Министерство образования Республики Беларусь

Учреждение образования

«Могилевский государственный университет имени А.А.Кулешова»

**удаление неудаляемых файлов**

Курсовой проект

по дисциплине «Компьютерные системы и сети»

студента 3 курса, специальности 1-40 01 01 «Программное обеспечение

информационных технологий», группы 5242201, заочной формы обучения

Владимира Сергеевича Папейко

Научный руководитель

кандидат физико-математических наук,

доцент кафедры программного

обеспечения информационных технологий

Елизавета Вячеславовна Черненко

Могилев

2023

**ОГЛАВЛЕНИЕ**

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

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

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

1.2.1 UNINSTALL TOOL………………………………………………………6

1.2.2 UNLOCKER………………………………………………………………7

1.2.3 IOBIT UNLOCKER……………………………………………………….7

1.2.4 REVO UNINSTALLER…………………………………………………...7  
ГЛАВА 2 ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ….... 9

2.1 КЛАСС ГЛАВНОЙ ФОРМЫ…………………………………………….. 9

2.2 МЕТОДЫ КЛАССА……………………………………………………….10  
ГЛАВА 3 ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ……….14  
ЗАКЛЮЧЕНИЕ………………………………………………………………..22  
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ………………………….23

**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 удаление не удаляемых файлов

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 10, в среде разработки Visual Studio 2019.

Объект исследования: использование среды Visual Studio 2019, возможностей языка С# и его библиотек для решения поставленной задачи.

Предмет исследования: удаление не удаляемых файлов.

# **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 Uninstall tool**

Uninstall tool — это условно-бесплатная утилита, предназначенная для корректного деинсталлирования программ, установленных в 32-битных и 64-разрядных операционных системах Microsoft Windows.

При запуске Uninstall tool сканирует систему и делает снимок файловой системы и системного реестра для последующего качественного удаления приложений, исходя из ранее полученных данных.

Программа производит мониторинг всех изменений в файловой системе и реестре на компьютере, когда на него пытаются установить программное обеспечение. Анализ установки приложения происходит до и после окончания инсталляции. Если пользователь намерен удалить установленную программу, то утилита производит сканирование всех изменений, которые сделала программа в системе, и выводит список, в котором отображаются ключи и значения в системном реестре, какие файлы и папки к ней относятся, и другие параметры. Из этого списка пользователь может удалить абсолютно всю информацию или выборочные элементы.

Uninstall tool имеет простой в использовании интерфейс, а в функциональные возможности включены такие опции, как пошаговые диалоги, детальная статистика всех изменений, восстановление системы, незаконченные операции с файлами и прочее.

* + 1. **Unlocker**

Unlocker — бесплатная утилита, разработанная Cedrick Collomb для ОС Microsoft Windows, которая позволяет разблокировать файлы, находящиеся в исключительном использовании программами, системными процессами или заблокированные системой. Позволяет определить процессы, обращающиеся к файлу, производить отложенное удаление файлов на стадии загрузки ОС.

Unlocker способна разблокировать почти любой файл в операционной системе, который используется другим процессом, интегрирует в контекстное меню каждого элемента Windows (папки, ярлыки и прочие) новый пункт «Unlocker», который оперативно снимает блокировку с файла, к которому невозможно получить доступ. Оснащён поддержкой многих языков (включая русский язык), а также существует портативная версия для 32-битных и 64-разрядных операционных систем Microsoft Windows.

**1.2.3 IObit Unlocker**

IObit Uninstaller — программа для полного удаления нежелательных программ, папок, панелей инструментов и плагинов браузеров от компании IObit. Удаляет файлы как с жёсткого диска, так и с реестра системы. Имеет ряд дополнительных функций по оптимизации работы Windows. Считается одной из лучших программ для деинсталляции, участвует в сравнительных рейтингах и имеет положительные и отрицательные отзывы на тематических сайтах.

* + 1. **Revo Uninstaller отступы с пред сравнить**

Revo Uninstaller — программа с закрытым кодом для замены стандартного апплета «Установка и удаление программ» операционных систем семейства Windows. Revo Uninstaller выпускается в двух вариантах — Free и Pro. В функциональность программы, помимо деинсталлятора, также входят: диспетчер автозагрузки, очистка ненужных файлов, очистка браузеров, очистка Microsoft Office, очистка Windows и удаление данных и файлов. Пользователи и тематические сайты выделяют функцию глубокого анализа и удаления оставшихся следов программы, как уникальную, поэтому Revo Uninstaller считается одной из лучших программ в своем сегменте.

**ГЛАВА 2**

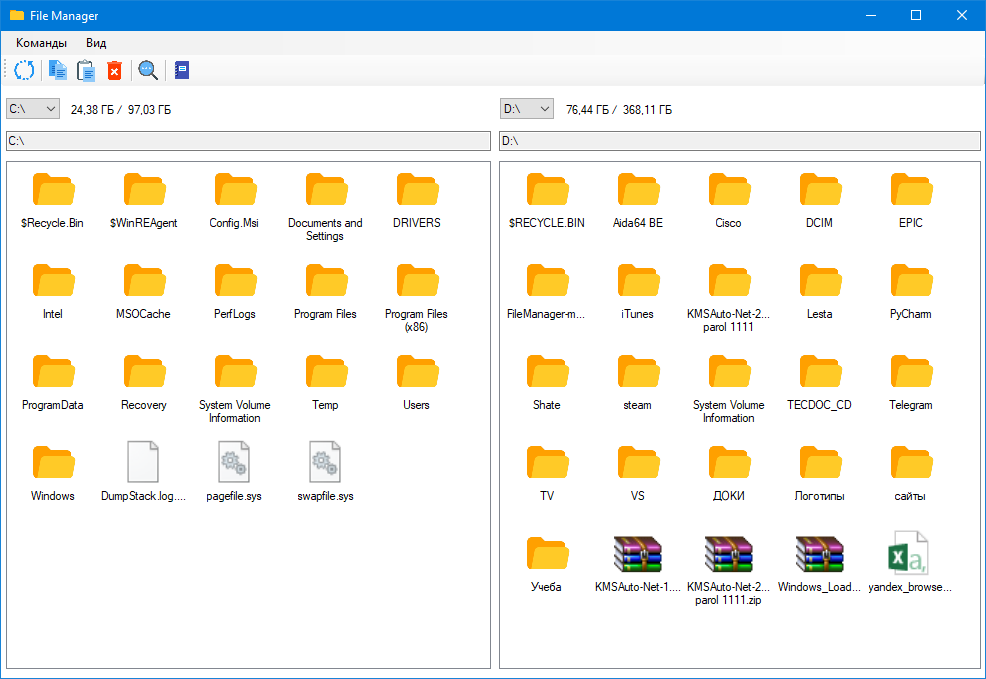
**ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ**

**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.

**2.2 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();

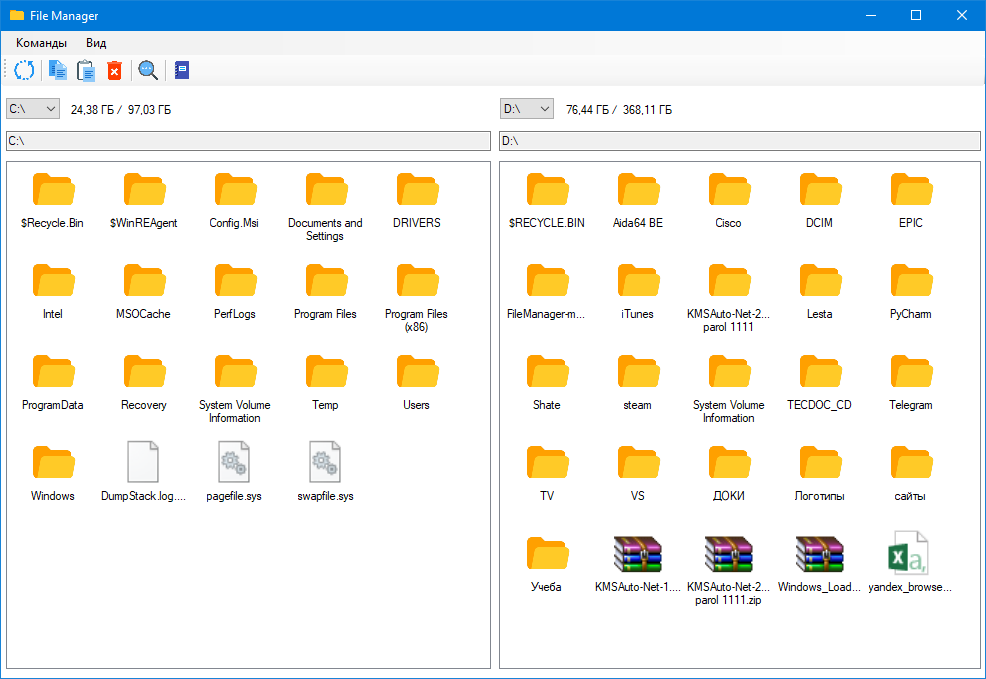
**ГЛАВА 3**

**ТЕСТИРОВАНИЕ ПРОГРАММНОГО ПРИЛОЖЕНИЯ**

При запуске приложения открывается главная форма. рисунок 3.1

Рисунок 3.1

Главная форма

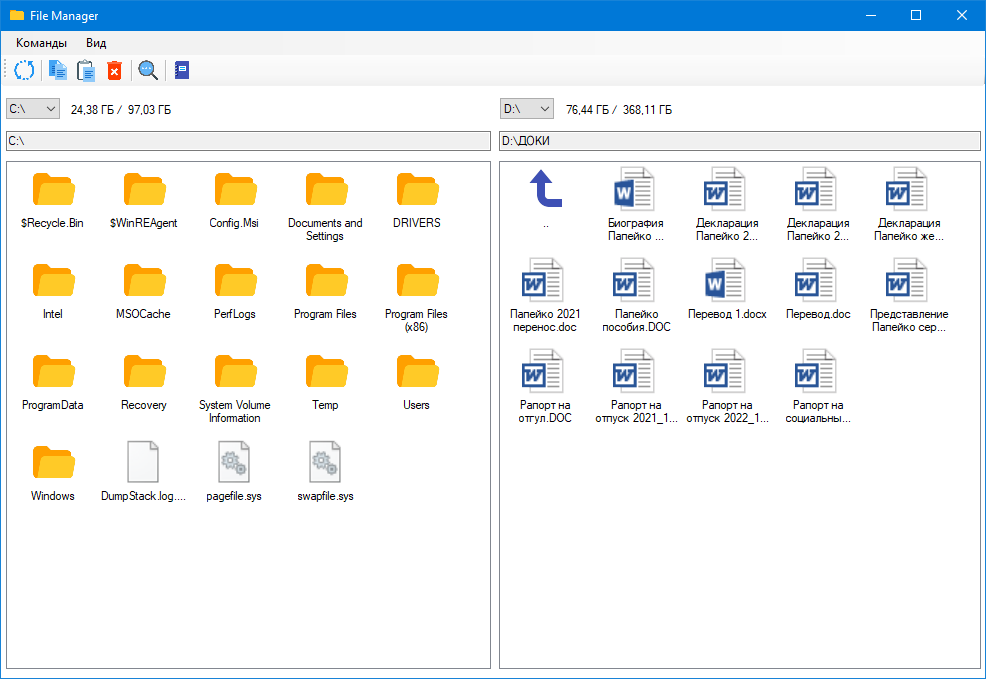


На главном окне есть адресная строка, кнопки представления, кнопки сортировки, представление элементов, строка для поиска элементов, а также кнопки перемещения по истории пользователя.

Режим представления данных “Большие значки” изображён на рисунке 3.2

Рисунок 3.2

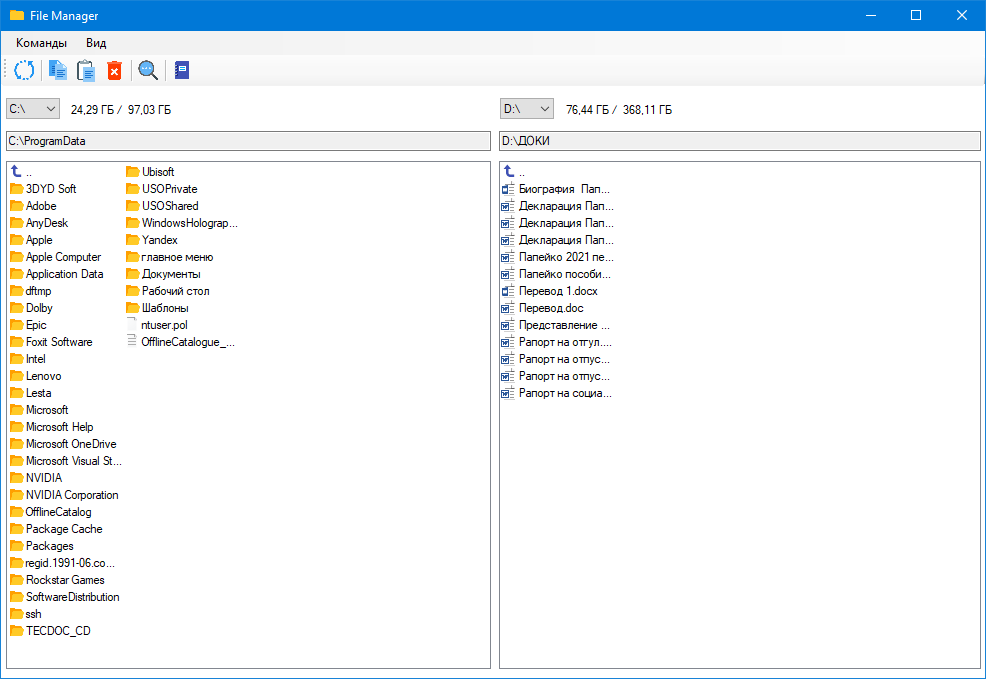
Представление “Большие значки”



Режим представления данных “Список” изображён на рисунке 3.3

Рисунок 3.3

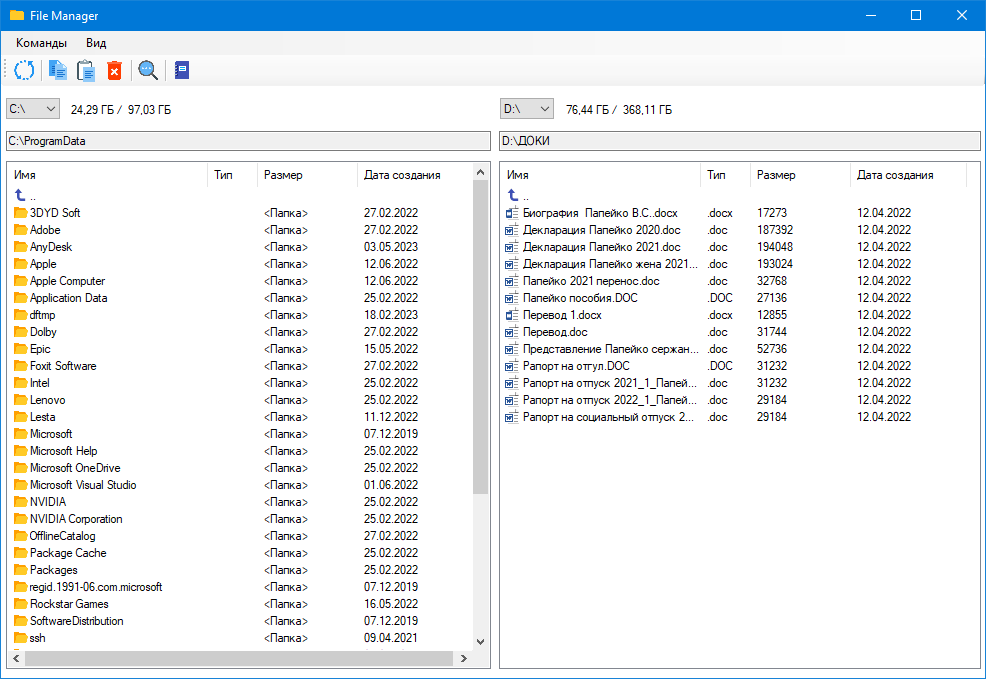
Представление “Список”



Режим представления данных “Таблица” изображён на рисунке 3.4

Рисунок 3.4

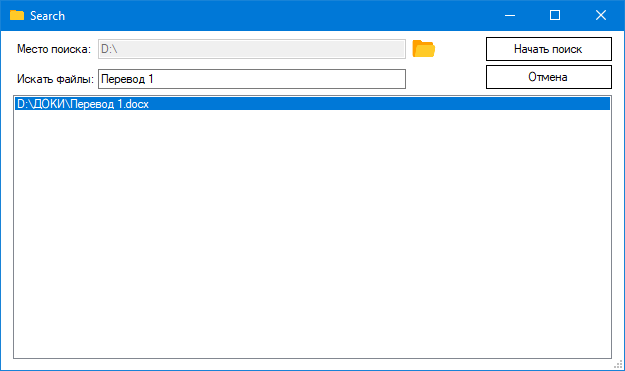
Представление “Таблица”



Поиск данных реализован на рисунке 3.5

Рисунок 3.5

Поиск данных



Функция создания элемента реализована на рисунках 3.6-3.8

Рисунок 3.6

Создание элемента

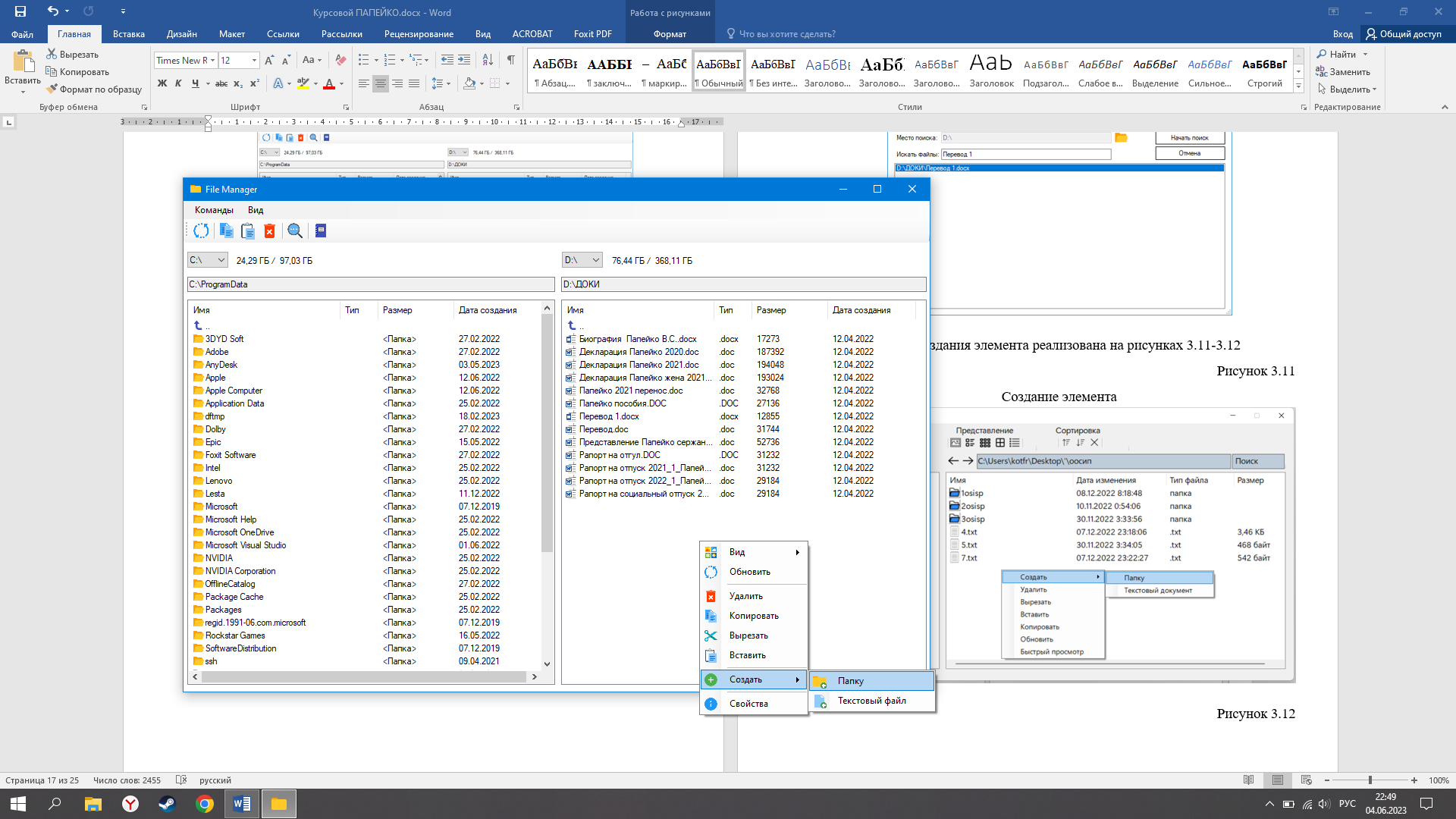


Рисунок 3.7

Создание папки

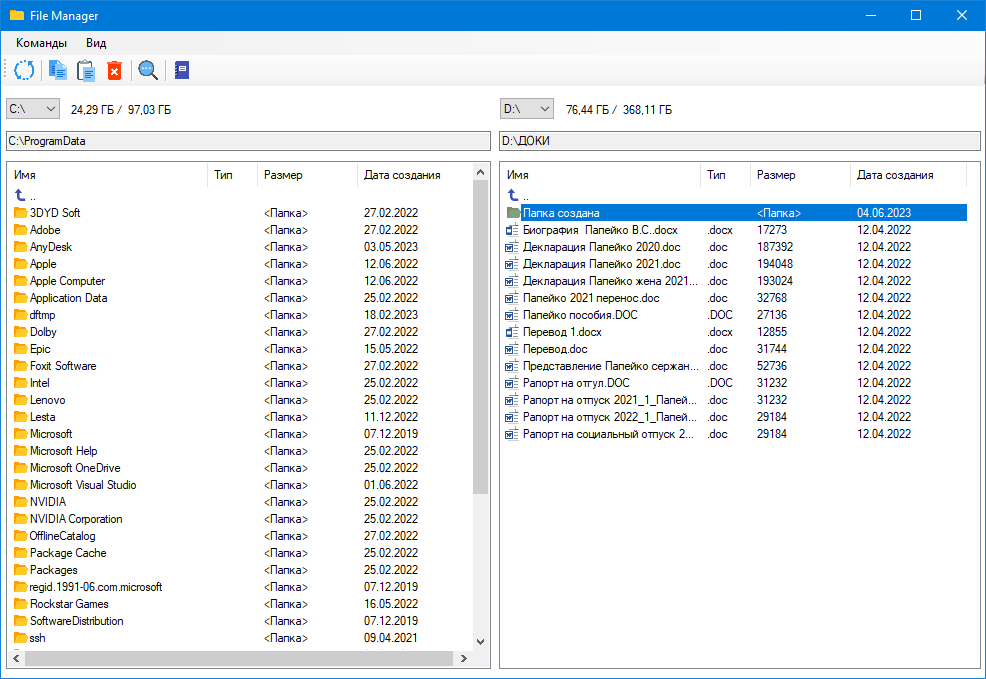
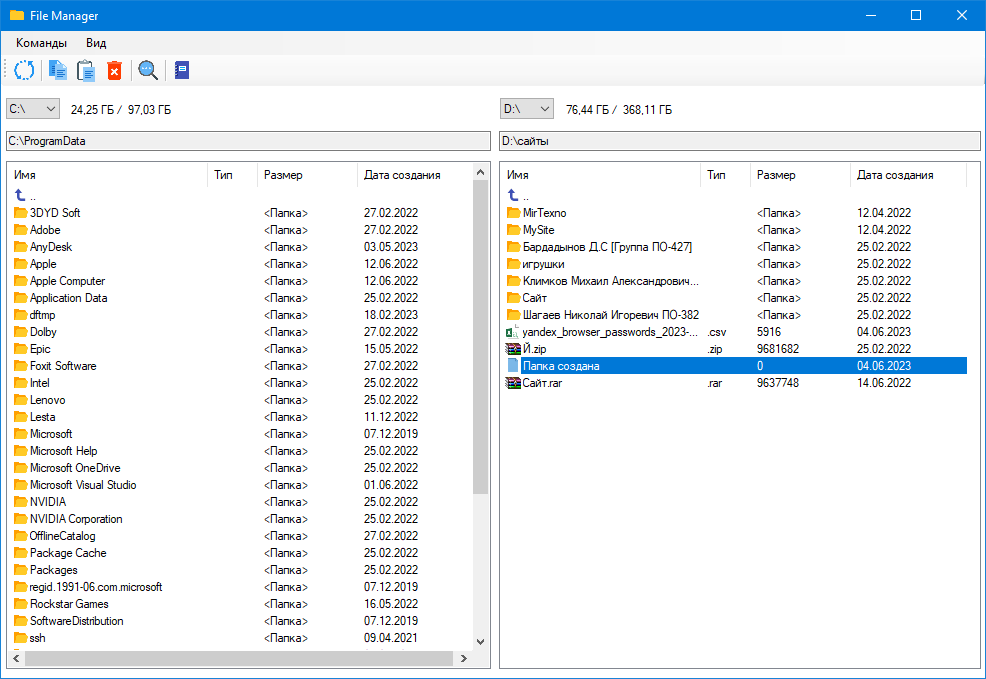


Рисунок 3.8

Создание файла



Функция удаления элемента реализована на рисунках 3.9-3.10

Рисунок 3.9

Удаление элемента

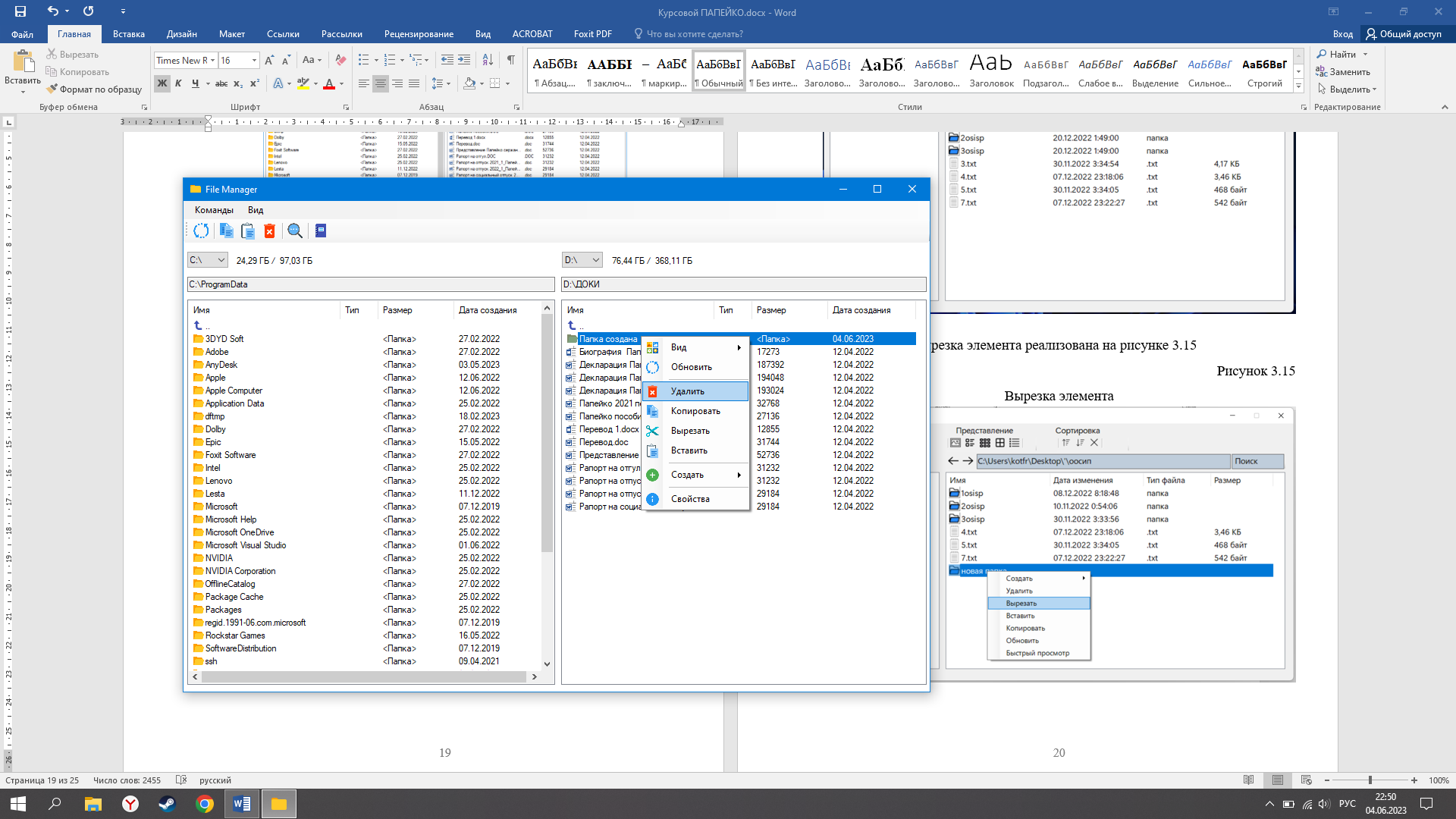
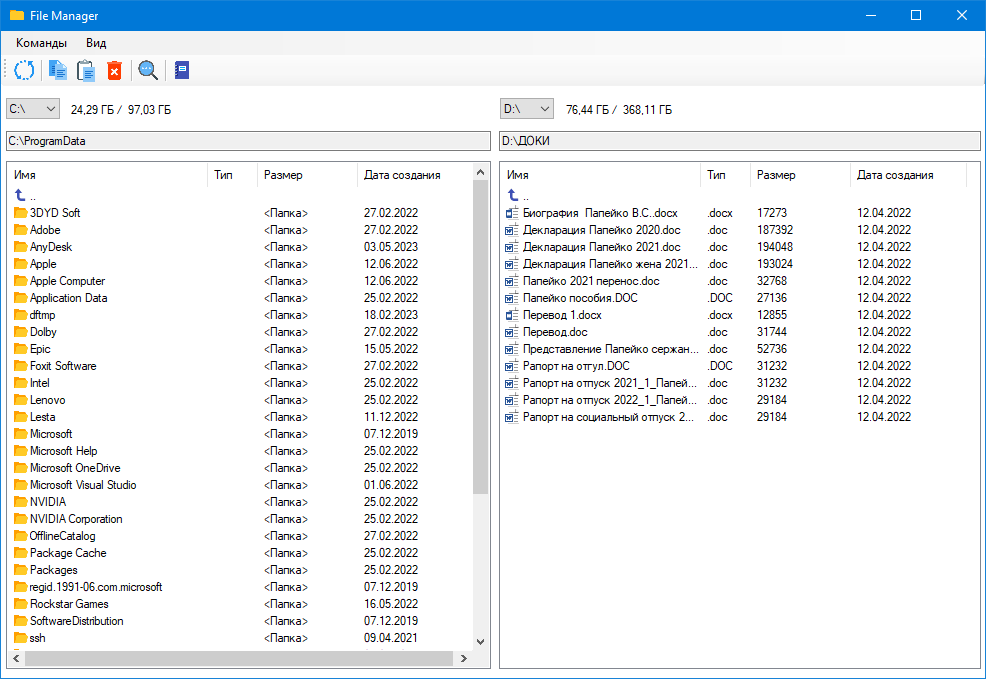


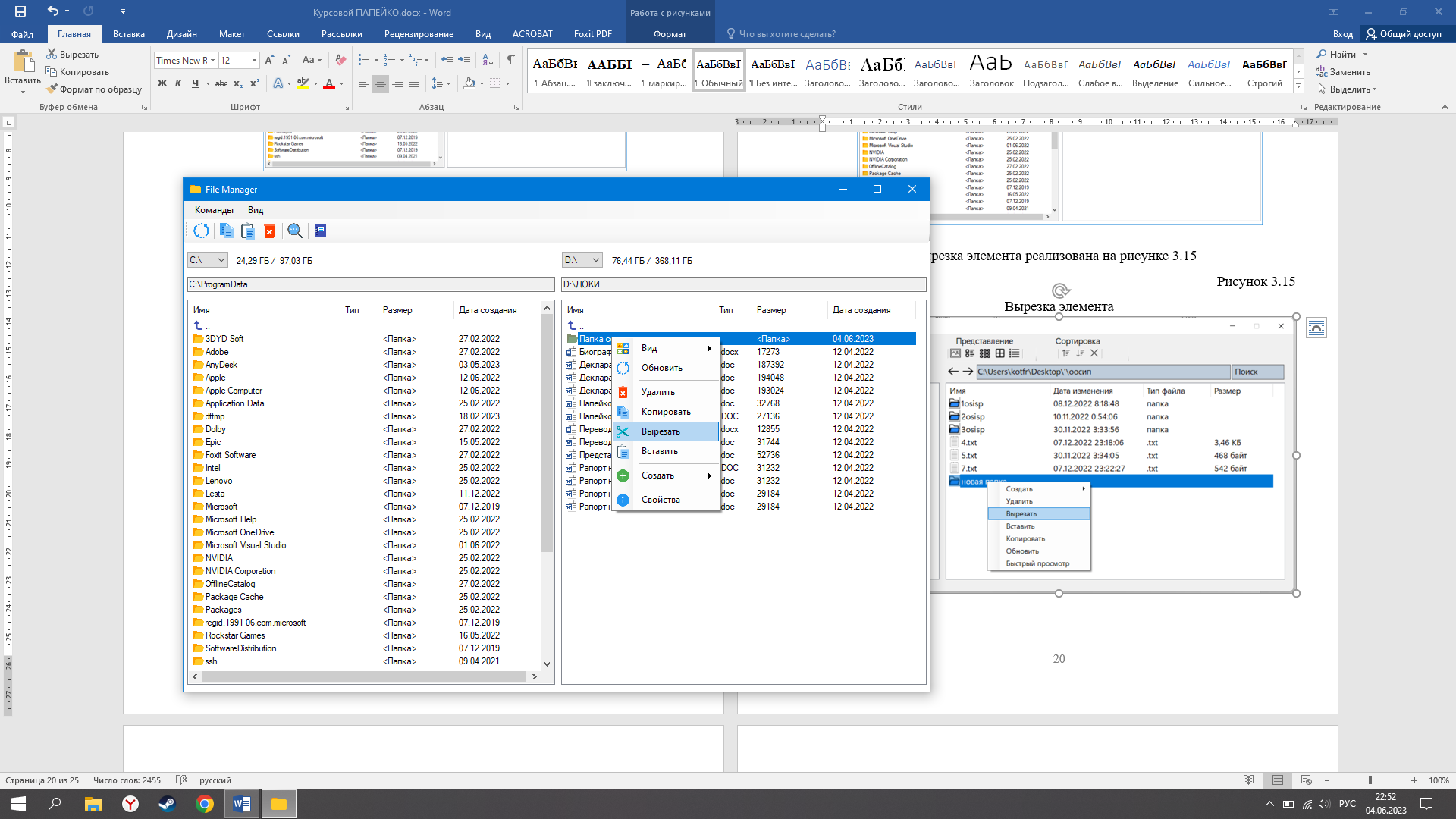
Рисунок 3.10



Функция вырезка элемента реализована на рисунке 3.11

Рисунок 3.11

Вырезка элемента



Функция вставки элемента реализована на рисунке 3.12-3.13

Рисунок 3.12

Вставка элемента

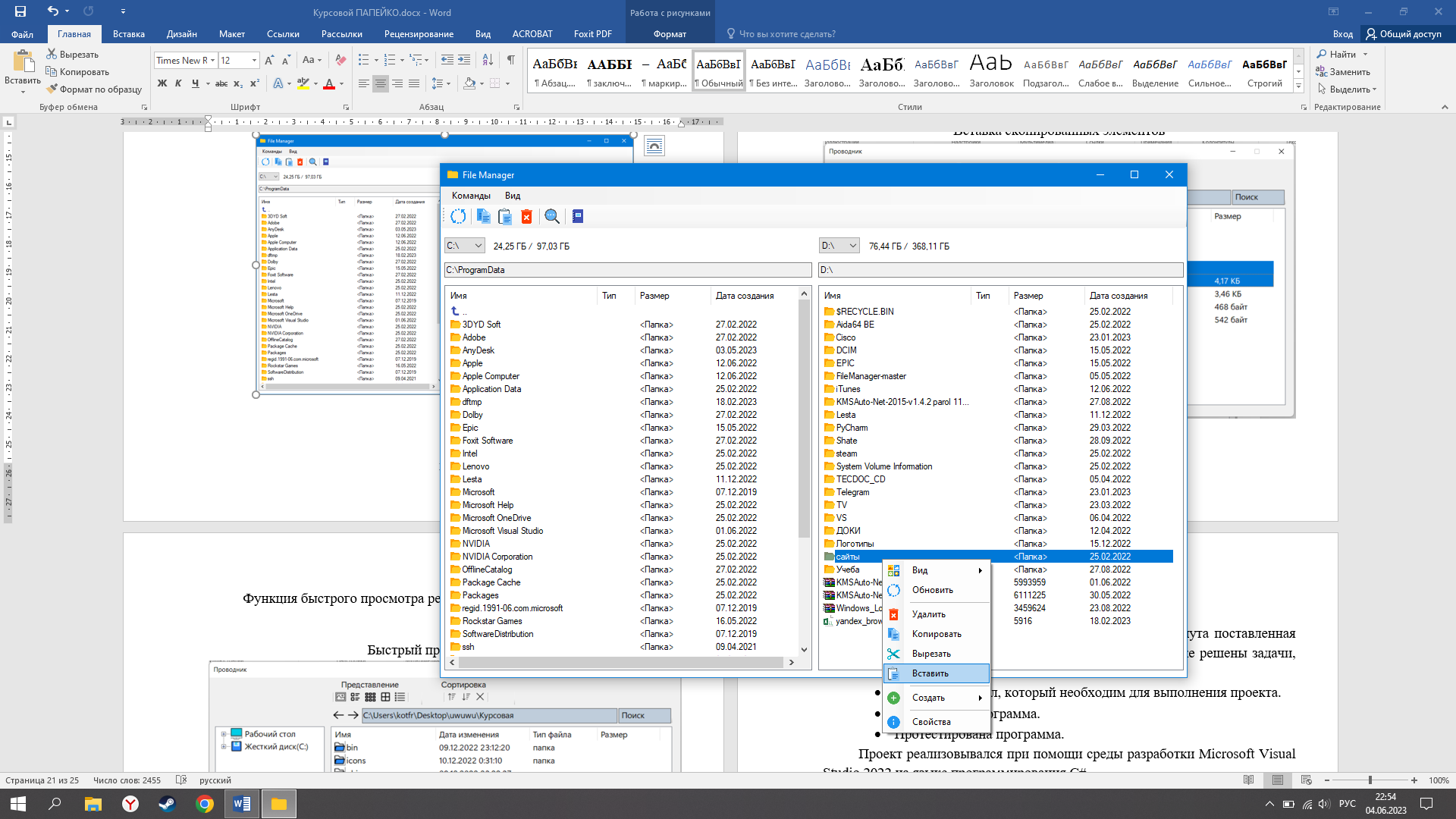
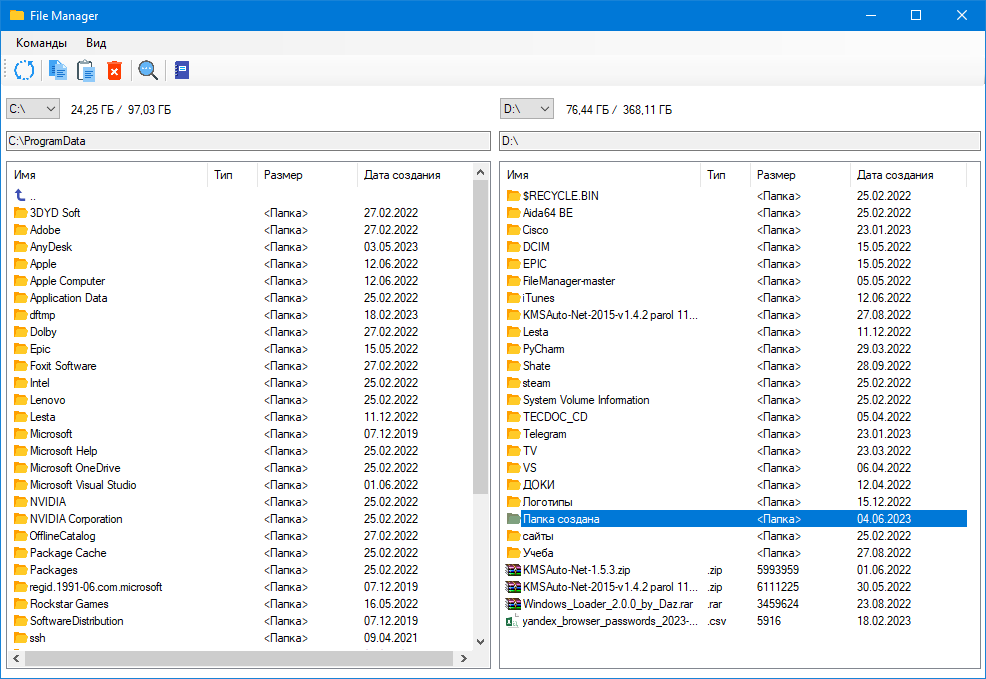


Рисунок 3.13

Вставка вырезанного элемента



Функция копирования элемента реализована на рисунках 3.14-3.15

Рисунок 3.14

Копирование элементов

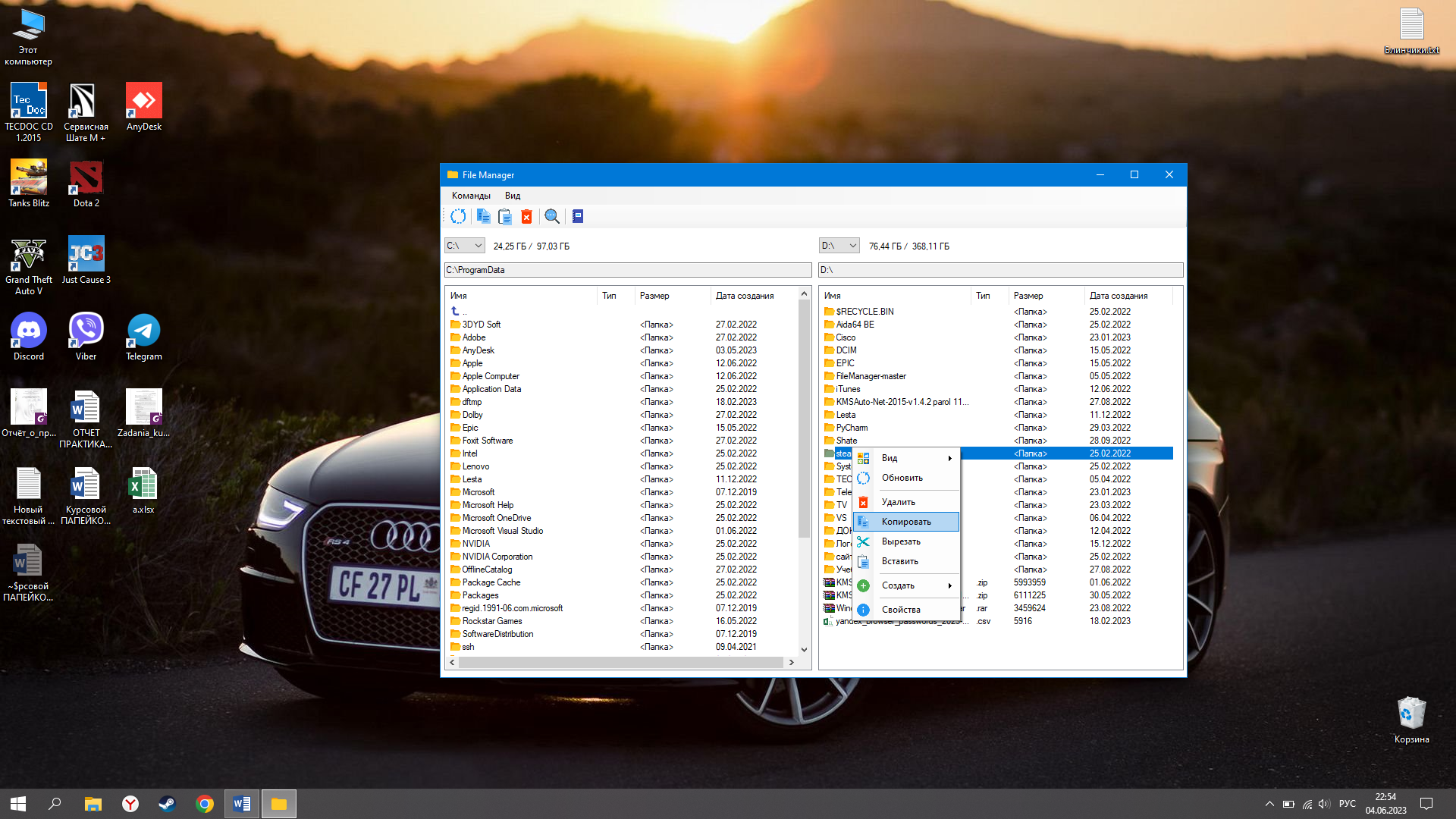
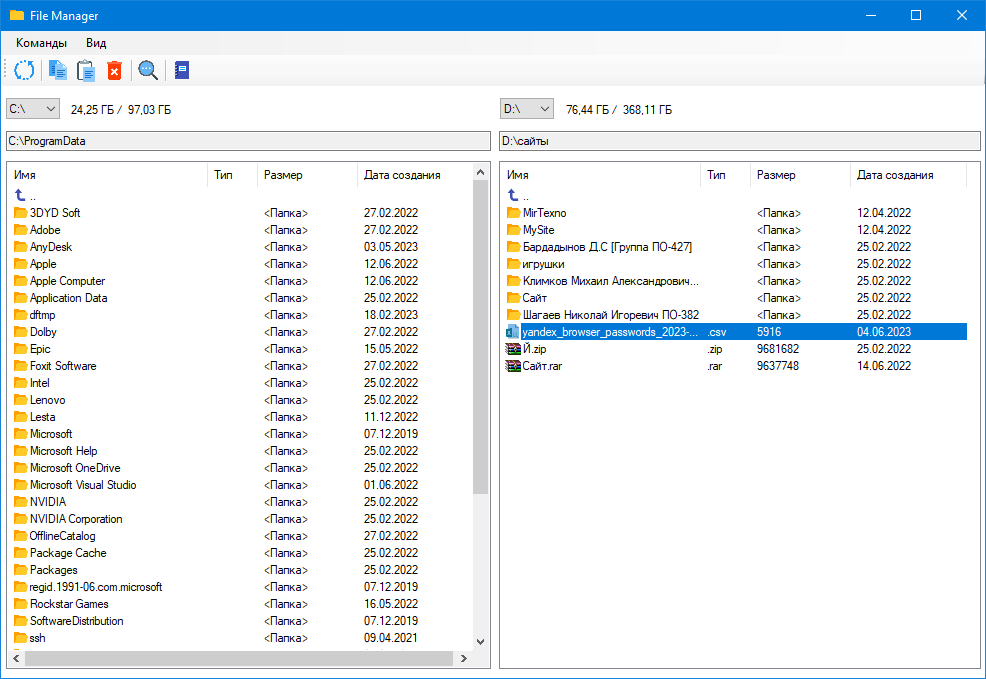


Рисунок 3.15

Вставка скопированных элементов



Заключение

В ходе выполнения курсового проекта была достигнута поставленная цель – удаление не удаляемого файла 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 2019 на языке программирования C#.

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

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

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

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

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

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

# 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.

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

# 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.