt(object sender, EventArgs e) используется для создания файлов в ListView (Листинг 3).

Листинг 3

Создание файла

int num = 1;

string newFile = "новый файл";

string path = Path.Combine(FilePath, newFile);

string newFilePath = path;

while (File.Exists(newFilePath + ".txt"))

{

newFilePath = path + "(" + num + ")";

num++;

}

File.Create(newFilePath + ".txt");

ListViewItem item = ListFiles.Items.Add("новый файл" + (num == 1 ? ("") : ("(" + (num - 1) + ")")), 4);

//Реальный путь

item.Tag = newFilePath;

Метод private void Insert\_Click(object sender, EventArgs e) используется для вставки файлов/папок (Листинг 4).

Листинг 4

Вставка файлов/папок

try

{

if (SelectedFiles[0] == null)

{

return;

}

else

{

for (int i = 0; SelectedFiles[i] != null; i++)

{

if (File.Exists(SelectedFiles[i]))

{

CutOrCopyFile(SelectedFiles[i]);

}

else if (Directory.Exists(SelectedFiles[i]))

{

CutOrCopyFolder(SelectedFiles[i]);

}

}

ShowFileList(FilePathTextBox.Text, false);

SelectedFiles = new string[200];

}

}

catch (Exception r)

{

MessageBox.Show(r.Message, "ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

Метод private void Delete\_Click(object sender, EventArgs e) используется для удаления файлов/папок (Листинг 5).

Листинг 5

Удаление файлов/папок

try

{

foreach (ListViewItem item in ListFiles.SelectedItems)

{

string path = item.Tag.ToString();

if (File.Exists(path))

{

File.Delete(path);

}

else if (Directory.Exists(path))

{

Directory.Delete(path, true);

}

ListFiles.Items.Remove(item);

}

ShowFileList(FilePath, false);

}

catch (Exception r)

{

MessageBox.Show(r.Message, "ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

Метод private void ListFiles\_ItemActivate(object sender, EventArgs e) позволяет открыть папку/файл (Листинг 6).

Листинг 6

Открытие папки/файла

FilePath = ListFiles.SelectedItems[0].Tag.ToString();

try

{

if (Directory.Exists(FilePath))

{

ShowFileList(FilePath, true);

}

else

{

Process.Start(FilePath);

}

}

catch (Exception r)

{

MessageBox.Show(r.Message, "ошибка", MessageBoxButtons.OK, MessageBoxIcon.Error);

}

Метод private void ReloadClick(object sender, EventArgs e) обеспечивает обновление данных в ListVeiw и TreeVeiw (Листинг 7).

Листинг 7

Обновление данных

FilePath = FilePathTextBox.Text;

ShowFileList(FilePath, false);

InitDisplay();

Метод private void FastView\_Click(object sender, EventArgs e) используется для быстрого просмотра файлов с расширением txt и cs (Листинг 8)

Листинг 8

Быстрый просмотр

if (ListFiles.SelectedItems.Count == 1)

{

var fileInfo = new FileInfo(ListFiles.SelectedItems[0].Text);

if (fileInfo.Extension == ".txt" || fileInfo.Extension == ".cs")

{

this.ClientSize = new System.Drawing.Size(1200, 510);

FastVeiw.Visible = true;

string file = ListFiles.SelectedItems[0].Tag.ToString();

FastVeiw.Text = File.ReadAllText(Path.Combine(FilePath, file));

}

}

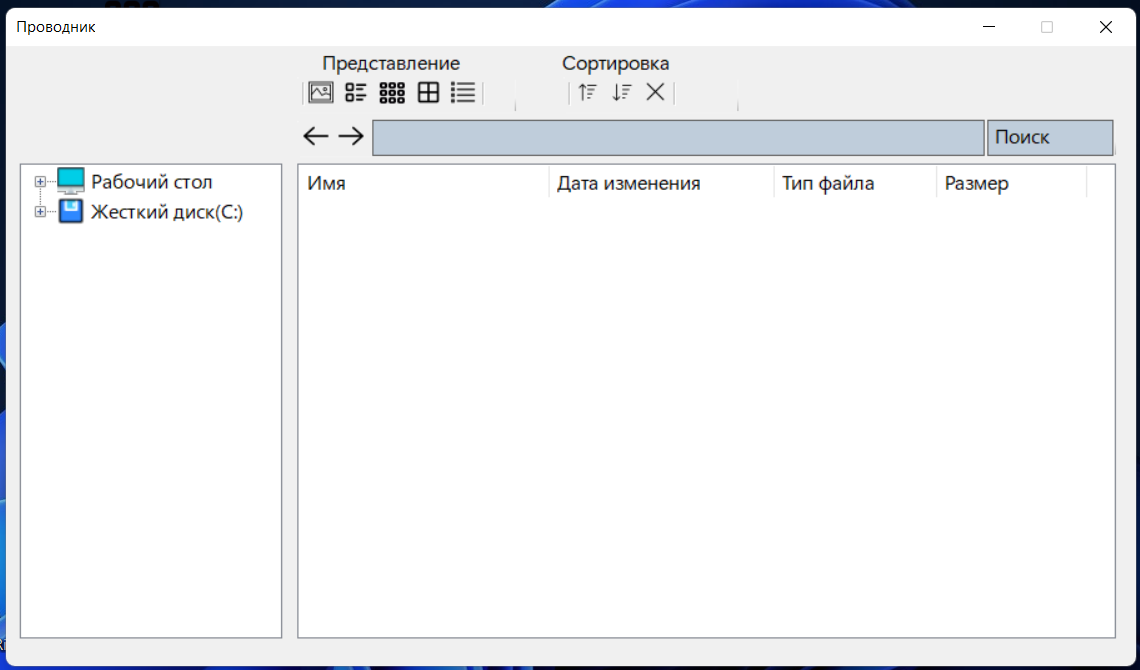
**ГЛАВА 3**

**ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ**

При запуске приложения открывается главная форма. рисунок 3.1

Рисунок 3.1

Главная форма

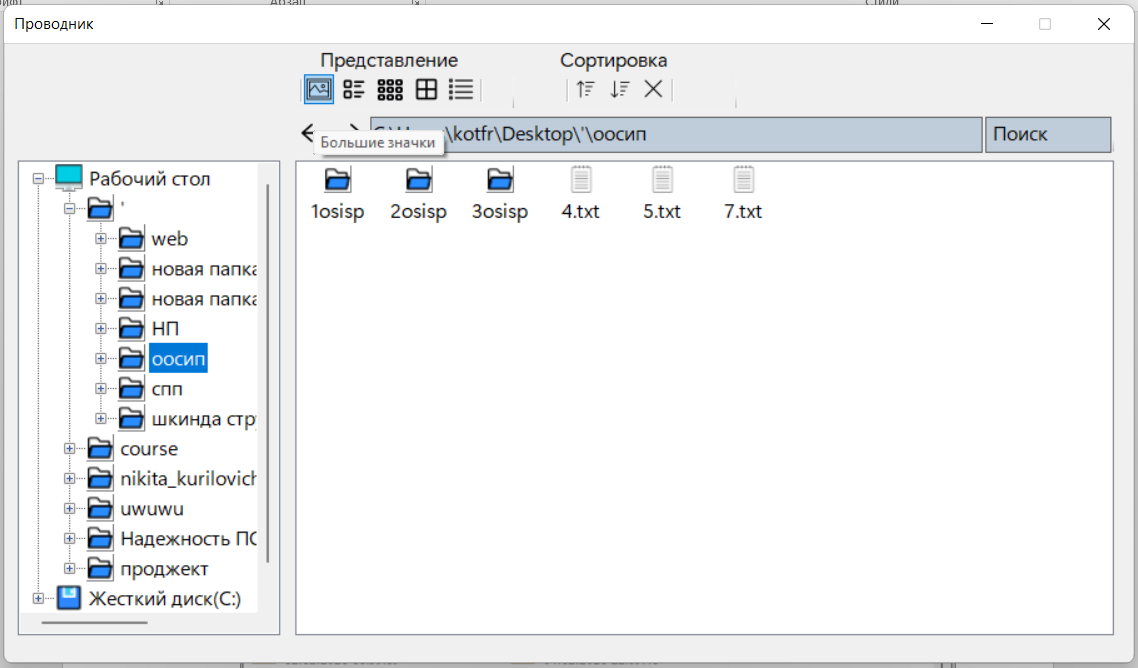


На главном окне есть адресная строка, кнопки представления, кнопки сортировки, представление элементов, строка для поиска элементов, а также кнопки перемещения по истории пользователя.

Режим представления данных “Большие значки” изображён на рисунке 3.2

Рисунок 3.2

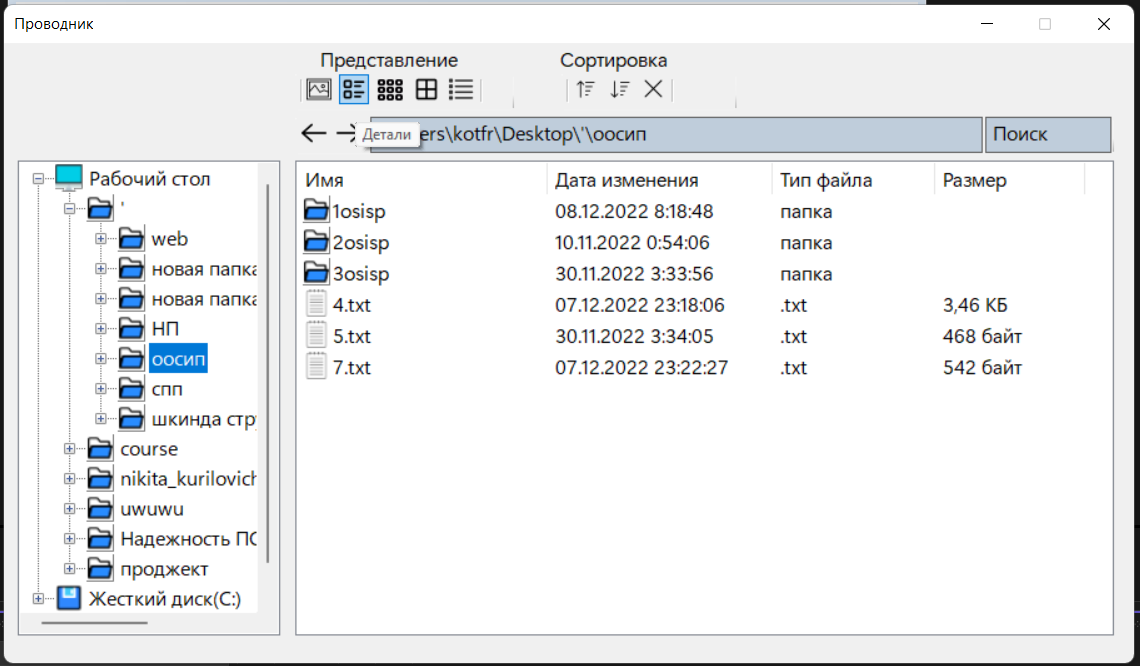
Представление “Большие значки”



Режим представления данных “Детали” изображён на рисунке 3.3

Рисунок 3.3

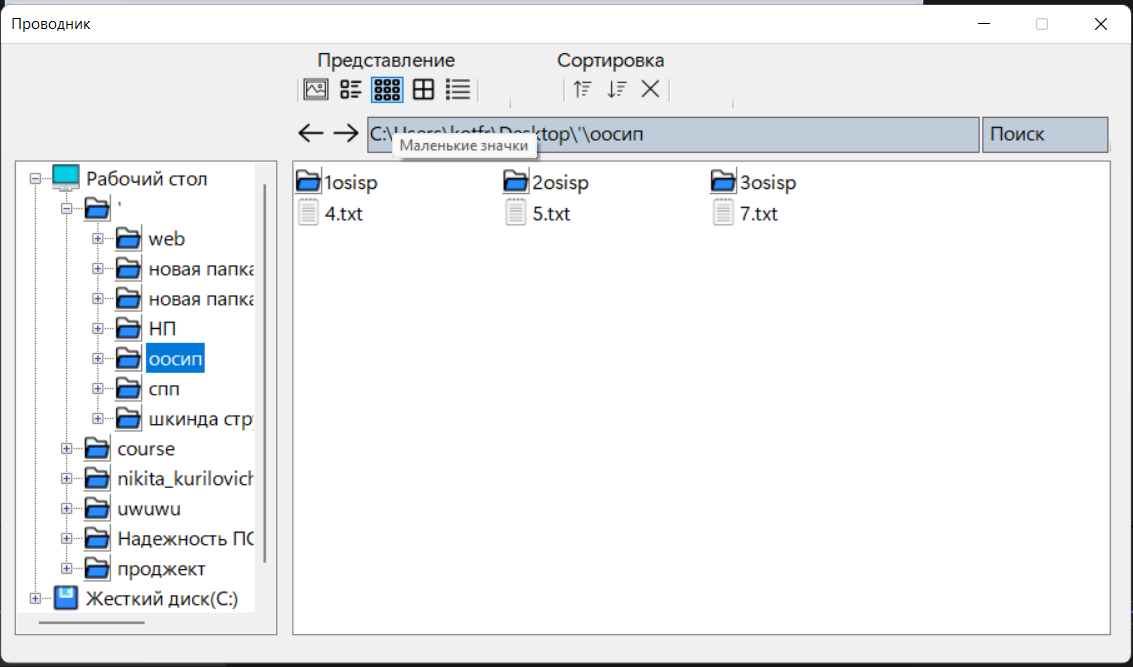
Представление “Детали”



Режим представления данных “Маленькие значки” изображён на рисунке 3.4

Рисунок 3.4

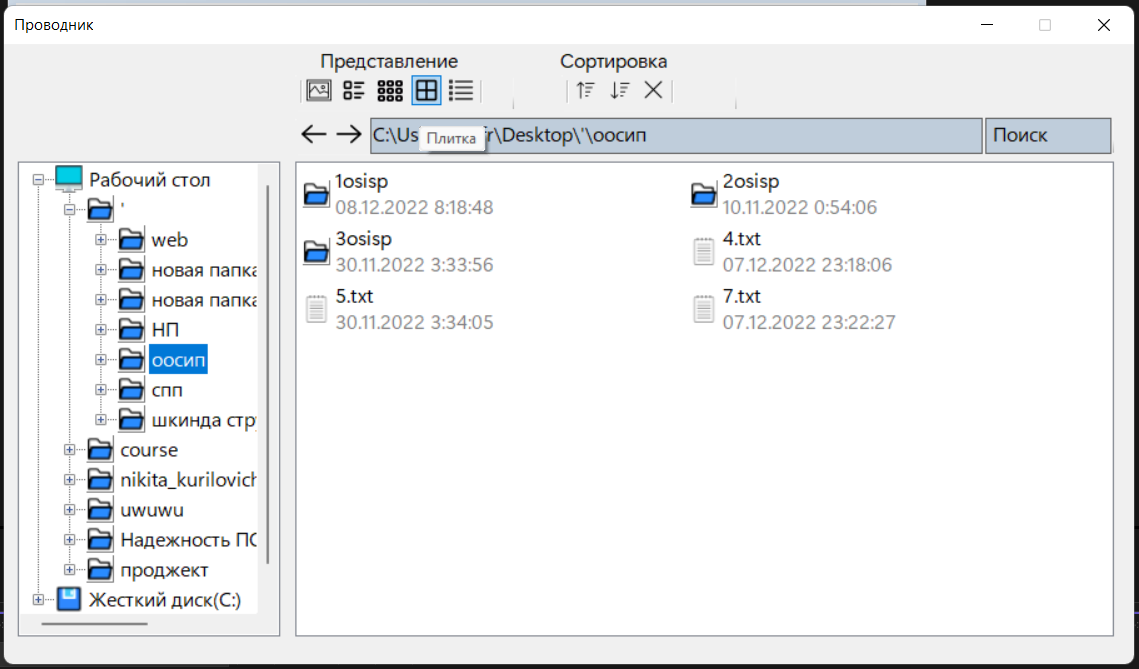
Представление “Маленькие значки”



Режим представления данных “Список” изображён на рисунке 3.5

Рисунок 3.5

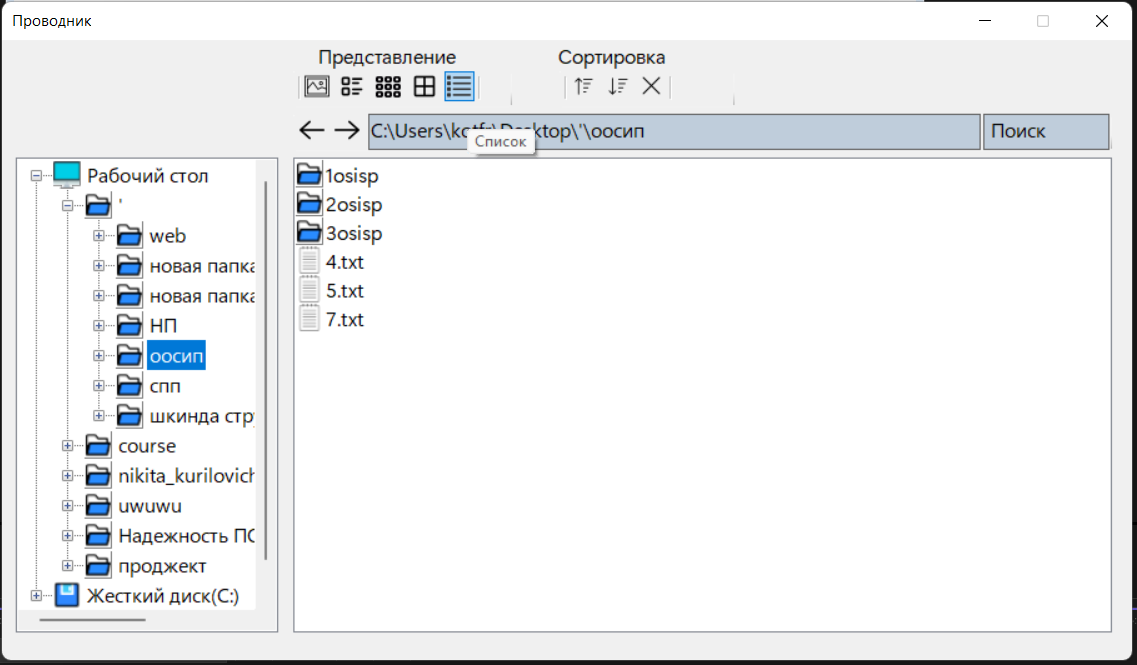
Представление “Плитка”



Режим представления данных “Список” изображён на рисунке 3.6

Рисунок 3.6

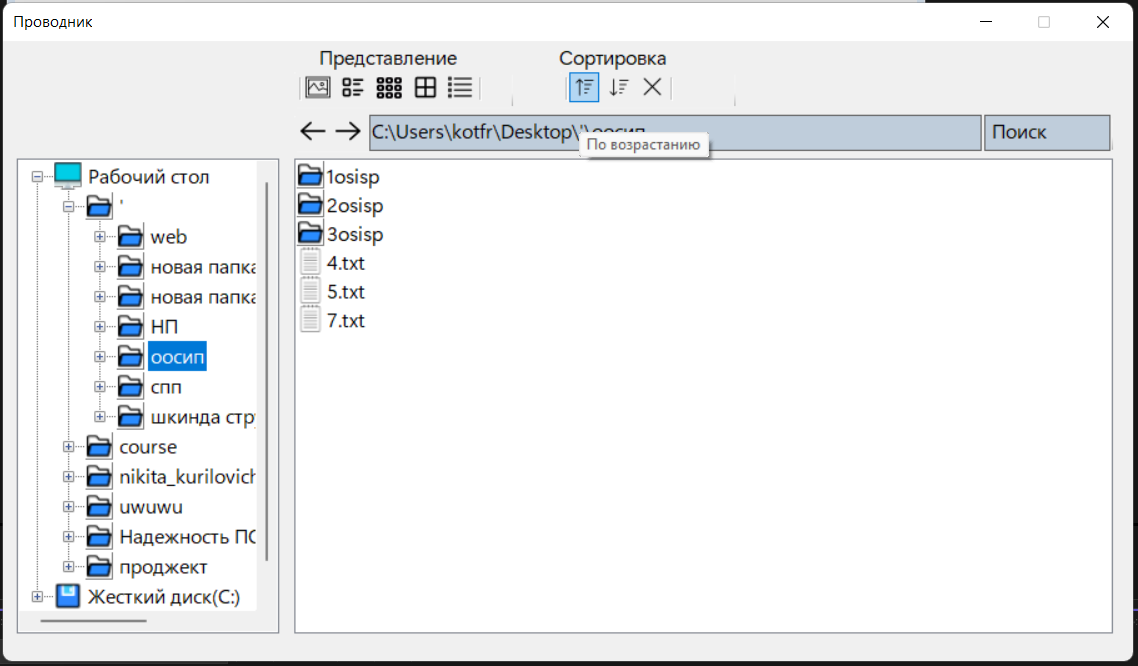
Представление “Список”



Кнопка сортировки данных по возрастанию изображена на рисунке 3.7

Рисунок 3.7

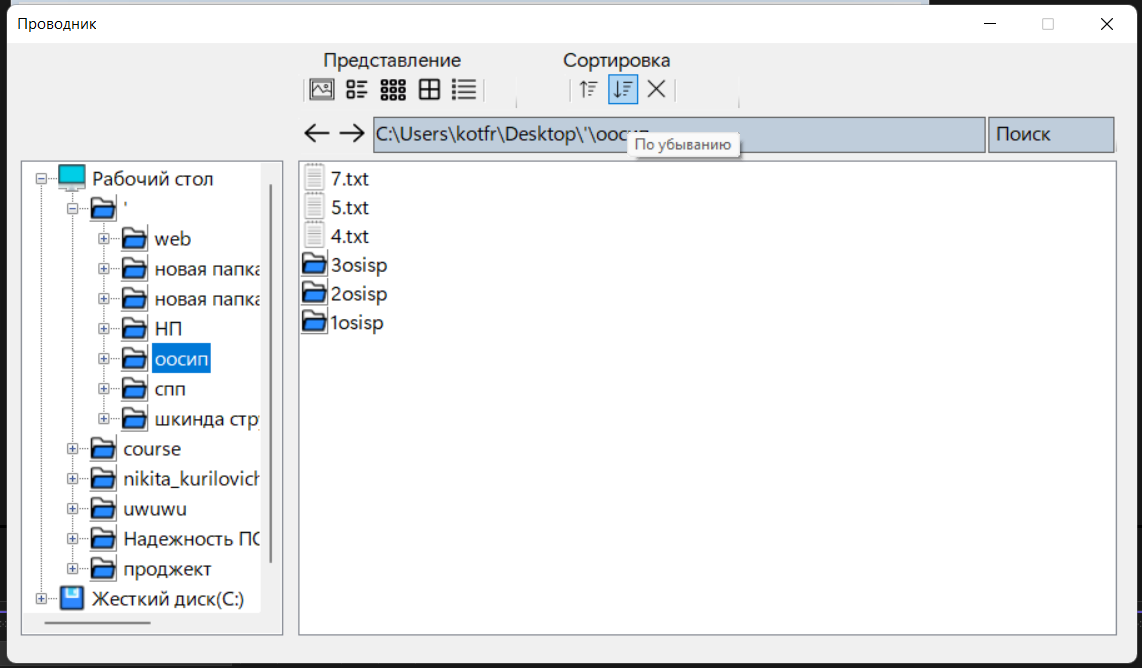
Сортировка по возрастанию



Кнопка сортировки данных по убыванию изображена на рисунке 3.8

Рисунок 3.8

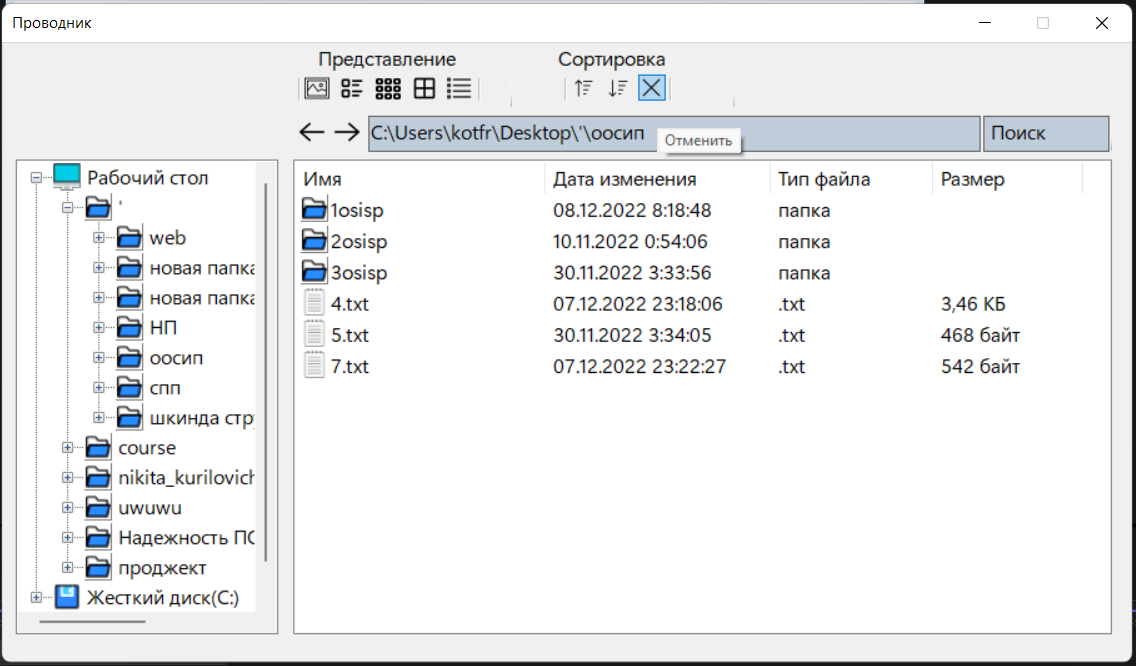
Сортировка по убыванию



Кнопка отвечающая за отмену сортировки изображена на рисунке 3.9

Рисунок 3.9

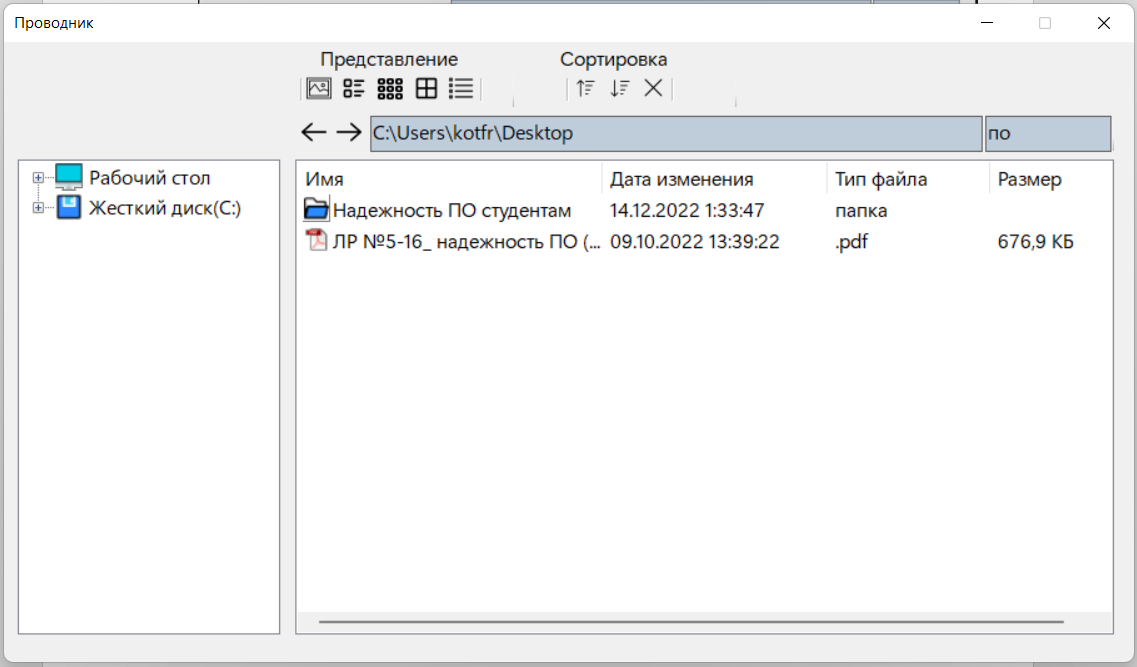
Отмена сортировки



Поиск данных реализован на рисунке 3.10

Рисунок 3.10

Поиск данных



Функция создания элемента реализована на рисунках 3.11-3.12

Рисунок 3.11

Создание элемента

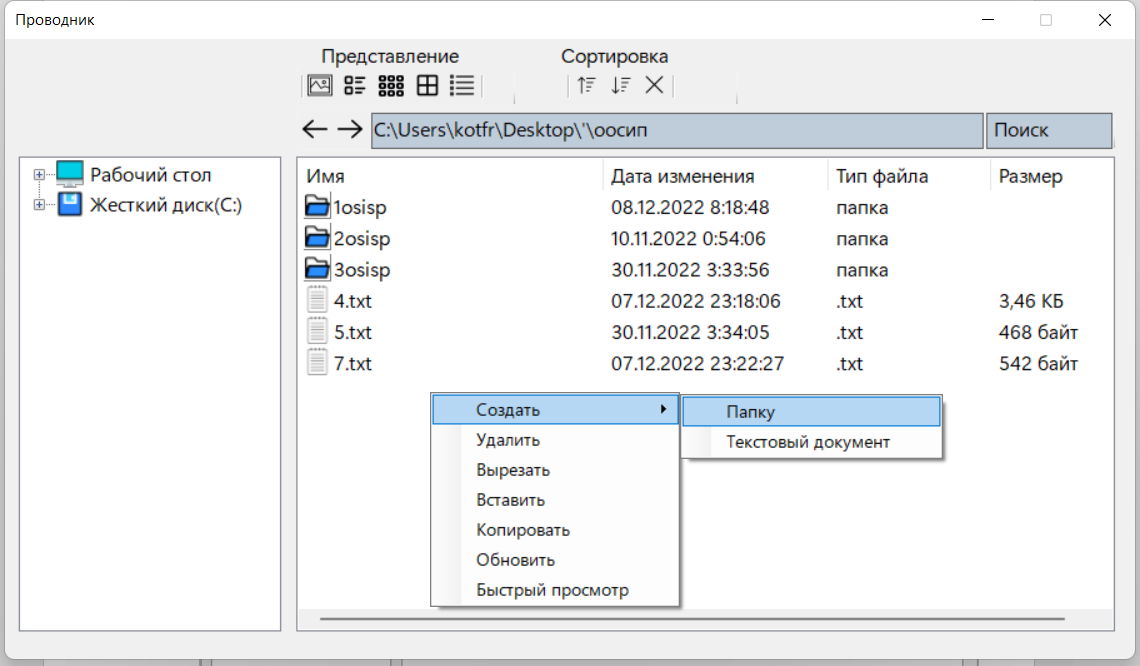
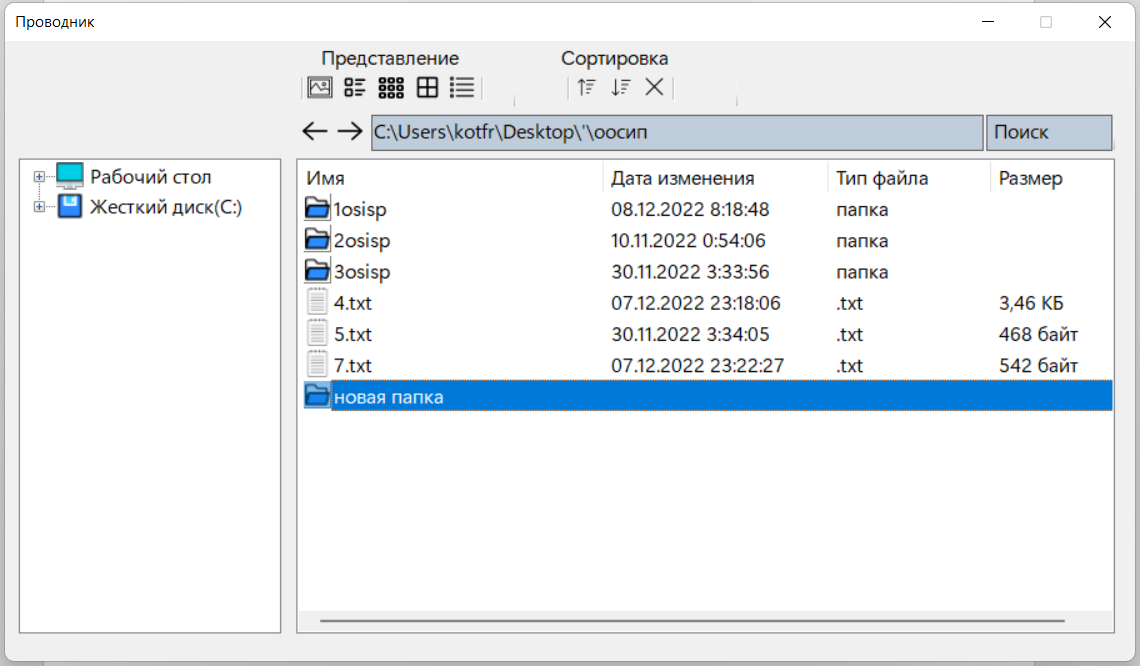


Рисунок 3.12



Функция удаления элемента реализована на рисунках 3.13-3.14

Рисунок 3.13

Удаление элемента

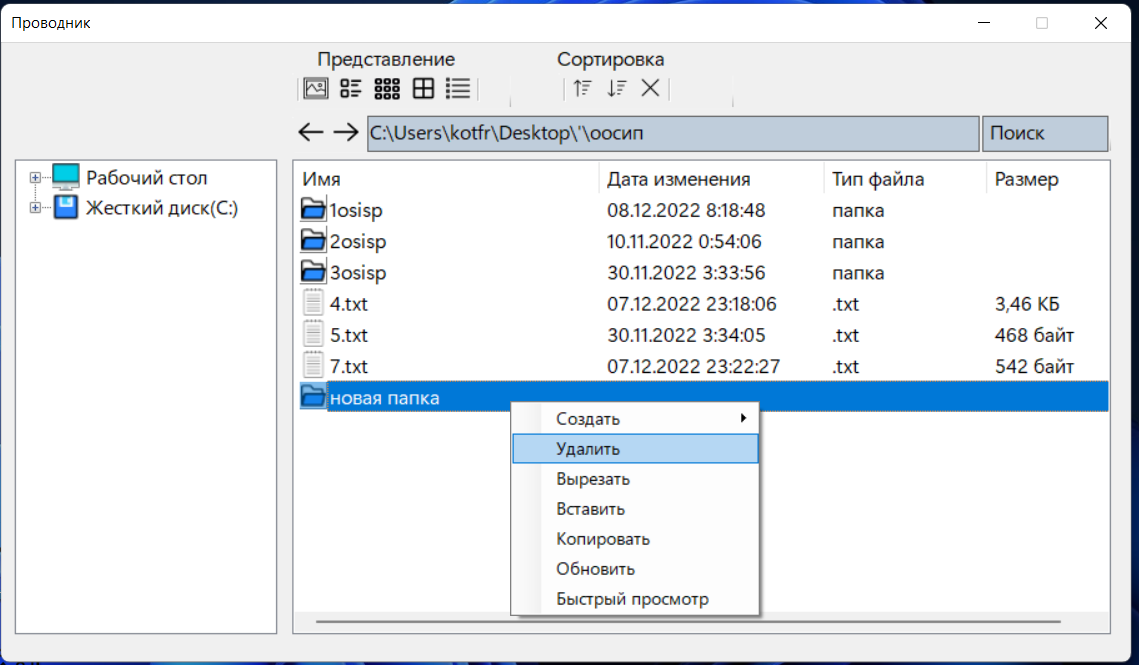
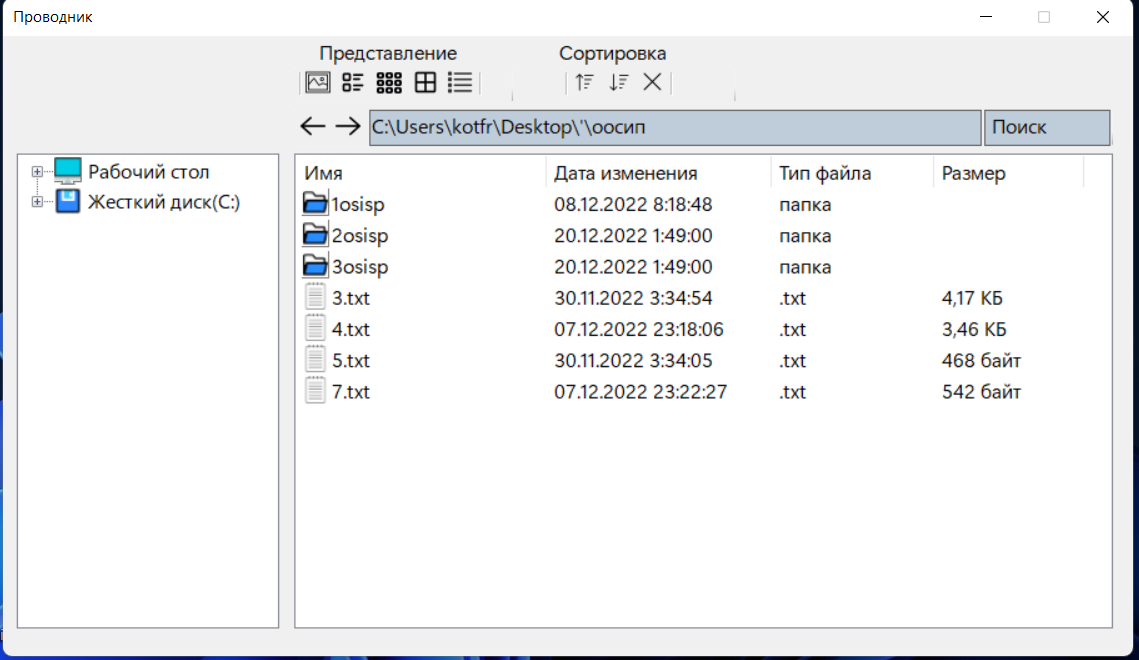


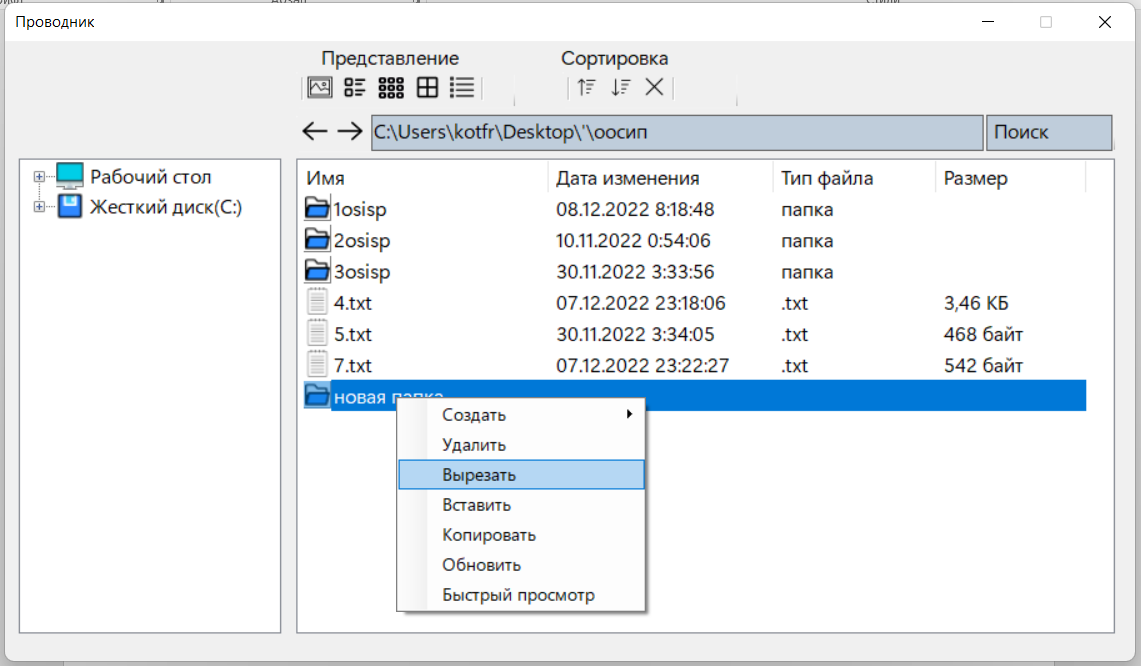
Рисунок 3.14



Функция вырезка элемента реализована на рисунке 3.15

Рисунок 3.15

Вырезка элемента



Функция вставки элемента реализована на рисунке 3.16-3.17

Рисунок 3.16

Вставка элемента

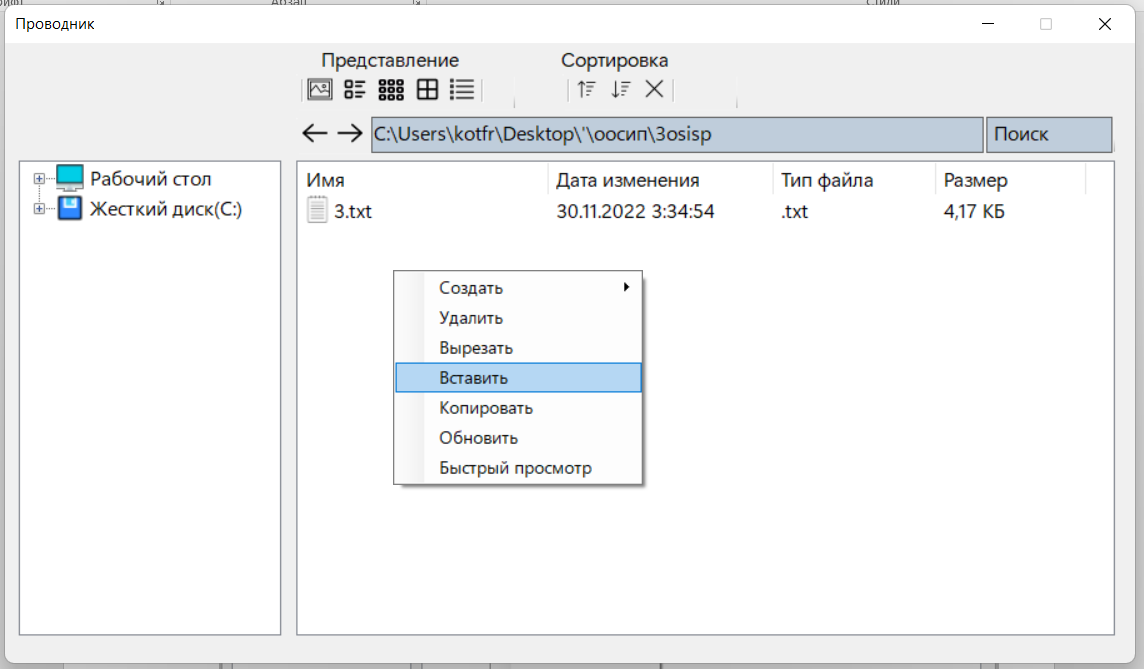
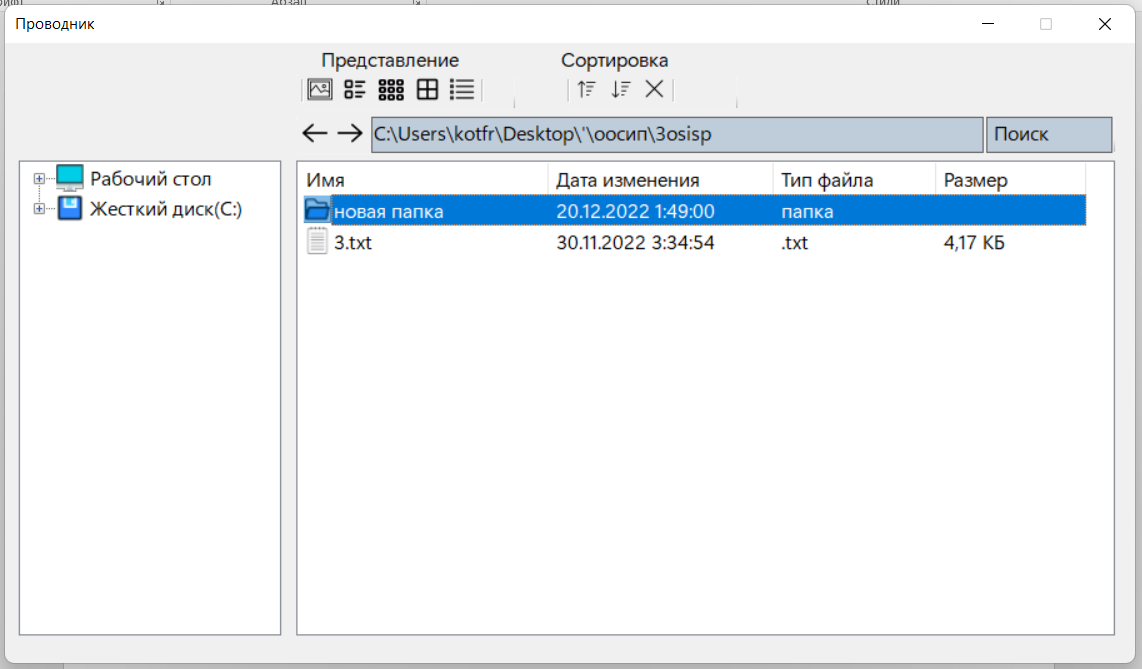


Рисунок 3.17

Вставка вырезанного элемента



Функция копирования элемента реализована на рисунках 3.18-3.19

Рисунок 3.18

Копирование элементов

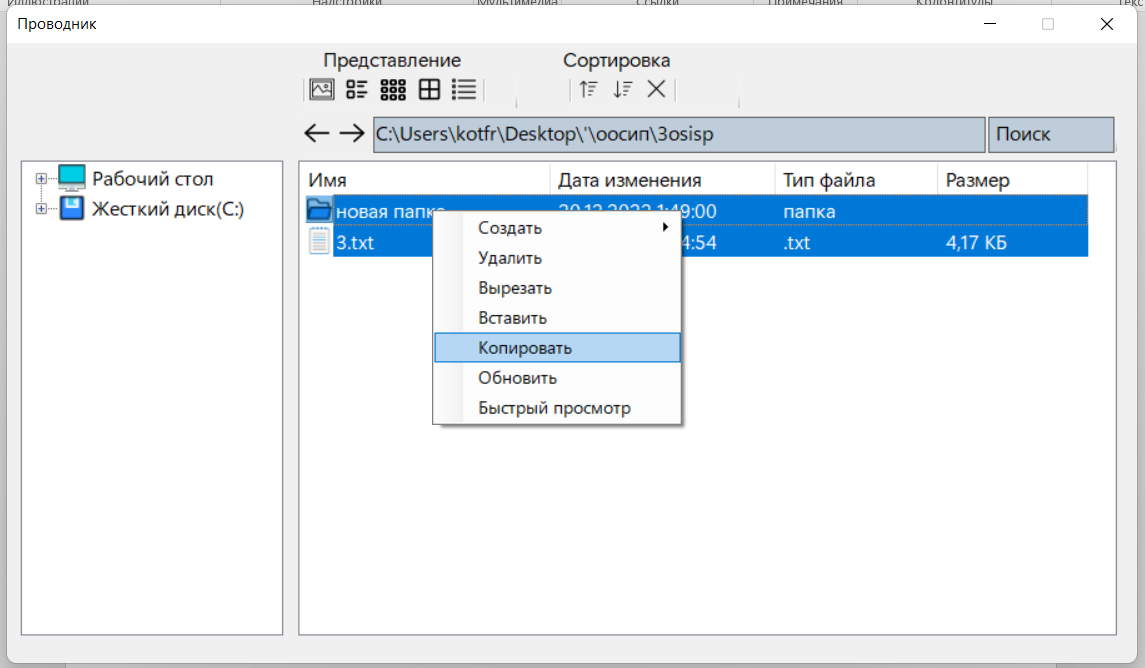
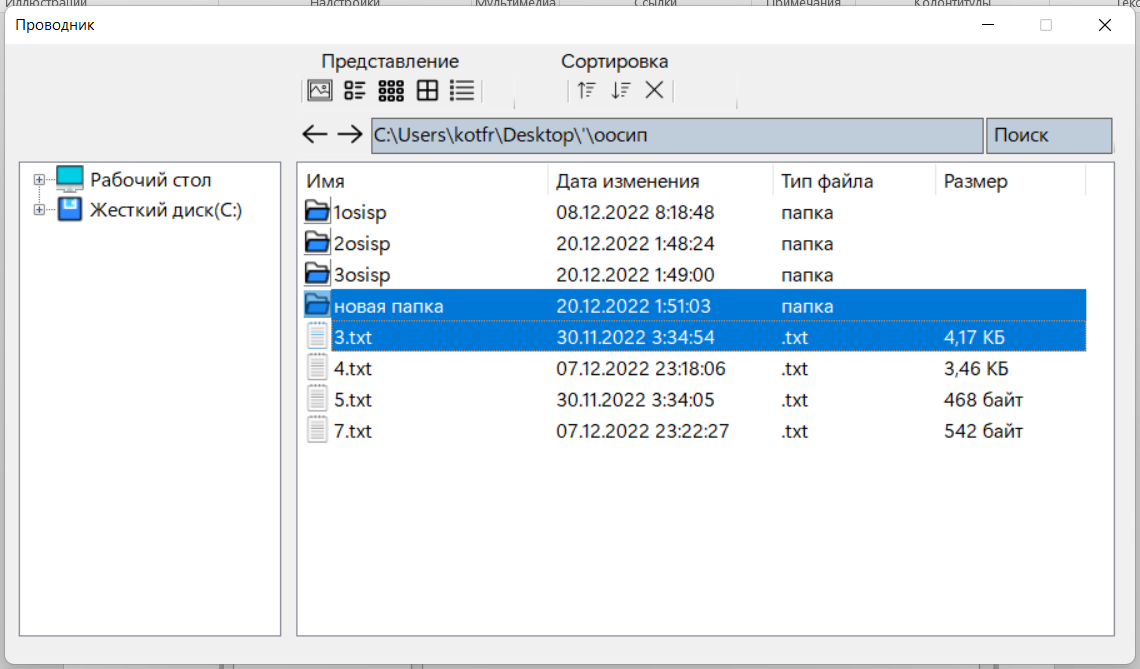


Рисунок 3.19

Вставка скопированных элементов



Функция быстрого просмотра реализована на рисунках 3.20-3.21

Рисунок 3.20

Быстрый просмотр файла

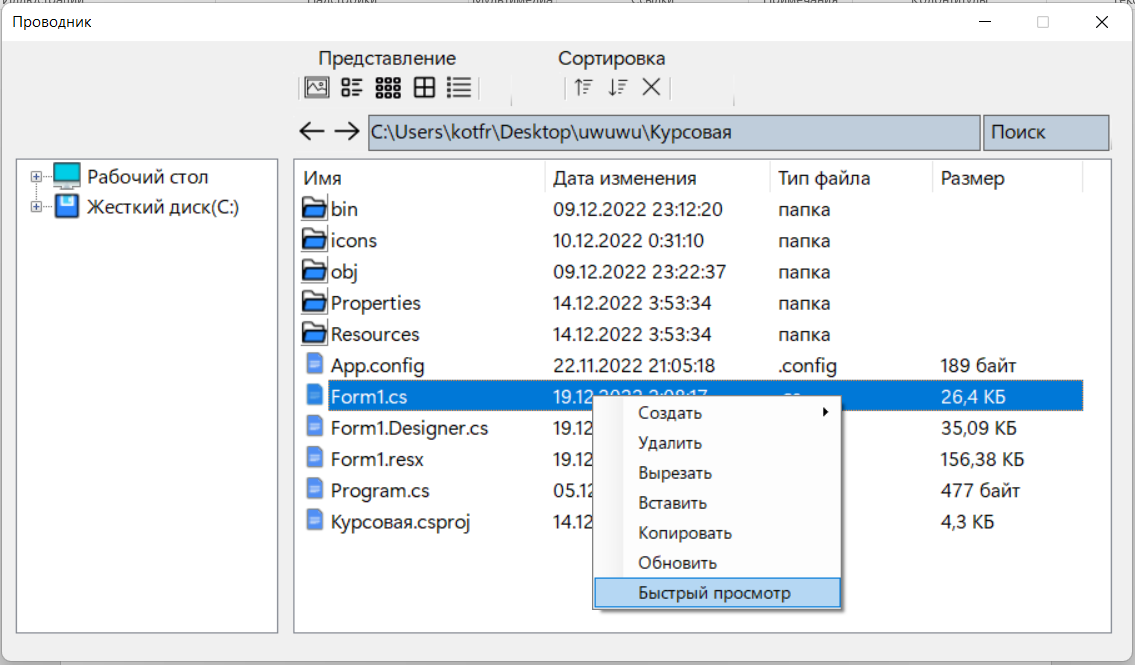
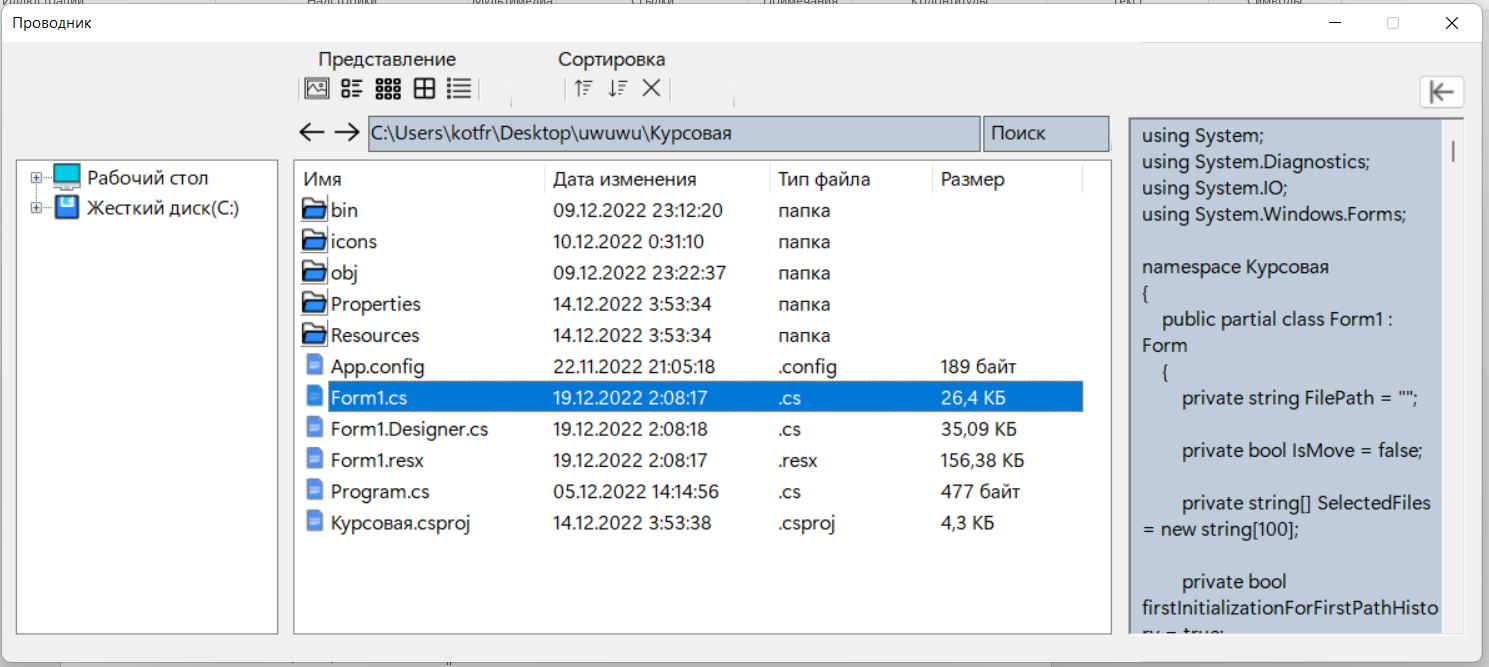


Рисунок 3.21



Заключение

В ходе выполнения курсового проекта была достигнута поставленная цель – создание файлового менеджера на языке C#, а также решены задачи, необходимые для достижения цели:

* Изучен материал, который необходим для выполнения проекта.
* Разработана программа.
* Протестирована программа.

Проект реализовывался при помощи среды разработки Microsoft Visual Studio 2022 на языке программирования C#.

В результате выполнения курсового проекта было создано приложение «Проводник Windows», которое является простым и удобным средством для работы с файлами.

Данное программное средство реализует следующие операции для работы с файлами и папками:

* Быстрый просмотр,
* Представление,
* Копирование,
* Удаление,
* Вырезка,
* Вставка,
* Сортировка,
* Обновление,
* Создание,
* Открытие,
* Поиск.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

# Албахари. C# 7.0 Карманный справочник. Издательство: Вильямс, 2018. - 224.

# Мороз, Л. А. Технологии программирования и методы алгоритмизации: контрольные задания / Л. А. Мороз. - Могилев: МГУ имени А. А. Кулешова, 2018. - 66 с.

# Мэтью Мак-Дональд, Windows Presentation Foundation в .NET 4.5 с примерами на C# 5.0 для профессионалов. Издательство: Вильямс, 2013. – 1024.

# Пахомов Б.И. C# для начинающих - Санкт-Петербург: Издательство БХВ-Петербург, 2014. - 432 с.

# Чарльз Петцольд, Программирование с использованием microsoft windows forms - Санкт-Петербург: Издательство Питер, Русская Редакция, 2006. - 432 c.

# Мюллер Джон Поль, C# для чайников. Издательство: Диалектика-Вильямс, 2019 – 608 c.

# Джеффри Рихтер, CLR via C# - Санкт-Петербург: Издательство: Питер, 2021. - 896 c.