Федеральное государственное автономное образовательное учреждение
высшего образования «Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»
Институт радиоэлектроники и информационных технологий – РТФ
Учебно-научный центр «Информационная безопасность»

На правах рукописи

Князева Наталия Сергеевна

РАЗРАБОТКА МЕТОДИКИ ИДЕНТИФИКАЦИИ ПОСЛЕДОВАТЕЛЬНОСТИ ВНЕШНИХ ВОЗДЕЙСТВИЙ НА ДИНАМИЧЕСКУЮ СИСТЕМУ, ИЗОМОРФНУЮ КОНЕЧНОМУ АВТОМАТУ (НА ПРИМЕРЕ ВОССТАНОВЛЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ФАЙЛОВЫХ ОПЕРАЦИЙ В ОПЕРАЦИОННОЙ СИСТЕМЕ)

Специальность 2.3.1. Системный анализ, управление
и обработка информации

ДИССЕРТАЦИЯ
на соискание ученой степени
кандидата технических наук

Научный руководитель:
доктор технических наук, доцент
Духан Евгений Изович

Екатеринбург – 2021

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Актуальность темы исследования.

Современные информационные технологии не только стимулируют активное развитие общества, но и способствуют росту преступлений, в которых компьютер выступает средством их совершения. Так, согласно официальным данным правоохранительных органов, в 2018 г. было зарегистрировано 174 723 преступлений, совершаемых с использованием информационных технологий [1], в 2019 г. зарегистрировано на 68,5 % больше — 294 409 аналогичных деяний [2]. В 2020 г. количество компьютерных преступлений выросло еще на 73,4 %, они составили 25 % от всех совершаемых преступлений. Больше половины компьютерных преступлений относится к категориям тяжких и особо тяжких. Четыре пятых таких преступлений совершаются путем кражи или мошенничества, почти каждое одиннадцатое с целью незаконного производства, сбыта или пересылки наркотических средств [3].

Одним из важных этапов успешного раскрытия компьютерного преступления является проведение компьютерно-технической экспертизы (КТЭ), в результате которой могут быть получены доказательственные данные. В отличие от традиционных видов экспертизы, где для формирования полного, достоверного, научно обоснованного заключения используются методики двадцатилетней давности, для КТЭ это невозможно, так как механизмы и методы совершения компьютерных преступлений постоянно модифицируются [4].

Объектами КТЭ являются документы, базы данных, фотографии, видео, программы и т.д., которые хранятся в виде файлов — поименованных наборов данных, расположенных на машинных носителях информации [5]. В ходе КТЭ часто требуется установить, какие действия пользователь производил с объектами:

– каким способом интересующие следствие файлы появлялись на исследуемом носителе (были созданы на этом носителе или скопированы с других носителей информации);

– имели ли место факты распространения файлов (копирование на внешний носитель информации, отправка по электронной почте);

– имели ли место факты работы с файлами (открытие, редактирование, печать).

Таким образом, обобщение экспертной практики показывает, что восстановление последовательности операций, совершенных пользователем над файлами, является одной из важнейших задач при проведении КТЭ.

Для решения этой задачи исследуется служебная информация, регистрируемая в файловой системе (ФС) компьютера. ФС представляет собой структурированное хранилище каталогов и файлов.

Во время активных действий пользователя компьютера временные отметки (ВО) файла (состояния ФС) изменяются. Обычно при той или иной файловой операции (ФОп) одновременно изменяются не одна, а несколько ВО. При этом нередко наблюдается, что одноименные ВО, сохраняемые в различных атрибутах файловой записи, расходятся в значениях. Кроме того, одна и та же ФОп, но выполненная различными приемами или с использованием разных утилит, может приводить к неравным результатам в отношении ВО.

Современный этап развития компьютерной криминалистики характеризуется отсутствием научно обоснованных и поддающихся автоматизации методик восстановления ФОп. В настоящее время на восстановление последовательности ФОп, совершенных над одним файлом, эксперт тратит от одного дня до недели, а достоверно выявляет лишь последнюю ФОп для небольшого списка исследованных операций. Обеспечение достоверности и глубины восстановления цепочки ФОп требуют большого объема ручной работы и высокой квалификации специалиста. При этом количество обрабатываемых пользователем файлов таково, что эффективное восстановление последовательности ФОп для каждого из них оказывается невозможным.

Таким образом, существует объективная потребность в изучении закономерностей изменения ВО при различных действиях над файлами и создании методики восстановления последовательности ФОп по ВО, а также реализующего эту методику программного обеспечения, адаптированного для использования в практической деятельности эксперта-криминалиста.

Гипотеза настоящего исследования состоит в том, что с точки зрения восстановления ФОп, файловую систему можно рассматривать как дискретную динамическую систему, которая характеризуется состояниями в некоторые моменты времени. Состояниями системы называют совокупность значений некоторых ее характеристик [6]. Применительно к ФС под состоянием системы следует понимать совокупность значений метаданных исследуемых файлов [5], а именно их ВО.

Восстановление цепочки ФОп сводится к обратной задаче, для решения которой необходимо определить траекторию движения между начальным и конечным состояниями системы.

Разработанность темы исследования. В настоящее время к исследованию методов и алгоритмов проведения КТЭ проявляется повышенный интерес со стороны мировой экспертной и научной общественности. Весомый вклад в развитие этого направления внесли российские ученые Федотов Н.Н. [7], Шелупанов А.А. [8], Смолина А.Р. [9].

Серьезное внимание исследователи уделяют способам восстановления последовательности ФОп. Основоположниками данного направления являются Carrier B., Chow K., Parsonage H. Обзор их работ [10–34] позволил подтвердить принципиальную возможность анализа ВО с целью восстановления некоторых ФОп. Однако использование теоретических результатов на практике остается весьма сложной задачей. Большинство работ ограничены наблюдением не за всеми потенциально имеющимися в распоряжении специалиста ВО и позволяют, как правило, по явным признакам определять выполнение только одной последней ФОп, а не цепочки событий. Так, если файл был отредактирован после копирования на носитель, то факт копирования может быть уже не

установлен. Из-за подобных ограничений могут быть допущены серьезные ошибки при проведении КТЭ, или не все поставленные на КТЭ вопросы окажутся проанализированы в полном объеме.

Существуют универсальные криминалистические программные комплексы [35–40], предназначенные для проведения КТЭ и позволяющие просматривать ВО файлов и выводить их в упорядоченном виде. Но функцией анализа ВО с целью восстановления последовательности ФОп они не обладают. Процесс восстановления хронологии ФОп в указанных комплексах не предусмотрен.

К текущему моменту специалистами накоплен некоторый раздел знаний о причинно-следственных связях между ФОп и изменениями ВО файлов, но попыток формализации этих знаний до сих пор не предпринималось. В открытых источниках не обнаружены методики и алгоритмы, позволяющие автоматизировать анализ ВО. Практикой компьютерной криминалистики востребована разработка научно обоснованной методики восстановления хронологии ФОп, пригодной для автоматизации.

Объект исследования. Процесс восстановления последовательности ФОп.

Предмет исследования. Закономерности изменения ВО при совершении ФОп.

Целью работы является повышение количества восстанавливаемых ФОп, увеличение длины последовательности восстанавливаемых ФОп при снижении времени, затрачиваемого на восстановление хронологии ФОп.

Научной задачей работы является разработка математической модели изменения значений ВО и методики восстановления последовательности ФОп на основе анализа ВО файлов.

Цель диссертационной работы достигается последовательным решением следующих **частных задач:**

1. Анализ состояния предметной области.

2. Разработка алгоритма проведения экспериментов для формирования базы знаний о механизмах изменений ВО файлов при выполнении различных ФОп.

3. Выявление закономерностей изменений ВО при совершении ФОп. Разработка математической модели изменения значений ВО файлов при проведении над ними операций. Оценка адекватности модели.

4. Разработка методики восстановления последовательности ФОп. Создание программного обеспечения, реализующего данную методику. Формирование рекомендаций по применению программного обеспечения.

В результате проведенного исследования на защиту выносятся следующие **положения:**

1. Динамика изменения ВО файлов в ОС Windows изоморфна динамике изменения состояний конечного автомата и может быть представлена в виде конечного автомата, где состояниями являются множество состояний ВО файлов, а входным алфавитом — множество совершаемых ФОп [85].

2. Разработанный алгоритм анализа изменения ВО файлов, включающий этап подготовки ВО файлов путем присваивания им уникальных значений, позволяет в автоматическом режиме формировать полные таблицы изменения ВО в ФС NTFS во всех версиях ОС Windows на заданных пространствах ФОп и типах файлов [81, 83, 84, 86].

3. Восстановление последовательности операций, совершенных над файлом, сопоставимо с поиском траекторий, по которым автомат мог прийти в конечное состояние, является обратной задачей и решается с помощью алгоритма обхода в глубину [82].

Методы исследования. Для решения поставленных задач в работе использовались методы системного анализа, моделирования, теория автоматов, теория графов. Моделирование производилось с использованием программной среды MATLAB.

Границы исследования:

– Операционные системы Windows XP, 7, 8, 10.

– Файловая система NTFS.

Обоснованность выбранных границ исследования подтверждает статистика использования ОС персональных компьютеров. По официальным

статистическим данным компании StatCounter на октябрь 2020 г. ОС Windows представляет 76,32% доли рынка ОС для настольных ПЭВМ в мире. В ОС Windows по умолчанию используется ФС NTFS.

В работе исследуются следующие ФОп: копирование, перемещение (переименование), удаление, просмотр (изменение) атрибутов, открытие, редактирование, исполнение (запуск), помещение в архив, разархивирование. Перечень анализируемых операций основан на вопросах, которые в большинстве случаев задают эксперту-криминалисту при постановке задачи на проведение КТЭ.

Степень достоверности результатов исследования. Достоверность результатов диссертационной работы обеспечивается применением корректных исходных данных, апробированных методов исследований, проверкой непротиворечивости и адекватности положений и выводов, экспериментальными данными, полученными при апробации программного обеспечения, реализующего методику восстановления последовательности ФОп.

Научная новизна. В рамках проведенного исследования получены следующие новые научные результаты:

1. Впервые предложена математическая модель изменения значений ВО при совершении ФОп в ОС Windows, основанная на гипотезе изоморфизма динамики изменения ВО файлов и динамики изменения состояний конечного автомата (соответствует п. 2 паспорта специальности).

2. Впервые разработан алгоритм анализа изменения ВО, отличающийся специальной подготовкой ВО файлов и выявляющий полный набор закономерностей изменения ВО при совершении ФОп в ФС NTFS во всех версиях ОС Windows на заданных пространствах ФОп и типов файлов (соответствует п. 5 паспорта специальности).

3. Разработана автоматическая методика восстановления последовательностей ФОп, которая впервые решает обратную задачу путем адаптации алгоритма обхода в глубину к особенностям решаемой задачи (соответствует п. 12 паспорта специальности).

Теоретическая значимость работы заключается в развитии научно-методического аппарата КТЭ в части восстановления последовательности совершенных над файлом операций, основанного на системном анализе изменения ВО, хранящихся в файловой записи таблицы MFT.

Практическая значимость результатов исследования заключается в разработке программы, позволяющей в автоматическом режиме проводить анализ ВО, расположенных внутри файловой записи, и восстанавливать хронологию ФОп. Разработаны рекомендации по использованию ВО, хранящихся во внутренней структуре файла, для повышения количества восстанавливаемых ФОп.

Апробация результатов. Основные результаты работы докладывались и обсуждались на Международной конференции Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT) (Екатеринбург, май 2020 г.).

Публикации. По теме диссертации опубликовано 6 научных работ, в том числе 5 научных статей в рецензируемых изданиях, определенных ВАК РФ и Аттестационным советом УрФУ, из них 2 статьи в изданиях, индексируемых в международной цитатно-аналитической базе Scopus.

Структура и объем диссертации. Диссертация содержит 156 страниц текста, 72 рисунка, 19 таблиц, и состоит из содержания, введения, 4 глав, заключения, списка литературы, 7 приложений.

Личный вклад. Все результаты диссертационной работы получены автором самостоятельно.

Краткое содержание диссертационной работы. Во введении обоснована актуальность диссертационной работы, проанализировано состояние исследуемой проблемы, сформулированы цели и задачи работы, перечислены основные научные результаты диссертации, определена научная новизна и практическая ценность результатов, представлены основные положения, выносимые на защиту, приведены сведения об апробации работы, публикациях по теме работы, а также структура диссертации и ее объем.

В первой главе даны определения КТЭ и цифрового «следа». На конкретных примерах продемонстрировано, что в основе следообразования в компьютерной системе лежат стабильные программно-алгоритмические решения различных приложений, которые нигде не документированы. Предложена гипотеза о том, что ФС с точки зрения компьютерной криминалистики можно представить в виде «черного ящика». На вход «черному ящику» поступают сигналы в виде действий пользователя или программ. На выходе «черного ящика» эксперт имеет состояния ФС, которые характеризуются наличием или отсутствием определенных «следов». Изложены общие теоретические сведения о ВО, приводится их классификация. Проведен обзор публикаций по тематике восстановления последовательности ФОп и функционала программных средств, направленных на проведение криминалистических исследований компьютерных систем.

Во второй главе представлен инструментарий и алгоритм проведения экспериментальных исследований изменений ВО в ОС Windows. Определены закономерности в изменении ВО при совершении ФОп. Представлены результаты анализа фальсификации ВО и описаны следы, позволяющие обнаружить эту фальсификацию.

В третьей главе описываются модель изменения значений ВО и методика восстановления хронологии ФОп, основанная на анализе ВО, хранящихся в файловой таблице MFT. Представлены результаты экспериментальной проверки адекватности модели.

В четвертой главе описана функция, разработанная на основе методики восстановления ФОп. Приведены рекомендации по использованию ВО, хранящихся во внутренней структуре файла. Представлены примеры восстановления последовательности ФОп.

В заключении сформулированы основные результаты, полученные в диссертационной работе.

В приложениях приводятся листинги программ и элементы модели изменения значений ВО: таблица переходов между векторами временных уровней и таблица возможных выявленных состояний ВО.

Благодарности. Автор выражает благодарность кандидату технических наук, доценту Бакланову Валентину Викторовичу, доктору технических наук, профессору, члену-корреспонденту Академии криптографии РФ Гайдамакину Николаю Александровичу, научному руководителю доктору технических наук, доценту Духану Евгению Изовичу за всестороннюю помощь и поддержку в проведении исследований. Автор благодарит своих коллег кандидата технических наук Хорькова Дмитрия Алексеевича и Черемных Ивана Витальевича за помощь в организации экспериментов и ряд ценных советов, позволивших улучшить разработанную методику восстановления ФОп.

1 АНАЛИЗ СОСТОЯНИЯ ПРЕДМЕТНОЙ ОБЛАСТИ. ПОСТАНОВКА ЗАДАЧ ИССЛЕДОВАНИЯ

1.1 Файловая система как дискретная динамическая система в задачах компьютерной криминалистики

Компьютерно-техническая экспертиза — это экспертиза, занимающаяся исследованием компьютерных средств и систем в целях отнесения исследуемого объекта к компьютерному средству, системе, установления его функциональных особенностей, работоспособности в целом и получения доступа к компьютерным носителям информации с последующим всесторонним исследованием их содержимого [41].

В общем случае объектом КТЭ является компьютерная система, которая включает в себя аппаратные, программные, информационные и сетевые компоненты, работающие в отлаженной взаимосвязи. На основе компонентного деления выделяют аппаратно-компьютерную, программно-компьютерную, информационно-компьютерную и компьютерно-сетевую экспертизы [42].

Аппаратно-компьютерная экспертиза направлена на решение вопросов, связанных с исследованием технической (аппаратной) части компьютерных средств. В рамках аппаратно-компьютерной экспертизы требуется определить:

– сведения об аппаратной конфигурации компьютера на момент инсталляции ОС и на текущий момент;

– сведения о подключениях к проводным (USB, IEEE 1394) или беспроводным (IrDA, Bluetooth, UWB) интерфейсам.

Программно-компьютерная экспертиза направлена на решение вопросов, связанных с исследованием программного обеспечения. В рамках программно-компьютерной экспертизы требуется определить:

– наличие инструментальных средств для создания и компиляции программ, и признаки их использования;

– остаточную информацию о ранее установленных и впоследствии удаленных программах;

– признаки автозапуска программ;

– историю запуска программ (полное имя, место и количество запусков, время последнего запуска, общая продолжительность активной работы с программой).

Информационно-компьютерная экспертиза решает задачи, связанные с получением и исследованием данных, находящихся на компьютерных носителях информации (поиск, обнаружение и анализ файлов). В рамках информационно-компьютерной экспертизы требуется определить:

– признаки работы с файлами на ПЭВМ (открытие, редактирование, копирование и удаление);

– признаки копирования файлов на другой носитель информации;

– признаки преднамеренного удаления следов работы пользователей;

Компьютерно-сетевая экспертиза занимается исследованием обстоятельств и фактов, связанных с применением сетевых технологий. В рамках компьютерно-сетевой экспертизы требуется определить то же, что и в рассмотренных выше видах экспертиз с той лишь разницей, что ее объекты функционируют в сети.

Для решения задач каждого типа компьютерной экспертизы исследуются компьютерные «следы», которые хранятся в компьютерной системе. Цифровой «след» представляет собой криминалистически значимую компьютерную информацию о событиях или действиях, отраженную в материальной среде, в процессе ее возникновения, обработки, хранения и передачи [43]. В ОС Windows «следы» могут храниться в файлах системного реестра, системных журналах, файлах конфигурации, файлах гибернации и подкачки, системных каталогах (Prefetch, Recent и др.), таблицах файловых систем FAT и NTFS и т.д. «Следы» в памяти компьютера остаются благодаря объективным факторам, в основе которых лежат стабильные и многократно проверенные программно-алгоритмические решения ОС, например:

1. При подключении к порту USB устройства в ОС инициируется серия запросов к устройству с различными типами запрашиваемых параметров. Данные, возвращаемые устройством, в виде ответов представляют собой идентификаторы, которые используются ОС в процессе опознавания устройств. После завершения опроса на основе полученных параметров в системе создаются идентификаторы, которые сохраняются в разделах системного реестра HKLM[1]\System\CurrentControlSet\Enum\USB, HKLM\System\CurrentControlSet\Enum\USBSTOR, файле Setupapi.dev.log, записях системных журналов System.evtx (коды событий: 10000, 20001, 20002), Microsoft-Windows-Kernel-PnP%4Configuration.evtx (коды событий: 400, 410, 420, 430).

2. При инсталляции программы в разделе системного реестра HKLM\Software\Microsoft\Windows\CurrentVersion\Uninstall создается подраздел с именем устанавливаемой программы, который хранит сведения о программе (название, разработчик, версия, дата установки, полный путь до каталога с конфигурационными файлами программы), в системном журнале Microsoft-Windows-Application-Experience%Program-Inventory.evtx создаются события с кодом 903, 904, 907, 908. Служба Superfetch, реализующая механизм предвыборки, обеспечивающий помещение в оперативную память информации о часто запускаемых программах, создает в каталоге Prefetch файл с расширением pf, имя которого частично совпадает с именем исполняемого файла (именем программы). Служба теневого копирования томов (Volume Shadow Service) создает теневую копию, что отражается в системном журнале Application.evtx (коды событий: 8194, 8212).

3. При открытии файла служба отслеживания изменившихся связей (Distributed Link Tracking Client) в каталоге Recent создает ярлык, имя которого совпадает с именем открываемого файла. Ярлык содержит полный путь до файла, время создания, изменения и последнего доступа к нему. При повторном открытии файла в уже существующем ярлыке обновляются только значения отметок времени.

[1] HKLM — HKEY_LOCAL_MACHINE

Как видно из приведенных примеров действия пользователей и программ отражаются в изменении содержимого уже существующих системных файлов или создании новых файлов. При этом детерминированность алгоритма следообразования в ОС позволяет определить причины, которые послужили созданию наблюдаемых изменений в ФС, например:

1. Наличие подразделов в разделах реестра HKLM\System\CurrentControlSet\Enum\USB, HKLM\System\CurrentControlSet\Enum\USBSTOR свидетельствует о подключении USB устройства.

2. Наличие подразделов в разделах реестра HKLM\Software\Microsoft\Windows\CurrentVersion\Uninstall свидетельствует об установке программы.

3. Наличие ярлыков в каталоге Recent свидетельствует об открытии файла.

Таким образом, ФС с точки зрения компьютерной криминалистики абстрактно можно представить в виде «черного ящика», на вход которому поступают сигналы в виде действий пользователя или программ, а на выходе имеется состояние ФС, наблюдаемое экспертом при проведении КТЭ и характеризуемое наличием или отсутствием определенных «следов».

Следовательно, процесс следообразования в ОС можно описать конечным автоматом, который в любой заданный момент времени однозначно находится в одном из состояний конечного множества. Под состояниями рассматриваются элементы ФС, которые изменяются при действиях пользователя, например, атрибуты и содержимое файлов, разделы или параметры системного реестра, записи журналов событий. С течением времени автомат может изменить свое состояние. Переходы между состояниями инициируются действиями программ или пользователя и точно определены. Автомат является детерминированным, так как из любого своего состояния по поступлению любого входного воздействия он может перейти лишь в одно состояние, а не во множество (состояния меняются одно за другим последовательно). Тогда при конечном известном множестве возможных состояний определение причин изменения состояний автомата, то есть действий пользователя, инициирующих эти изменения,

является обратной задачей и решается методом прямого перебора всех возможных маршрутов до конечного состояния.

Применительно к задаче восстановления последовательности операций, совершенных пользователем над файлами, под состояниями должны рассматриваться элементы ФС, которые достаточно описывают систему в моменты времени между переходами. Такими элементами являются ВО файлов, которые хранятся в области метаданных ФС, а также в системных файлах специальных форматов.

Для одного файла в ФС может храниться значительное количество ВО. Так, например, сравнение содержимого вкладок «Общие» и «Подробно» окна «Свойства» оболочки Explorer (рис. 1.1) позволяет сделать вывод о том, что информация о ВО файла «Пример.xlsx» для этих вкладок получена из разных источников. Более того, эти ВО не совпадают между собой, что несет информацию о той или иной совершенной ФОп. Восстановление интересующей следствие хронологии событий возможно на основе тщательного исследования соотношений указанных ВО.

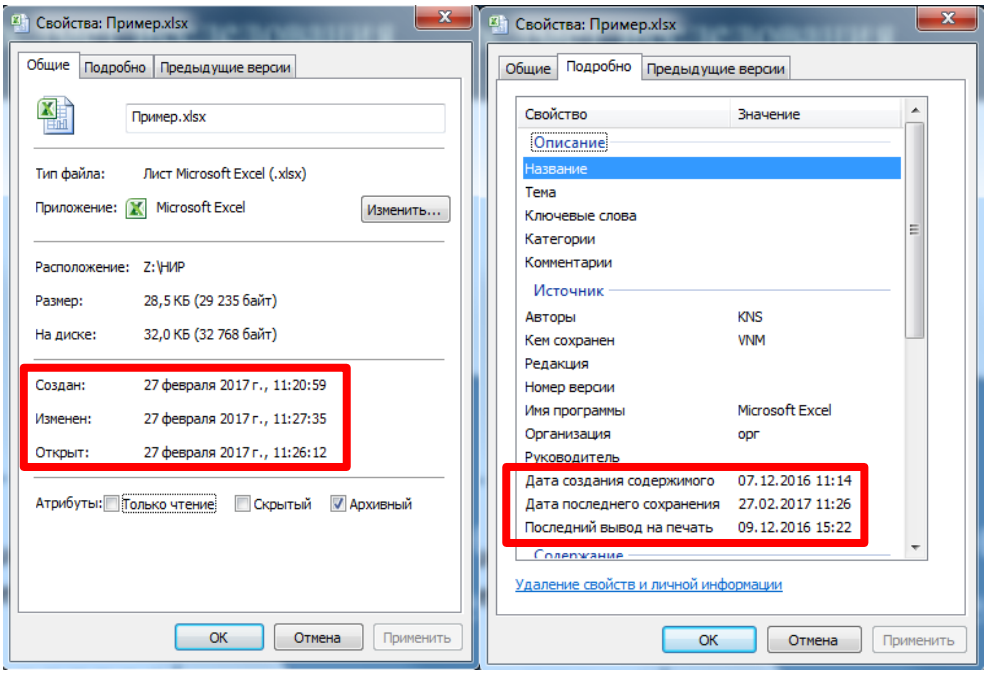


Рис. 1.1. Окно «Свойства» для файла «Пример.xlsx»

1.2 Феноменология ВО в ФС NTFS в ОС Windows

Временная отметка (метка времени, timestamp) — значение момента времени, зафиксированное в виде последовательности символов или закодированной информации. ВО может иметь полное представление, включающее все элементы даты и времени дня, или комбинированное [45].

В ОС Windows существует множество способов представления ВО. Тактика фиксации времени осуществляется разработчиками программного обеспечения на этапе проектирования. Ниже перечислены самые распространенные форматы представления времени:

– текстовое представление;

– преставление в форме отсчетов;

– представление в двоичном формате.

В текстовом представлении значение ВО содержит дату и время, а также может фиксировать указание на часовой пояс. Примером текстового представления является формат ISOString [45, 46] — строка вида ГГГГ-ММ-ДДТчч:мм:сс, где: ГГГГ — год, ММ — месяц, ДД — день, чч — часы, мм — минуты, сс — секунды.

ВО в форме отсчетов фиксируют количество отрезков времени, прошедших с заданного момента времени:

$$T = T_0 + Q \cdot \text{ВО}, \tag{1.1}$$

где значением T обозначено фиксируемое время произошедшего события;

T_0 — начальный момент времени;

Q — размер промежутка времени (квант времени);

ВО — хранимая временная отметка — количество квантов времени Q, прошедших с начального момента времени T_0.

Для разных типов ВО значения T_0 и Q различаются.

Примером представления ВО в форме отсчетов является формат FILETIME — 64-разрядные значения, представляющие количество сотен

наносекунд, прошедших с 1 января 1601 г. [10]. Формат FILETIME фиксирует ВО без поправок на часовой пояс (UTC+0).

ВО в двоичном формате содержит закодированные дату и время без поправки на часовой пояс (UTC+0). Примерами такого формата являются FTIME MS-DOS [10] и MS-DOS: wFatDate wFatTime. В формате FTIME MS-DOS временные отметки — 32-разрядные значения, где время — 16 разрядное значение, состоящее из трех частей: с 00 по 04 — количество пар секунд, с 05 по 10 — количество минут, с 11 по 15 — количество часов; дата — 16 разрядное значение, состоящее из трех частей: с 00 по 04 — день, с 05 по 08 — месяц, с 09 по 15 — количество лет, прошедших с 1980-го г. Формат MS-DOS: wFatDate wFatTime отличается от FTIME MS-DOS порядком следования даты и времени: сначала дата, а затем время (рис. 1.2).

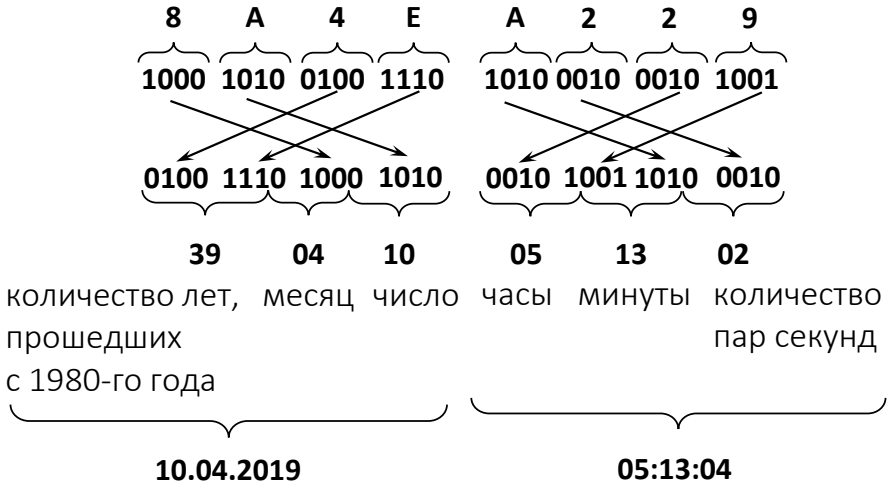


Рис. 1.2. Преобразование ВО из формата MS-DOS: wFatDate wFatTime

На основании изученных работ [10–34] можно выделить три основных места хранения ВО в ОС Windows:

− в главной файловой таблице MFT (Master File Table): в файловых записях файлов и в индексных записях родительских каталогов. Такие ВО контролируются и управляются ОС. Далее будем называть их *внешними ВО.*

− в структуре самого файла. Такие ВО обновляются конкретной прикладной программой, «закрепленной» за данным форматом файла. В то же время существуют файлы, в чьих структурах ВО не сохраняются. Далее будем называть их *внутренними ВО.*

− в специальных системных файлах, необходимых для функционирования ОС: ярлыках и списках переходов (JumpLists), в ульях реестра (SYSTEM, SAM, SOFTWARE и др.), в журналах событий (System.evtx, OAlerts.evtx и др.). Такие ВО контролируются и управляются ОС и приложениями. Далее будем называть их *дополнительными ВО.*

Далее рассмотрим каждую категорию ВО на целесообразность их использования для восстановления последовательности ФОп.

1.2.1 Внешние ВО

Для каждого файла в MFT создается как минимум одна файловая запись (далее — запись), размер которой составляет 1 Кбайт. Запись хранит основную информацию о файле (рис. 1.3). Первые 42 байт — заголовок записи, который содержит сигнатуру (ASCII-строка «FILE» или «BAAD») и основные сведения об объекте, например, флаг использования записи, флаг определения типа (каталог или файл) и другое [10]. Остальные 982 байта делятся между атрибутами — структурами данных — и неиспользуемым пространством.

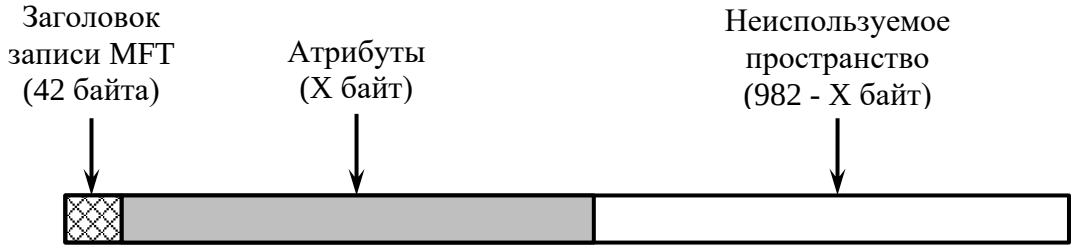


Рис. 1.3. Структура записи MFT

В ОС Windows предусмотрено большое количество атрибутов, каждый из которых обладает собственной внутренней структурой, тем не менее, все атрибуты содержат две части: заголовок и содержимое. Заголовок универсален и стандартен. Его содержимое зависит от типа атрибута и может иметь произвольный размер. Заголовок определяет тип атрибута, его размер и имя. Тип атрибута представляет собой числовой код, ассоциированный с типом хранящихся данных, и обладает именем, которое состоит из прописных букв и начинается со знака «$» (табл. 1.1).

Таблица 1.1
Стандартные типы атрибутов

Идентификатор типа	Имя	Содержимое
16	$STANDARD_INFORMATION	Флаги, ВО, владелец, идентификатор системы безопасности
32	$ATTRIBUTE_LIST	Список других атрибутов файла
48	$FILE_NAME	Имя файла в Unicode, ВО
128	$DATA	Содержимое файла
144	$INDEX_ROOT	Корневой узел индексного дерева
160	$INDEX_ALLOCATION	Узлы индексного дерева, корень которого определяется атрибутом $INDEX_ROOT

Атрибуты могут быть резидентными (содержимое атрибутов хранится в записях MFT после заголовка) и нерезидентными (содержимое хранится во внешних кластерах ФС, адрес которых содержится в заголовке). Пример типовой записи MFT представлен на рис. 1.4.

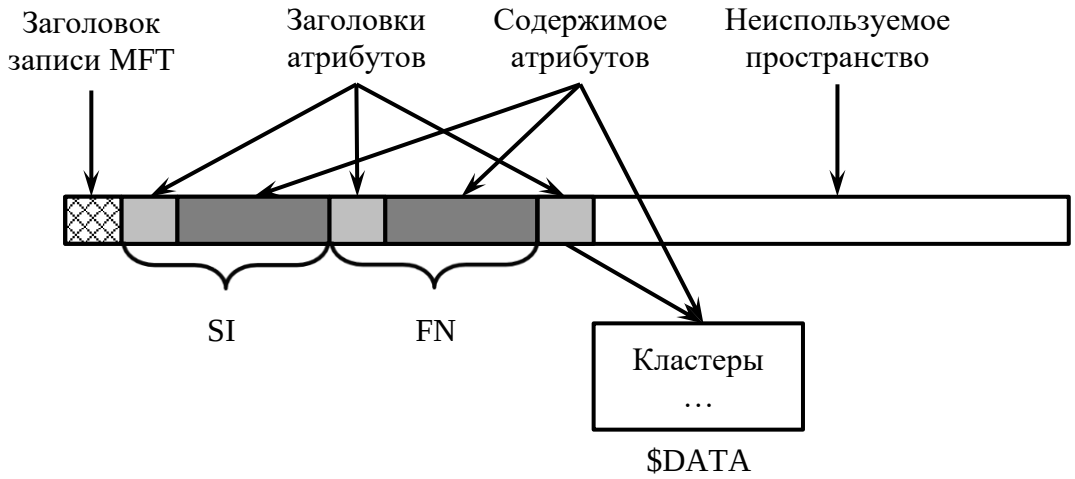


Рис. 1.4. Пример записи MFT

Почти все записи MFT содержат атрибуты типов $STANDARD_INFORMATION (SI) и $FILE_NAME (FN). Единственное исключение составляют не базовые дополнительные записи, ассоциируемые с файлом. Они создаются в том случае, если для описания атрибутов файла недостаточно выделенного пространства в 1 Кбайт. Таким образом, все файлы имеют атрибуты SI и FN. Оба атрибута являются резидентными и содержат по четыре ВО:

1. *Create (C)* — время, в которое была создана файловая запись.

2. *Modify (M)* — время последнего изменения атрибута $DATA (содержимого файла).

3. *Время модификации MFT (X)* — время последней модификации метаданных файла.

4. *Access (A)* — время последнего обращения к содержимому файла.

В таблице MFT ВО хранятся в формате FILETIME. На рис. 1.5 представлено содержимое записи MFT типичного файла, где контурами на красном фоне выделены ВО из атрибута SI, контурами на сером фоне — ВО из атрибута FN.

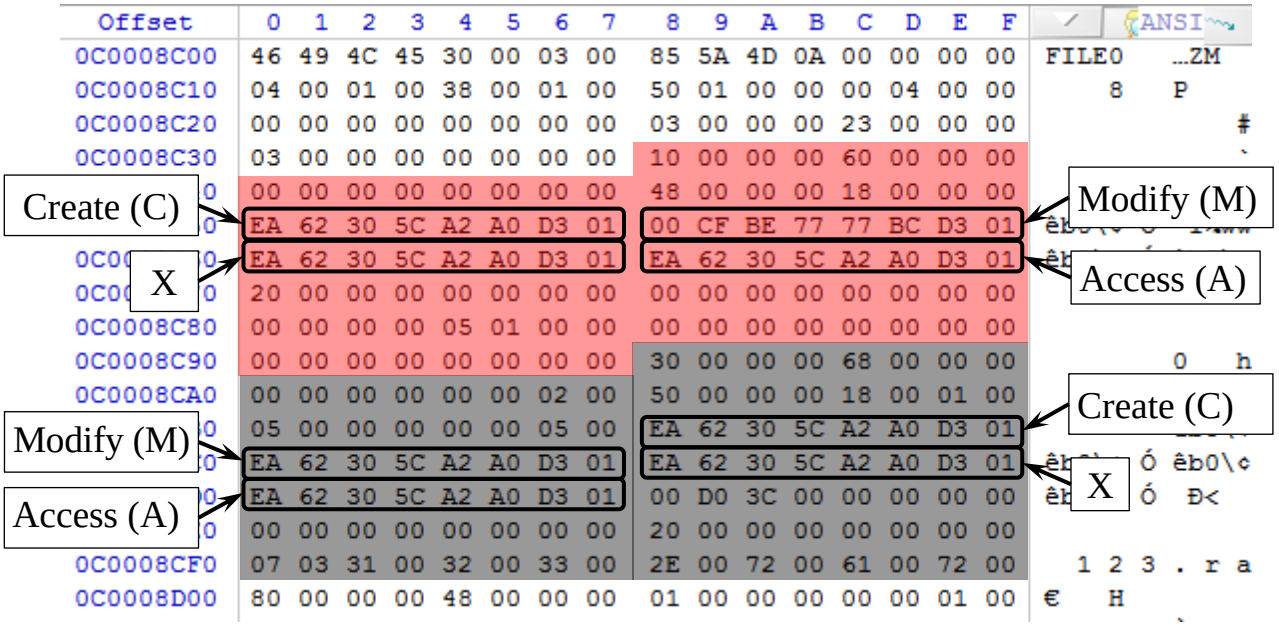


Рис. 1.5. ВО файла, расположенные в записи MFT

ВО файлов также можно найти в индексных записях родительских каталогов. Индексные записи представляют собой структурированные списки, которые используются для упорядочения содержимого каталогов. Они располагаются в атрибутах $INDEX_ROOT и $INDEX_ALLOCATION записей MFT. При этом $INDEX_ROOT всегда является резидентным, а $INDEX_ALLOCATION — нерезидентным, то есть содержимое этого атрибута хранится вне таблицы MFT.

Индексная запись начинается с заголовка, который содержит сигнатуру «INDX» и служебные данные. Затем следуют заголовок индексного узла и список индексных элементов (далее — элементов), где пустой элемент обозначает конец списка (рис. 1.6).

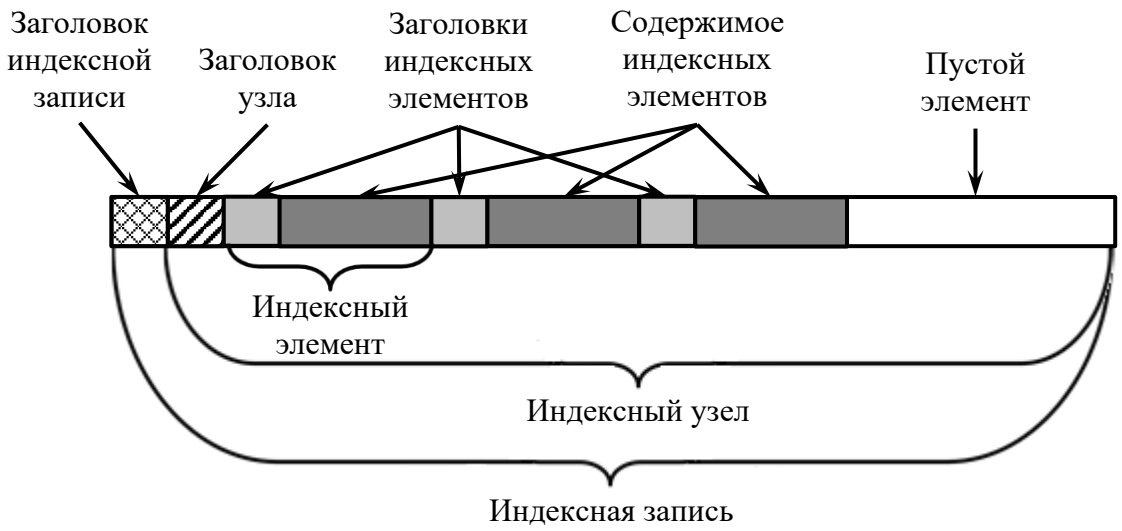


Рис. 1.6. Структура индексной записи

Заголовок узла хранит смещения начала и конца списка элементов. Индексные элементы в свою очередь состоят из заголовка и содержимого. Один элемент соответствует одному файлу (рис. 1.7).

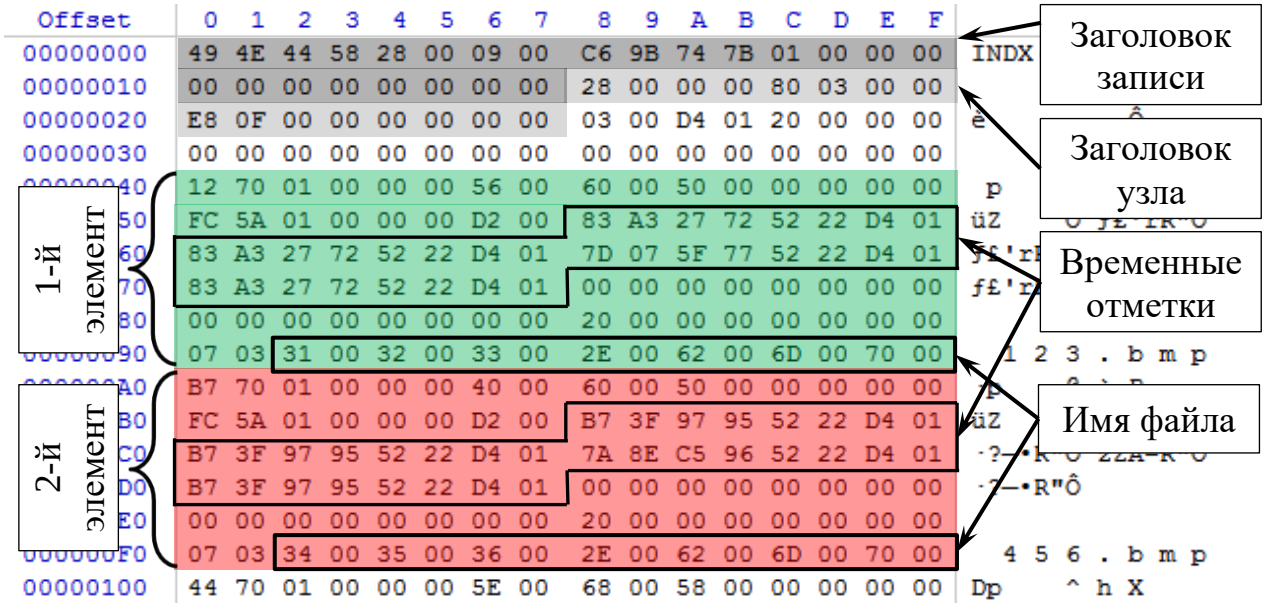


Рис. 1.7. ВО файлов,
расположенные в индексной записи каталога

Заголовок элемента определяет адрес записи MFT соответствующего файла и размер содержимого элемента. Содержимое элемента представляет собой атрибут FN, в котором хранятся имя файла и его ВО (рис. 1.7).

Таким образом, для одного файла существуют двенадцать обязательных ВО: в атрибутах файловой записи SI и FN файла и в индексных записях каталога, в котором находится файл, хранятся по четыре ВО каждого типа:

1. время, в которое была создана файловая запись; 2. время последнего изменения содержимого файла; 3. время последней модификации метаданных файла; 4. время последнего обращения к содержимому файла.

С помощью штатных средств ОС данные ВО невозможно удалить или изменить. Для считывания трех ВО, а именно: создания, изменения и доступа к файлу из атрибута SI, используется штатная функция Windows API GetFileTime (рис. 1.8). В инсталляции ОС Windows штатных утилит для извлечения всех двенадцати ВО файлов не существует.

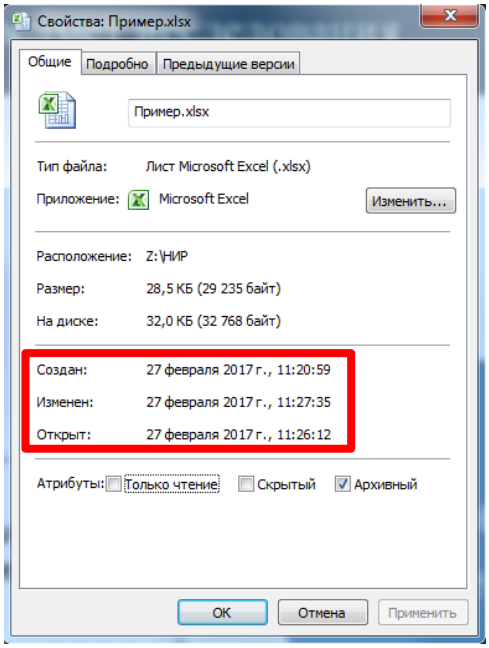


Рис. 1.8. Окно «Свойства» файла

На сегодняшний день существует несколько работ, посвященных изучению процессов изменения внешних ВО. В этих работах используется единый подход к исследованию, который состоит в том, что закономерности в изменениях ВО выявляются экспериментальным путем. Авторы фиксируют и сравнивают значения ВО до и после совершения анализируемой ФОп. Следует добавить, что подобные исследования выполняются вручную и требуют колоссальных временных затрат и высокой квалификации эксперта. Ниже приведен краткий обзор наиболее информативных исследований.

Б. Кэрриэ (B. Carrier) в работе [10] одним из первых упоминает о существовании ВО файлов и каталогов. Он проводил наблюдения за четырьмя ВО из атрибута SI в ОС Windows XP. Опция обновления ВО последнего доступа (A)

была включена[1]. В результате исследований Б. Кэрриэ определил, как изменяются ВО при совершении 4 ФОп: создание, копирование, перемещение, удаление. При создании файла его ВО устанавливаются равными текущему системному времени. При копировании для исходного файла обновляется значение ВО последнего обращения, а у копии — значения ВО последнего обращения и создания. При перемещении файла изменяются значения ВО обращения. При удалении файла значения ВО не обновляются.

К. Чоу, Ф. Лоу, М. Кван, К. Лаи (K. Chow, F. Law, M. Kwan, K. Lai) в статье [11] исследуют три ВО из атрибута SI главной файловой таблицы MFT в ОС Windows XP. Проводя эксперименты над текстовыми файлами и изображениями, авторы приходят в основном к тем же выводам, что и Б. Кэрриэ. В развитие идей Б. Кэрриэ авторы описывают факторы, которые влияют на корректность значений ВО, например изменение системного времени или состояние опции обновления ВО A. Авторы обнаруживают, что при включенной опции ВО A, она будет изменяться при любом взаимодействии с файлом, даже при наведении курсора на его пиктограмму. Следует учесть, что для ОС Windows начиная с Windows 7 значение данного параметра по умолчанию равно 1.

Т. Кнутсон (Tony Knutson) в статье [12] аккумулирует знания, полученные предыдущими авторами, подкрепляя их собственными экспериментами. Т. Кнутсон проводил наблюдения за четырьмя ВО из атрибута SI главной файловой таблицы MFT в ОС Windows XP, 7, 8. Для извлечения и отображения ВО использовалась программа FTK Imager (версия 3.1.1.8). Опция обновления ВО A в ОС Windows 7, 8 была выключена, а в ОС Windows XP — включена. В результате исследований Т. Кнутсон определил, как изменяются ВО при

[1] В файловой системе NTFS существует возможность отключать обновление времени последнего доступа к файлам. Согласно документации Microsoft, эта возможность предназначалась для увеличения быстродействия. Чтобы активировать эту опцию необходимо параметр HKLM\System\CurrentControlSet\Control\FileSystem\NtfsDisableLastAccessUpdate установить в значение 0 [44]. В ОС Windows 7, 8, 10 данный параметр по умолчанию выключен. В ходе экспериментов выявлено, что характер изменений ВО зависит от состояния этой опции.

совершении 3 ФОп: копирование, перемещение, редактирование. В работе перемещение и копирование проводилось как в пределах одного тома с ФС NTFS, так и между томами.

В.С. Матвеева в статье [13] проводила наблюдения за 8 ВО из атрибутов SI и FN в ОС Windows XP, 7. Опция обновления ВО A была включена. Для извлечения и отображения ВО использовалась команда «istat» в программе The Sleuth Kit (TSK). В результате исследований В. Матвеева определила характер изменений ВО при совершении 7 ФОп: перемещение, копирование, удаление, открытие, редактирование, просмотр и изменение атрибутов файла. Перемещение и копирование проводилось как в пределах одного тома, так и между томами. Кроме того, Матвеева считает, что по сравнению одноименных ВО из атрибутов SI и FN можно выявлять факты их фальсификации, основываясь на том, что ВО из атрибута FN не могут превосходить ВО из атрибута SI.

Г. Чо (Gyusang Cho) в работе [14] проводил наблюдения за 8 ВО из атрибутов SI и FN в ОС Windows 7. Опция обновления ВО A была выключена. В результате исследований GS. Cho определил, как изменяются ВО при совершении 5 ФОп: переименование, перемещение, копирование, редактирование, изменение атрибутов файла. Перемещение и копирование проводилось как в пределах одного тома, так и между томами.

Таким образом, считается целесообразным исследовать изменения внешних ВО с целью установления закономерностей между выполняемыми ФОп и изменениями ВО:

− в вышеперечисленных работах продемонстрировано, что по ВО, хранящимся в файловой записи таблицы MFT, можно определить ряд значимых ФОп, таких как копирование, удаление, редактирование, перемещение, открытие;

− таблица MFT является основной структурой данных ФС NTFS. Каждый реально существующий файл и некоторые удаленные файлы (пока файловая запись не перезаписалась) имеют в таблице MFT файловые записи, которые в свою очередь хранят внешние ВО с высокой точностью — 10^{-7} с;

− с помощью штатных средств ОС Windows нельзя модифицировать или удалить внешние ВО.

1.2.2 Внутренние ВО

Как было ранее сказано некоторые форматы файлов предусматривают наличие ВО внутри своей структуры. Виды представления внутренних ВО, создаваемых прикладным ПО, и место их размещения в файлах различных форматов сведены в табл. 1.2.

Таблица 1.2

Виды представления внутренних ВО в файлах различных форматов

Формат	Вид представления	Формат представления ВО	Место размещения
1	2	3	4
.doc	Шестнадцатеричное с прямым порядком байт (little endian)	FILETIME	1. После сигнатуры «Microsoft Office Word» + 29 байт — ВО «Последний вывод на печать» 2. После сигнатуры «Microsoft Office Word» + 39 байт — ВО «Дата создания содержимого» 3. После сигнатуры «Microsoft Office Word» + 51 байт — ВО «Дата последнего сохранения»
.xls	Шестнадцатеричное с прямым порядком байт (little endian)	FILETIME	1. После сигнатуры «Microsoft Excel» + 13 байт — ВО «Последний вывод на печать» 2. После сигнатуры «Microsoft Excel» + 25 байт — ВО «Дата создания содержимого» 3. После сигнатуры «Microsoft Excel» + 37 байт — ВО «Дата последнего сохранения»
.ppt	Шестнадцатеричное с прямым порядком байт (little endian)	FILETIME	1. После сигнатуры «Microsoft Office PowerPoint» + 25 байт — ВО «Последний вывод на печать» 2. После сигнатуры «Microsoft Office PowerPoint» + 37 байт — ВО «Дата создания содержимого» 3. После сигнатуры «Microsoft Office PowerPoint» + 49 байт — ВО «Дата последнего сохранения»
.jpg (фото-графии)	Текстовая строка в ASCII-кодировке	YYYY:MM:DD HH:mm:ss	После сигнатуры «DataTime Original» — «Дата съемки»
.pdf	Текстовая строка в ASCII-кодировке	YYYYMMDD HHmmss (YYYY-MM-DDT HH:mm:ss)	1. После сигнатуры «CreationDate» («CreateDate») — ВО создания файла 2. После сигнатуры «ModDate» («ModifyDate», «LastModified») — ВО модификации файла

Формат	Вид представления	Формат представления ВО	Место размещения
.docx, xlsx, pptx	Текстовая строка в UTF-8-кодировке	YYYY-MM-DDThh:mm:ss.sTZD	В файле «core.xml»: 1. После сигнатуры «lastPrinted» — ВО «Последний вывод на печать» 2. После сигнатуры «created» — ВО «Дата создания содержимого» 3. После сигнатуры «modified» — ВО «Дата последнего сохранения»

Внутренние ВО отображаются на вкладке «Подробно» окна «Свойства». Кроме того, внутренние ВО файлов отображаются в программах, ориентированных на работу с данным форматов файлов. Например, программы Foxit Reader на вкладке «Description» или Adobe Reader на вкладке «Properties» раздела главного меню «File» позволяют просмотреть внутренние ВО файлов с расширением .pdf. Просмотр свойств изображений можно производить с использованием специализированного ПО, ориентированного на работу с EXIF-заголовками, например программы ExifReader, либо с использованием программ для работы с изображениями: AdobePhotoshop, ACDSee, FastStone Image Viewer, поддерживающими стандарт EXIF.

Э. Дидриксен (Espen Didriksen) исследовал формат файлов Office Open XML в работе [18] и определил, что в структуре файлов данного формата фиксируются время создания содержимого документа, последнего вывода на печать и последнего сохранения. В результате исследований этих ВО Э. Дидриксен сформулировал следующие закономерности:

1. ВО, хранящиеся внутри файлов Office Open XML, округляются до единиц минут;

2. Если ВО создания из файловой записи таблицы MFT и ВО «Дата создания содержимого», хранящаяся внутри файлов Office Open XML, совпадают, то файл не является копией;

3. Если ВО создания из файловой записи таблицы MFT установлена позже ВО «Дата создания содержимого», хранящейся внутри файлов Office Open XML, то файл является копией;

4. ВО «Последний вывод на печать», хранящаяся внутри файлов Office Open XML, не всегда фиксируется во внутренней структуре документа. Для

того чтобы время печати было зафиксировано, необходимо, чтобы после печати документ был сохранен. И даже наличие такой отметки не позволяет выявить, был ли осуществлен вывод документа в твердой копии или была осуществлена печать в файл;

5. Если ВО «Последний вывод на печать» установлена ранее чем ВО «Дата создания содержимого», то файл является копией. Установление ВО в таком порядке происходит следующим образом: когда первая версия документа была напечатана и сохранена с использованием команды «Сохранить», то установилась ВО операции «Последний вывод на печать», потом документ был сохранен при помощи команды «Сохранить как…» под другим именем, после чего установилась ВО «Дата создания содержимого».

Кроме того, было определено, что, если у файла имеются внутренние ВО, то в большинстве случаев их нельзя удалить или модифицировать с помощью штатных средств ОС Windows. Так, например, на вкладке «Подробно» окна «Свойства» имеется кнопка «Удаление свойств и личной информации», после нажатия которой появляется новое окно «Удаление свойств». Для текстовых документов с помощью этого диалогового окна нельзя удалить внутренние ВО, а для изображений можно удалить лишь «Дату съемки» (рис. 1.9).

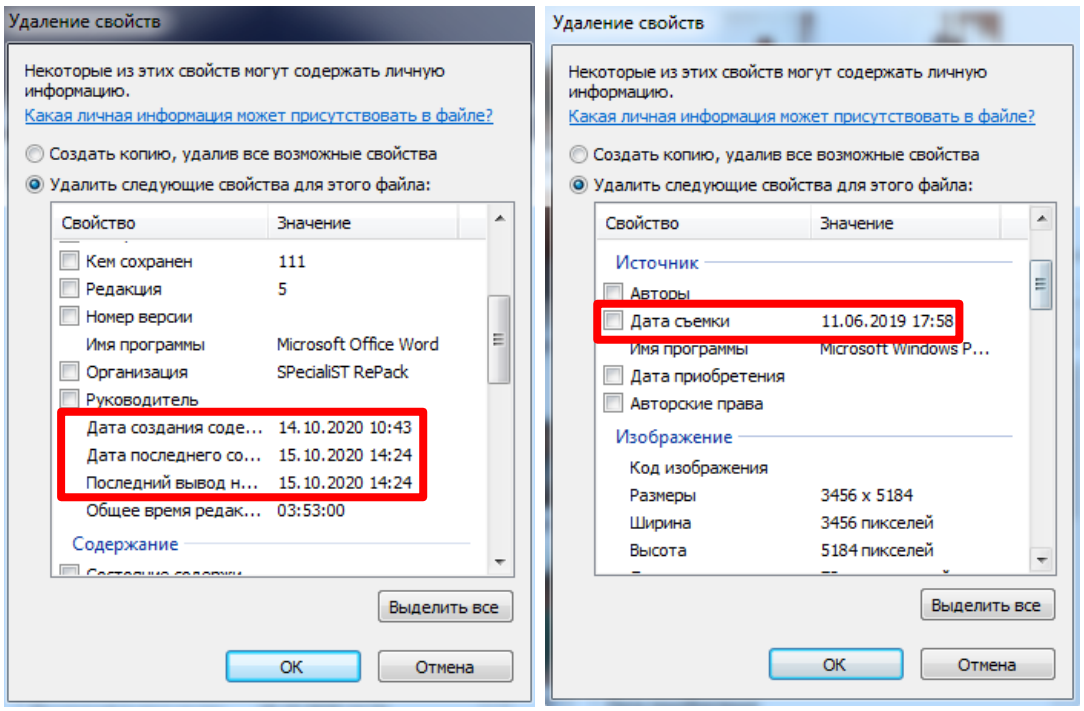


Рис. 1.9. Диалоговое окно «Удаление свойств»

Тем не менее, использовать внутренние ВО для восстановления хронологии совершения ФОп нецелесообразно, так как не каждый файл обладает ими. Если файл имеет внутренние ВО, то их может быть не более 3. Внутренние ВО могут быть полезны в качестве дополнительного источника информации, их наличие у файла позволяет:

– уточнять время создания файла. Особенно это актуально, если целевой файл многократно копировался с одного носителя на другой и, как следствие, внешние ВО не несут актуальной информации о дате и времени создания файла;

– выяснять, распечатывался ли файл. Только ограниченный круг форматов файлов предусматривает сохранение времени последней печати, к ним относятся файлы документов пакетов Microsoft Office 97–2003 и документов пакетов Microsoft Office 2007–2010, базирующихся на универсальном языке разметки XML;

– уточнять дату и время последней модификации файла.

1.2.3 Дополнительные ВО

Под дополнительными ВО в данной работе рассматриваются ВО, содержащиеся в системных и служебных файлах и каталогах. В большинстве случаев дополнительные ВО фиксируются в ФС после открытия и редактирования файлов.

Широко известными источниками дополнительных ВО являются ярлыки. Ярлык — это ссылка на объект (файл, папку), указывающая на его местоположение. Ярлыки создаются при открытии файла и сохраняются в каталоге «\.\Users\<имя_пользователя>\AppData\Roaming\Microsoft\Windows\Recent».
Структура ярлыка представлена на рис. 1.10 и описана в документе «[MS-SHLLINK] Shell Link (.LNK) Binary File Format» [47].

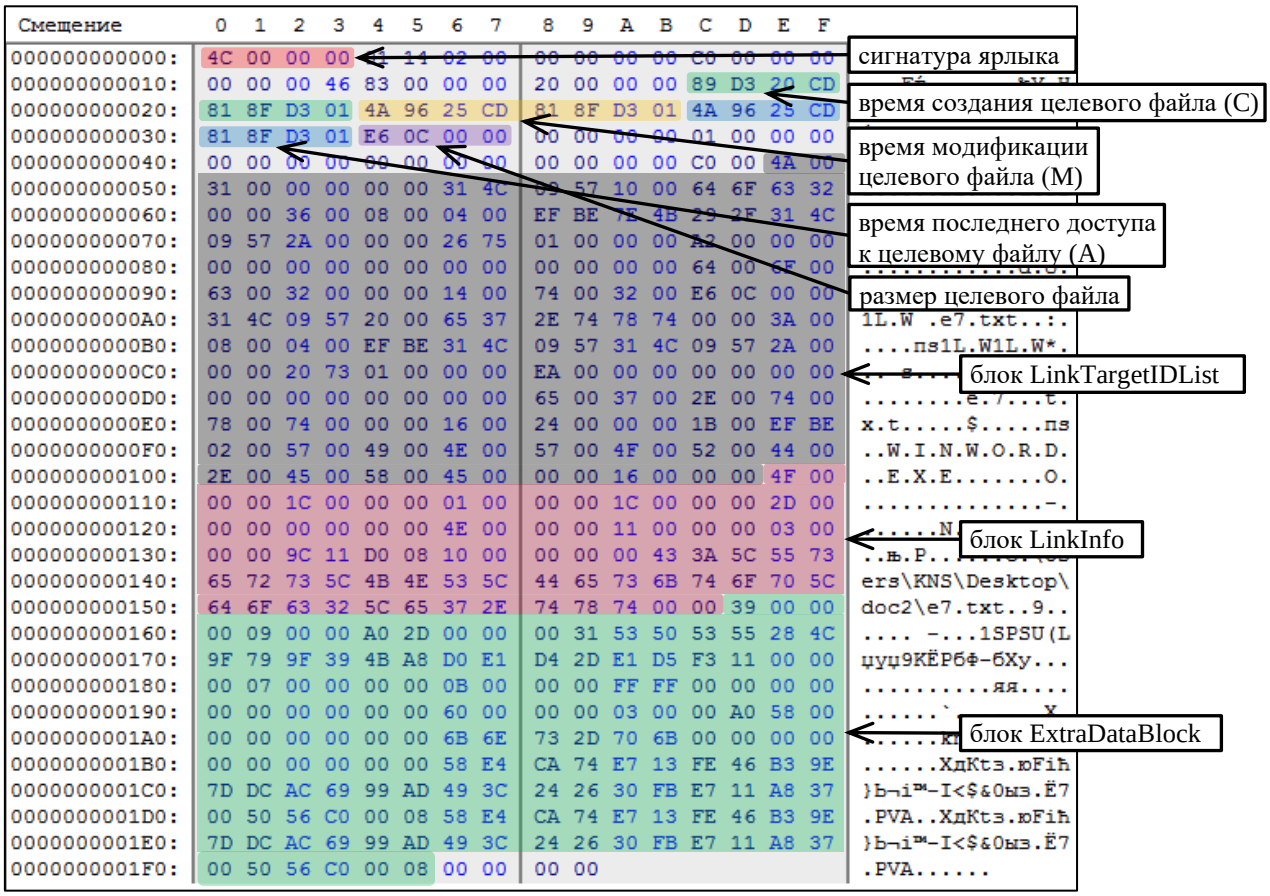


Рис. 1.10. Структура ярлыка

Г. Парсонейдж (Harry Parsonage) в статье [21] приводит аргументы в пользу использования ВО файлов-ярлыков (далее — ярлыков[1]) с целью установления факта открытия или редактирования файла после его скачивания из сети Интернет или копирования с другого носителя. Для решения поставленной задачи автор использует девять ВО, три из которых являются отметками ярлыка: время создания, время модификации и время доступа; еще три ВО содержатся внутри ярлыка и принадлежат целевому файлу (файл, на который был создан ярлык). Если целевой файл существует на носителе, то он будет иметь свои собственные три ВО.

Ярлык состоит из заголовка и блоков. В заголовке хранятся ВО создания, модификации и последнего доступа к целевому файлу (файл, на который создался ярлык). Эти отметки совпадают с ВО, содержащимися в атрибуте SI

[1] При открытии или запуске файлов и каталогов в ОС Windows автоматически генерируются ярлыки на открываемый объект, которые сохраняются в каталоге «\.\Users\<имя_пользователя>\AppData\Roaming\Microsoft\Windows\Recent». Структура ярлыка и ВО, хранящиеся внутри ярлыка, изучены в п.1.2 диссертации.

файловой записи (внешние ВО), в момент последнего открытия целевого файла.

Важной для криминалистического исследования особенностью ярлыков является то, что они остаются в ФС даже после удаления целевого файла. Ниже приведены примеры, когда можно установить факт копирования файла при наличии или отсутствии исходного файла и его копии в зависимости от того, открывались данные файлы или нет (то есть создавались или нет ярлыки, указывающие на эти файлы) (табл. 1.3).

Таблица 1.3

Установление факта копирования файла по наличию ярлыка

1	~~исходный файл~~ ярлык есть	→	~~копия файла~~ ярлык есть	— можно установить
2	~~исходный файл~~ ярлык есть	→	копия файла ярлыка нет	— можно установить
3	исходный файл ярлыка нет	→	~~копия файла~~ ярлык есть	— можно установить
4	~~исходный файл~~ ярлыка нет	→	копия файла ярлык есть	— можно установить
5	исходный файл ярлык есть	→	~~копия файла~~ ярлыка нет	— **нельзя** установить

~~~~ — означает, что файл (копия) отсутствует на исследуемом машинном носителе информации

Таким образом, если файл копировался на внешний машинный носитель информации и открывался на нем, то данный факт можно установить по наличию ярлыка (первая и третья строки табл. 1.3). Если же файл копировался на внешний машинный носитель информации и не открывался на нем, то при отсутствии машинного носителя определить факт копирования на него нельзя (пятая строка табл. 1.3).

Ярлыки можно удалить штатными средствами ОС Windows, не имея полномочий администратора. Кроме того, настройками системного реестра можно предотвратить создание ярлыков в каталоге «Recent» для файлов с конкретным расширением. Для этого в параметре EditFlags необходимо установить значение 0x00000000 в ветке реестра HKEY_CLASSES_ROOT\<ProgID>, где
~~~~

<ProgID> — класс файлов. Например, для файлов с расширением txt ветка реестра будет принимать значение txtfile (рис. 1.11).

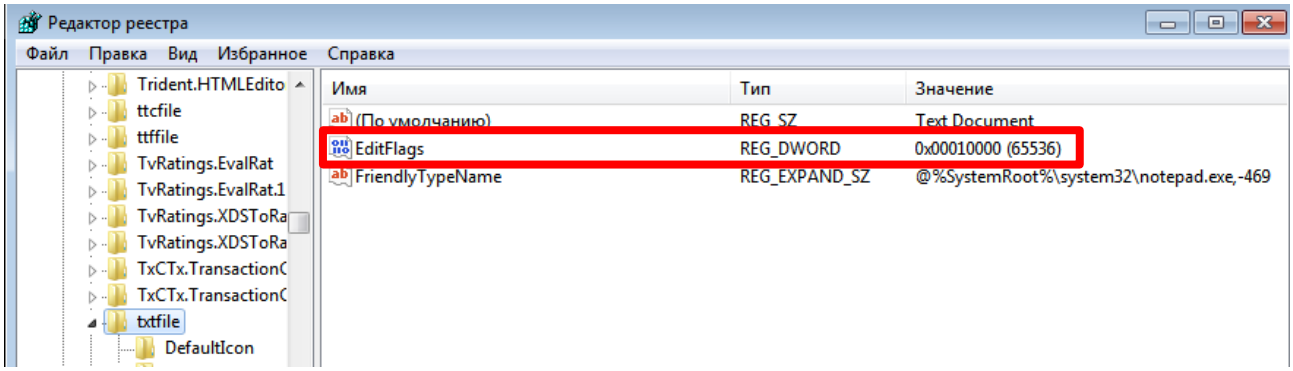


Рис. 1.11. HKEY_CLASSES_ROOT\txtfile

Тем не менее, наличие ярлыков позволяет достоверно определять ФОп первого и последнего открытия.

Следующим широко известным местом хранения криминалистически значимой информации является системный реестр — центральная иерархическая база данных, используемая в ОС Windows для хранения сведений, необходимых для настройки системы для одного или более пользователей, приложений и аппаратных устройств. Реестр содержит два базовых типа сущностей — ключ («раздел») и параметр. Раздел является аналогом каталога ФС и может содержать в себе другие подразделы (подобно подкаталогам), а также параметры.

В системном реестре существуют разделы, в названиях которых встречается аббревиатура MRU (Most Recently Used) — список недавно использованных элементов. В этих ветках в качестве параметров хранятся имена последних запускаемых или открываемых файлов.

Например, в разделе реестра HKEY_CURRENT_USER\Software\Microsoft\Office\Word для каждого запускаемого приложения Microsoft Office Word хранятся списки последних открытых документов (File MRU) и последних открытых каталогов (Place MRU). Эти списки отображаются в меню «Файл\Последние\Последние документы» и «Файл\Последние\Последние места» приложения. Кроме того, у каждого файла в этих разделах фиксируется ВО его последнего открытия в формате FILETIME.

Данные списки можно очистить как через меню приложения Microsoft Office, так и в разделах реестра, полномочий администратора для этого не требуется.

Последним рассматриваемым местом хранения ВО являются журналы событий (системные журналы) — файлы, в которых централизованно хранится информации о важных программных и аппаратных событиях приложений и ОС. Основные системные журналы хранятся в каталогах «\.\Windows\System32\winevt\Logs\» и «\.\Windows\inf\». Преимущественно журналы хранят записи, связанные с системными событиями: вход/выход пользователя, установка программ, подключение устройств. Но некоторые приложения имеют свои собственные журналы, в которых фиксируются события, связанные с их работой.

В рамках данного исследования представляют интерес события, формируемые приложениями, которые непосредственно обрабатывают файлы. На данный момент известно, что только Microsoft Office имеет свой собственный журнал «OAlerts.evtx» (или «Microsoft Office Alerts.evtx»), в котором хранятся события, сформированные в процессе вывода диалогового окна программами из его пакета (рис. 1.12).

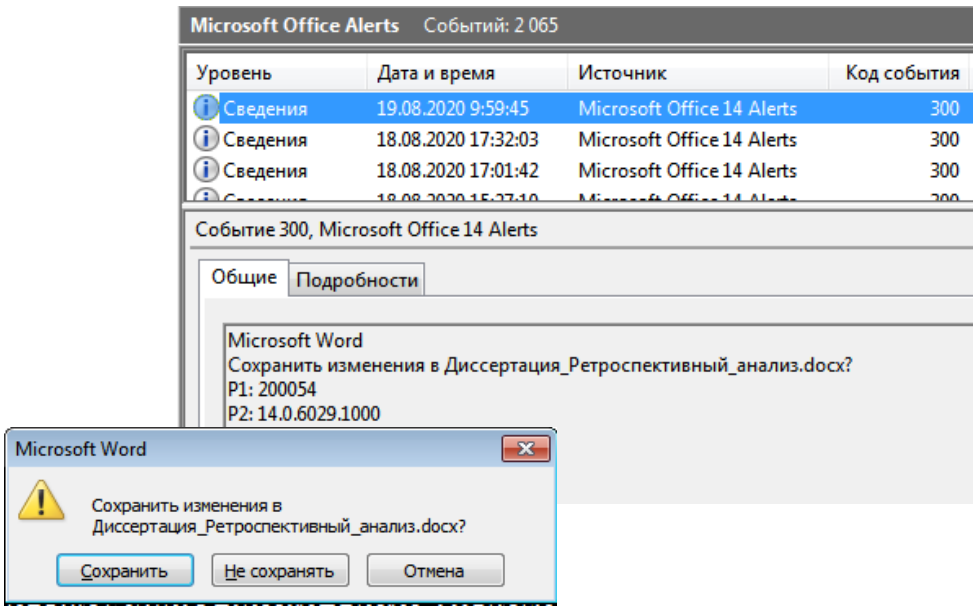


Рис. 1.12. События из журнала «Microsoft Office Alerts.evtx»

Данные события имеют код «300» и источник «Microsoft Office 14 Alerts». Фиксируются следующие ФОп:

– печать документов, если область печати выходит за границы разделов документа;

– сохранение изменений в документе, если пользователь закрывает измененный документ без предварительного сохранения;

– сохранение документа в другой формат;

– открытие документа, если производилась проверка правописания и найдено большое количество орфографических или грамматических ошибок;

– неудачные попытки открытия ссылки внутри документа.

При этом фиксируются только имена файлов без полного пути к ним. Полная идентификация файла возможна только при наличии уникального имени. Поэтому события из журнала можно использовать в качестве дополнения к внутренним ВО для уточнения, например, ВО печати или ВО модификации. Кроме того, журналы событий можно очистить штатными средствами ОС Windows.

После рассмотрения мест хранения и способов формирования дополнительных ВО, можно резюмировать следующее:

– дополнительные ВО достаточно легко удалять;

– некоторые дополнительные ВО формируются только у файлов, обрабатываемых определенными приложениями;

– дополнительные ВО позволяют достоверно устанавливать 3 ФОп: открытие, редактирование, копирование, только в том случае, если файл открывался (копирование определяется, если оба файла: копия и оригинал открывались на исследуемой ОС).

Таким образом, дополнительные ВО целесообразно рассматривать в качестве дополнительной (уточняющей) информации к результатам анализа внешних ВО.

1.2.4 Сравнение существующих способов восстановления последовательности ФОп по ВО в ОС Windows

В результате обзора работ [10–30], доступных для анализа и посвященных изучению ВО, можно предложить классификацию способов восстановления последовательности ФОп, оценить возможности и недостатки каждого из них (таблица 1.4).

Таблица 1.4

Существующие способы восстановления последовательности ФОп

Основоположник способа	Источник ВО				Исследуемая версия ОС	Опция обновления ВО А	Возможности способа
	атрибуты файловой записи		внутри формата Office Open XML	ярлыки			
	SI	FN					
Б. Кэрриэ	+	–	–	–	Windows XP	вкл.	Определение 4 ФОп: 1. копирование (если после копирования файл не редактировали), 2. перемещение в пределах тома, 3. редактирование, 4. удаление
В. Матвеева	+	+	–	–	Windows XP, 7	вкл.	Определение 7 ФОп: 1. копирование (если после копирования файл не редактировали), 2. перемещение в пределах тома и между томами, 3. редактирование, 4. удаление, 5. открытие, 6. просмотр атрибутов файла 7. изменение атрибутов файла
Э. Дидриксен	+ (только 3 ВО)	–	+	–	–	–	Определение 2 ФОп: 1. копирование, 2. печать файла
Г. Парсонейдж	+ (только 3 ВО)	–	–	+	Windows XP	вкл.	Определение 4 ФОп: 1. копирование (если файл-оригинал и файл-копия открывались), 2. редактирование, 3. удаление, 4. открытие

Представленные исследования позволили их авторам выявить ряд закономерностей процесса изменения ВО при выполнении над файлами различных операций и предложить частные методики восстановления последовательности ФОп, основанные на сравнительном анализе ВО. Данные исследования могут оказаться полезными экспертам-специалистам, однако эти изыскания носят весьма разрозненный, не системный характер и не обеспечивают полноту исследований и широту охвата разнообразия ФОп и вариантов их выполнения, поэтому не могут являться основой для создания автоматизированного инструментария для восстановления последовательности ФОп. Тем не менее, вышеописанные результаты исследований позволяют говорить о целесообразности системного подхода к анализу ВО.

1.3 Обзор программ, имеющих средства просмотра и анализа внешних ВО файлов

Специализированных программ, позволяющих восстанавливать действия пользователя с файлами, в доступных источниках найдено не было. Так, например, фирма NirSoft [38] разработала коллекцию небольших и полезных программ для эксперта-криминалиста, которые исследуют активность ФС, используя для этого информацию, извлеченную из системного реестра, журналов событий и других системных файлов. Утилиты дают возможность просматривать историю подключения накопителей, установки программного обеспечения, но не позволяют восстанавливать хронологию действий над пользовательскими файлами. Другие существующие программные продукты предназначены для проведения комплексной криминалистической экспертизы [35-37, 39]. В их инструментарий практически всегда входит средство просмотра содержимого ФС исследуемого носителя с различной степенью детализации. Например, The Sleuth Kit отображает ВО из атрибутов FN и SI, при этом новые версии данной программы выводят ВО с точностью до 100 наносекунд (рис. 1.13). А программа Autopsy отображает ВО из атрибута SI с точностью до секунд (рис. 1.14).

```
$STANDARD_INFORMATION Attribute Values:
Flags: Archive
Owner ID: 0
Security ID: 660  (S-1-5-21-2571853146-3837938598-3442622739-1000)
Last User Journal Update Sequence Number: 23639496
Created:        2018-11-06 16:07:53.829370000 (ALM)
File Modified:  2018-10-29 13:53:25.147464300 (ALM)
MFT Modified:   2018-11-06 16:15:44.412996800 (ALM)
Accessed:       2018-11-06 16:07:53.829370000 (ALM)

$FILE_NAME Attribute Values:
Flags: Archive
Name: 0013~1.JPG
Parent MFT Entry: 19279         Sequence: 2
Allocated Size: 2445312         Actual Size: 2441397
Created:        2018-11-06 16:07:53.829370000 (ALM)
File Modified:  2018-10-29 13:53:25.147464300 (ALM)
MFT Modified:   2018-11-06 16:08:35.247442700 (ALM)
Accessed:       2018-11-06 16:07:53.829370000 (ALM)
```

Рис. 1.13. Вывод ВО в ПО The Sleuth Kit

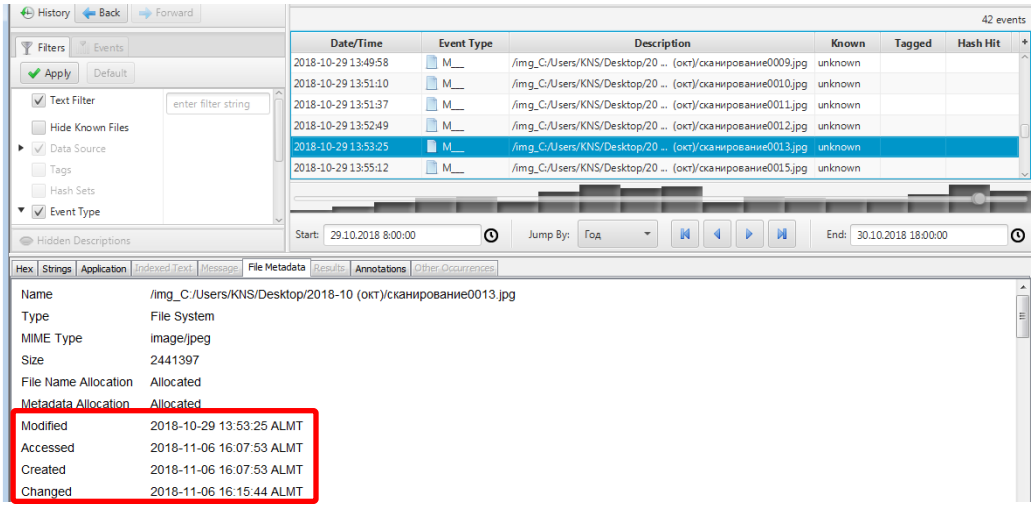


Рис. 1.14. Вывод ВО в ПО Autopsy

Некоторые программные средства позволяют визуализировать временную шкалу (timeline), то есть располагать события определенным образом согласно времени, в которое они произошли. Так, например, ПО Autopsy отображает ВО в виде диаграммы (рис. 1.15). А программа R-studio [39] сортирует файлы по времени создания, изменения и последнего доступа в виде дерева папок, названия которых соответствуют году, месяцу и числу (рис. 1.16).

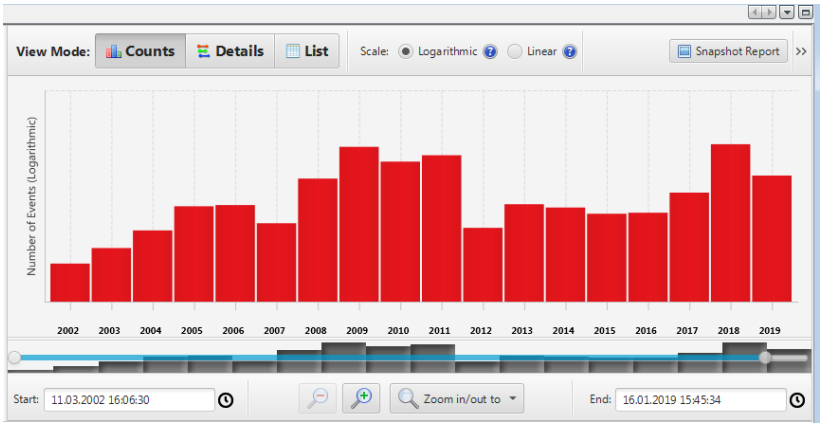


Рис. 1.15. Визуализация временной шкалы в ПО Autopsy

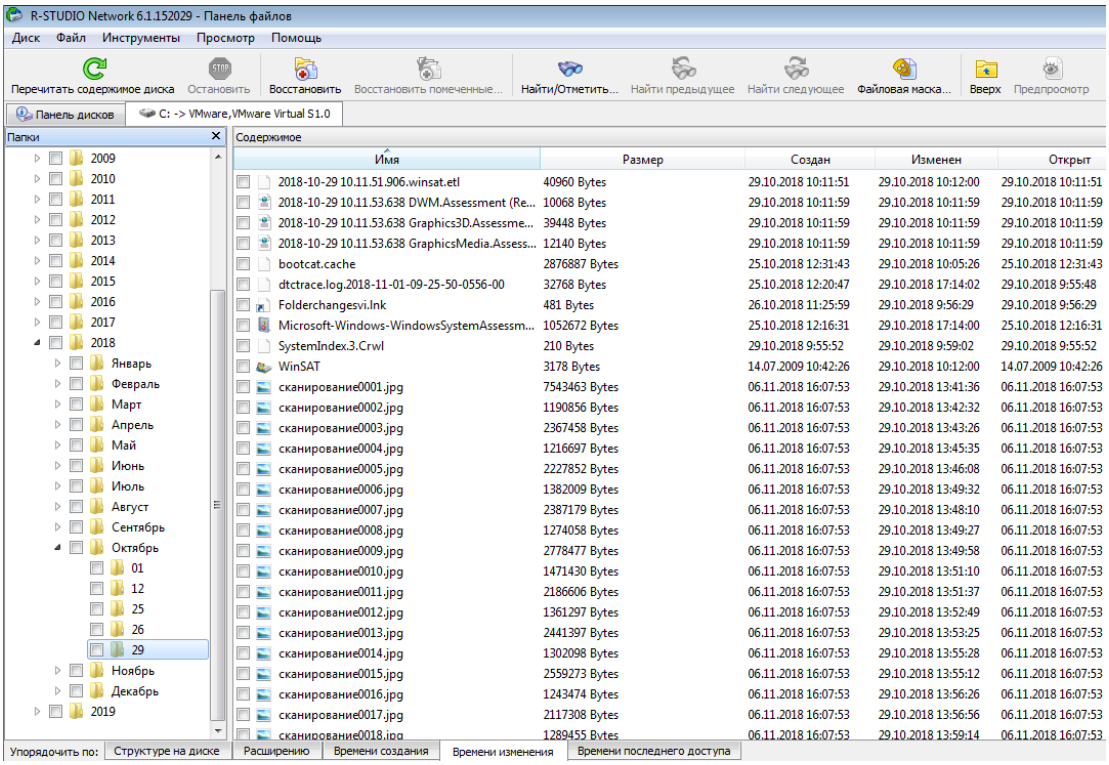


Рис. 1.16. Визуализация временной шкалы в ПО R-studio

Для построения временной шкалы существует специализированные утилиты командной строки log2timeline.exe и psort.exe, разработанные К. Гаджонсоном (Kristenn Gudjonsson) [40]. Программа log2timeline.exe находит ВО, хранящиеся в таблице MFT, журналах событий и внутренних структурах файлов. Программа psort.exe сортирует результаты, полученные после выполнения log2timeline.exe, и обладает большим набором аргументов для настройки фильтра отображаемых данных, в том числе вывод ВО за определенную дату. Наглядной визуализации эти инструменты не предоставляют, но в них реализована возможность экспорта данных в различные форматы,

такие как csv, xml, SQLite и др. Файлы с расширением csv удобнее всего просматривать в Microsoft Excel (рис. 1.17).

	A	B	C	D	F	J	K	M
1	date	time	timezone	MACB	sourcetype	short	desc	filename
789	05/25/2018	8:23:02	UTC	MA..	NTFS Content Modification Time	/123	TSK:/123 Type: directory	TSK:/123
790	05/28/2018	2:44:34	UTC	..C.	NTFS Metadata Modification Time	/123	TSK:/123 Type: directory	TSK:/123
791	05/25/2018	8:15:31	UTC	.A.B	NTFS Creation Time	/123/132.txt	TSK:/123/132.txt Type: file	TSK:/123/132.txt
792	05/25/2018	8:15:48	UTC	M...	NTFS Content Modification Time	/123/132.txt	TSK:/123/132.txt Type: file	TSK:/123/132.txt
793	05/28/2018	2:44:34	UTC	..C.	NTFS Metadata Modification Time	/123/132.txt	TSK:/123/132.txt Type: file	TSK:/123/132.txt
794	05/25/2018	8:23:00	UTC	...B	NTFS Creation Time	/123/3	TSK:/123/3 Type: directory	TSK:/123/3
795	05/25/2018	8:23:19	UTC	MA..	NTFS Content Modification Time	/123/3	TSK:/123/3 Type: directory	TSK:/123/3
796	05/28/2018	2:44:34	UTC	..C.	NTFS Metadata Modification Time	/123/3	TSK:/123/3 Type: directory	TSK:/123/3
797	05/25/2018	8:15:48	UTC	M...	NTFS Content Modification Time	/123/3/132.txt	TSK:/123/3/132.txt Type: file	TSK:/123/3/132.txt
798	05/25/2018	8:23:03	UTC	.A.B	NTFS Creation Time	/123/3/132.txt	TSK:/123/3/132.txt Type: file	TSK:/123/3/132.txt
799	05/28/2018	2:44:34	UTC	..C.	NTFS Metadata Modification Time	/123/3/132.txt	TSK:/123/3/132.txt Type: file	TSK:/123/3/132.txt
800	10/18/2018	6:39:32	UTC	...B	NTFS Creation Time	/12345	TSK:/12345 Type: directory	TSK:/12345
801	10/18/2018	6:39:32	UTC	MAC.	NTFS Content Modification Time	/12345	TSK:/12345 Type: directory	TSK:/12345
802	07/23/2018	6:57:39	UTC	M...	NTFS Content Modification Time	/12345/123.bmp	TSK:/12345/123.bmp Type: file	TSK:/12345/123.bmp
803	10/18/2018	6:39:32	UTC	.ACB	NTFS Creation Time	/12345/123.bmp	TSK:/12345/123.bmp Type: file	TSK:/12345/123.bmp
804	07/26/2018	3:29:00	UTC	...B	Open XML Metadata	Author: KNS	Creating App: Microsoft Office Word Ap	TSK:/12345/123.docx
805	07/26/2018	3:30:00	UTC	M...	Open XML Metadata	Author: KNS	Creating App: Microsoft Office Word Ap	TSK:/12345/123.docx

Рис. 1.17. Просмотр результатов выполнения утилит log2timeline.exe и psort.exe в Microsoft Excel

На рис. 1.17 представлен результат выполнения утилит log2timeline.exe и psort.exe. Первые два столбца «date» и «time» содержат дату и время соответственно. Третий столбец «timezone» указывает часовой пояс (UTC — по Гринвичу). В четвертом столбце «MACB» указан символьный идентификатор ВО, где M — время изменения данных, C — время модификации метаданных файла, A — время последнего обращения к содержимому файла, B — время, в которое была создана файловая запись. В пятом столбце «sourcetype» приводится расшифровка типа ВО и источника извлечения (NTFS — таблица MFT, Open XML Metadata — внутренняя ВО офисного документа). Восьмой столбец «filename» содержит имя файла.

Характеристики рассмотренных программ приведены в таблице 1.5.

Таблица 1.5

Основные характеристики программ, имеющих средства просмотра ВО

Название программы	Выводимые ВО	Точность ВО	Особенности работы
The Sleuth Kit	Из атрибутов SI и FN файловой записи	До 100 нсек	Работает из командной строки
Autopsy	Из атрибута SI файловой записи	До секунды	Есть необходимость установки программы
R-studio	Из атрибута SI файловой записи	До секунды	Есть необходимость установки программы
log2timeline.exe и psort.exe	Из атрибута SI файловой записи	До секунды	Работает из командной строки

Из вышеописанных программ только The Sleuth Kit отображает ВО файла из атрибутов SI и FN файловой записи с точностью до 100 нсек, но не выводит информацию о ВО из индексных записей родительских каталогов. Более того, ни одна из рассмотренных программ не проводит анализа ВО с целью восстановления действий пользователя над файлами.

1.4 Постановка задач исследования

Восстановление последовательности ФОп является одной из важных и сложных задач компьютерной криминалистики. При проведении КТЭ следует использовать множество ВО, описывающих события, произошедшие в системе.

ВО могут быть представлены в различных форматах и располагаться как внутри формата файла, так и в специальных системных файлах. В ФС NTFS основным местом хранения ВО является таблица MFT. Каждый хранимый в системе файл имеет в таблице MFT собственную файловую запись, в которой фиксируются так называемые внешние ВО.

В отечественной и зарубежной научно-технической литературе имеется достаточно большое количество публикаций, посвященных исследованию закономерностей изменений ВО файлов при совершении над ними операций. Авторам работ удалось выявить ряд важных закономерностей, в соответствии с которыми модифицируются ВО файлов. Исследователи на конкретных примерах иллюстрируют возможность восстановления отдельных действий пользователя путем трудозатратного ручного анализа, который основан на опыте и знаниях, полученных в ходе их собственных наблюдений за поведением ВО.

В то же время научно обоснованных методик и алгоритмов, позволяющих автоматизировать процесс восстановления последовательности событий, а также приемлемых программных решений представленный в первой главе диссертации анализ предметной сферы не выявил. Это объяснимо рядом недостатков, присущих предшествующим изысканиям:

− отсутствие у исследователей приемлемого программного инструментария для получения и анализа всех типов ВО;

− ограничение количества одновременно анализируемых ВО — в большинстве работ рассматриваются всего три или четыре внешние ВО для каждого файла;

− отсутствие формализованной модели процесса изменения ВО файлов при проведении над ними ФОп.

С учетом изложенного для повышения количества восстанавливаемых ФОп и длины последовательности восстанавливаемых ФОп необходимо решить следующие **задачи:**

1. Разработать программный инструментарий для наблюдения за ВО при выполнении ФОп и провести эксперименты в объеме, достаточном для исследования влияния ФОп на изменение внешних ВО.

2. Определить причинно-следственные связи между ФОп и изменениями ВО файлов, на основании которых разработать математическую модель изменения значений ВО. Провести экспериментальную оценку адекватности модели.

3. Разработать методику восстановления последовательности ФОп и реализовать ее на программном уровне.

4. Сформировать рекомендации по применению программного обеспечения.

2 ИССЛЕДОВАНИЕ МЕХАНИЗМА ИЗМЕНЕНИЯ ВНЕШНИХ ВО ФАЙЛОВ

2.1 Программный инструментарий для наблюдения за изменениями внешних ВО

Для наблюдения за формированием и обновлением ВО файлов в NTFS в ходе настоящего исследования Хорьковым Д.А. разработана консольная программа FTA (File Time Analyzer), которая для каждого файла находит и выводит двенадцать ВО из таблицы MFT. Программа содержит собственный драйвер файловой системы NTFS, который взаимодействует с разделом как с двоичным массивом (файлом) в режиме «только чтение» и предоставляет все необходимые функции по доступу к системным областям ФС, включая атрибуты файловых записей MFT.

Для корректной работы программы в ОС Windows ее требуется запустить из командной строки с полномочиями администратора (рис. 2.1).

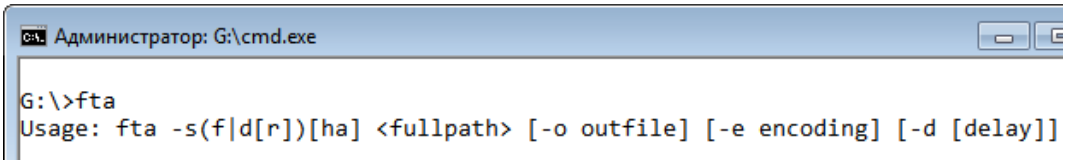


Рис. 2.1. Запуск программы FTA в командной строке

Программа имеет следующие параметры:

–sf — вывод ВО для одного файла или каталога, полное имя которого записывается после аргумента;

–sd — вывод ВО всех файлов и каталогов, находящихся в указанном каталоге (ВО самого каталога не выводится);

–sdr — вывод ВО всех файлов и каталогов, находящихся в указанном каталоге, рекурсивно;

h — вывод ВО в хронологическом порядке;

a — вывод ВО с использованием режима аппроксимации (значения ВО, разность которых меньше 1 секунды, интерпретируются как одно значение),

fullpath — полный путь до файла и каталога;

–o outfile — установка файла для вывода результатов;

–e encoding — установка кодировки для вывода информации (по умолчанию установлено значение 866);

–d delay — установка задержки в милисекундах при выводе ВО следующего файла (по умолчанию установлено значение 60000).

Программа FTA выводит информацию о ВО файлов в виде таблицы из 12 строк и 5 столбцов (рис. 2.2).

```
G:\>fta -sf X:\123\132.txt

Текущее время (UTC): 17.10.2018 10:25:55

Временные отметки файла <\.\123\132.txt>
SIC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
SIM: Время посл. модификации    25.05.2018  8:15:48  M = 131717097483581209
SIA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
SIX: Время модификации записи   28.05.2018  2:44:34  X = 131719490746574357
FNC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
FNM: Время посл. модификации    25.05.2018  8:15:31  M = 131717097314632912
FNA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
FNX: Время модификации записи   25.05.2018  8:15:31  X = 131717097314632912
DRC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
DRM: Время посл. модификации    25.05.2018  8:15:48  M = 131717097483581209
DRA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
DRX: Время модификации записи   28.05.2018  2:44:34  X = 131719490746574357
```

1	2	3	4	5

Рис. 2.2. Пример вывода программы FTA

Здесь и далее для удобства введена следующая аббревиатура. Символами «C», «M», «A», «X» отображены соответственно отметки создания, модификации и последнего доступа к файлу, а также отметка последней модификации метаданных. Символами «SI» обозначены ВО, извлеченные из атрибута $STANDARD_INFORMATION, символами «FN» — ВО, извлеченные из атрибута $FILE_NAME, символами «DR» — ВО, извлеченные из $INDEX_ROOT и $INDEX_ALLOCATION. Таким образом, например, аббревиатура «SIC» обозначает ВО создания файла в атрибуте $STANDARD_INFORMATION, а «FNM» — ВО последней модификации файла из атрибута $FILE_NAME.

В заголовочной строке вывода программы с точностью до секунды указывается дата и время извлечения ВО. Первый столбец содержит символьный идентификатор ВО, три символа которого указывают на место хранения и тип ВО. Во втором столбце приводится расшифровка типа ВО. Третий и четвертый столбцы содержат соответственно дату и время в универсальном формате UTC+0 (по Гринвичу). В пятом столбце ВО приведены в количестве 100-наносекундных интервалов с 00:00:00 01.01.1601 — целое 18-разрядное число.

Таким образом, разработанная программа выполняет требования к инструменту для наблюдения за изменениями ВО файлов: работает в режиме «только чтение», не модифицируя ВО файла, выводит информацию о ВО из атрибутов SI и FN файловой записи и из индексных записей родительских каталогов с точностью 10^{-7} с.

2.2 Алгоритм экспериментального исследования механизма изменения внешних ВО файлов

Исследования изменений ВО при выполнении ФОп проводились на нескольких компьютерах с различной аппаратной конфигурацией под управлением ОС Windows XP, 7, 8 и 10 в двух возможных режимах: с выключенной (NtfsDisableLastAccessUpdate = 1) и включенной (NtfsDisableLastAccessUpdate = 0) ВО последнего доступа (SIA).

Цель данного этапа исследований заключалась в фиксации фактов изменения внешних ВО после применения различных ФОп.

Эксперимент проводился в следующем порядке:

1. Подготовка файлов для исследования;
2. Выполнение ФОп;
3. Фиксирование изменений ВО.

Фиксирование ВО производилось с помощью утилиты FTA (п. 2.1 диссертации).

1. Подготовка файлов для исследования

На дисковом разделе с ФС NTFS создавался специальный каталог с именем «poligon». В этом каталоге создавались несколько подкаталогов, которым присваивались числовые обозначения (имена). В каталогах создавались по несколько файлов различных форматов, которые заполнялись произвольным содержимым. Наблюдения производились над файлами длиной 0 байт, более 0 байт, но менее 700 байт[1] (содержимое файла хранится в файловой записи таблицы MFT), более 700 байт (содержимое файла хранится во внешних кластерах файловой системы). В рамках работы исследовались файлы следующих форматов:

- документы (txt, odt, pdf, rtf, djvu, djv, dot, doc, docx, xls, xlsx, ppt, pptx);
- исполняемые файлы (exe, dll, com, sys);
- изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff);
- видеофайлы (mov, avi, mpeg, mpg, mkv, mp4);
- звуковые файлы (mp3, wav);
- архивы (zip, rar, 7z);
- интернет-файлы (url, eml).

Кроме того, у файлов устанавливались различные атрибуты («архивный» («a»), «только чтение» («r»), «системный» («s») или «скрытый» («h»)).

С помощью шестнадцатеричного редактора в файловых записях MFT, созданных для исследования файлов, производились предварительные упорядоченные изменения ВО во всех атрибутах. На рис. 2.3 и рис. 2.4 изображены таблицы ВО файла до и после упорядочивания. У файла, который был только создан, ВО совпадают между собой (рис. 2.3). В случае, если какие-либо ВО наследуют значения других ВО при совершении ФОп, то такие изменения не будут обнаружены по ВО, представленным на рис. 2.3, и будут обнаружены, если значения ВО до совершения ФОп были разными как на рис. 2.4.

[1] Размер определен экспериментально

```
Временные отметки файла <\.\poligon\2\aaadd.docx>
SIC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
SIM: Время посл. модификации    25.05.2018  8:15:31  M = 131717097314632912
SIA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
SIX: Время модификации записи   25.05.2018  8:15:31  X = 131717097314632912
FNC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
FNM: Время посл. модификации    25.05.2018  8:15:31  M = 131717097314632912
FNA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
FNX: Время модификации записи   25.05.2018  8:15:31  X = 131717097314632912
DRC: Время создания файла       25.05.2018  8:15:31  C = 131717097314632912
DRM: Время посл. модификации    25.05.2018  8:15:31  M = 131717097314632912
DRA: Время последнего доступа   25.05.2018  8:15:31  A = 131717097314632912
DRX: Время модификации записи   25.05.2018  8:15:31  X = 131717097314632912
```

Рис. 2.3. ВО файла aaadd.docx до упорядочивания

```
Временные отметки файла <\.\poligon\2\aaadd.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    1.01.1601  0:00:00  X = 4
FNC: Время создания файла        1.01.1601  0:00:00  C = 5
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 6
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 7
FNX: Время модификации записи    1.01.1601  0:00:00  X = 8
DRC: Время создания файла        1.01.1601  0:00:00  C = 9
DRM: Время посл. модификации     1.01.1601  0:00:00  M = 10
DRA: Время последнего доступа    1.01.1601  0:00:00  A = 11
DRX: Время модификации записи    1.01.1601  0:00:00  X = 12
```

Рис. 2.4. ВО файла aaadd.docx после упорядочивания

Изменения ВО производились после создания файла и перед выполнением над ним операций, что не влияло на результаты экспериментов. ВО, хранимая в атрибутах файловой записи, представляет количество сотен наносекунд, прошедших с 1 января 1601 г., что и отображается в пятом столбце вывода программы, третий и четвертый столбцы — дата и время в универсальном формате. Перед исследованием ВО были присвоены значения: $1 \cdot 10^{-7}$ с, $2 \cdot 10^{-7}$ с, $3 \cdot 10^{-7}$ с, $4 \cdot 10^{-7}$ с и т. д. (пятый столбец), третий и четвертый столбцы при этом отображают одно и то же значение, так как дата и время в универсальном формате указывается с точностью до секунд. Наблюдения производились только за значениями из пятого столбца.

2. Выполнение ФОп

Над подготовленными файлами выполнялись одиночные ФОп, которые реализовывались либо вручную, либо программным путем и различными способами (табл. 2.1).

Таблица 2.1
Способы выполнения ФОп

ФОп	Выполняется с помощью
копирование	- меню файловых менеджеров, - оболочки Explorer, - команд copy, xcopy в командной строке
перемещение (переименование)	- меню файловых менеджеров, - оболочки Explorer, - команд ren, rename, move в командной строке
удаление	- меню файловых менеджеров, - оболочки Explorer, - команд del, erase в командной строке, - приложений
просмотр атрибутов	- меню файловых менеджеров, - оболочки Explorer, - команды attrib в командной строке
изменение атрибутов	- меню файловых менеджеров, - оболочки Explorer, - команды attrib ±R в командной строке
открытие	- меню файловых менеджеров, - оболочки Explorer, - приложений
редактирование	- приложений, - команды echo в командной строке
исполнение (запуск)	- файловых менеджеров, - оболочки Explorer, - командной строки
помещение в архив	- архиваторов
разархивирование	

В качестве файловых менеджеров использовались три наиболее популярных: Far Manager [48], Total Commander [49] и FreeCommander [50].

В качестве приложений для удаления файлов использовались файловые шредеры[1]: File Shredder [51], Privazer [52], Recuva [53], SDelete [54].

В качестве приложений для открытия и редактирования текстовых документов использовались программы: Блокнот, WordPad [55], Microsoft Office [56], Adobe Reader [57], Foxit Reader [58], изображений: Paint, GNU Image

[1] файловые шредеры — специальные программы, целевым предназначением которых является гарантированное удаление файлов без возможности восстановления их содержимого

Manipulation Program (GIMP) [59], Movavi фоторедактор [60], PhotoScape [61], PixBuilder Studio [62], видео- и звуковых файлов: Media Player Classic [63], Windows Media Player, интернет файлов: Internet Explorer, Opera [64], Mozilla [65], Chrome [66].

В качестве архиваторов использовались три наиболее популярных: WinRAR [67], 7-zip [68] и встроенный архиватор Windows.

Копирование и перемещение файлов производилось как в пределах одного логического диска, так и между разделами с различными ФС, а именно: из FAT32 в NTFS, из exFAT в NTFS, из CDFS в NTFS, из UDF в NTFS.

Ниже приведен пример проведения экспериментов (рис. 2.5–2.7).

```
Временные отметки файла <\.\poligon\1\aaaee.exe>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    1.01.1601  0:00:00  X = 4
FNC: Время создания файла        1.01.1601  0:00:00  C = 5
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 6
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 7
FNX: Время модификации записи    1.01.1601  0:00:00  X = 8
DRC: Время создания файла        1.01.1601  0:00:00  C = 9
DRM: Время посл. модификации     1.01.1601  0:00:00  M = 10
DRA: Время последнего доступа    1.01.1601  0:00:00  A = 11
DRX: Время модификации записи    1.01.1601  0:00:00  X = 12
```

Рис. 2.5. ВО подготовленного файла до перемещения

```
Временные отметки файла <\.\poligon\1\aaaee.exe>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    1.01.1601  0:00:00  X = 4
FNC: Время создания файла        1.01.1601  0:00:00  C = 5
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 6
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 7
FNX: Время модификации записи    1.01.1601  0:00:00  X = 8
DRC: Время создания файла        1.01.1601  0:00:00  C = 1
DRM: Время посл. модификации     1.01.1601  0:00:00  M = 2
DRA: Время последнего доступа    1.01.1601  0:00:00  A = 3
DRX: Время модификации записи    1.01.1601  0:00:00  X = 4
```

Рис. 2.6. ВО файла после выделения его пиктограммы левой кнопкой мыши

```
Временные отметки файла <\.\poligon\2\aaaee.exe>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    6.03.2019 11:30:41  X = 131963454418788009
FNC: Время создания файла        1.01.1601  0:00:00  C = 1
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 2
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 3
FNX: Время модификации записи    1.01.1601  0:00:00  X = 4
DRC: Время создания файла        1.01.1601  0:00:00  C = 1
DRM: Время посл. модификации     1.01.1601  0:00:00  M = 2
DRA: Время последнего доступа    1.01.1601  0:00:00  A = 3
DRX: Время модификации записи    6.03.2019 11:30:41  X = 131963454418788009
```

Рис. 2.7. ВО после перемещения файла

Из рисунков видно, что после выделения левой кнопкой мыши пиктограммы файла ВО из атрибута DR присваиваются значения одноименных ВО из атрибута SI (рис. 2.6). После перемещения файла ВО из атрибута FN присваиваются значения ВО из атрибута SI, ВО SIX и DRX изменяются на время выполнения операции (рис. 2.7).

3. Фиксирование изменений ВО

После выполнения каждой ФОп фиксировались только факты изменения ВО с помощью программы FTA в табличную форму, внешний вид которой представлен в табл. 2.2.

Выше отмечалось, что ФОп представляют собой многоэтапные элементарные действия, и основные из них приводят к изменению ВО, поэтому в таблице предусмотрено разделение каждой ФОп на такие этапы. Это позволяет наглядно отображать моменты обновления конкретных ВО.

Таблица 2.2
Изменение ВО при копировании файла

Элементарные действия	Момент обновления ВО	Временные отметки											
		SI				FN				DR			
		C	M	A	X	C	M	A	X	C	M	A	X
Копирование файла в пределах одного логического диска													
открытие каталогов	T1												
выделение файла	T2									SIC	SIM	SIA	SIX
копирование файла	T3												

В ячейках таблицы указывается либо момент времени (T1, T2, T3, …), либо предыдущее состояние ВО из других атрибутов (SIX, SIM, …). Серые ячейки указывают на синхронное изменение ВО после выполнения ФОп. Пустые ячейки указывают на то, что ВО не изменились. Моменты времени T1, T2, T3 и т. д. отражают не абсолютное значение времени обновления ВО, а временной порядок. При этом T1 — момент времени, когда совершалась первое элементарное действие, T2 — момент времени совершения второго действия и т.д.

Ниже представлены два примера.

Пример 1. При создании нового файла с помощью контекстного меню оболочки Explorer ФОп состоит из двух этапов (табл. 2.3). После нажатия правой кнопки мыши выводится контекстное меню, из которого выбирается команда «Создать» → «Текстовый документ». После завершения этапа создания файла все ВО устанавливаются равными T1 (текущее время — время создания файла). Затем пользователь вводит имя вновь создаваемого файла. В момент завершения ввода имени (нажатие клавиши «Enter») устанавливается T2 (время присвоения имени файлу), в этот момент синхронно изменяются отметки SIX и DRX. Таким образом, у файла формируются следующие ВО: SIC = SIM = SIA = FNC = FNM = FNA = FNX = DRC = DRM = DRA = T1; ВО SIX = DRX = T2.

Таблица 2.3
Изменение ВО при создании файла

Элементарные действия	Момент обновления ВО	Временные отметки											
		SI				FN				DR			
		C	M	A	X	C	M	A	X	C	M	A	X
Создание файла с помощью контекстного меню оболочки Explorer													
создание файла	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1	T1
присвоение имени	T2				T2								T2

Пример 2. Переименование файла происходит в три этапа (табл. 2.4). На первом этапе выделяется пиктограмма файла, что приводит к уравниванию ВО DRC = SIC, DRM = SIM, DRA = SIA, DRX = SIX и к синхронному измене-

нию ВО DRX = SIX = T1. На втором этапе в момент запроса имени ВО из атрибута $FILE_NAME присваивают значения ВО из атрибута каталога. После ввода имени ВО DRA становится равна ВО SIA, а ВО SIX = DRX синхронно изменяются и равны времени переименования файла.

Таблица 2.4
Изменение ВО при переименовании файла

Элементарные действия	Момент обновления ВО	Временные отметки											
		SI				FN				DR			
		C	M	A	X	C	M	A	X	C	M	A	X
Переименование файла с помощью контекстного меню оболочки Explorer													
выделение файла	T1									SIC	SIM	SIA	SIX
запрос имени	T2				T2	SIC	SIM	SIA	SIX				T2
ввод имени	T3				T3								T3

2.3 Выявление закономерностей изменения внешних ВО

Анализ результатов серии экспериментов, проведенных по вышеописанному алгоритму, позволил сформулировать ряд важных выводов.

1. Если одноименные ВО из атрибутов $STANDARD_INFORMATION и $INDEX различны, то при наведении курсора на находящуюся на рабочем столе или в окне каталога пиктограмму файла происходит их уравнивание, т. е. отметкам DR* присваивается значение отметок SI*. Аналогичный результат наблюдается при выделении пиктограммы файла (однократный щелчок мыши) и при совершении любой ФОп с файлом. Многочисленные эксперименты показали, что одноименные ВО из атрибутов SI* и DR* синхронно изменяются при совершении любого действия над файлом.

По этой причине ВО DR* из области данных атрибутов $INDEX_ROOT и $INDEX_ALLOCATION в дальнейшем рассматривать не имеет смысла.

2. Если продолжительность регистрации ФОп меньше длины элементарного временного интервала (100 наносекунд), то значения соответствующих ВО изменяются синхронно (совпадают). Если на ФОп накладывается прерывание системного таймера, то между отметками точного времени появляется интервал

$\Delta T = n \cdot \tau$, кратный значению системного тика τ (разрешающей способности системного таймера) [69].

Для получения значения системного тика можно воспользоваться программой Clockres от SysInternals [70], которая работает из командной строки и выводит три значения: максимальное (Maximum timer interval), минимальное (Minimum timer interval) и текущее (Current timer interval) (рис. 2.8).

```
ClockRes v2.0 - View the system clock resolution
Copyright (C) 2009 Mark Russinovich
SysInternals - www.sysinternals.com

Maximum timer interval: 15.600 ms
Minimum timer interval: 0.500 ms
Current timer interval: 15.600 ms
```

Рис. 2.8. Информация, выводимая утилитой Clockres в ОС Windows 7

Величина τ обусловлена частотой прерываний используемого системного таймера и зависит от аппаратно-программной платформы. На рис. 2.9 представлена проверка стабильности данной величины.

```
 Set timeBeginPeriod(16);
Sleep(1) time = 15596 microseconds
Sleep(1) time = 15598 microseconds
Sleep(1) time = 15598 microseconds
Sleep(1) time = 15580 microseconds
Sleep(1) time = 15603 microseconds
Sleep(1) time = 15603 microseconds
Sleep(1) time = 15595 microseconds
Sleep(1) time = 15588 microseconds
Sleep(1) time = 15637 microseconds
Sleep(1) time = 15581 microseconds
Sleep(1) time = 15610 microseconds
Sleep(1) time = 15581 microseconds
Sleep(1) time = 15592 microseconds
Sleep(1) time = 15601 microseconds
Sleep(1) time = 15604 microseconds
Sleep(1) time = 15607 microseconds
Sleep(1) time = 15596 microseconds
Sleep(1) time = 15617 microseconds
Sleep(1) time = 15600 microseconds
Sleep(1) time = 15600 microseconds
Sleep(1) time = 15604 microseconds
Sleep(1) time = 15598 microseconds
Sleep(1) time = 15605 microseconds
Sleep(1) time = 15601 microseconds
```

Рис. 2.9. Стабильность системного таймера в ОС Windows 7

Из рис. 2.9 видно, что в процессе работы системного таймера могут присутствовать промежутки времени, когда приращения довольно стабильны и равны значению, выводимому утилитой Clockres. Однако имеют место такие

промежутки времени, на которых присутствует разброс приращения в пределах ± 40 мкс.

На рис. 2.10 представлены ВО файла, который редактировался в программе Microsoft Office. Значения ВО SIM, SIA, SIX, FNM, FNA, FNX совпадают с точностью до секунд, но при этом значения ВО, приведенные с точностью до 100-наносекунд, различаются.

```
Временные отметки файла <\.\poligon\1\aaaww.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     6.03.2019 11:05:54  M = 131963439542155380
SIA: Время последнего доступа    6.03.2019 11:05:54  A = 131963439541375378
SIX: Время модификации записи    6.03.2019 11:05:54  X = 131963439543559382
FNC: Время создания файла        1.01.1601  0:00:00  C = 1
FNM: Время посл. модификации     6.03.2019 11:05:54  M = 131963439542155380
FNA: Время последнего доступа    6.03.2019 11:05:54  A = 131963439541375378
FNX: Время модификации записи    6.03.2019 11:05:54  X = 131963439543559382
```

Рис. 2.10. ВО редактированного файла

Разница между отметками M и A (рис. 2.10):

ΔT_1 = T(M) – T(A) = 13196343954,2155380 с – 13196343954,1375378 с = 0,0780002 с.

Разница между ВО X и M (рис. 2.10):

ΔT_2 = T(X) – T(M) = 13196343954,3559382 с – 13196343954,2155380 с = 0,1404002 с.

С учетом того, что в исследуемой ОС величина системного тика τ = 15,600 мс = 0,0156 с, то количество системных тиков между отметками M и A равно:

$$n_1 = \Delta T_1/\tau = 0{,}0780002 \text{ с} / 0{,}0156 \text{ с} = 5.$$

Количество системных тиков между ВО X и M:

$$n_2 = \Delta T_2/\tau = 0{,}1404002 \text{ с} / 0{,}0156 \text{ с} = 9.$$

Зафиксированная длительными наблюдениями кратность интервала ΔT находилась в пределах n = 0 ... 9.

Таким образом, при выполнении ФОп могут появляться расхождения в значениях ВО, не превышающие 150 мс, при этом они не оказывают существенного влияния при восстановлении хронологии ФОп, и в дальнейшем такие расхождения учитываться не будут.

3. При копировании файла у исходного файла BO не изменяются, кроме случая, когда копирование происходит в ОС Windows 7 с включенной BO SIA — в этом случае изменяется BO SIA.

4. При копировании файла у файла-копии синхронно изменяются BO SIC = SIA = FNC = FNM = FNA = FNX, а BO SIM наследуется от копируемого файла. Изменение BO SIX зависит от применяемого приложения и размера копируемого файла:

– если копирование происходит в ОС Windows XP, 8, 10 или в файловом менеджере Total Commander, то BO SIX наследуется от копируемого файла;

– если копирование происходит в ОС Windows 7, то BO SIX = SIC = SIA = FNC = FNM = FNA = FNX, при этом возможно отставание BO SIX от BO SIC = SIA = FNC = FNM = FNA = FNX на несколько секунд (рис. 2.11).

```
Временные отметки файла <\.\poligon\3\aaacc.docx>
SIC: Время создания файла        6.03.2019  8:17:28  C = 131963338489305734
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    6.03.2019  8:17:28  A = 131963338489305734
SIX: Время модификации записи    6.03.2019  8:17:31  X = 131963338519569787
FNC: Время создания файла        6.03.2019  8:17:28  C = 131963338489305734
FNM: Время посл. модификации     6.03.2019  8:17:28  M = 131963338489305734
FNA: Время последнего доступа    6.03.2019  8:17:28  A = 131963338489305734
FNX: Время модификации записи    6.03.2019  8:17:28  X = 131963338489305734
```

Рис. 2.11. BO файла-копии

Копирование приводит к наибольшей неопределенности в отношении предыдущих состояний файла, так как остаются неизменными только BO: SIM и SIX. Характерным признаком файла-копии является одновременное и одинаковое изменение BO FN*.

5. При копировании файла из ФС FAT, FAT32, CDFS, UDF в ФС NTFS у файла-копии синхронно изменяются BO SIC = SIA = FNC = FNM = FNA = FNX, а BO SIM наследуется от копируемого файла. Так как в NTFS точность фиксирования BO равна 10^{-7} с, а в других ФС — 1 с, то BO SIM файла-копии округляется до секунд (нули в семи младших разрядах точной BO) (рис. 2.12).

```
Временные отметки файла <\.\poligon\2\aaapp.docx>
SIC: Время создания файла       11.03.2019  8:06:20  C = 131967651800030499
SIM: Время посл. модификации    11.03.2019  6:52:30  M = 131967607500000000
SIA: Время последнего доступа   11.03.2019  8:06:20  A = 131967651800030499
SIX: Время модификации записи   11.03.2019  8:06:20  X = 131967651800030499
FNC: Время создания файла       11.03.2019  8:06:20  C = 131967651800030499
FNM: Время посл. модификации    11.03.2019  8:06:20  M = 131967651800030499
FNA: Время последнего доступа   11.03.2019  8:06:20  A = 131967651800030499
FNX: Время модификации записи   11.03.2019  8:06:20  X = 131967651800030499
```

Рис. 2.12. ВО файла-копии (скопированного из раздела FAT)

6. Операция переименования файла по характеру преобразования ВО полностью тождественна операции перемещения файла. Это объясняется тем, что обе операции используют одни и те же системные вызовы. Критерием для различения операций перемещения или переименования служит неравенство ВО FN. В момент ввода имени или при выполнении команды «Вырезать» файла ВО из атрибута FN наследуют значения ВО из атрибута SI (рис. 2.13). Таким образом, по имеющимся ВО файла можно определить факт состоявшегося переименования или перемещения, но отличить одно от другого нельзя.

```
Временные отметки файла <\.\poligon\2\aaadd.docx>
SIC: Время создания файла       1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации    1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа   1.01.1601  0:00:00  A = 3
SIX: Время модификации записи   6.03.2019  8:27:03  X = 131963344231211825
FNC: Время создания файла       1.01.1601  0:00:00  C = 1
FNM: Время посл. модификации    1.01.1601  0:00:00  M = 2
FNA: Время последнего доступа   1.01.1601  0:00:00  A = 3
FNX: Время модификации записи   1.01.1601  0:00:00  X = 4
```

Рис. 2.13. ВО перемещенного файла

7. При перемещении (см. п. 6) файла с установленными атрибутами «r», «s» или «h» в файловом менеджере Total Commander ВО FNC, FNM, FNA наследуют значения ВО SIC, SIM, SIA, соответственно, а ВО SIX = FNX (синхронно изменяются).

```
Временные отметки файла <\.\poligon\3\aaall.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи   11.03.2019 10:30:44  X = 131967738445003563
FNC: Время создания файла        1.01.1601  0:00:00  C = 1
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 2
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 3
FNX: Время модификации записи   11.03.2019 10:30:44  X = 131967738445003563
```

Рис. 2.14. ВО перемещенного файла в Total Commander
с установленными атрибутами «r», «s» или «h»

8. При перемещении файла из раздела FAT у файла синхронно изменяются ВО SIA = SIX = FNC = FNM = FNA = FNX, а ВО SIC, SIM наследуются от исходного файла. ВО SIM округляется до секунд (см. п. 5), ВО SIC округляется до миллисекунд (нули в пяти младших разрядах точной ВО), так как точность фиксирования ВО создания в FAT равна 10^{-2} с.

```
Временные отметки файла <\.\poligon\3\aaapp.docx>
SIC: Время создания файла       11.03.2019  6:52:04  C = 131967607242900000
SIM: Время посл. модификации    11.03.2019  6:52:30  M = 131967607500000000
SIA: Время последнего доступа   11.03.2019  8:07:43  A = 131967652634981988
SIX: Время модификации записи   11.03.2019  8:07:43  X = 131967652634981988
FNC: Время создания файла       11.03.2019  8:07:43  C = 131967652634981988
FNM: Время посл. модификации    11.03.2019  8:07:43  M = 131967652634981988
FNA: Время последнего доступа   11.03.2019  8:07:43  A = 131967652634981988
FNX: Время модификации записи   11.03.2019  8:07:43  X = 131967652634981988
```

Рис. 2.15. ВО перемещенного файла из раздела FAT

9. При операции замены файла одноименным (путем копирования из другого каталога) вновь полученный файл (рис. 2.16) наследует ВО SIC, SIA, FNC, FNM, FNA, FNX от заменяемого файла (рис. 2.17), и SIM от заменяющего файла (рис. 2.18). ВО SIX фиксирует время замены файла.

```
Временные отметки файла <\.\poligon\2\aaadd.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 18
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    6.03.2019  8:36:46  X = 131963350060284283
FNC: Время создания файла        1.01.1601  0:00:00  C = 5
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 6
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 7
FNX: Время модификации записи    1.01.1601  0:00:00  X = 8
```

Рис. 2.16. ВО файла после операции заменить (путем копирования)

```
Временные отметки файла <\.\poligon\2\aaadd.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 2
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 3
SIX: Время модификации записи    1.01.1601  0:00:00  X = 4
FNC: Время создания файла        1.01.1601  0:00:00  C = 5
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 6
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 7
FNX: Время модификации записи    1.01.1601  0:00:00  X = 8
```

Рис. 2.17. ВО заменяемого файла

```
Временные отметки файла <\.\poligon\1\aaadd.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 17
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 18
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 19
SIX: Время модификации записи    1.01.1601  0:00:00  X = 20
FNC: Время создания файла        1.01.1601  0:00:00  C = 21
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 22
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 23
FNX: Время модификации записи    1.01.1601  0:00:00  X = 24
```

Рис. 2.18. ВО заменяющего файла

10. При операции замены файла одноименным (путем перемещения из другого каталога) вновь полученный файл (рис. 2.19) наследует ВО SIC, SIM, SIA, FNC, FNM, FNA, FNX от заменяющего файла. ВО SIX — время замены файла.

```
Временные отметки файла <\.\poligon\1\aaatt.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 17
SIM: Время посл. модификации     1.01.1601  0:00:00  M = 18
SIA: Время последнего доступа    1.01.1601  0:00:00  A = 19
SIX: Время модификации записи   11.03.2019  8:50:11  X = 131967678118851525
FNC: Время создания файла        1.01.1601  0:00:00  C = 17
FNM: Время посл. модификации     1.01.1601  0:00:00  M = 18
FNA: Время последнего доступа    1.01.1601  0:00:00  A = 19
FNX: Время модификации записи    1.01.1601  0:00:00  X = 20
```

Рис. 2.19. ВО файла после операции заменить (путем перемещения)

11. При просмотре атрибутов файла во всех версиях ОС Windows:

– с включенной ВО SIA изменяется ВО SIA;

– с выключенной ВО SIA никаких изменений ВО не происходит.

12. При изменении атрибутов файла во всех версиях ОС Windows изменяется ВО SIX.

13. При открытии или исполнении (запуске) файла в ОС Windows XP в оболочке Explorer с выключенной ВО SIA изменятся ВО SIX.

14. При первом открытии или исполнении (запуске) файла в ОС Windows XP в оболочке Explorer с включенной ВО SIA синхронно изменяются ВО SIA = SIX.

При каждом последующем запуске исполняемого файла (начиная с третьего) в момент запуска изменяются ВО SIX, а ВО SIA наследуют время предыдущего запуска с точностью до 1 секунды. Таким образом, ВО исполняемого файла содержат время последнего и предпоследнего запусков.

15. В ОС Windows XP в оболочке Explorer повторное обновление ВО SIA и SIX происходит только через час после последнего обновления этих ВО.

16. При открытии или исполнении (запуске) файла в ОС Windows 7, 8, 10 и в ОС Windows XP (не в оболочке Explorer):

– с включенной ВО SIA изменяется ВО SIA;

– с выключенной ВО SIA никаких изменений ВО не происходит.

17. При редактировании файла:

– с включенной ВО SIA синхронно изменяются ВО SIM = SIA = SIX;

– с выключенной ВО SIA синхронно изменяются ВО SIM = SIX.

18. При редактировании файла в приложении Microsoft Office синхронно изменяются ВО SIM = SIA = SIX = FNM = FNA = FNX, но с некоторыми временными погрешностями в пределах секунды (рис. 2.20). Это объясняется затянутым процессом сохранения файла на основе глобального шаблона Normal.dot.

```
Временные отметки файла <\.\poligon\1\aaaww.docx>
SIC: Время создания файла        1.01.1601  0:00:00  C = 1
SIM: Время посл. модификации     6.03.2019 11:05:54  M = 131963439542155380
SIA: Время последнего доступа    6.03.2019 11:05:54  A = 131963439541375378
SIX: Время модификации записи    6.03.2019 11:05:54  X = 131963439543559382
FNC: Время создания файла        1.01.1601  0:00:00  C = 1
FNM: Время посл. модификации     6.03.2019 11:05:54  M = 131963439542155380
FNA: Время последнего доступа    6.03.2019 11:05:54  A = 131963439541375378
FNX: Время модификации записи    6.03.2019 11:05:54  X = 131963439543559382
```

Рис. 2.20. ВО редактированного файла

19. При разархивировании у файла-архива во всех версиях ОС Windows:

– с включенной ВО SIA изменяется ВО SIA;

– с выключенной ВО SIA изменений не происходит.

20. При разархивировании архиватором WinRAR во всех версиях ОС Windows файлов с расширением 7z, rar, zip у вновь полученных файлов синхронно изменяются ВО SIC = SIA = SIX = FNC = FNM = FNA = FNX. При этом возможно отставание ВО SIX на несколько секунд. А ВО SIM округляется до секунд при разархивировании файлов с расширением rar.

21. При разархивировании архиваторами 7-ZIP и встроенным архиватором во всех версиях ОС Windows:

– файлов с расширением 7z, rar у вновь формируемых файлов синхронно изменяются ВО SIC = SIA = SIX = FNC = FNM = FNA = FNX;

– файлов с расширением zip у вновь формируемых файлов синхронно изменяются ВО SIX = FNC = FNM = FNA = FNX.

В обоих случаях возможно отставание ВО SIX на несколько секунд.

22. При удалении файлов с помощью штатных средств во всех версиях ОС Windows происходит следующее:

– файловая запись удаленного файла помечается как свободная (флаг использования, находящийся по смещению 0x16 от начала записи (сигнатура «FILE»), устанавливается в значение «0x00»);

– кластеры, соответствующие нерезидентным атрибутам файловой записи, помечаются как свободные. При этом их содержимое не затирается, и в файловой записи сохраняются указатели на кластеры;

– с включенной ВО SIA синхронно изменяются ВО SIA=SIX;

– с выключенной ВО SIA изменяется ВО SIX.

23. При удалении файлов с помощью файловых шредеров во всех версиях ОС Windows происходят следующие изменения в файловых записях (табл. 2.5).

Таблица 2.5
Сводная таблица изменений в файловых записях при удалении с помощью файлового шредера

ПО	Содержимое файла	Файловая запись		
		Имя файла	Временные отметки	Размер файла
File Shredder	перезаписывается нулями или случайной последовательностью символов	изменяется	SIM=SIX=FNM=FNX — время удаления файла, SIC, SIA, FNC, FNA не изменяются	0
Privazer	перезаписывается нулями	изменяется	SIX — время удаления файла, FNC, FNM, FNA, FNX не изменяются, SIC=SIM=SIA — случайная дата	0
Recuva	перезаписывается нулями, если содержимое файла хранится в отдельных кластерах; не перезаписывается, если содержимое файла хранится в файловой записи	не изменяется	не изменяются	не изменяется
SDelete	перезаписывается случайной последовательностью символов	изменяется	SIM=SIX=FNM=FNX — время удаления файла, SIC, SIA, FNC, FNA не изменяются	не изменяется
Far Manager	перезаписывается нулями	изменяется	SIM=SIX=FNM=FNX — время удаления файла, SIC, SIA, FNC, FNA не изменяются	0

Прикладным программам, в том числе предназначенным для гарантированного удаления файлов, установлен запрет на модификацию (редактирование) файловых записей в таблице MFT. Поэтому программы, предназначенные для надежного удаления файлов, прежде чем непосредственно удалить файл, редактируют его содержимое, записывая нули или случайную последовательность символов, и затем переименовывают файл (каждый по своему принципу). При этом ВО будут изменяться следующим образом:

– при редактировании файла синхронно изменяются ВО SIM=SIX;

– при переименовании файла ВО из атрибута FN наследуют значения ВО из атрибута SI, ВО SIX изменяется на время переименования;

– при удалении файла с включенной ВО SIA синхронно изменяются ВО SIA=SIX;

– при удалении файла с выключенной ВО SIA изменяется ВО SIX.

Так как все вышеперечисленные операции файловыми шредерами выполняются одномоментно, то у файловой записи удаленного файла ВО SIM=SIX=FNM=FNX соответствуют времени удаления, а ВО SIC, SIA, FNC, FNA остаются прежними (до удаления файла).

Выявленные закономерности наблюдались многократно в ОС Windows XP, 7, 8 и 10 с различной программно-аппаратной конфигурацией. В результате исследований процессов изменения ВО были получены следующие результаты:

1. Подтверждены результаты исследований Г. Чо, Т. Кнутсона, В. Матвеевой для ФОп: копирование, перемещение, редактирование, открытие, удаление файла, просмотр и изменение его атрибутов.

2. Уточнены изменения ВО для ФОп: редактирование в пакете программ Microsoft Office и перемещение между томами (из FAT в NTFS).

Ранее Т. Кнутсон определил, что при перемещении из FAT в NTFS у файла ВО SIA и SIX синхронно изменяются, а ВО SIC и SIM наследуются от исходного файла. В результате использования представленного алгоритма дополнительно определено, что ВО SIM округляется до секунд, ВО SIC округляется до миллисекунд.

Г. Чо при исследовании операции редактировании файла в приложении Microsoft Office обнаружил, что у файла синхронно изменяются ВО SIM = SIA = SIX. В результате использования алгоритма дополнительно определено, что изменившиеся ВО не абсолютно идентичны, а имеют некоторые отличия в десятых долях секунд.

3. Уточнен характер изменения ВО с включенной и выключенной опцией обновления ВО SIA при выполнении всех ФОп для разных версий ОС Windows.

4. Обнаружены новые значимые комбинации изменения ВО.

При перемещении файла с установленными атрибутами «только чтение», «системный» или «скрытый» в файловом менеджере Total Commander ВО FNC, FNM, FNA наследуют значения ВО SIC, SIM, SIA соответственно, а ВО SIX=FNX синхронно изменяются.

Ранее не было определено, как изменяются ВО при разархивировании файла. В результате использования алгоритма установлено, что характер изменений ВО при разархивировании зависит как от применяемого архиватора, так и от формата архива (zip, rar).

5. Установлено, что для всех ФОп, кроме разархивирования, формат файла не влияет на характер изменения ВО.

В результате исследований был получен большой объем данных, который был систематизирован и представлен в виде сводной таблицы (см. табл. 2.6), удобной для их дальнейшего анализа. В табл. 2.6 «вкл. SIA» обозначает, что ВО последнего доступа включена, а «выкл. SIA» — ВО последнего доступа выключена. Серые ячейки указывают на синхронное изменение ВО после выполнения ФОп. Белые ячейки — на отсутствие изменений. Буква T в ячейках обозначает момент выполнения ФОп, аббревиатуры SIC, SIM, SIA, SIX — наследование ВО значений из атрибутов $STANDARD_INFORMATION.

Таблица 2.6
Таблица изменений ВО при различных ФОп

ФОп	**SI**				**FN**			
	C	**M**	**A**	**X**	**C**	**M**	**A**	**X**
Копирование в ОС Windows 7 (вкл. SIA) (исходный объект[1])			T					
Копирование в ОС Windows 7 (выкл. SIA) (исходный объект)								
Копирование в ОС Windows XP, Windows 8, 10 (вкл./выкл. SIA) (исходный объект)								
Копирование в ОС Windows 7 (вкл./выкл. SIA) (новый объект[2])	T		T	T	T	T	T	T
Копирование в ОС Windows XP, Windows 8, 10 или в файловых менеджерах Total Commander, Far Manager (вкл./выкл. SIA) (новый объект)	T		T		T	T	T	T
Перемещение/переименование (вкл./выкл. SIA) (новый объект)				T	SIC	SIM	SIA	SIX
Перемещение/переименование в файловом менеджере Total Commander для файлов с установленными атрибутами «только чтение», «системный» или «скрытый» (вкл./выкл. SIA) (новый объект)				T	SIC	SIM	SIA	T

[1] файл, над которым выполняли ФОп
[2] новый файл, который был создан в результате выполнения ФОп над исходным объектом

ФОп	SI				FN			
	C	M	A	X	C	M	A	X
Перемещение/переименование из файловой системы FAT в файловую систему NTFS (вкл./выкл. SIA) (новый объект)			T	T	T	T	T	T
Удаление (вкл. SIA)			T	T				
Удаление (выкл. SIA)				T				
Просмотр атрибутов (вкл. SIA)			T					
Просмотр атрибутов (выкл. SIA)								
Изменение атрибутов (вкл./выкл. SIA)				T				
Открытие в ОС Windows XP в оболочке Explorer (вкл. SIA)			T	T				
Открытие в ОС Windows 7, 8, 10 и в ОС Windows XP (не в оболочке Explorer) (вкл. SIA)			T					
Открытие в ОС Windows 7 (выкл. SIA)				T				
Открытие в ОС Windows XP, Windows 8, 10 (выкл. SIA)								
Редактирование (вкл. SIA)		T	T	T				
Редактирование (выкл. SIA)		T		T				
Редактирование в пакете Microsoft Office (вкл./выкл. SIA)		T	T	T		T	T	T
Исполнение (запуск) в ОС Windows XP в оболочке Explorer (вкл. SIA)			T	T				
Исполнение (запуск) в ОС Windows 7, 8, 10 и в ОС Windows XP (не в оболочке Explorer) (вкл. SIA)			T					
Исполнение (запуск) в ОС Windows 7 (выкл. SIA)				T				
Исполнение (запуск) в ОС Windows XP, Windows 8, 10 (выкл. SIA)								
Разархивирование (вкл. SIA) (исходный объект)			T					
Разархивирование (выкл. SIA) (исходный объект)								
Разархивирование архиватором 7-Zip и встроенным архиватором Windows файлов с расширением 7z, rar (вкл./выкл. SIA) (новый объект)	T		T	T	T	T	T	T
Разархивирование архиваторами 7-Zip и встроенным архиватором Windows файлов с расширением zip (вкл./выкл. SIA) (новый объект)				T	T	T	T	T
Разархивирование архиватором WinRAR файлов с расширением zip, 7z, rar (вкл./выкл. SIA) (новый объект)	T		T	T	T	T	T	T

Для удобства последующего анализа изменений ВО файловые операции, которые приводят к синхронному изменению одинаковых ВО, объединены в группы (табл. 2.7). Каждая группа содержит одну или две ФОп. Список групп

может пополняться при обнаружении ФОп, под воздействием которых будут возникать отличные от уже известных изменения ВО, например, в случае выпуска новых версий ОС Windows.

Таблица 2.7

Таблица изменений ВО с группированием ФОп

ФОп, объединенные в группы	**SI**				**FN**			
	C	**M**	**A**	**X**	**C**	**M**	**A**	**X**
Копирование в ОС Windows XP, 8, 10 или в файловых менеджерах Total Commander, Far Manager (вкл./выкл. SIA) (новый объект)	T		T		T	T	T	T
Копирование в ОС Windows 7 (вкл./выкл. SIA) (новый объект); Разархивирование архиватором 7-Zip и встроенным архиватором Windows файлов с расширением 7z, rar (вкл./выкл. SIA) (новый объект); Разархивирование архиватором WinRAR файлов с расширением zip, 7z, rar (вкл./выкл. SIA) (новый объект)	T		T	T	T	T	T	T
Редактирование (вкл. SIA)		T	T	T				
Редактирование (выкл. SIA)		T		T				
Редактирование в пакете Microsoft Office (вкл./выкл. SIA)		T	T	T		T	T	T
Перемещение/переименование (вкл./выкл. SIA) (новый объект)				T	SIC	SIM	SIA	SIX
Перемещение/переименование из файловой системы FAT в файловую систему NTFS (вкл./выкл. SIA) (новый объект)			T	T	T	T	T	T
Разархивирование архиваторами 7-Zip и встроенным архиватором Windows файлов с расширением zip (вкл./выкл. SIA) (новый объект)				T	T	T	T	T
Копирование в ОС Windows 7 (вкл. SIA) (исходный объект); Просмотр атрибутов (вкл. SIA); Запуск/открытие в ОС Windows 7, 8, 10 и в ОС Windows XP (не в оболочке Explorer) (вкл. SIA); Разархивирование (вкл. SIA) (исходный объект)			T					
Изменение атрибутов (вкл./выкл. SIA); Удаление (выкл. SIA); Запуск/открытие в ОС Windows XP в оболочке Explorer (выкл. SIA)				T				
Запуск/открытие в ОС Windows XP в оболочке Explorer (вкл. SIA); Удаление (вкл. SIA)			T	T				
Перемещение/переименование в файловом менеджере Total Commander для файлов с установленными атрибутами «только чтение», «системный» или «скрытый» (вкл./выкл. SIA) (новый объект)				T	SIC	SIM	SIA	T

На основе сформированной обобщенной таблицы изменения ВО (табл. 2.7) возможно восстанавливать последнюю ФОп, совершенную над файлом. Например, если у исследуемого файла ВО SIM=SIA=SIX, и они

изменились позже ВО SIC, FNC, FNM, FNA, FNX, то можно определить, что данной комбинации ВО соответствует операция «редактирование».

Для того, чтобы восстановить цепочку из нескольких ФОп, необходимо знать возможные комбинации ВО, которые могут возникать при последовательном выполнении ФОп. С этой целью необходимо на основе табл. 2.7 разработать модель изменения ВО на основе теории конечных автоматов. При этом в качестве входных символов, которые подаются на вход автомата, рассматривать ФОп, в качестве состояний автомата — комбинации ВО. Функция переходов между состояниями будет формироваться на основе таблицы изменений ВО и описывать, каким образом ФОп изменяют состояния ВО.

2.4 Анализ демаскирующих признаков подделки внешних ВО

При восстановлении последовательности ФОп важно знать, что извлеченные ВО являются достоверными, так как они могут быть изменены, в том числе с целью сокрытия следов преступных (неправомочных) действий. Изменять ВО можно специальными утилитами и в файловых менеджерах, которые имеют подобный функционал.

Для изучения и нахождения признаков, позволяющих выявлять факты фальсификации ВО, выбраны три свободно распространяемые программы: файловые менеджеры Far Manager, Total Commander и утилита командной строки TimeStomp [71]. После применения каждой программы с помощью программы FTA фиксировались и подвергались дальнейшему анализу измененные значения ВО.

Файловый менеджер Total Commander имеет специальное меню для изменения атрибутов файлов (рис. 2.21).

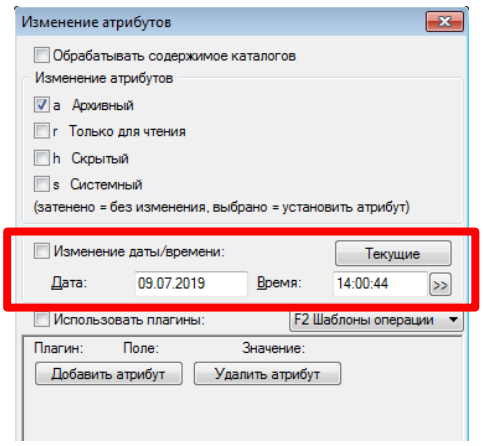


Рис. 2.21. Изменение ВО SIM с помощью файлового менеджера Total Commander

Total Commander позволяет изменять только три ВО: SIC, SIM, SIA (рис. 2.22).

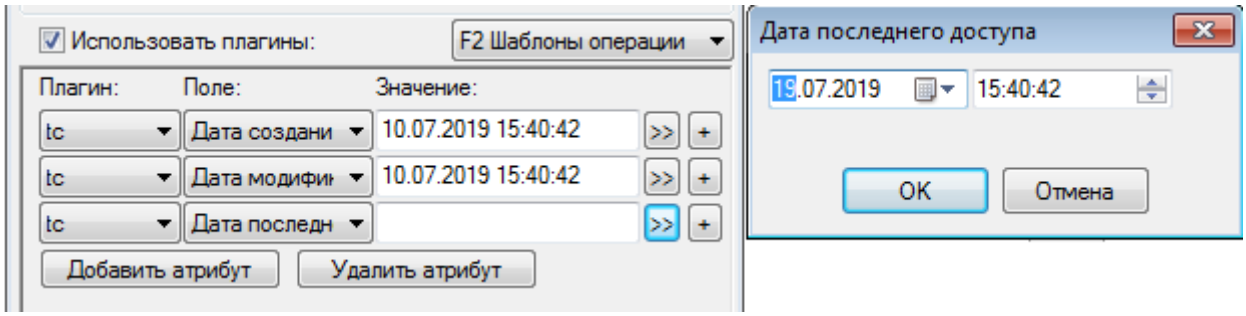


Рис. 2.22. Изменение ВО с помощью файлового менеджера Total Commander

На рис. 2.23 и 2.24 приведены ВО файла до и после их фальсификации. Из рисунков видно, что после изменения ВО в семи младших разрядах ее точного значения устанавливаются нули, поскольку файловый менеджер производит округление ВО до секунд. При этом ВО SIX соответствует времени произведения фальсификации атрибутов.

```
Временные отметки файла <\.\poligon\1\3\test.txt>
SIC: Время создания файла       19.07.2019  9:40:42  C = 132080028422075622
SIM: Время посл. модификации    19.07.2019  9:40:42  M = 132080028422075622
SIA: Время последнего доступа   19.07.2019  9:40:42  A = 132080028422075622
SIX: Время модификации записи   19.07.2019  9:40:45  X = 132080028452183675
FNC: Время создания файла       19.07.2019  9:40:42  C = 132080028422075622
FNM: Время посл. модификации    19.07.2019  9:40:42  M = 132080028422075622
FNA: Время последнего доступа   19.07.2019  9:40:42  A = 132080028422075622
FNX: Время модификации записи   19.07.2019  9:40:42  X = 132080028422075622
```

Рис. 2.23. ВО файла «test.txt» до изменения атрибутов

```
Временные отметки файла <\.\poligon\1\3\test.txt>
SIC: Время создания файла       10.07.2019  9:40:42  C = 132072252420000000
SIM: Время посл. модификации    10.07.2019  9:40:42  M = 132072252420000000
SIA: Время последнего доступа   10.07.2019  9:40:42  A = 132072252420000000
SIX: Время модификации записи   19.07.2019  9:44:48  X = 132080030887815953
FNC: Время создания файла       19.07.2019  9:40:42  C = 132080028422075622
FNM: Время посл. модификации    19.07.2019  9:40:42  M = 132080028422075622
FNA: Время последнего доступа   19.07.2019  9:40:42  A = 132080028422075622
FNX: Время модификации записи   19.07.2019  9:40:42  X = 132080028422075622
```

Рис. 2.24. ВО файла «test.txt» после изменения атрибутов
с помощью файлового менеджера Total Commander

Аналогичные возможности по изменению ВО файлов предоставляет утилита TimeStomp. Фальсификация может быть обнаружена по округлению ВО до секунд

Наиболее универсальным для изменения ВО является файловый менеджер Far Manager, так как позволяет осуществлять модификацию четырех ВО: SIC, SIM, SIA и SIX с точностью до миллисекунд (рис. 2.25–2.27).

```
                                    DD.MM.YYYYY hh:mm:ss,ms
Last write time:                    10.07.2019  15:48:05,393
Creation time:                      10.07.2019  15:48:05,393
Last access time:                   10.07.2019  15:48:05,393
Change time:                        10.07.2019  15:48:08,591
              [ Original ] [ Current ] [ Blank ]
```

Рис. 2.25. Изменение ВО с помощью
файлового менеджера Far Manager

```
Временные отметки файла <\.\poligon\1\4\test.txt>
SIC: Время создания файла       19.07.2019  9:48:05  C = 132080032853935412
SIM: Время посл. модификации    19.07.2019  9:48:05  M = 132080032853935412
SIA: Время последнего доступа   19.07.2019  9:48:05  A = 132080032853935412
SIX: Время модификации записи   19.07.2019  9:48:08  X = 132080032885915468
FNC: Время создания файла       19.07.2019  9:48:05  C = 132080032853935412
FNM: Время посл. модификации    19.07.2019  9:48:05  M = 132080032853935412
FNA: Время последнего доступа   19.07.2019  9:48:05  A = 132080032853935412
FNX: Время модификации записи   19.07.2019  9:48:05  X = 132080032853935412
```

Рис. 2.26. ВО файла «test.txt» до изменения атрибутов

```
Временные отметки файла <\.\poligon\1\4\test.txt>
SIC: Время создания файла       10.07.2019  9:48:05  C = 132072256853930000
SIM: Время посл. модификации    10.07.2019  9:48:05  M = 132072256853930000
SIA: Время последнего доступа   10.07.2019  9:48:05  A = 132072256853930000
SIX: Время модификации записи   10.07.2019  9:48:08  X = 132072256885910000
FNC: Время создания файла       19.07.2019  9:48:05  C = 132080032853935412
FNM: Время посл. модификации    19.07.2019  9:48:05  M = 132080032853935412
FNA: Время последнего доступа   19.07.2019  9:48:05  A = 132080032853935412
FNX: Время модификации записи   19.07.2019  9:48:05  X = 132080032853935412
```

Рис. 2.27. ВО файла «test.txt» после изменения атрибутов
с помощью файлового менеджера Far Manager

Из рис. 2.27 видно, что после изменения ВО имеются нули в четырех младших разрядах точной ВО, так как происходит их округление до миллисекунд.

Таким образом, модификация ВО при помощи специализированных программ и файловых менеджеров сопровождается округлением соответствующих ВО до секунд и миллисекунд. Следовательно, по данному признаку можно уверенно выявлять факты фальсификации ВО. А для определения времени умышленной модификации ВО использовать ВО SIX.

2.5 Выводы

1. Разработан алгоритм экспериментального исследования механизмов изменения ВО, которая позволяет устанавливать характер изменений ВО при выполнении ФОп. Алгоритм заключается в подготовке файлов специальным образом, выполнении ФОп над подготовленными файлами и фиксировании изменений ВО после выполнения каждой операции. В алгоритме используется программный инструментарий, позволяющий выводить ВО файлов, хранящиеся в таблице MFT ФС NTFS в атрибутах SI и FN и индексных записях родительских каталогов, в режиме «только чтение». Алгоритм опробован на компьютерах с различной аппаратной конфигурацией под управлением ОС Windows XP, 7, 8 и 10. Отличием предложенного алгоритма от существующих является системность, полнота исследований и широта охвата разнообразия ФОп и вариантов их выполнения, а также возможность его автоматизации. При появлении новых версий ОС Windows рекомендуется использовать разработанный алгоритм для обнаружения новых закономерностей в изменениях ВО.

2. Применение алгоритма позволило выявить характерные признаки ранее не исследованных ФОп: перемещение/переименование в файловом менеджере Total Commander, разархивирование; уточнить закономерности изменения ВО для ФОп: редактирование в пакете программ Microsoft Office, перемещение между томами (из FAT в NTFS); уточнить изменения ВО

с включенной и выключенной опцией обновления ВО SIA при выполнении всех ФОп для разных версий ОС Windows.

3. Составлена обобщенная таблица изменений ВО при выполнении ФОп. В таблицу входят 12 групп ФОп, выявленных по результатам экспериментальных исследований, под воздействием которых изменяются ВО. Каждая группа содержит одну или две ФОп, которые приводят к одинаковому изменению ВО. Таблица изменений ВО наглядно демонстрирует закономерности в изменениях ВО и позволяет определять последнюю операцию, совершенную над файлом. Кроме того, таблица может быть положена в основу модели, описывающей изменения ВО при совершении ФОп над файлами.

4. Проведен анализ изменений ВО при попытке их умышленной модификации с помощью специализированных программ. Анализ выявил следующие признаки фактов фальсификации ВО: округление ВО, которые умышленно изменялись программами, до секунд и миллисекунд; установка ВО последней модификации метаданных файла (SIX) на момент времени применения программы для умышленной модификации ВО.

3 РАЗРАБОТКА МОДЕЛИ ИЗМЕНЕНИЯ ЗНАЧЕНИЙ ВНЕШНИХ ВО И МЕТОДИКИ ВОССТАНОВЛЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ФОП

3.1 Разработка модели изменения значений внешних ВО

Анализ результатов экспериментов, представленных во 2 главе диссертации, позволяет утверждать, что при выполнении операций над файлами ВО изменяются по некоторым фиксированным «закономерностям», запрограммированным в ядре ОС. Закономерности процесса изменения ВО разработчиками ОС Windows и прикладных пакетов программного обеспечения не документированы. Математических моделей, описывающих процесс изменения ВО при осуществлении пользователем ПЭВМ ФОп, представленный в 1 главе рукописи, информационный поиск не выявил.

С целью сопоставления возможных последовательностей ФОп наблюдаемым вариантам состояний ВО файлов разработана модель изменения значений ВО при выполнении ФОп (далее — модель изменения значений ВО), основанная на математическом аппарате теории автоматов.

Детерминированным конечным автоматом (КА) называется следующий набор объектов [72–75]:

1. $X = \{x_1, x_2, \ldots, x_n\}$ — множество символов (входной алфавит). Каждый отдельный символ алфавита называется буквой. Последовательность букв конечной длины — словом. Число букв в слове — длиной.
2. $Y = \{y_1, y_2, \ldots, y_m\}$ — множество символов (выходной алфавит).
3. $S = \{s_0, s_2, \ldots, s_k\}$ — множество состояний.
4. f_s — функция переходов КА из одного состояния в другое.
5. f_y — функция выходов.
6. s_0 — начальное состояние, $s_0 \in S$.

Указанные объекты связаны выражением $KA = (X, Y, S, f_s, f_y, s_0)$, при этом X, Y, S — непустые множества.

Функция переходов каждой паре «входной символ x_j — состояние s_i» ставит в соответствие новое состояние s_l и записывается следующим образом:

$$s_l = f_s(x_j, s_i),$$

где: x_j — входной символ;

s_i — текущее состояние автомата;

s_l — новое состояние автомата.

Функция выходов каждой паре «входной символ x_j — состояние s_i» ставит в соответствие выходной символ x_l и записывается следующим образом:

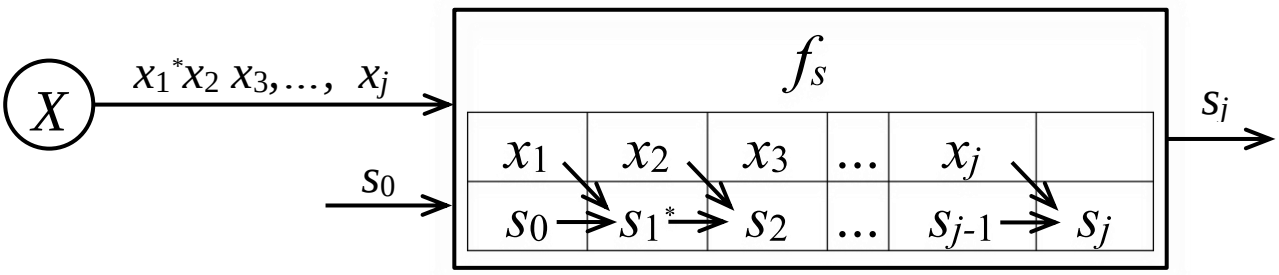
$$y_l = f_y(x_j, s_i),$$

где: x_j — входной символ;

s_i — текущее состояние автомата;

y_l — выходной символ.

Если выходные символы совпадают с состояниями, то автомат называется автоматом без выходного преобразователя и описывается четырьмя элементам $KA = (X, S, f_s, s_0)$. Модель изменения значений ВО может быть представлена в виде конечного автомата без выходного преобразователя (рис. 3.1).



X — множество ФОп,
$x_1\ x_2\ x_3, \ldots, x_j$ — последовательность выполненных ФОп,
j — количество выполненных ФОп,
S — множество состояний ВО файлов (векторы временных уровней),
$s_0, s_1, s_2, \ldots, s_{j-1}, s_j \in S$,
s_0 — начальное состояние ВО,
s_j — конечное состояние ВО,
f_s — функция переходов,
$s_1 = f_s(s_0, x_1), s_2 = f_s(s_1, x_2), \ldots, s_j = f_s(s_{j-1}, x_j)$

Рис. 3.1. Математическая модель изменения значений ВО

* Индексами 1, 2, 3 … j обозначены порядковые номера выполненных ФОп и полученных состояний ВО

Выбор детерминированного конечного автомата в качестве математической модели изменения значений ВО обусловлен следующими причинами:

1. Множество символов и множество состояний конечны. Множество символов — множество ФОп, которые подаются на вход автомата, определено в границах исследования. Множество состояний ВО ограничено количеством ВО, фиксируемых для одной файловой записи в таблице MFT ОС Windows.

2. Переходы из состояния в состояние однозначны. При совершении операций ВО изменяются по строго определенному закону, что было установлено в ходе экспериментов (см. п. 2.3 рукописи).

Для описания модели (рис. 3.1) необходимо определить множества X, S и задать функцию переходов f_s.

В качестве элементов входного алфавита X будем рассматривать ФОп, которые совершаются над файлами и приводят к изменению их ВО (табл. 3.1).

Таблица 3.1
Элементы входного алфавита X

X	ФОп
x_1	Копирование в ОС Windows XP, 8, 10 или в файловых менеджерах Total Commander, Far Manager (вкл./выкл. SIA) (новый объект)
x_2	Копирование в ОС Windows 7 (вкл./выкл. SIA) (новый объект); Разархивирование архиватором 7-Zip и встроенным архиватором Windows файлов с расширением 7z, rar (вкл./выкл. SIA) (новый объект); Разархивирование архиватором WinRAR файлов с расширением zip, 7z, rar (вкл./выкл. SIA) (новый объект)
x_3	Редактирование (вкл. SIA)
x_4	Редактирование (выкл. SIA)
x_5	Редактирование в пакете Microsoft Office (вкл./выкл. SIA)
x_6	Перемещение/переименование (вкл./выкл. SIA) (новый объект)
x_7	Перемещение/переименование из файловой системы FAT в файловую систему NTFS (вкл./выкл. SIA) (новый объект)
x_8	Разархивирование архиваторами 7-Zip и встроенным архиватором Windows файлов с расширением zip (вкл./выкл. SIA) (новый объект)
x_9	Копирование в ОС Windows 7 (вкл. SIA) (исходный объект); Просмотр атрибутов (вкл. SIA); Запуск/открытие в ОС Windows 7, 8, 10 и в ОС Windows XP (не в оболочке Explorer) (вкл. SIA); Разархивирование (вкл. SIA) (исходный объект)

X	ФОп
x_{10}	Изменение атрибутов (вкл./выкл. SIA); Удаление (выкл. SIA); Запуск/открытие в ОС Windows XP в оболочке Explorer (выкл. SIA)
x_{11}	Запуск/открытие в ОС Windows XP в оболочке Explorer (вкл. SIA); Удаление (вкл. SIA)
x_{12}	Перемещение/переименование в файловом менеджере Total Commander для файлов с установленными атрибутами «только чтение», «системный» или «скрытый» (вкл./выкл. SIA) (новый объект)

Табл. 3.1 построена на основе табл. 2.7 (п. 2.3), в ней учтены все ФОп, которые приводят к изменению значений ВО. Таким образом, множество X состоит из 12 элементов, $|X| = 12$.

Множеству состояний автомата S поставим в соответствие множество состояний ВО, полученных от воздействия ФОп. В главе 2 диссертации показано, что ВО имеют числовые значения, что делает возможным сравнивать их между собой. На рис. 3.2 представлены ВО файла «111.txt».

```
Временные отметки файла <\.\111111\aaaa\111.txt>
SIC: Время создания файла        2.11.2017  8:37:57  C = 131540854773326225
SIM: Время посл. модификации     2.11.2017  8:49:23  M = 131540861630942270
SIA: Время последнего доступа    2.11.2017  8:37:57  A = 131540854773326225
SIX: Время модификации записи    2.11.2017  8:51:13  X = 131540862732844210
FNC: Время создания файла        2.11.2017  8:37:57  C = 131540854773326225
FNM: Время посл. модификации     2.11.2017  8:49:23  M = 131540861630942270
FNA: Время последнего доступа    2.11.2017  8:37:57  A = 131540854773326225
FNX: Время модификации записи    2.11.2017  8:49:34  X = 131540861741462469
```

Рис. 3.2. ВО файла «111.txt»

Для файла «111.txt» справедливы следующие отношения между ВО:

$$\begin{aligned} &SIC = SIA = FNC = FNA \\ &SIM = FNM \\ &SIC < SIM \\ &SIC < SIX \\ &SIC < FNX \\ &SIM < SIX \\ &SIM < FNX \\ &FNX < SIX \end{aligned} \qquad (3.1)$$

Соотношения ВО показывают порядок их изменения во времени. Для построения модели эти соотношения представим в виде *временных уровней*, где самой «ранней» ВО присвоим значение — «1», а более поздним — соответ-

ственно «2», «3», «4», «5», «6», «7», «8». Например, на основании (3.1) ВО файла «111.txt» будут присвоены временные уровни в соответствии с табл. 3.2.

Таблица 3.2

Соответствие между ВО и временными уровнями для файла «111.txt»

Временная отметка	SIC	SIM	SIA	SIX	FNC	FNM	FNA	FNX
Временной уровень	1	2	1	4	1	2	1	3

Полученную последовательность временных уровней запишем в одну строку, которую будем называть *вектором временных уровней (ВВУ)*. Таким образом, ВВУ состоит из 8 временных уровней, каждый из которых соответствует типу ВО и имеет однозначное местоположение в записи: (SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX). Значения векторов временных уровней являются элементами множества состояний *S*. Для файла «111.txt» ВВУ принимает вид (1,2,1,4,1,2,1,3). Согласно теории комбинаторики, множество *S* может состоять из $8^8 = 16\ 777\ 216$ элементов. Однако далее при построении таблицы переходов были выявлены только 458 элемента.

Функция переходов между состояниями f_s описывает, каким образом ФОп изменяют состояния ВО. Функция переходов сформирована на основе таблицы изменений ВО (табл. 2.7) и представлена *таблицей переходов*.

Алгоритм построения *таблицы переходов между состояниями* (значениями ВВУ) s_i приведен ниже.

Начальным состоянием будем считать s_0 = (1,1,1,1,1,1,1,1), когда все ВО равны. Оно означает, что файл был создан и ФОП над ним не совершались. К такому файлу будем применять все ФОп из входного алфавита *X* (табл. 3.1), под воздействием которых ВО изменяются по правилам (выявленным закономерностям), представленным на рис. 3.3, где красным цветом выделены ВО, которые обновились одномоментно. Например, после воздействия операции x_5 у файла синхронно изменятся шесть ВО: SIM, SIA, SIX, FNM, FNA и FNX.

$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x1} (\boldsymbol{SIC}, SIM, \boldsymbol{SIA}, SIX, \boldsymbol{FNC}, \boldsymbol{FNM}, \boldsymbol{FNA}, \boldsymbol{FNX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_2} (\boldsymbol{SIC}, SIM, \boldsymbol{SIA}, \boldsymbol{SIX}, \boldsymbol{FNC}, \boldsymbol{FNM}, \boldsymbol{FNA}, \boldsymbol{FNX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_3} (SIC, \boldsymbol{SIM}, \boldsymbol{SIA}, \boldsymbol{SIX}, FNC, FNM, FNA, FNX)$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_4} (SIC, \boldsymbol{SIM}, SIA, \boldsymbol{SIX}, FNC, FNM, FNA, FNX)$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_5} (SIC, \boldsymbol{SIM}, \boldsymbol{SIA}, \boldsymbol{SIX}, FNC, \boldsymbol{FNM}, \boldsymbol{FNA}, \boldsymbol{FNX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_6} (SIC, SIM, SIA, \boldsymbol{SIX}, \boldsymbol{SIC}, \boldsymbol{SIM}, \boldsymbol{SIA}, \boldsymbol{SIX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_7} (SIC, SIM, \boldsymbol{SIA}, \boldsymbol{SIX}, \boldsymbol{FNC}, \boldsymbol{FNM}, \boldsymbol{FNA}, \boldsymbol{FNX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_8} (SIC, SIM, SIA, \boldsymbol{SIX}, \boldsymbol{FNC}, \boldsymbol{FNM}, \boldsymbol{FNA}, \boldsymbol{FNX})$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_9} (SIC, SIM, \boldsymbol{SIA}, SIX, FNC, FNM, FNA, FNX)$,
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_{10}} (SIC, SIM, SIA, \boldsymbol{SIX}, FNC, FNM, FNA, FNX)$.
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_{11}} (SIC, SIM, \boldsymbol{SIA}, \boldsymbol{SIX}, FNC, FNM, FNA, FNX)$.
$(SIC, SIM, SIA, SIX, FNC, FNM, FNA, FNX) \xrightarrow{x_{12}} (SIC, SIM, SIA, \boldsymbol{SIX}, \boldsymbol{SIC}, \boldsymbol{SIM}, \boldsymbol{SIA}, \boldsymbol{FNX})$.

Рис. 3.3. Правила изменения ВО

В результате применения каждой операции для состояния s_0 по правилу изменения ВО сформируются новые состояния:

$$
\begin{aligned}
&f_s(x_1, s_0) = s_1, && s_1 = (2,1,2,1,2,2,2,2),\\
&f_s(x_2, s_0) = s_2, && s_2 = (2,1,2,2,2,2,2,2),\\
&f_s(x_3, s_0) = s_3, && s_3 = (1,2,2,2,1,1,1,1),\\
&f_s(x_4, s_0) = s_4, && s_4 = (1,2,1,2,1,1,1,1),\\
&f_s(x_5, s_0) = s_5, && s_5 = (1,2,2,2,1,2,2,2),\\
&f_s(x_6, s_0) = s_6, && s_6 = (1,1,1,2,1,1,1,1),\\
&f_s(x_7, s_0) = s_7, && s_7 = (1,1,2,2,2,2,2,2),\\
&f_s(x_8, s_0) = s_8, && s_8 = (1,1,1,2,2,2,2,2),\\
&f_s(x_9, s_0) = s_9, && s_9 = (1,1,2,1,1,1,1,1),\\
&f_s(x_{10}, s_0) = s_6, && s_6 = (1,1,1,2,1,1,1,1),\\
&f_s(x_{11}, s_0) = s_{10}, && s_{10} = (1,1,2,2,1,1,1,1),\\
&f_s(x_{12}, s_0) = s_{11}, && s_{11} = (1,1,1,2,1,1,1,2).
\end{aligned}
\qquad (3.2)
$$

Начальное состояние s_0 записывается в верхнюю строку первого столбца таблицы. В остальные ячейки столбца записываются состояния, полученные от воздействия ФОп. Полученные таким образом состояния s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11} являются исходными для построения следующих столбцов таблицы (рис. 3.4).

x(t)	s(t)											
	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11
x1	s1											
x2	s2											
x3	s3											
x4	s4											
x5	s5											
x6	s6											
x7	s7											
x8	s8											
x9	s9											
x10	s6											
x11	s10											
x12	s11											

Рис. 3.4. Формирование таблицы переходов между состояниями

К состояниям, записанным в верхних ячейках новых столбцов, применяются ФОп из множества X (табл. 3.1), в результате чего образуются как новые, так и уже известные состояния. Например, для s_1 при воздействии ФОп x_1, x_2, x_3, x_4, x_5 по правилу изменения ВО сформируются уже известные состояния s_1, s_2, s_3, s_4, s_5, а при воздействии ФОп x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12} появятся новые состояния s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, s_{17}, s_{18}:

$$
\begin{aligned}
&f_s(x_1, s_1) = s_1, && s_1 = (2,1,2,1,2,2,2,2),\\
&f_s(x_2, s_1) = s_2, && s_2 = (2,1,2,2,2,2,2,2),\\
&f_s(x_3, s_1) = s_3, && s_3 = (1,2,2,2,1,1,1,1),\\
&f_s(x_4, s_1) = s_4, && s_4 = (1,2,1,2,1,1,1,1),\\
&f_s(x_5, s_1) = s_5, && s_5 = (1,2,2,2,1,2,2,2),\\
&f_s(x_6, s_1) = s_{12}, && s_{12} = (2,1,2,3,2,1,2,1),\\
&f_s(x_7, s_1) = s_{13}, && s_{13} = (2,1,3,3,3,3,3,3),\\
&f_s(x_8, s_1) = s_{14}, && s_{14} = (2,1,2,3,3,3,3,3),\\
&f_s(x_9, s_1) = s_{15}, && s_{15} = (2,1,3,1,2,2,2,2),\\
&f_s(x_{10}, s_1) = s_{16}, && s_{16} = (2,1,2,3,2,2,2,2),\\
&f_s(x_{11}, s_1) = s_{17}, && s_{17} = (2,1,3,3,2,2,2,2),\\
&f_s(x_{12}, s_1) = s_{18}, && s_{18} = (2,1,2,3,2,1,2,3).
\end{aligned}
\qquad (3.3)
$$

Обнаруженные новые состояния являются исходными для построения следующих столбцов таблицы. Данный процесс итерационен, он выполняется для всех появляющихся новых состояний.

Для автоматизации генерирования таблицы переходов написана функция Table.m, листинг которой приведен в Приложении А. Фрагмент таблицы

переходов, сформированной функцией Table.m, представлен на рис. 3.5 (полная таблица приведена в приложении Б).

Количество строк в полной таблице переходов соответствует количеству элементов множества X (табл. 3.1), количество столбцов — количеству элементов множества S (табл. 3.3, полная таблица приведена в приложении В). Для исследуемых двенадцати групп ФОп всего было выявлено 458 элементов из 16 777 216 ($|S| = 458$).

$x(t)$	$S(t)$															
	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	…	s_{457}
x_1	s_1	s_1	s_{19}	s_1	s_1	s_1	s_{19}	s_{19}	s_{19}	s_1	s_{19}	s_{19}	s_{19}	s_{19}	…	s_{19}
x_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	…	s_2
x_3	s_3	s_3	s_3	s_3	s_3	s_{34}	s_3	s_{40}	s_{40}	s_3	s_3	s_{53}	s_{57}	s_{40}	…	s_{348}
x_4	s_4	s_4	s_4	s_{22}	s_4	s_{35}	s_4	s_{41}	s_{49}	s_{22}	s_{22}	s_{54}	s_{58}	s_{41}	…	s_{431}
x_5	s_5	s_5	s_5	s_5	s_5	s_5	s_5	s_{42}	s_{42}	s_5	s_5	s_5	s_5	s_{42}	…	s_5
x_6	s_6	s_{12}	s_{20}	s_{23}	s_{30}	s_{23}	s_{38}	s_{43}	s_{38}	s_{51}	s_{43}	s_{38}	s_{59}	s_{62}	…	s_{211}
x_7	s_7	s_{13}	s_{13}	s_{24}	s_{24}	s_{24}	s_7	s_7	s_7	s_7	s_7	s_7	s_{13}	s_{13}	…	s_{24}
x_8	s_8	s_{14}	s_{14}	s_{25}	s_{31}	s_{25}	s_8	s_{44}	s_8	s_{44}	s_{44}	s_8	s_{14}	s_{63}	…	s_{90}
x_9	s_9	s_{15}	s_{21}	s_{26}	s_{26}	s_{36}	s_{39}	s_{45}	s_{45}	s_9	s_{39}	s_{55}	s_{60}	s_{64}	…	s_{457}
x_{10}	s_6	s_{16}	s_{16}	s_{27}	s_{32}	s_{23}	s_6	s_{46}	s_{50}	s_{52}	s_{52}	s_{38}	s_{12}	s_{65}	…	s_{456}
x_{11}	s_{10}	s_{17}	s_{17}	s_{28}	s_{28}	s_{37}	s_{10}	s_{47}	s_{47}	s_{10}	s_{10}	s_{56}	s_{61}	s_{66}	…	s_{434}
x_{12}	s_{11}	s_{18}	s_{18}	s_{29}	s_{33}	s_{29}	s_{11}	s_{48}	s_{11}	s_{48}	s_{48}	s_{11}	s_{18}	s_{67}	…	s_{94}

Рис. 3.5. Фрагмент таблицы переходов

Таблица 3.3

Элементы множества S

S	ВВУ
s_0	(1,1,1,1,1,1,1,1)
s_1	(2,1,2,1,2,2,2,2)
s_2	(2,1,2,2,2,2,2,2)
s_3	(1,2,2,2,1,1,1,1)
s_4	(1,2,1,2,1,1,1,1)
s_5	(1,2,2,2,1,2,2,2)
s_6	(1,1,1,2,1,1,1,1)
s_7	(1,1,2,2,2,2,2,2)
s_8	(1,1,1,2,2,2,2,2)
s_9	(1,1,2,1,1,1,1,1)

S	ВВУ
s_{10}	(1,1,2,2,1,1,1,1)
s_{11}	(1,1,1,2,1,1,1,2)
s_{12}	(2,1,2,3,2,1,2,1)
s_{13}	(2,1,3,3,3,3,3,3)
s_{14}	(2,1,2,3,3,3,3,3)
s_{15}	(2,1,3,1,2,2,2,2)
s_{16}	(2,1,2,3,2,2,2,2)
s_{17}	(2,1,3,3,2,2,2,2)
s_{18}	(2,1,2,3,2,1,2,3)
s_{19}	(3,1,3,2,3,3,3,3)

S	ВВУ	…
s_{20}	(2,1,2,3,2,1,2,2)	…
s_{21}	(2,1,3,2,2,2,2,2)	…
s_{22}	(1,3,2,3,1,1,1,1)	…
s_{23}	(1,2,2,3,1,2,2,2)	…
s_{24}	(1,2,3,3,3,3,3,3)	…
s_{25}	(1,2,2,3,3,3,3,3)	…
s_{26}	(1,2,3,2,1,1,1,1)	…
s_{27}	(1,2,2,3,1,1,1,1)	…
s_{28}	(1,2,3,3,1,1,1,1)	…
s_{29}	(1,2,2,3,1,2,2,3)	…

S	ВВУ
s_{448}	(1,4,6,5,1,1,3,2)
s_{449}	(2,6,5,7,2,1,4,3)
s_{450}	(2,5,6,7,2,1,4,3)
s_{451}	(2,5,7,6,2,1,4,3)
s_{452}	(3,6,5,7,3,1,4,2)
s_{453}	(3,5,6,7,3,1,4,2)
s_{454}	(3,5,7,6,3,1,4,2)
s_{455}	(1,6,5,7,1,2,4,3)
s_{456}	(1,5,6,7,1,2,4,3)
s_{457}	(1,5,7,6,1,2,4,3)

В результате анализа таблицы переходов выявлен ряд закономерностей.

1. $\forall\, i \in [0;457],\ j \in [1;12]\ f_s(x_j, s_i) \neq s_0$ — для любого состояния переход в состояние s_0 не возможен.

2. $\forall\, i \in [1;457]\ \exists\, j \in [1;12]\ f_s(x_j, s_i) = s_i$ — в каждое состояние существует переход из этого же состояния, за исключением s_0.

3. $\forall\, i \in [0;457]\ f_s(x_2, s_i) = s_2$ — для любого состояния под воздействием ФОп x_2 происходит переход в состояния s_2.

4. $\forall\, i \in [0;457]\ f_s(x_2, s_i) = s_1 \vee s_{19}$ — для любого состояния под воздействием ФОп x_1 происходит переход в состояние s_1 или s_{19}.

5. $\forall\, i \in [0;457]\ f_s(x_5, s_i) = s_5 \vee s_{42}$ — для любого состояния под воздействием ФОп x_5 происходит переход в состояние s_5 или s_{42}.

6. $\forall\, i \in [0;457]\ f_s(x_7, s_i) = s_7 \vee s_{13} \vee s_{24}$ — для любого состояния под воздействием ФОп x_7 происходит переход в состояние s_7, s_{13} или s_{24}.

7. Состояния повторяются только по горизонтали для всех ФОп, за исключением x_6 и x_{10}.

Выявленные закономерности позволяют сформулировать ряд выводов, важных для восстановления последовательности ФОп.

1. Состояние s_0 будем считать начальным состоянием. Данное состояние использовалось как исходное для построения таблицы переходов, оно свойственно файлам, которые были только что созданы и над которыми ФОп еще не производились. Переход в состояние s_0 из любого другого состояния не возможен (подтверждается закономерностью № 1).

2. Определить точное количество повторов подряд одной ФОп невозможно (следует из закономерности № 2).

3. Состояния s_1, s_2 и s_{19} также будем считать начальными состояниями, так как данные состояния являются результатом выполнения ФОп копирования x_1, x_2 (следует из закономерностей № 3 и № 4). Файлы, являющиеся результатом копирования, с точки зрения ФС — вновь созданные файлы.

4. Однозначно определяется последняя ФОп в случае, если это была одна из операций: x_1, x_2, x_3, x_4, x_5, x_7, x_8, x_9, x_{11}, x_{12} (следует из закономерности № 7).

Конечный автомат, описываемый множеством ФОп, множеством состояний ВО и таблицей переходов, позволяет смоделировать процесс изменения значений ВО. Например, при начальном состоянии s_0 в результате последова-

тельного воздействия ФОп x_2 (копирование), x_5 (редактирование в пакете Microsoft Office), x_9 (открытие (вкл. А)) будет происходить следующая смена состояний ВО s_0 (1,1,1,1,1,1,1,1) → s_2 (2,1,2,2,2,2,2,2) → s_5 (1,2,2,2,1,2,2,2) → s_{36} (1,2,3,2,1,2,2,2) (рис. 3.6):

- $f_s(x_2, s_0) = s_2$;
- $f_s(x_5, s_2) = s_5$;
- $f_s(x_9, s_5) = s_{36}$.

x(t)	S(t)															
	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	…	s_{457}
x_1	s_1	s_1	s_{19}	s_1	s_1	s_1	s_{19}	s_{19}	s_{19}	s_1	s_{19}	s_{19}	s_{19}	s_{19}	…	s_{19}
x_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	…	s_2
x_3	s_3	s_3	s_3	s_3	s_3	s_{14}	s_3	s_{40}	s_{40}	s_3	s_3	s_{53}	s_{57}	s_{40}	…	s_{348}
x_4	s_4	s_4	s_4	s_{22}	s_4	s_{15}	s_4	s_{41}	s_{49}	s_{22}	s_{22}	s_{54}	s_{58}	s_{41}	…	s_{431}
x_5	s_5	s_5	s_5	s_5	s_5	s_5	s_5	s_{42}	s_{42}	s_5	s_5	s_5	s_5	s_{42}	…	s_5
x_6	s_6	s_{12}	s_{20}	s_{23}	s_{30}	s_{23}	s_{38}	s_{43}	s_{38}	s_{51}	s_{43}	s_{38}	s_{59}	s_{62}	…	s_{211}
x_7	s_7	s_{13}	s_{13}	s_{24}	s_{24}	s_{24}	s_7	s_7	s_7	s_7	s_7	s_7	s_{13}	s_{13}	…	s_{24}
x_8	s_8	s_{14}	s_{14}	s_{25}	s_{31}	s_{25}	s_8	s_{44}	s_8	s_{44}	s_{44}	s_8	s_{14}	s_{63}	…	s_{90}
x_9	s_9	s_{15}	s_{21}	s_{26}	s_{32}	s_{36}	s_{39}	s_{45}	s_{45}	s_9	s_{39}	s_{55}	s_{60}	s_{64}	…	s_{457}
x_{10}	s_6	s_{16}	s_{16}	s_{27}	s_{32}	s_{23}	s_6	s_{46}	s_{50}	s_{52}	s_{52}	s_{38}	s_{12}	s_{65}	…	s_{456}
x_{11}	s_{10}	s_{17}	s_{17}	s_{28}	s_{28}	s_{37}	s_{10}	s_{47}	s_{47}	s_{10}	s_{10}	s_{56}	s_{61}	s_{66}	…	s_{434}
x_{12}	s_{11}	s_{18}	s_{18}	s_{29}	s_{33}	s_{29}	s_{11}	s_{48}	s_{11}	s_{48}	s_{48}	s_{11}	s_{18}	s_{67}	…	s_{94}

Рис. 3.6. Моделирование изменения значений ВО

Следует отметить, что предложенная модель изменения значений ВО (рис. 3.1) может быть использована по отношению к другим ОС и ВО. Для этого в исследуемых ОС необходимо провести эксперименты по алгоритму, описанному в п. 2.2 диссертации, в результате которых будут определены правила изменения значений ВО при совершении ФОп. Затем на основе полученных правил по алгоритму, предложенному в данном параграфе, сформировать таблицу переходов и ВВУ.

3.2 Экспериментальная оценка адекватности модели изменения значений ВО

Для оценки адекватности предложенной модели изменения значений ВО реально происходящим процессам времяобразования в ОС Windows была проведена проверка ВВУ файлов реально эксплуатируемых ПЭВМ. Для этого были произвольно выбраны 20 компьютеров с различной аппаратной конфигу-

рацией под управлением ОС Windows XP, 7 и 10, на которых достаточно интенсивно обрабатывалась компьютерная информация (табл. 3.4).

Таблица 3.4
Параметры ОС исследуемых ПЭВМ

№ ПЭВМ	Наименование ОС	Версия ОС	Номер сборки	Опция обновления ВО SIA
1,10,11	Windows 7 Professional	6.1	7601	выкл.
2	Windows 10 Professional	6.3	19043	вкл.
3	Windows XP Professional	5.1	2600	вкл.
4	Windows 10 Home Single Language	6.3	19042	вкл.
5	Windows 10 Education	6.3	18363	выкл.
6-9	Windows 10 Home	6.3	19042	вкл.
12-14	Windows 10 Education	6.3	19042	вкл.
15	Windows 10 Professional	6.3	19041	вкл.
16-18	Windows 10 Professional	6.3	19042	вкл.
19	Microsoft Windows 7 Ultimate	6.1	7600	выкл.
20	Windows XP Home	5.1	2600	вкл.

Из ФС проверяемых компьютеров с помощью программы FTA были извлечены ВО файлов, форматы которых были исследованы в п. 2.2 диссертации. Из извлеченных ВО были сформированы ВВУ (экспериментально полученные ВВУ), которые затем были сравнены с теоретически рассчитанными ВВУ, т.е. полученными в результате математического моделирования (см. приложение В).

В результате сравнения выявлены как известные, так и неизвестные ВВУ. Количество известных ВВУ и их процент по отношению ко всем экспериментально полученным ВВУ для исследуемых форматов файлов приведены в таблице приложения Г, фрагмент которой представлен в табл. 3.5.

Таблица 3.5
Фрагмент таблицы с подсчетом количества и процента известных ВВУ для исследуемых ПЭВМ

Форматы файлов	ПЭВМ № 1			ПЭВМ № 2			ПЭВМ № 3		
	Количество экспериментально полученных ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученных ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученных ВВУ	Количество известных ВВУ	Процент известных ВВУ
Документы (txt, odt, pdf, rtf, djvu, djv, dot, doc, docx, xls, xlsx, ppt, pptx)	2 855	2 838	**99,40 %**	4 605	4 581	**99,47 %**	2 606	2 512	**96,39 %**
Исполняемые файлы (exe, dll, com, sys)	88	88	**100 %**	3 833	3 821	**99,68 %**	1 575	1 572	**99,81 %**
Изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff)	2 393	2 393	**100 %**	40 786	40 616	**99,58 %**	16 787	16 708	**99,53 %**
Видеофайлы (mov, avi, mpeg, mpg, mkv, mp4)	3	3	**100 %**	736	733	**99,59 %**	46	46	**100 %**
Звуковые файлы (mp3, wav)	30	30	**100 %**	1 193	1 180	**98.91 %**	768	766	**99,73 %**
Архивы (zip, rar, 7z)	105	103	**98,09 %**	371	368	**99,19 %**	145	142	**97,93 %**
Интернет-файлы (url, eml)	4	4	**100 %**	10	10	**100 %**	36	36	**100 %**
Итого:	5 478	5 459	**99,65 %**	51 534	51 309	**99,56 %**	21 963	21 782	**99.18 %**

Данный результат позволяет сделать обоснованный вывод об адекватности разработанной модели, экспериментально полученные ВВУ совпадают с теоретически рассчитанными в более чем 99 % случаев. Небольшой объем выборки файлов компенсируется разными техническими характеристиками и программным оснащением исследуемых ПЭВМ.

Далее был проведен ручной анализ выявленных неизвестных ранее ВВУ, позволивший выделить следующие типичные причины, по которым экспериментально полученные ВВУ не совпадают с теоретически рассчитанными:

1. Неправильно установленное системное время. На одной из исследуемых ПЭВМ системное время отставало на 1 час от реального времени, вследствие чего при копировании файлов с других носителей на исследуемую ПЭВМ время редактирования устанавливалось позже, чем время создания. ВВУ скопированных на данную ПЭВМ файлов выглядели следующим образом: (1,2,1,1,1,1,1,1), тогда как на ПЭВМ с правильно установленным системным временем ВВУ скопированных файлов равен (2,1,2,2,2,2,2,2).

2. Обработка файлов специализированными приложениями, которые не были исследованы в рамках данной работы.

3. Появление новых закономерностей изменений ВО в обновленных версиях (сборках) ОС Windows. На одной из исследуемых ПЭВМ была установлена ОС Windows 10 (сборка 19043). В данной сборки при перемещении файлов с внешних носителей на ПЭВМ обновляются ВО SIA = FNC = FNM = FNA = FNX, тогда как в предыдущих сборках обновляются только ВО SIA = SIX = FNC = FNM = FNA = FNX.

Описанные проблемные случаи № 2 и № 3 были учтены путем автоматизированного проведения экспериментов по алгоритму, приведенному в главе 2. В результате по полученным дополнительным закономерностям была создана новая таблица переходов размерностью 13×474, где множество X состоит из 13 групп ФОп ($|X| = 13$), множество S из 474 состояний ВО ($|S| = 474$).

Проблемный случай № 1 выходит за границы исследования, так как в рамках данной работы не рассматриваются ситуации, когда файл перемещается между ОС с некорректно установленным системным временем.

3.3 Разработка методики восстановления последовательности ФОп по внешним ВО

Модель изменения значений внешних ВО, разработанная в п. 3.1 диссертации, наглядно демонстрирует, к каким состояниям ВО могут приводить различные варианты последовательностей ФОп. На рис. 3.7 изображен пример моделирования процесса изменения значений ВО, осуществляемый при помощи таблицы переходов состояний ВО. В примере последовательное выполнение ФОп $x_6 \rightarrow x_8 \rightarrow x_{10}$ над файлом с изначальным состоянием ВО s_0 привело к конечному состоянию ВО s_{46}: $s_0\,(x_6) \rightarrow s_6$, $s_6\,(x_8) \rightarrow s_8$, $s_8\,(x_{10}) \rightarrow s_{50}$.

x(t)	S(t)															
	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	…	s_{457}
x_1	s_1	s_1	s_{19}	s_1	s_1	s_1	s_{19}	s_{19}	s_{19}	s_1	s_{19}	s_{19}	s_{19}	s_{19}	…	s_{19}
x_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	…	s_2
x_3	s_3	s_3	s_3	s_3	s_3	s_{34}	s_3	s_{40}	s_{40}	s_3	s_3	s_{53}	s_{57}	s_{40}	…	s_{348}
x_4	s_4	s_4	s_4	s_{22}	s_4	s_{35}	s_4	s_{41}	s_{49}	s_{22}	s_{22}	s_{54}	s_{58}	s_{41}	…	s_{431}
x_5	s_5	s_5	s_5	s_5	s_5	s_5	s_5	s_{42}	s_{42}	s_5	s_5	s_5	s_5	s_{42}	…	s_5
x_6	s_6	s_{12}	s_{20}	s_{23}	s_{30}	s_{23}	s_{38}	s_{43}	s_{38}	s_{51}	s_{43}	s_{38}	s_{59}	s_{62}	…	s_{211}
x_7	s_7	s_{13}	s_{13}	s_{24}	s_{24}	s_{24}	s_7	s_7	s_7	s_7	s_7	s_7	s_{13}	s_{13}	…	s_{24}
x_8	s_8	s_{14}	s_{14}	s_{25}	s_{31}	s_{25}	s_8	s_{44}	s_8	s_{44}	s_{44}	s_8	s_{14}	s_{63}	…	s_{90}
x_9	s_9	s_{15}	s_{21}	s_{26}	s_{26}	s_{36}	s_{39}	s_{45}	s_{45}	s_9	s_{39}	s_{55}	s_{60}	s_{64}	…	s_{457}
x_{10}	s_{10}	s_{16}	s_{16}	s_{27}	s_{32}	s_{23}	s_{40}	s_{46}	s_{50}	s_{52}	s_{52}	s_{38}	s_{12}	s_{65}	…	s_{456}
x_{11}	s_{10}	s_{17}	s_{17}	s_{28}	s_{28}	s_{37}	s_{10}	s_{47}	s_{47}	s_{10}	s_{10}	s_{56}	s_{61}	s_{66}	…	s_{434}
x_{12}	s_{11}	s_{18}	s_{18}	s_{29}	s_{33}	s_{29}	s_{11}	s_{48}	s_{11}	s_{48}	s_{48}	s_{11}	s_{18}	s_{67}	…	s_{94}

Рис. 3.7. Моделирование процесса изменения значений ВО при выполнении последовательности ФОп $x_6 \rightarrow x_8 \rightarrow x_{10}$

В ходе проведения КТЭ эксперт может получить только конечные состояния ВО файлов. Поэтому методика восстановления ФОп заключается в определении конечного состояния (известного эксперту), а затем поиске ФОп и состояний, которые могли привести к имеющемуся конечному состоянию ВО по таблице переходов (рис. 3.8). При этом некоторые состояния могут быть обнаружены в таблице переходов на пересечении нескольких ФОп и состояний ВО, что приведет к появлению различных вариантов последовательностей ФОп.

x(t)	S(t)															
	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	…	s_{457}
x_1	s_1	s_1	s_{19}	s_1	s_1	s_1	s_{19}	s_{19}	s_{19}	s_1	s_{19}	s_{19}	s_{19}	s_{19}	…	s_{19}
x_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	s_2	…	s_2
x_3	s_3	s_3	s_3	s_3	s_3	s_{34}	s_3	s_{40}	s_{40}	s_3	s_3	s_{53}	s_{57}	s_{40}	…	s_{348}
x_4	s_4	s_4	s_4	s_{22}	s_4	s_{35}	s_4	s_{41}	s_{49}	s_{22}	s_{22}	s_{54}	s_{58}	s_{41}	…	s_{431}
x_5	s_5	s_5	s_5	s_5	s_5	s_5	s_5	s_{42}	s_{42}	s_5	s_5	s_5	s_5	s_{42}	…	s_5
x_6	s_6	s_{12}	s_{20}	s_{23}	s_{30}	s_{23}	s_{38}	s_{43}	s_{38}	s_{51}	s_{43}	s_{38}	s_{59}	s_{62}	…	s_{211}
x_7	s_7	s_{13}	s_{13}	s_{24}	s_{24}	s_{24}	s_7	s_7	s_7	s_7	s_7	s_7	s_{13}	s_{13}	…	s_{24}
x_8	s_8	s_{14}	s_{14}	s_{25}	s_{31}	s_{25}	s_8	s_{44}	s_8	s_{44}	s_{44}	s_8	s_{14}	s_{63}	…	s_{90}
x_9	s_9	s_{15}	s_{21}	s_{26}	s_{26}	s_{36}	s_{39}	s_{45}	s_{45}	s_9	s_{39}	s_{55}	s_{60}	s_{64}	…	s_{457}
x_{10}	s_6	s_{16}	s_{16}	s_{27}	s_{32}	s_{23}	s_6	s_{46}	s_{50}	s_{52}	s_{52}	s_{38}	s_{12}	s_{65}	…	s_{456}
x_{11}	s_{10}	s_{17}	s_{17}	s_{28}	s_{28}	s_{37}	s_{10}	s_{47}	s_{47}	s_{10}	s_{10}	s_{56}	s_{61}	s_{66}	…	s_{434}
x_{12}	s_{11}	s_{18}	s_{18}	s_{29}	s_{33}	s_{29}	s_{11}	s_{48}	s_{11}	s_{48}	s_{48}	s_{11}	s_{18}	s_{67}	…	s_{94}

Рис. 3.8. Определение ФОп и состояния ВО, на пересечении которых находится конечное состояние ВО s_{50}

Задачу поиска промежуточных состояний можно решить, используя теорию графов в два этапа. Для этого на первом этапе по таблице переходов необходимо построить граф переходов [73], где вершины — состояния s_i, а ребра — операции x_j, производимые над файлом, петли свидетельствуют об операциях, возможно, производившихся над файлом неопределенное количество раз. На втором этапе в построенном орграфе переходов произвести полный перебор всех возможных маршрутов между конечной вершиной s_i, представляющей конечное состояние ВО, и начальной вершиной s_0, соответствующей начальному состоянию ВО.

Построенный на первом этапе граф является однонаправленным (орграф) и наглядно представляет переходы между состояниями s_i. Так, переходу к состоянию s_{50}, представленному на рис. 3.8, в графическом виде соответствует орграф, приведенный на рис. 3.9.

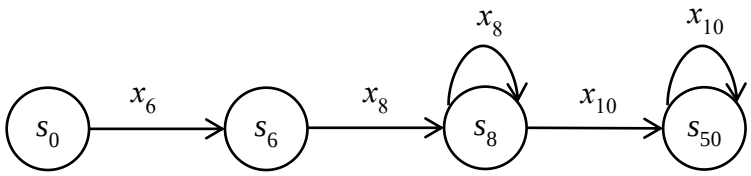


Рис. 3.9. Орграф перехода из состояния s_0 в состояние s_{50}

Исчерпывающий орграф переходов может быть построен по полной таблице переходов в программной среде MATLAB с применением функции digraph [76]. В качестве входных аргументов функции необходимо использо-

вать матрицу смежности, сгенерированную на основе таблицы переходов. Матрица смежности графа G с конечным числом вершин n — это квадратная матрица размера $n \times n$, в которой значение элемента a_{ij} равно числу ребер из i-й вершины графа в j-ю вершину. Если ребер между вершинами i и j не существует, то элемент a_{ij} равен нулю. При построении матрицы смежности необходимо учесть следующие особенности таблицы переходов:

− Состояния s_1, s_2 и s_{19} были определены как начальные в п. 3.1 диссертации, поэтому в матрице смежности переходы к состояниям s_1, s_2, s_{19} из множества других состояний необходимо исключить путем обнуления соответствующих элементов, оставив лишь переходы из s_0 в s_1, s_2 и из s_2 в s_{19}.

− Состояния s_5 и s_{42} являются результатом выполнения операции редактирования файла в пакете Microsoft Office (x_5). В состояние s_5 возможен переход из 394 состояний под воздействием x_5, в состояние s_{42} возможен переход из 64 состояний под воздействием x_5. Так как при совершении этой операции одновременно обновляются 6 ВО, и затираются предыдущие значения ВО, то будет определено, что до операции x_5 могли произойти любые другие ФОп. Поэтому целесообразно, для минимизации количества ребер в графе, исключить в матрице смежности переходы к состояниям s_5, s_{42} из множества других состояний, оставив переходы из s_0 в s_5 и из s_7 в s_{42}.

В результате сформированный орграф (рис. 3.10) будет иметь 458 вершин и 3654 ребра.

Рис. 3.10. Орграф переходов, построенный по полной таблице переходов

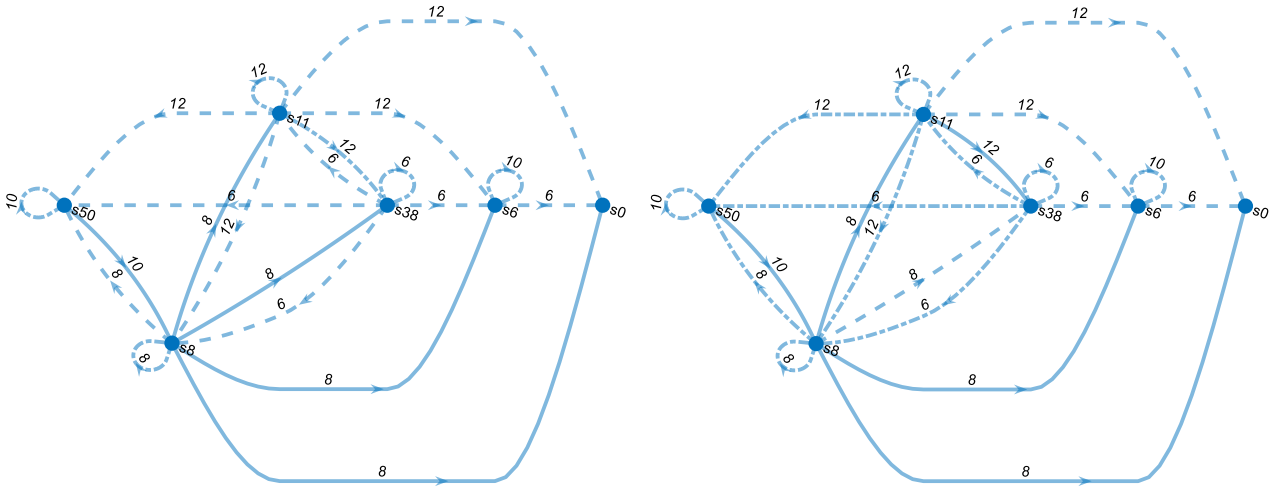
На втором этапе для полного перебора всех возможных маршрутов можно применять широко распространенные алгоритмы обхода графов: поиск в ширину и глубину [77]. Обход графа — это переход от одной его вершины к другой в поисках связей этих вершин, где в качестве связей используются ребра графа. В процессе обхода вершины могут находиться в трех состояниях: непросмотренные, просмотренные, использованные. Изначально все вершины имеют статус непросмотренных.

Алгоритм поиска в ширину (breadth-first search, BFS) подразумевает поуровневое исследование графа. Для описания поиска в ширину вводится очередь Q для хранения вершин. Поиск начинается с некоторой начальной вершины u. Эта вершина помещается в очередь Q и с этого момента считается *просмотренной*. Смежные с u вершины u_1, u_2, …, u_p, помещаются в очередь и получают статус *просмотренных*, а вершина u удаляется из очереди и получает статус *использованной*. Далее смежные с u_1, u_2, …, u_p вершины

помещаются в очередь и получают статус *просмотренных*, а вершины u_1, u_2, …, u_p удаляются из очереди и получают статус *использованных*. Вершины просматриваются в порядке возрастания их расстояния от начальной вершины. В тот момент, когда очередь Q окажется пустой, поиск в ширину обойдет все вершины графа.

При поиске в глубину (depth-first search, DFS) производится перечисление вершин «вглубь», пока это возможно. Поиск начинается с некоторой начальной вершины u и с этого момента, она считается *просмотренной*. Если среди вершин, смежных с u, существует еще непросмотренная вершина w, тогда w объявляется *просмотренной*, и поиск продолжается из этой вершины. Если все вершины, смежные с u, просмотрены, тогда u объявляется *использованной* вершиной. Если в графе не осталось непросмотренных вершин, то поиск заканчивается. Если же осталась непросмотренная вершина, то поиск продолжается из нее.

На рис. 3.11 представлен процесс обхода орграфа (рис. 3.10) алгоритмами поиска в ширину (а) и в глубину (б) из вершины s_{50}. Данные алгоритмы были реализованы в MATLAB с помощью функций bfsearch [78] и dfsearch [79] соответственно.



а) б)

Рис. 3.11. Процесс обхода орграфа алгоритмами поиска в ширину и в глубину

На рис. 3.11 сплошной линией выделены ребра, которые ведут к непросмотренным вершинам; штрихпунктирной линией — ребра, которые ведут

к просмотренным вершинам; штриховой линией — ребра, которые ведут к использованным вершинам. Пошаговый процесс обхода в ширину и глубину графа представлен в таблицах 3.6 и 3.7 соответственно.

Таблица 3.6
Пошаговый процесс обхода в ширину

№ шага	Вершины, находящиеся в очереди Q	Статус вершины	Смежные вершины	Статус смежной вершины	Ребро, соединяющее вершины из очереди и смежную	Тип линии на рис. 3.11, а
1.	s_{50}	просмотренная				
2.	s_{50}		s_8	непросмотренная	$s_{50}s_8$	сплошная
3.	s_{50}	использованная				
4.	s_8	просмотренная				
5.			s_0	непросмотренная	s_8s_0	сплошная
6.			s_6	непросмотренная	s_8s_6	сплошная
7.			s_{11}	непросмотренная	s_8s_{11}	сплошная
8.			s_{38}	непросмотренная	s_8s_{38}	сплошная
9.			s_{50}	использованная	s_8s_{50}	штриховая
10.	s_8	использованная				
11.	s_0	просмотренная				
12.	s_6	просмотренная				
13.	s_{11}	просмотренная				
14.	s_{38}	просмотренная				
15.	s_0		—			
16.	s_0	использованная				
17.	s_6		s_0	использованная	s_6s_0	штриховая
18.	s_6	использованная				
19.	s_{11}		s_0	использованная	$s_{11}s_0$	штриховая
20.			s_6	использованная	$s_{11}s_6$	штриховая
21.			s_8	использованная	$s_{11}s_8$	штриховая
22.			s_{38}	просмотренная	$s_{11}s_{38}$	штрихпунктирная
23.			s_{50}	использованная	$s_{11}s_{50}$	штриховая
24.	s_{11}	использованная				
25.	s_{38}		s_6	использованная	$s_{38}s_6$	штриховая
26.			s_8	использованная	$s_{38}s_8$	штриховая
27.			s_{11}	использованная	$s_{38}s_{11}$	штриховая
28.			s_{50}	использованная	$s_{38}s_{50}$	штриховая
29.	s_{38}	использованная				

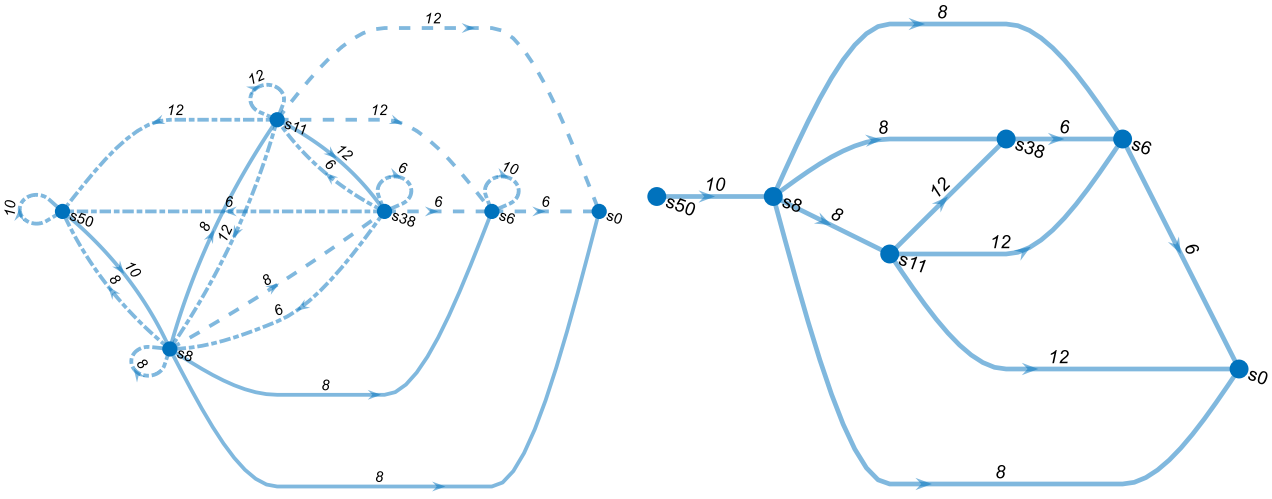
Таблица 3.7
Пошаговый процесс обхода в глубину

№ шага	Рассматриваемые вершины	Статус вершины	Смежные вершины	Статус смежной вершины	Ребро, соединяющее вершины из очереди и смежную	Тип линии на рис. 3.11, б
1.	s_{50}	просмотренная				
2.	s_{50}		s_8	непросмотренная	$s_{50}s_8$	сплошная
3.	s_8	просмотренная				
4.	s_8		s_0	непросмотренная	s_8s_0	сплошная
5.	s_0	просмотренная				
6.	s_0		—			
7.	s_0	использованная				
8.	s_8		s_6	непросмотренная	s_8s_6	сплошная
9.	s_6	просмотренная				
10.	s_6		s_0	использованная	s_6s_0	штриховая
11.	s_6	использованная				
12.	s_8		s_{11}	непросмотренная	s_8s_{11}	сплошная
13.	s_{11}	просмотренная				
14.	s_{11}		s_0	использованная	$s_{11}s_0$	штриховая
15.			s_6	использованная	$s_{11}s_6$	штриховая
16.			s_8	просмотренная	$s_{11}s_8$	штрихпунктирная
17.			s_{38}	непросмотренная	$s_{11}s_{38}$	сплошная
18.	s_{38}	просмотренная				
19.	s_{38}		s_6	использованная	$s_{38}s_6$	штриховая
20.			s_8	просмотренная	$s_{38}s_8$	штрихпунктирная
21.			s_{11}	просмотренная	$s_{38}s_{11}$	штрихпунктирная
22.			s_{50}	просмотренная	$s_{38}s_{50}$	штрихпунктирная
23.	s_{38}	использованная				
24.	s_{11}		s_{50}	просмотренная	$s_{11}s_{50}$	штрихпунктирная
25.	s_{11}	использованная				
26.	s_8		s_{38}	использованная	s_8s_{38}	штриховая
27.			s_{50}	просмотренная	s_8s_{50}	штрихпунктирная
28.	s_8	использованная				
29.	s_{50}	использованная				

Оба алгоритма приводят к идентичному результату, имеют одинаковую трудоемкость и позволяют обнаруживать все маршруты, соединяющие две вершины: начальную и конечную. При этом поиск в ширину зачастую используется для нахождения кратчайшего маршрута между точками. Поиск

в глубину применяется для определения любого возможного маршрута между двумя вершинами и обнаружения циклов на графе.

Следует обратить внимание, что специфика орграфа перехода между состояниями ВО такова, что оба алгоритма приводят к появлению циклов. С точки зрения восстановления последовательности ФОп циклы и петли интерпретируются как повторение некоторых ФОп неопределенное количество раз. С одной стороны для полноты картины эту информацию надо учитывать. С другой стороны, в результате проведения КТЭ следствие интересует в первую очередь последовательность и факт выполнения ФОп, а не количество повторения однотипных операций. Поэтому, чтобы избежать избыточности информации циклы и петли целесообразно исключать, что возможно при применении поиска в глубину. Для этого исключаются ребра, которые ведут к просмотренным вершинам (штрихпунктирная линия). На рис. 3.12, б изображен орграф процесса обхода в глубину без циклов и петель.



а) б)

Рис. 3.12. Процесс обхода орграфа алгоритмом поиска в глубину с циклами и петлями (а) и без циклов и петель (б)

Орграф, построенный на рис. 3.12, б наглядно демонстрирует, какие ФОп могли быть совершены над исследуемым файлом. При этом последовательность ФОп восстанавливается от последней выполненной над файлом операции к первой. Между конечной и начальной вершинами могут быть обнаружены несколько маршрутов, что приведет к выявлению различных вариантов

последовательностей ФОп. Так, для примера, приведенного на рис. 3.12, б между вершинами, соответствующими состояниям ВО s_{50} и s_0, возможны следующие варианты маршрутов:

$$
\begin{aligned}
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_6,\ s_6(x_6) \rightarrow s_0.\\
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_{38},\ s_{38}(x_6) \rightarrow s_6,\ s_6(x_6) \rightarrow s_0.\\
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_{11},\ s_{11}(x_{12}) \rightarrow s_{38},\ s_{38}(x_6) \rightarrow s_6,\ s_6(x_6) \rightarrow s_0.\\
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_{11},\ s_{11}(x_{12}) \rightarrow s_6,\ s_6(x_6) \rightarrow s_0.\\
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_{11},\ s_{11}(x_{12}) \rightarrow s_0.\\
&s_{50}(x_{10}) \rightarrow s_8,\ s_8(x_8) \rightarrow s_0.
\end{aligned}
\tag{3.4}
$$

В (3.4) все варианты маршрутов начинаются с операций x_{10} и x_8. Таким образом, для состояния ВО s_{50} однозначно оказываются определены последняя ФОп (x_{10}) и предпоследняя ФОп (x_8).

Автоматическое описание всех маршрутов может быть реализовано в MATLAB с помощью функции allpaths [80], где в качестве входных параметров используется G — граф, s — начальная вершина, t — конечная вершина. Результат ее выполнения для состояния s_{50} (рис. 3.12, б) представлен на рис. 3.13

	1	2	3	4	5	6
1	's50'	's8'	's0'	[]	[]	[]
2	's50'	's8'	's6'	's0'	[]	[]
3	's50'	's8'	's11'	's0'	[]	[]
4	's50'	's8'	's11'	's6'	's0'	[]
5	's50'	's8'	's11'	's38'	's6'	's0'
6	's50'	's8'	's38'	's6'	's0'	[]

Рис. 3.13. Результат использования функции allpaths для состояния s_{50}

На рис. 3.14 представлен пример процесса обхода орграфа из вершины, соответствующей состоянию ВО s_{32}.

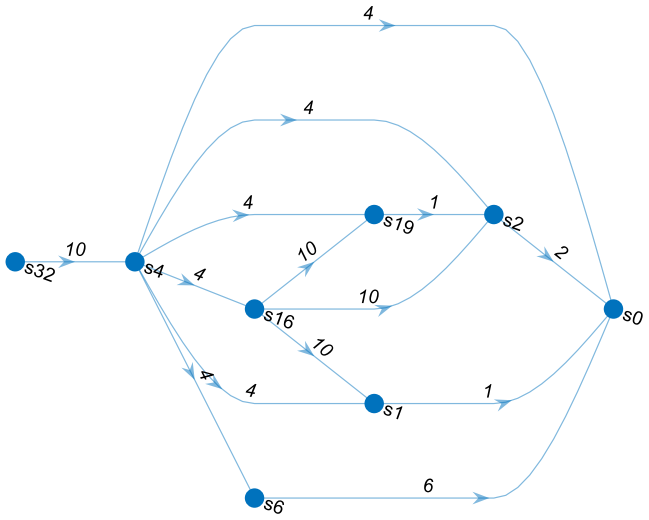
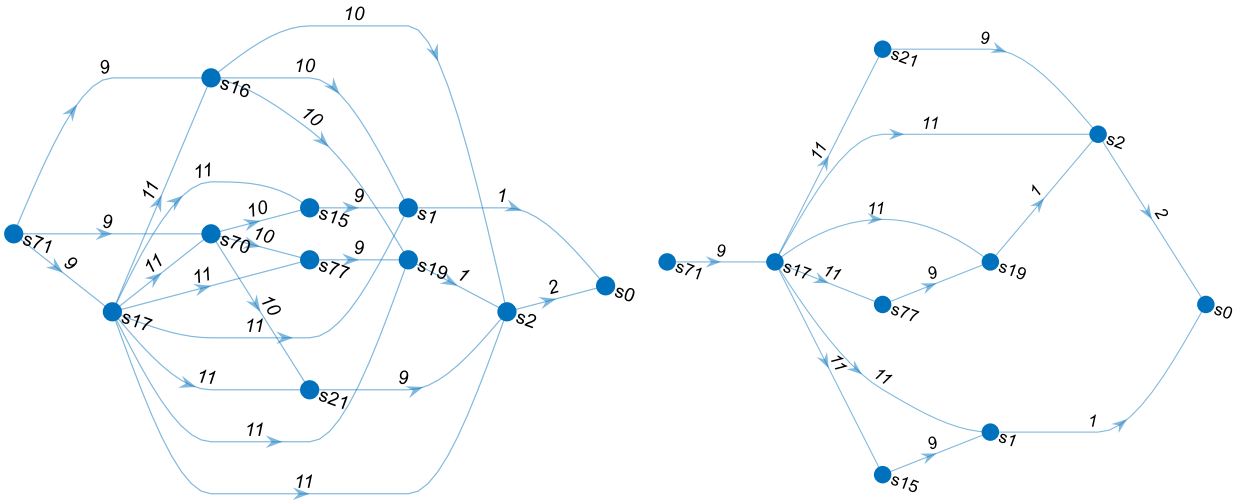


Рис. 3.14. Процесс обхода орграфа алгоритмом поиска в глубину из вершины, соответствующей состоянию ВО s_{32}

Пример на рис. 3.14 демонстрируют однозначное определение последней и предпоследней ФОп, что очевидно следует из особенностей таблицы переходов: состояния ВО повторяются только по горизонтали для всех ФОп, за исключением x_6 и x_{10}.

Для уменьшения количества вариантов возможных маршрутов можно произвести их постобработку путем исключения некоторых ребер и вершин по имеющимся дополнительным исходным условиям. Например, если известно, что все операции производились в ОС Windows 7, тогда можно исключить ребра, соответствующие ФОп, производимым в других версиях ОС Windows. В примере на рис. 3.15, а можно исключить все ребра, соответствующие ФОп x_{10}, в результате количество маршрутов уменьшится на 6 (рис. 3.15, б).



а) б)

Рис. 3.15. Процесс обхода орграфа алгоритмом поиска в глубину из вершины s_{71} без постобработки (а) и с постобработкой (б)

Таким образом, согласно табл. 3.1 в качестве исходных условий, которые позволят производить постобработку графа, могут быть версия ОС (Windows XP, 7, 8 или 10) и состояние опции обновления ВО SIA (вкл. или выкл.).

Для определения времени совершения выявленных операций необходимо ребрам графа поставить в соответствие значение ВО. При этом ребер значительно больше чем значений ВО в ВВУ, так как ВО перезаписывают предыдущие значения ВО при совершении ФОп. Для наглядности разберем пример определения времени совершения ФОп. При состоянии ВО s_{32} возможны следующие варианты маршрутов:

$$
\begin{aligned}
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_2,\ s_2(x_2) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_1,\ s_1(x_1) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_6,\ s_6(x_6) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_{16},\ s_{16}(x_{10}) \rightarrow s_2,\ s_2(x_2) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_{16},\ s_{16}(x_{10}) \rightarrow s_1,\ s_1(x_1) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_{19},\ s_{19}(x_1) \rightarrow s_2,\ s_2(x_2) \rightarrow s_0.\\
&s_{32}(x_{10}) \rightarrow s_4,\ s_4(x_4) \rightarrow s_{16},\ s_{16}(x_{10}) \rightarrow s_{19},\ s_{19}(x_1) \rightarrow s_2,\ s_2(x_2) \rightarrow s_0.
\end{aligned}
\qquad (3.5)
$$

Первый маршрут из (3.5) описывает последовательное выполнение ФОп (маршрут читается справа налево): x_4 (редактирование) и x_{10} (изменение атрибутов), что приводит к следующим сменам состояний ВО:

$$s_0\,(1,1,1,1,1,1,1,1) \xrightarrow{x4} s_4\,(1,2,1,2,1,1,1,1) \xrightarrow{x10} s_{32}\,(1,2,1,3,1,1,1,1)$$

Значение ВО, соответствующее временному уровню «3» в s_{32} — время изменения атрибутов файла (x_{10}), значение ВО, соответствующее временному уровню «2» в s_{32} — время редактирования файла (x_4), значение ВО, соответствующее временному уровню «1» в s_{32} — время создания файловой записи.

Второй маршрут из (3.5) описывает последовательное выполнение ФОп: x_2 (копирование (новый объект)), x_4 (редактирование) и x_{10} (изменение атрибутов), что приводит к следующим сменам состояний ВО:

$$s_0\,(1,1,1,1,1,1,1,1) \xrightarrow{x2} s_2\,(2,1,2,2,2,2,2,2) \xrightarrow{x4} s_4\,(1,2,1,2,1,1,1,1) \xrightarrow{x10} s_{32}\,(1,2,1,3,1,1,1,1)$$

Аналогично предыдущему варианту (первый маршрут): временной уровень «3» — время изменения атрибутов файла (x_{10}), временной уровень «2» — время редактирования файла (x_4), временной уровень «1» — время создания файловой записи. Только в данном варианте файловая запись была создана методом копирования.

Пятый маршрут из (3.5) описывает последовательное выполнение ФОп: x_2 (копирование (новый объект)), x_{10} (изменение атрибутов), x_4 (редактирование) и x_{10} (изменение атрибутов), что приводит к следующим сменам состояний ВО:

$$s_0\,(1,1,1,1,1,1,1,1) \xrightarrow{x2} s_2\,(2,1,2,2,2,2,2,2) \xrightarrow{x10} s_{16}\,(2,1,2,3,2,2,2,2) \xrightarrow{x4}$$

$$s_4\,(1,2,1,2,1,1,1,1) \xrightarrow{x10} s_{32}\,(1,2,1,3,1,1,1,1)$$

В данном варианте появилась еще одна ФОп x_{10} между x_2 и x_4. Время ее совершения неизвестно, так как ВО ее выполнения (временной уровень «3» в s_{16}) перезаписалась при совершении ФОп x_4.

Цепочка восстановленных ФОп может быть сколь угодно длинной, но время совершения можно установить для первой (время создания файловой записи или время последнего редактирования (для файлов-копий)) и последних $(k-1)$ ФОп, где k — максимальное значение временного уровня в ВВУ

($k = max(s_i)$). При этом ВО, соответствующая временному уровню k — время совершения последней ФОп, ВО, соответствующая временному уровню $(k-1)$ — время совершения предпоследней ФОп, и так далее, а ВО, соответствующая временному уровню 1 — время совершения первой ФОп. Время совершения промежуточных ФОп восстановить не удастся, только факт их возможного совершения.

Таким образом, решение задачи по восстановлению последовательности ФОп сводится к выполнению трех этапов. Первые два этапа производятся всего один раз: 1. сформировать орграф переходов по полной таблице переходов (рис. 3.10), который является универсальным для 12 ФОп в ОС Windows; 2. произвести поиск в глубину и исключить циклы и петли для всех вершин орграфа (состояний ВО).

Данный процесс был запрограммирован в MATLAB, листинг функции Transition_digraph.m представлен в приложении Д. Для имеющейся таблицы переходов построение орграфа и расчет маршрутов для всех вершин занимает 35 минут на ПЭВМ (Intel(R) Core(TM) i3-8130U CPU @ 2,20GHz, 2,21 GHz, ОЗУ 8,00 ГБ). В результате выполнения функции формируется таблица соответствия между состояниями ВО и набором возможных вариантов последовательностей ФОп.

Третий этап заключается в постобработке маршрутов на основе известных исходных условий. Данный этап реализуется в функции Retro.m, описание которой приведено в следующей главе, а листинг представлен в приложении Е.

3.4 Выводы

1. Разработана модель процесса изменения значений ВО файлов на основе теории автоматов. Элементами модели являются множество ФОп, множество ВВУ (состояний ВО) и таблица переходов, в которой задаются правила изменений ВО. Модель позволяет теоретически рассчитывать возможные состояния ВО файла при разном последовательном воздействии ФОп на файл, что может быть положено в основу методики восстановления

последовательности ФОп. В ходе моделирования изменений ВО для 12 групп ФОп в исследуемых ОС Windows XP, 7, 8, 10 выявлено 458 возможных состояний ВО. Предложенная модель может применяться при обновлении ОС Windows, при этом количество возможных состояний ВО возрастет.

2. Подтверждена экспериментально адекватность модели путем сравнения ВВУ файловых записей реально эксплуатируемых ПЭВМ с ВВУ, полученными в результате математического моделирования. Выявлено, что доля «неизвестных» ВВУ, полученных при проведении экспериментов, составляет менее 1 %. Одной из основных причин обнаружения «неизвестных» ВВУ является появление новых закономерностей изменений ВО в обновленных версиях (сборках) ОС Windows. Для устранения отставания модели от современных версий ОС Windows следует проводить эксперименты по алгоритму, приведенному в главе 2, для определения новых закономерностей изменений ВО, по которым формировать обновленные таблицу переходов и множество возможных состояний ВО.

3. Разработана методика восстановления ФОп на основе модели изменений ВО с применением алгоритма поиска в глубину. Методика позволяет однозначно определять порядок следования ФОп и делится на три этапа. В результате выполнения первых двух этапов формируется таблица соответствия между состояниями ВО и набором возможных вариантов последовательностей ФОп. Данный процесс автоматизирован в функции Transition_digraph.m. На третьем этапе происходит исключение вариантов последовательностей ФОп, которые не могли производиться, исходя из наличия дополнительных начальных условий (версия ОС и состояние опции обновления ВО SIA).

4. Применение методики позволяет восстанавливать цепочку ФОп, состоящую из двух и более операций, тогда как сейчас эксперты восстанавливают только последнюю операцию.

4 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ МЕТОДИКИ ВОССТАНОВЛЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ ФОП

4.1 Назначение и описание функции Retro.m

Для практического использования описанной в п. 3.3 методики восстановления хронологий ФОп была разработана функция Retro.m в программной среде MATLAB. Выбор программной среды был обусловлен наличием средств работы с функциями построения и анализа графов. Листинг функции Retro.m представлен в приложении Е.

Функция Retro.m имеет следующие возможности:

1. Извлечение ВО объектов ФС NTFS.

2. Составление ВВУ из извлеченных ВО.

3. Поиск полученных ВВУ объектов среди теоретически рассчитанных.

3.1. Если ВВУ не обнаружено, вывод сообщения о неизвестном ВВУ.

3.2. Если ВВУ обнаружено, определение вариантов последовательностей ФОп, соответствующих данному ВВУ.

4. Исключение вариантов последовательностей ФОп по имеющимся дополнительным исходным условиям (состояние опции обновления ВО последнего доступа и версия ОС).

5. Отображение вариантов последовательностей ФОп для каждого файла в виде таблицы с указанием времени совершения ФОп.

6. Сохранение результатов анализа в текстовом формате, пригодном для дальнейшего импорта в табличный или текстовый редактор.

Функция Retro.m не имеет параметров. При ее запуске появляется диалоговое окно выбора объекта исследования (рис. 4.1). В качестве объекта исследования должна быть представлена главная файловая таблица MFT, которая располагается в файле «*$MFT»,* или ее фрагмент, хранящий одну или несколько файловых записей.

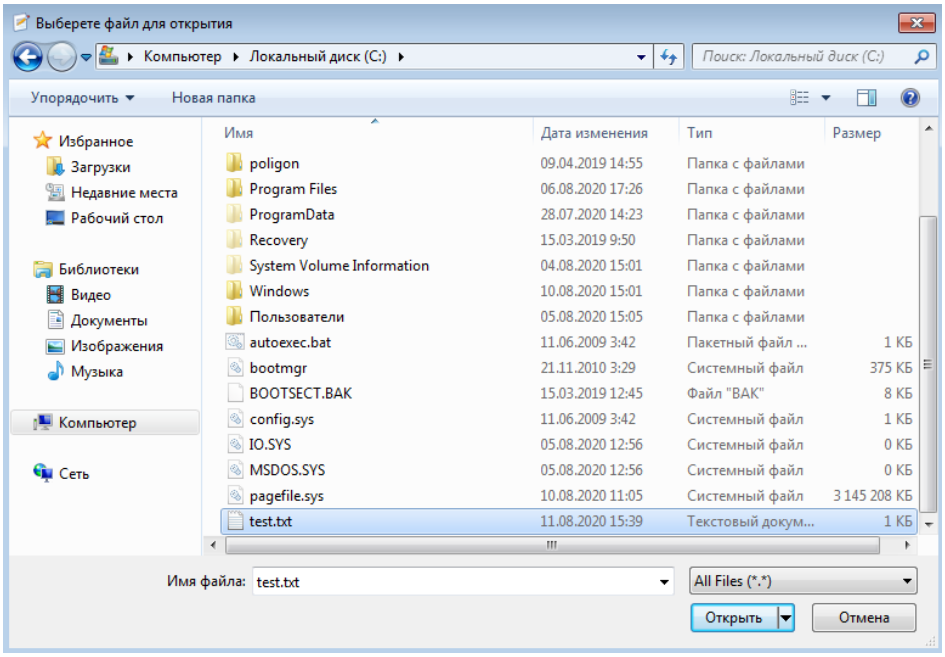


Рис. 4.1. Диалоговое окно выбора объекта исследования

В результате выполнения функции для каждого файла создается таблица, отражающая последовательность ФОп (рис. 4.2).

Имя файла: check_print.pdf	Файловая операция	Временная отметка
Размер файла: 69933 байт	Вариант 1	
ВВУ = (2,1,3,4,2,1,3,1)	файл был перемещен/переименован	13.06.2021 05:15:01
	файл копировался или был разархивирован или был запущен/открыт	28.05.2021 00:53:08
	файл был создан методом копирования	26.05.2021 18:20:37
	исходный файл редактировался	22.05.2021 14:02:07

Рис. 4.2. Результат выполнения функции Retro.m

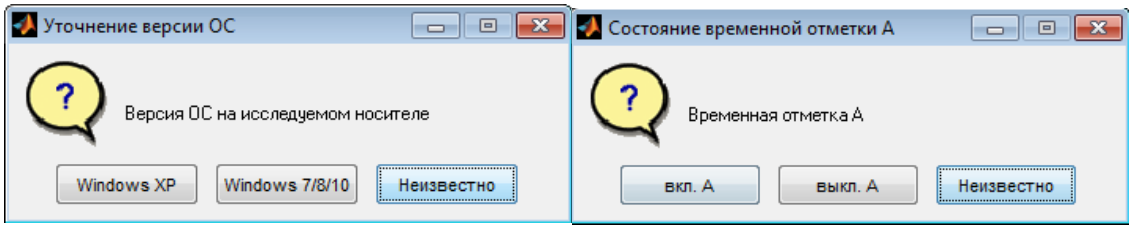
Последовательность ФОп отображается сверху вниз от последней совершенной над файлом операции к первой. Первый столбец содержит имя файла, размер файла и ВВУ. Во втором столбце приводится название ФОп. Третий столбец отображает вычисленное время совершения ФОп. Для ФОп с неизвестным временем их совершения указывается «н/д» (недостаточно данных).

В случае, когда состоянию ВО соответствуют несколько вариантов последовательностей ФОп они будут пронумерованы (рис. 4.3).

Имя файла: письмо.txt	Файловая операция	Временная отметка
Размер файла: 2711 байт	Вариант 1	
ВВУ = (1,2,2,3,3,3,3,3)	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	создана файловая запись	11.06.2021 06:49:52
	Вариант 2	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл был создан методом копирования	11.06.2021 06:49:52
	Вариант 3	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл был создан методом копирования или извлечен из архива	11.06.2021 06:49:52
	Вариант 4	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл был перемещен/переименован	н/д
	создана файловая запись	11.06.2021 06:49:52
	Вариант 5	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл копировался или был разархивирован или был запущен/открыт	н/д
	создана файловая запись	11.06.2021 06:49:52
	Вариант 6	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл запускался/открывался (вкл. А)	н/д
	создана файловая запись	11.06.2021 06:49:52
	Вариант 7	
	файл был извлечен из архива	18.06.2021 08:06:52
	файл редактировался (вкл. А)	13.06.2021 05:44:22
	файл запускался/открывался (вкл. А)	н/д
	файл был перемещен/переименован	н/д
	создана файловая запись	11.06.2021 06:49:52

Рис. 4.3. Результат выполнения функции Retro.m для файла с несколькими вариантами последовательностей ФОп

Функция позволяет уменьшить количество вариантов последовательностей ФОп путем уточнения версии ОС (рис. 4.4, а) и состояния опции обновления ВО последнего доступа в ОС (рис. 4.4, б), в которой хранились файловые записи.



а) б)

Рис. 4.4. Диалоговые окна с уточняющими вопросами

Если у файла какая-либо ВО является округленной по причине копирования/перемещения файла из другой ФС или изменения в специализированных программах, то после перечисления ФОп будет указано «! Округлена ВО …» (рис. 4.5).

Имя файла: img024.pdf	Файловая операция	Временная отметка
Размер файла: 733634 байт	Вариант 1	
ВВУ = (2,1,2,2,2,2,2,2)	файл был создан методом копирования или извлечен из архива	22.05.2021 14:03:55
	исходный файл редактировался	14.04.2021 17:28:10
	! Округлена временная отметка изменения	

Рис. 4.5. Результат выполнения функции Retro.m
для файла с округленной ВО SIM

Если для файла будет обнаружен неизвестный ВВУ, то будет указано «Для файла «…» неизвестный вектор временных уровней» и значение ВВУ (рис. 4.6).

Для файла "RU2.docx" неизвестный вектор временных уровней
(2,1,5,4,2,1,4,3)

Рис. 4.6. Результат выполнения функции Retro.m
для файла с неизвестным ВВУ

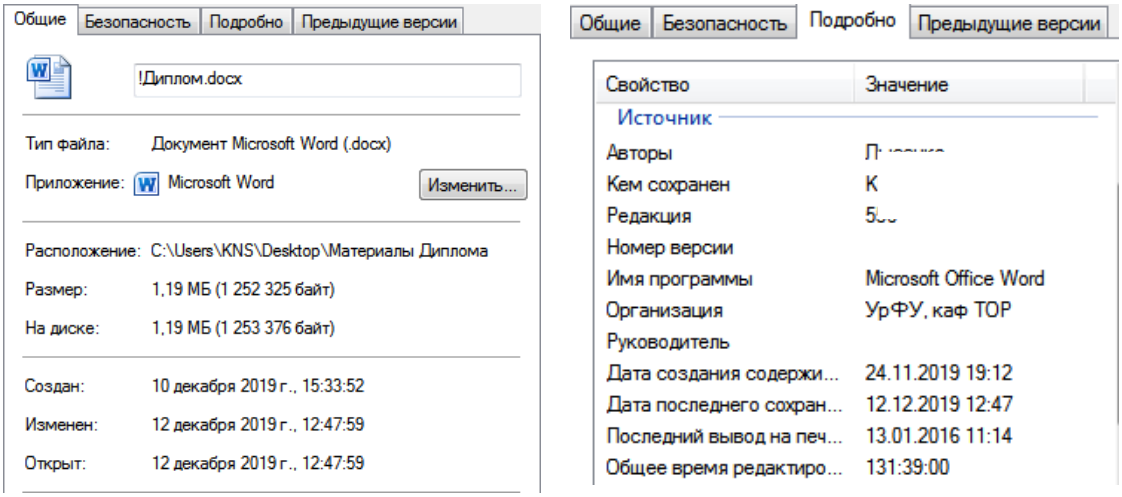
4.2 Рекомендации по использованию внутренних ВО

Несмотря на то, что методика восстановления последовательности ФОп и реализующая ее функция Retro.m в большинстве случаев позволяют экспертам давать однозначные ответы в отношении совершения ФОп над файлом, необходимо отметить, что возникают трудности в определении операции копирования для файлов, являющимися копиями. Операция копирование является сложной в определении, так как при копировании файла у файла-

копии одновременно обновляются 7[1] из 8 внешних ВО. Только ВО изменения наследуется от файла-оригинала, и, следовательно, отстает от ВО создания. Если после копирования файл-копия редактируется, то ВО изменения обновляется, и один из явных признаков файлов-копий — отставание ВО изменения от ВО создания — утрачивается. В таких ситуациях рекомендуется использовать ВО, хранящиеся во внутренней структуре файлов.

В работах [18–20] изложено, как с помощью сравнения ВО, содержащихся в структуре файлов, возможно определить факт копирования файла и факт фальсификации ВО.

Для отображения внутренних ВО можно воспользоваться штатными средствами ОС Windows (окно «Свойства» файла, вкладка «Подробно», рис. 4.7 (б)). При сравнении используются три внешние ВО: создания, редактирования, последнего доступа из атрибута SI файловой записи (эти ВО отображаются в окне «Свойства» файла на вкладке «Общие», рис. 4.7 (а)), и две-три внутренние ВО: «дата создания содержимого» (ВО создания), «дата последнего сохранения» (ВО изменения), «последний вывод на печать[2]» (может отсутствовать).



а) б)

Рис. 4.7. ВО, отображаемые в окне «Свойства» файла на вкладках «Общие» (а) и «Подробно» (б)

[1] для ОС Windows 7 (для ОС Windows XP, 8, 10 обновляются 6 ВО)

[2] Данная ВО существует только для файлов, созданных программами из пакета Microsoft Office

Возможные соотношения между внешними и внутренними ВО приведены ниже:

– если внутренняя и внешняя ВО изменения файла не совпадают, то ВО были изменены с помощью специальных утилит;

– если внутренняя и внешняя ВО создания файла совпадают, то файл не является копией;

– если внутренняя и внешняя ВО создания файла не совпадают и внешняя ВО создания установлена позже чем внутренняя ВО создания, то файл является копией;

– если внутренняя и внешняя ВО создания файла не совпадают и внешняя ВО создания установлена раньше, чем внутренняя ВО создания, то ВО были изменены с помощью специальных утилит;

– если внутренняя ВО печати файла установлена раньше, чем внутренняя ВО создания, то файл является копией;

– если внутренние ВО изменения и создания файла совпадают, то файл был сохранен с помощью команды «Сохранить как…».

Сравнительный анализ внутренних ВО был реализован в функции Retro.m. После реализации основной методики восстановления последовательности ФОп на основе внешних ВО функция задает вопросы к эксперту о наличии внутренних ВО (рис. 4.8).

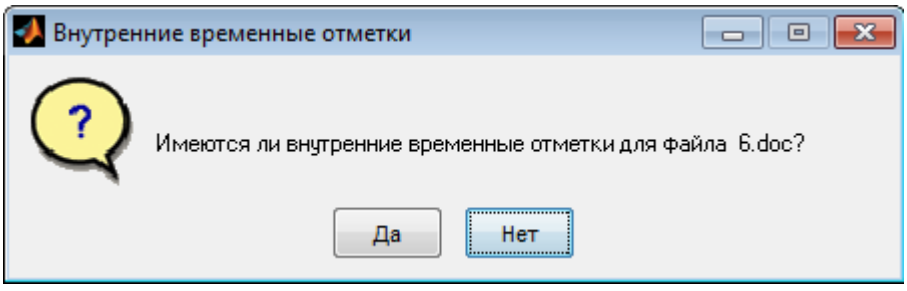


Рис. 4.8. Диалоговое окно с вопросом о наличии внутренних ВО

Если таковые имеются, то появляется диалоговое окно для введения внутренних ВО (рис. 4.9).

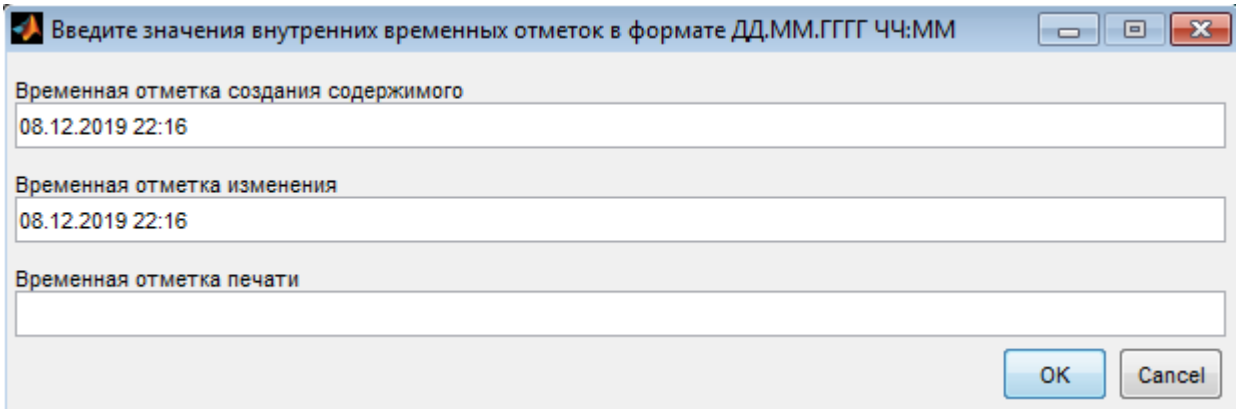


Рис. 4.9. Диалоговое окно ввода внутренних ВО

Функция Retro.m выдает результат сравнения внутренних ВО следующей строкой после анализа внешних ВО (рис. 4.10).

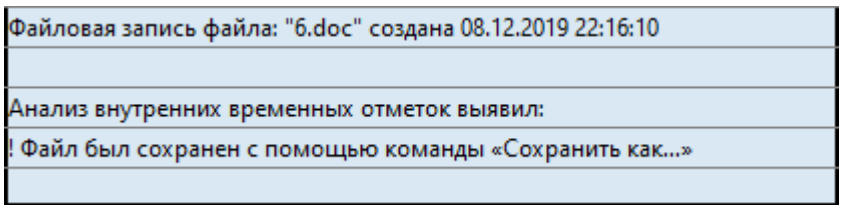


Рис. 4.10. Диалоговое окно ввода внутренних ВО

4.3 Примеры восстановления последовательности ФОп, оценка результатов восстановления

Пример 1:

Требуется определить последовательность ФОп для файла «Реферат.docx».

Исходные данные:

– фрагмент файловой таблицы, содержащей файловую запись файла «Реферат.docx»;

– состояние опции обновления ВО последнего доступа неизвестно;

– версия ОС, на которой обрабатывался файл, неизвестна;

– файл имеет две внутренние ВО: «дата создания содержимого», «дата последнего сохранения».

Результат выполнения функции Retro.m отображен на рис. 4.11.

Имя файла: Реферат.docx	Файловая операция	Временная отметка
Размер файла: 67653 байт	Вариант 1	
ВВУ = (1,2,2,2,1,2,2,2)	файл редактировался в пакете Microsoft Office	19.06.2021 06:01:33
	создана файловая запись	15.06.2021 17:33:55
	Анализ внутренних временных отметок выявил:	
	! Файл является копией	
	Время создания содержимого:	05.05.2021 18:57:00

Рис. 4.11. Результат работы функции Retro.m для файла «Реферат.docx»

В результате выполнения функции Retro.m восстановлена последовательность из 3 ФОп: «файл редактировался в пакете Microsoft Office», «создана файловая запись», «создание содержимого» (дополнительно установлено по анализу внутренних ВО).

Оценка результатов восстановления:

Данный пример иллюстрирует восстановление операции копирование по дополнительному анализу внутренних ВО, что было реализовано в функции Retro.m. Способ восстановления последовательности ФОп, основанный на сравнении двух внешних ВО создания и изменения (предложенный Б. Кэрриэ), не позволит выявить файл-копию.

Пример 2:

Требуется определить последовательность ФОп для файла «img02.jpg».

Исходные данные:

– фрагмент файловой таблицы, содержащей файловую запись файла «img02.jpg»;

– опция обновления ВО последнего доступа включена;

– версия ОС, на которой обрабатывался файл, — Windows 10;

– внутренние ВО неизвестны.

Результат выполнения функции Retro.m отображен на рис. 4.12.

Имя файла: img02.jpg	Файловая операция	Временная отметка
Размер файла: 3541 байт	Вариант 1	
ВВУ = (2,3,3,3,2,1,2,2)	файл редактировался (вкл. А)	18.06.2021 18:39:10
	файл был перемещен/переименован	н/д
	файл был создан методом копирования или извлечен из архива	09.12.2020 19:18:14
	исходный файл редактировался	11.02.2020 18:59:00
	Вариант 2	
	файл редактировался (вкл. А)	18.06.2021 18:39:10
	файл копировался или был разархивирован или был запущен/открыт	н/д
	файл был перемещен/переименован	н/д
	файл был создан методом копирования или извлечен из архива	09.12.2020 19:18:14
	исходный файл редактировался	11.02.2020 18:59:00
	Вариант 3	
	файл редактировался (вкл. А)	18.06.2021 18:39:10
	файл копировался или был разархивирован или был запущен/открыт	н/д
	файл запускался/открывался (вкл. А)	н/д
	файл был перемещен/переименован	н/д
	файл был создан методом копирования или извлечен из архива	09.12.2020 19:18:14
	исходный файл редактировался	11.02.2020 18:59:00
	Вариант 4	
	файл редактировался (вкл. А)	18.06.2021 18:39:10
	файл запускался/открывался (вкл. А)	н/д
	файл был перемещен/переименован	н/д
	файл был создан методом копирования или извлечен из архива	09.12.2020 19:18:14
	исходный файл редактировался	11.02.2020 18:59:00

Рис. 4.12. Результат работы функции Retro.m для файла «img02.jpg»

В результате выполнения функции Retro.m отображаются 4 варианта последовательностей ФОп. В каждом варианте содержатся операции: «файл редактировался (вкл. А)», «файл был перемещен/переименован», «файл был создан методом копирования или извлечен из архива», «исходный файл редактировался». Разница между вариантами заключается в появлении одной-двух ФОп между операциями «файл был создан методом копирования или извлечен из архива» и «файл редактировался (вкл. А)» с неопределенным временем совершения.

Оценка результатов восстановления:

Данный пример демонстрирует, что при наличии нескольких вариантов последовательностей операций 4 ФОп («файл редактировался», «файл был перемещен/переименован», «файл был создан методом копирования или извлечен из архива», «исходный файл редактировался») наблюдаются в каждом из вариантов. Неопределенными остаются ФОп, расположенные в середине

восстановленных цепочек. Эксперту необходимо иметь ввиду, что над файлом могли совершаться данные ФОп. Определение того, что файл является копией, стало возможным без внутренних ВО, так как после операции копирование следовала операция перемещение/переименование, повлекшая наследование значений ВО. Способ восстановления последовательности ФОп, основанный на сравнении двух внешних ВО создания и изменения, не позволил бы восстановить операции: копирование и перемещение/переименование.

Пример 3:

Требуется определить последовательность ФОп для файла «applsci-10-04686.pdf».

Исходные данные:

– фрагмент файловой таблицы, содержащей файловую запись файла «applsci-10-04686.pdf»;

– состояние опции обновления ВО последнего доступа неизвестно;

– версия ОС, на которой обрабатывался файл, неизвестна;

– внутренние ВО неизвестны.

Результат выполнения функции Retro.m отображен на рис. 4.13

Имя файла: applsci-10-04686.pdf	Файловая операция	Временная...
Размер файла: 2889149 байт	Вариант 1	
ВВУ = (2,1,3,3,3,3,3,3)	файл был перемещен/переименован из файловой системы FAT	29.05.2021 ...
	файл был создан методом копирования	07.05.2021 ...
	исходный файл редактировался	28.04.2021 ...
	! Округлена временная отметка создания	
	! Округлена временная отметка изменения	

Рис. 4.13. Результат работы функции Retro.m для файла «applsci-10-04686.pdf»

В результате выполнения функции Retro.m восстановлена последовательность из 3 ФОп: «файл был перемещение/переименование из файловой системы FAT», «файл был создан методом копирования» и «исходный файл редактировался». Дополнительно установлено, что ВО создания и ВО изменения округлены, что характерно при перемещении файла из ФС FAT.

Оценка результатов восстановления:

В данном примере восстановлена ФОп «файл был перемещен/переименован из файловой системы FAT». Определение данной операции стало возможно благодаря анализу 8 ВО, хранящихся в файловой записи, и по дополнительному определению округленных ВО, что не проверяется в способах восстановления, предложенных Б. Кэрриэ, Г. Чо, Т. Кнутсона, В. Матвеевой.

4.4 Выводы

1. Разработана функция Retro.m, которая автоматически извлекает ВО из файловых записей; формирует на их основе ВВУ; производит поиск сформированных ВВУ в таблице соответствия между ВВУ и набором возможных вариантов последовательностей ФОп; исключает последовательности ФОп исходя из дополнительных исходных условий; отображает в табличном виде полученные варианты последовательностей ФОп.

2. Приведены рекомендации по использованию ВО, хранящихся во внутренней структуре файла (внутренних ВО) для уточнения анализа. В функцию Retro.m добавлен функционал, связанный со сравнительным анализом внутренних ВО.

3. Приведены примеры восстановления последовательностей ФОп с применением функции Retro.m. Показано, что применение предложенной методики и реализующей ее функции Retro.m позволяет восстанавливать операции, которые не выявляются существующими способами восстановления.

4. Применение функции Retro.m сводит к минимуму время, затрачиваемое экспертами на проведение непосредственно анализа ВО. Эксперт подает на вход функции файловую запись объекта исследования, функция автоматически по ВО файловой записи формирует вектора временных уровней и производит их поиск в таблице соответствия между векторами и набором вариантов последовательностей ФОп, которая хранится в функции. На выход функция возвращает возможные варианты последовательностей ФОп из таблицы. Данная процедура для одного файла занимает доли секунды.

ЗАКЛЮЧЕНИЕ

В результате проведения диссертационного исследования частные задачи выполнены в полном объеме:

1. Проведен анализ существующих способов восстановления ФОп в ОС Windows, в результате которого обоснована целесообразность изучения изменений ВО для решения задачи восстановления последовательности ФОп.

2. Предложен алгоритм проведения экспериментальных исследований процесса изменений внешних ВО. Исследования проводились над файлами с различной наполняемостью и разных форматов в ОС Windows XP, 7, 8 и 10. При проведении экспериментов использовалась программа, которая работает в режиме «только чтение» и выводит информацию обо всех ВО, хранящихся в файловой записи файла. В результате проведения экспериментальных исследований выявлены закономерности изменений ВО при совершении ФОп. Закономерности обобщены в таблицу, представленную в разделе 2.3.

3. Обоснован изоморфизм динамики изменения ВО файлов в ОС Windows и динамики изменения состояний КА. Представлена модель, описывающая закономерности процесса изменения ВО при выполнении операций над файлами, в виде конечного автомата, где состояниями являются множество состояний ВО файлов, а входным алфавитом — множество совершаемых ФОп. Адекватность модели подтверждена экспериментально.

4. Предложена методика восстановления последовательности ФОп, основанная на разработанной модели. Методика позволяет достоверно восстанавливать последовательность ФОп и определять последнюю совершенную над файлом операцию.

5. Разработана и протестирована функция, позволяющая автоматизировать методику восстановления хронологии ФОп, что значительно сокращает временные затраты при исследовании большого количества файлов. Сформированы рекомендации по применению функции и использованию внутренних ВО для уточнения анализа.

Прагматическая цель работы, которая заключалась в разработке математической модели изменения значений ВО и программного обеспечения для восстановления последовательности ФОп на основе анализа ВО файлов, которые поспособствуют повышению количества восстанавливаемых ФОп, увеличению длины последовательности восстанавливаемых ФОп и уменьшению времени, затрачиваемому на восстановление хронологии ФОп, достигнута.

В результате применения алгоритма экспериментального исследования обнаружены изменения ВО для ФОп, которые ранее не были известны. На основе модели изменения значений ВО разработана методика восстановления последовательности ФОп, которая позволяет восстанавливать цепочку ФОп, состоящую из двух и более операций. Функция Retro.m, реализующая методику восстановления ФОп, уменьшает время, затрачиваемое экспертами при проведении анализа ВО.

Направление дальнейших исследований

Перспективным направлением дальнейших исследований является разработка методики восстановления последовательности ФОп, основанной на анализе ВО и журналов событий ОС Windows, с учетом возможной модификации системного времени.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

API — Application Programming Interface
ASCII — American standard code for information interchange
EXIF — Exchangeble Image Format
FAT — File Allocation Table
FTA — File Time Analyzer
MFT — Master File Table
NTFS — New Technology File System
UTC — Universal Time Coordinated
DR — атрибуты $INDEX_ROOT и $INDEX_ALLOCATION
FN — атрибут $FILE_NAME
SI — атрибут $STANDARD_INFORMATION
ВВУ — вектор временных уровней
ВО — временная отметка
КТЭ — компьютерно-техническая экспертиза
ОС — операционная система
ПО — программное обеспечение
ПЭВМ — персональная электронно-вычислительная машина
ФОп — файловая операция
ФС — файловая система

СПИСОК ЛИТЕРАТУРЫ

1. Краткая характеристика состояния преступности в Российской Федерации за январь - декабрь 2018 года. [Электронный ресурс]. — Режим доступа : https://мвд.рф/reports/item/16053092. (дата обращения: 03.06.2021).

2. Краткая характеристика состояния преступности в Российской Федерации за январь - декабрь 2019 года. [Электронный ресурс]. — Режим доступа : https://мвд.рф/reports/item/19412450. (дата обращения: 03.06.2021).

3. Краткая характеристика состояния преступности в Российской Федерации за январь - декабрь 2020 года. [Электронный ресурс]. — Режим доступа : https://мвд.рф/reports/item/22678184. (дата обращения: 03.06.2021).

4. Шелупанов А.А., Смолина А.Р. Методика проведения подготовительной стадии исследования при производстве компьютерно-технической экспертизы. — Доклады Томского государственного университета систем управления и радиоэлектроники. — 2016. — № 1. — С. 31-34.

5. ГОСТ Р 57429—2017. Судебная компьютерно-техническая экспертиза. Термины и определения.

6. Гайдук А.Р. Непрерывные и дискретные динамические системы. — М.: УМ и ИЦ «Учебная литература», 2004. — 252 с.

7. Федотов Н.Н. Форензика — компьютерная криминалистика. — М.: Юридический мир, 2007. — 432 с.

8. Шелупанов А.А., Смолина А.Р. Форензика. Теория и практика расследования киберпреступлений. — М.: Горячая линия – Телеком, 2020. — 104 с.

9. Смолина А.Р., Шелупанов А.А. Классификация методик производства компьютерно-технической экспертизы с помощью подхода теории графов. — Безопасность информационных технологий. — № 2016-2. — С. 73-77.

10. Кэрриэ Б. Криминалистический анализ файловых систем. — СПб. : Питер, 2007. — 480 с.

11. Chow K., Law F., Kwan M., Lai K. The Rules of Time on NTFS File System // Second International Workshop on Systematic Approaches to Digital Forensic

Engineering. — 2007. [Электронный ресурс]. — Режим доступа : i.cs.hku.hk/cisc/forensics/papers/RuleOfTime.pdf (дата обращения: 03.06.2021).

12. Knutson T. Filesystem Timestamps: What Makes Them Tick? — 2016. [Электронный ресурс]. — Режим доступа : https://www.sans.org/reading-room/whitepapers/forensics/filesystem-timestamps-tick-36842 (дата обращения: 03.06.2021).

13. Матвеева В.С. Криминалистический подход к анализу временных атрибутов файлов в операционной системе семейства Microsoft Windows и файловой системе NTFS // Безопасность информационных технологий. — 2013. — Вып. 1.

14. Cho GS. An Intuitive Computer Forensic Method by Timestamp Changing Patterns // Eight International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. — 2014. — pp.542-548.

15. Cho GS. A computer forensic method for detecting timestamp forgery in NTFS // Computer & Security, 2013. — Vol. 34. — pp.36-46.

16. Ding X., Zou H. Reliable Time Based Forensics in NTFS. — 2010. [Электронный ресурс]. — Режим доступа : https://www.acsac.org/2010/program/posters/ding.pdf (дата обращения: 03.06.2021).

17. Neuner S., Voyiatzis A. G., Schmiedecker M., Weippl E. Timestamp hiccups: Detecting manipulated filesystem timestamps on NTFS. // Proceedings of the 12th International Conference on Availability, Reliability and Security, 2017. — pp. 1–6.

18. Didriksen E. Forensic Analysis of OOXML Documents // Master’s Thesis. [Электронный ресурс]. — Режим доступа : https://ntnuopen.ntnu.no/ntnu-xmlui/bitstream/handle/11250/198656/EDidriksen.pdf?sequence=1 (дата обращения: 03.06.2021).

19. Gungor A. Word Forensic Analysis And Compound File Binary Format. [Электронный ресурс]. — Режим доступа : https://www.forensicfocus.com/articles/word-forensic-analysis-and-compound-file-binary-format (дата обращения: 03.06.2021).

20. Godiah D.O. Forensic analysis of office open XML spreadsheets (Thesis). — [Электронный ресурс]. — Режим доступа : https://su-plus.strathmore.edu/handle/11071/5614 (дата обращения: 03.06.2021).

21. Parsonage H. The Meaning of Linkfiles In Forensic Examinations. [Электронный ресурс]. — Режим доступа : http://computerforensics.parsonage.co.uk/downloads/TheMeaningofLIFE.pdf (дата обращения: 03.06.2021).

22. Rusbarsky K. A Forensic Comparison of NTFS and FAT32 File Systems. — 2012. [Электронный ресурс]. — Режим доступа : http://www.marshall.edu/forensics/files/RusbarskyKelsey_Research-Paper-Summer-2012.pdf (дата обращения: 03.06.2021).

23. Casey E. Errors, Uncertainty and Loss in Digital Evidence // International Journal on Digital Evidence. — 2002. — Vol. 1(2). [Электронный ресурс]. — Режим доступа : https://pdfs.semanticscholar.org/35d7/b7386ee91fc4576966259daf360b4754c1d0.pdf?_ga=2.92825179.718869684.1501156263-1854761517.1501156263 (дата обращения: 03.06.2021).

24. Brian D., Eugene H. Defining event reconstruction of a digital crime scene // Journal of Forensic Sciences. — 2004. — Vol. 6(49). [Электронный ресурс]. — Режим доступа : https://pdfs.semanticscholar.org/fa3e/6b4d398dc0afeab038fc9e267f63c6226914.pdf (дата обращения: 03.06.2021).

25. Tanushree R., Aruna J. Windows Registry Forensics: An Imperative Step in Tracking Data Theft via USB Devices // International Journal of Computer Science and Information Technologies. — 2012. — Vol. 3(3). [Электронный ресурс]. — Режим доступа : http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.301.4521&rep=rep1&type=pdf (дата обращения: 03.06.2021).

26. Minnaard W. Timestomping NTFS. — 2014. — [Электронный ресурс]. — Режим доступа : http://www.delaat.net/rp/2013-2014/p48/report.pdf (дата обращения: 03.06.2021).

27. Antonovich C. Jump List Forensics. // Patrick Leahy Center for Digital Investigation (LCDI), Champlain College Miller Center, Burlington, USA, April, 2014.

28. Lallie H. and Bains P. An overview of the jump list configuration file in Windows 7 // Journal of Digital Forensics, Security and Law, 2012. — Vol. 7(1). — pp. 15–28.

29. Lyness R. Forensic Analysis of Windows 7 Jump Lists. [Электронный ресурс]. — Режим доступа : https://articles.forensicfocus.com/2012/10/30/forensic-analysis-of-windows-7-jump-lists/ (дата обращения: 03.06.2021).

30. Kishore A. Delete or Clear Jump List Recent Items in Windows 7, 8 & 10. [Электронный ресурс]. — Режим доступа : https://www.online-tech-tips.com/computer-tips/clear-recent-items-windows-jumplists/ (дата обращения: 03.06.2021).

31. Cho GS., Rogers M. Finding Forensic Information on Creating a Folder in $LogFile of NTFS // Social Informatics and Telecommunications Engineering, 2012. — pp.211-225.

32. Xiaoyu H., Shunxiang W. Vista Event Log File Parsing Based on XML Technology // Proceedings of 4th International Conference on Computer Science & Education, 2009. — pp.1186-1190.

33. Murphey R. Automated Windows event log forensics // Digital Investigation 4S, 2007. — pp. 92-100.

34. Dwyer J., Truta T.M. Finding in Windows Event Logs Using Standard Deviation // 9th IEEE Inernational Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2013. — pp. 563-570.

35. FTK® Imager | AccessData. [Электронный ресурс]. — Режим доступа : https://accessdata.com/products-services/forensic-toolkit-ftk/ftkimager (дата обращения: 10.08.2020).

36. The Sleuth Kit. [Электронный ресурс]. — Режим доступа : https://sleuthkit.org/sleuthkit/download.php (дата обращения: 10.08.2020).

37. Autopsy | Digital Forensics. [Электронный ресурс]. — Режим доступа : https://www.autopsy.com (дата обращения: 10.08.2020).

38. NirSoft - freeware utilities: password recovery, system utilities, desktop utilities. [Электронный ресурс]. — Режим доступа : https://www.nirsoft.net (дата обращения: 15.05.2019).

39. Программа Восстановления Данных для ОС Windows. [Электронный ресурс]. — Режим доступа : https://www.r-studio.com/ru (дата обращения: 10.08.2020).

40. Plaso/tools at master · log2timeline/plaso · GitHub [Электронный ресурс]. — Режим доступа : https://github.com/log2timeline/plaso/blob/master/tools/psort.py (дата обращения: 10.08.2020).

41. Усов А.И. Концептуальные основы судебной компьютерно-технической экспертизы : автореф. дис. ... д-ра юрид. наук : 12.00.09 / Моск. ин-т МВД России. М., 2002.

42. Усов А.И. Судебная компьютерно-техническая экспертиза: становление, развитие, методическое обеспечение // Теория и практика судебной экспертизы. — № 3 (11). 2008. — С. 10-22.

43. Россинская Е. Р., Рядовский И. А. Концепция цифровых следов в криминалистике // Аубакировские чтения : материалы Международной научно-практической конференции (19 февраля 2019 г.). Алматы, 2019. — С. 6-8.

44. NtfsDisableLastAccessUpdate. [Электронный ресурс]. — Режим доступа : https://technet.microsoft.com/en-us/library/cc959914.aspx (дата обращения: 03.06.2021).

45. ГОСТ ИСО 8601–2001. Представление дат и времени. Общие требования. — Взамен ГОСТ 7.64–90 ; введ. 2002–07–01. — Минск : Межгос. совет по стандартизации, метрологии и сертификации. — 20 с. — (Система стандартов по информации, библиотечному и издательскому делу).

46. JavaScript метод Date.toISOString. [Электронный ресурс]. — Режим доступа : https://basicweb.ru/javascript/js_date.php (дата обращения: 10.08.2020).

47. [MS-SHLLINK] — v20170915 Shell Link (.LNK) Binary File Format. [Электронный ресурс]. — Режим доступа: https://msdn.microsoft.com/en-us/library/dd871305.aspx (дата обращения: 02.04.2018).

48. FAR Manager — скачать бесплатно с официального сайта. [Электронный ресурс]. — Режим доступа: https://farmanager.ru (дата обращения: 10.06.2021).

49. Total Commander – Download. [Электронный ресурс]. — Режим доступа: https://www.ghisler.com/download.htm (дата обращения: 10.06.2021).

50. FreeCommander официальный сайт, бесплатно скачать FreeCommander XE. [Электронный ресурс]. — Режим доступа: https://freecommander.ru (дата обращения: 10.06.2021).

51. File Shredder. [Электронный ресурс]. — Режим доступа: https://www.fileshredder.org (дата обращения: 10.06.2021).

52. Free PC cleaner & Privacy tool – Download. [Электронный ресурс]. — Режим доступа: https://privazer.com/download.php (дата обращения: 10.06.2021).

53. Recuva - бесплатная программа для восстановления файлов. [Электронный ресурс]. — Режим доступа: http://recuva.su (дата обращения: 10.06.2021).

54. SDelete - Windows Sysinternals | Microsoft Docs. [Электронный ресурс]. — Режим доступа: https://docs.microsoft.com/en-us/sysinternals/downloads/sdelete (дата обращения: 10.06.2021).

55. WordPad официальный сайт, бесплатно скачать текстовый редактор Ворд Пад на русском. [Электронный ресурс]. — Режим доступа: https://wordpad.ru (дата обращения: 10.06.2021).

56. Office 365 | Microsoft Office. [Электронный ресурс]. — Режим доступа: https://www.office.com (дата обращения: 10.06.2021).

57. Adobe Acrobat Reader DC (Россия). [Электронный ресурс]. — Режим доступа: https://acrobat.adobe.com/ru/ru/acrobat/pdf-reader.html (дата обращения: 10.06.2021).

58. Бесплатная загрузка PDF Reader и PDF Viewer | Foxit Software. [Электронный ресурс]. — Режим доступа: https://www.foxit.com/ru/pdf-reader (дата обращения: 10.06.2021).

59. GNU Image Manipulation Program (GIMP). [Электронный ресурс]. — Режим доступа: https://www.gimp.org (дата обращения: 10.06.2021).

60. Скачать бесплатный фоторедактор на русском языке – Movavi. [Электронный ресурс]. — Режим доступа: https://www.movavi.ru/photosuite-download (дата обращения: 10.06.2021).

61. Photoscape бесплатный графический редактор с русским интерфейсом. [Электронный ресурс]. — Режим доступа: https://photoscape.su (дата обращения: 10.06.2021).

62. PixBuilder Studio - Free photo editor. [Электронный ресурс]. — Режим доступа: https://www.wnsoft.com/en/pixbuilder (дата обращения: 10.06.2021).

63. Media Player Classic официальный сайт! Лучший видеоплеер 2018 года для Windows: XP, Vista, 7, 8, 8.1, 10! [Электронный ресурс]. — Режим доступа: http://mediaplayerclassic.ru (дата обращения: 10.06.2021).

64. Браузер Opera | Быстрее, безопаснее, умнее | Opera. [Электронный ресурс]. — Режим доступа: https://www.opera.com/ru (дата обращения: 10.06.2021).

65. Интернет для людей, а не для прибыли — Mozilla. [Электронный ресурс]. — Режим доступа: https://www.mozilla.org/ru (дата обращения: 10.06.2021).

66. Веб-браузер Google Chrome. [Электронный ресурс]. — Режим доступа: https://www.google.ru/chrome (дата обращения: 10.06.2021).

67. WinRAR download free and support: WinRAR. [Электронный ресурс]. — Режим доступа : http://www.win-rar.ru (дата обращения: 10.06.2021).

68. 7-Zip. [Электронный ресурс]. — Режим доступа : https://www.7-zip.org (дата обращения: 10.06.2021).

69. Руссинович М., Соломон Д., Ионеску А., Йосифович П. Внутреннее устройство Windows. 7-е изд. — СПб. : Питер, 2018. — 944 с.

70. ClockRes - Windows Sysinternals | Microsoft Docs. [Электронный ресурс]. — Режим доступа : https://docs.microsoft.com/en-us/sysinternals/downloads/clockres (дата обращения: 03.06.2021).

71. Timestomp-GUI - Browse /Windows at SourceForge.net. [Электронный ресурс]. — Режим доступа : https://sourceforge.net/projects/timestomp-gui/files/Windows/Timestomp-GUI.exe/download (дата обращения: 10.08.2020).

72. Перегудов Ф.И., Тарасенко Ф.П. Основы системного анализа. — Томск: Изд-во НТЛ, 1997. — 396 с.

73. Советов Б.Я. Моделирование систем : Учебник для вузов / Б.Я. Советов, С.А. Яковлев — 5-е изд., стер. — М. : Высш. шк., 2007. — 343 с.

74. Пестриков В.М. Дискретная математика : учеб. пособие /сост. В.М. Пестриков, В.С. Дудкин , Г.А. Петров — СПб.: СПб ГТУРП, 2013. — 136 с.

75. Карпов Ю.Г. Теория автоматов : Учебник для вузов. — СПб. : Питер, 2002. — 224 с.

76. Graph with directed edges – MATLAB [Электронный ресурс]. — Режим доступа : https://ch.mathworks.com/help/matlab/ref/digraph.html (дата обращения: 01.04.2021).

77. Асанов М.О., Баранский В.А., Расин В.В. Дискретная математика: графы матроиды, алгоритмы. — Ижевск: НИЦ «РХД». 2001. — 288 с.

78. Breadth-first graph search – MATLAB bfsearch [Электронный ресурс]. — Режим доступа : https://www.mathworks.com/help/matlab/ref/graph.bfsearch.html (дата обращения: 01.04.2021).

79. Depth-first graph search – MATLAB dfsearch [Электронный ресурс]. — Режим доступа : https://www.mathworks.com/help/matlab/ref/graph.dfsearch.html (дата обращения: 01.04.2021).

80. Find all paths between two graph nodes - MATLAB allpaths [Электронный ресурс]. — Режим доступа : https://www.mathworks.com/help/matlab/ref/graph.allpaths.html (дата обращения: 01.04.2021).

Список публикаций автора по теме диссертации

Статьи, опубликованные в рецензируемых научных журналах и изданиях, определенных ВАК РФ и Аттестационным советом УрФУ:

81. Духан Е.И. Князева Н.С. Анализ результатов исследования изменений временных отметок файлов // Вестник УрФО. Безопасность в информационной сфере. — 2021. — Вып. 39. — № 1. — С. 21-26 (ВАК).

82. Князева Н.С. Восстановление последовательности файловых операций с применением теории графов при проведении компьютерных исследований // Вестник УрФО. Безопасность в информационной сфере. — 2021. — Вып. 40. — № 2. — С. 14-21 (ВАК).

83. Духан Е.И., Князева Н.С. Методика и результаты исследования изменений временных отметок файловых объектов // М.: Радиотехника, 2020. — т. 84. — № 2(4). — С. 64-72 (ВАК).

84. Natalia Knyazeva, Dmitry Khorkov, Elena Vostretsova. Building Knowledge Bases for Timestamp Changes Detection Mechanisms in MFT Windows OS // 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), 2020. — С. 553-556 (Scopus).

85. Natalia Knyazeva, Evgeny Dukhan. Timestamp Change Model in Windows OS // 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT), 2020. — С. 623-626 (Scopus).

Другие публикации:

86. Бакланов В.В., Князева Н.С., Хорьков Д.А. Анализ временных отметок файловой системы NTFS в операционной системе Microsoft Windows XP // Проблемы информационной безопасности. Компьютерные системы. Выпуск № 4. — Санкт-Петербург, 2012. — С. 25-32.

ПРИЛОЖЕНИЕ А. ЛИСТИНГ ФУНКЦИИ TABLE.M

```
clear
A0=[1 1 1 1 1 1 1 1];%начальное состояние
A(1,:)=A0;

F(1,:)=[1 1 1 1 1 1 1 1];% начальное состояние
F(2,:)=[1 0 1 0 1 1 1 1];% копирование (новый объект) 10101111
F(3,:)=[1 0 1 1 1 1 1 1];% копирование (новый объект) 10111111
F(4,:)=[0 1 1 1 0 0 0 0];% редактирование (вкл. A) 01110000
F(5,:)=[0 1 0 1 0 0 0 0];% редактирование (выкл. A) 01010000
F(6,:)=[0 1 1 1 0 1 1 1];% редактирование в Office 01110111
F(7,:)=[0 0 0 1 0 0 0 0];% перемещение (переименование) 0001XXXX (где X -
значения BO из атрибута SI до произведенной операции)
F(8,:)=[0 0 1 1 1 1 1 1];% перемещение (переименование) из ФС FAT в ФС NTFS
00111111
F(9,:)=[0 0 0 1 1 1 1 1];% разархивирование встроенным архиватором Windows
(новый объект) 00011111
F(10,:)=[0 0 1 0 0 0 0 0];% просмотр атрибутов, открытие, копирование,
разархивирование (вкл. A) 00100000
F(11,:)=[0 0 0 1 0 0 0 0];% изменение атрибутов, открытие (выкл. A) 00010000
F(12,:)=[0 0 1 1 0 0 0 0];% запуск/открытие в ОС Windows XP в оболочке Explorer,
удаление (вкл. A)  00110000
F(13,:)=[0 0 0 1 0 0 0 1];% перемещение/переименование в файловом менеджере
Total Commander 0001XXX1

%Массив M(функция,вектор,состояние);
sa=1;
while sa<=size(A,1)
    d=max(A(sa,:))+1;
    % 11111111
    M(1,1,sa)=A(sa,1);
    M(1,2,sa)=A(sa,2);
    M(1,3,sa)=A(sa,3);
    M(1,4,sa)=A(sa,4);
    M(1,5,sa)=A(sa,5);
    M(1,6,sa)=A(sa,6);
    M(1,7,sa)=A(sa,7);
    M(1,8,sa)=A(sa,8);
    % 10101111
    M(2,1,sa)=d;
    M(2,2,sa)=A(sa,2);
    M(2,3,sa)=d;
    M(2,4,sa)=A(sa,4);
    M(2,5,sa)=d;
    M(2,6,sa)=d;
    M(2,7,sa)=d;
    M(2,8,sa)=d;
    % 10111111
    M(3,1,sa)=d;
    M(3,2,sa)=A(sa,2);
    M(3,3,sa)=d;
    M(3,4,sa)=d;
    M(3,5,sa)=d;
    M(3,6,sa)=d;
    M(3,7,sa)=d;
    M(3,8,sa)=d;
    % 01110000
    M(4,1,sa)=A(sa,1);
    M(4,2,sa)=d;
    M(4,3,sa)=d;
    M(4,4,sa)=d;
```

```
M(4,5,sa)=A(sa,5);
M(4,6,sa)=A(sa,6);
M(4,7,sa)=A(sa,7);
M(4,8,sa)=A(sa,8);
% 01010000
M(5,1,sa)=A(sa,1);
M(5,2,sa)=d;
M(5,3,sa)=A(sa,3);
M(5,4,sa)=d;
M(5,5,sa)=A(sa,5);
M(5,6,sa)=A(sa,6);
M(5,7,sa)=A(sa,7);
M(5,8,sa)=A(sa,8);
% 01110111
M(6,1,sa)=A(sa,1);
M(6,2,sa)=d;
M(6,3,sa)=d;
M(6,4,sa)=d;
M(6,5,sa)=A(sa,5);
M(6,6,sa)=d;
M(6,7,sa)=d;
M(6,8,sa)=d;
% 0001XXXX (где X - значения BO из атрибута SI)
M(7,1,sa)=A(sa,1);
M(7,2,sa)=A(sa,2);
M(7,3,sa)=A(sa,3);
M(7,4,sa)=d;
M(7,5,sa)=A(sa,1);
M(7,6,sa)=A(sa,2);
M(7,7,sa)=A(sa,3);
M(7,8,sa)=A(sa,4);
% 00111111
M(8,1,sa)=A(sa,1);
M(8,2,sa)=A(sa,2);
M(8,3,sa)=d;
M(8,4,sa)=d;
M(8,5,sa)=d;
M(8,6,sa)=d;
M(8,7,sa)=d;
M(8,8,sa)=d;
% 00011111
M(9,1,sa)=A(sa,1);
M(9,2,sa)=A(sa,2);
M(9,3,sa)=A(sa,3);
M(9,4,sa)=d;
M(9,5,sa)=d;
M(9,6,sa)=d;
M(9,7,sa)=d;
M(9,8,sa)=d;
% 00100000
M(10,1,sa)=A(sa,1);
M(10,2,sa)=A(sa,2);
M(10,3,sa)=d;
M(10,4,sa)=A(sa,4);
M(10,5,sa)=A(sa,5);
M(10,6,sa)=A(sa,6);
M(10,7,sa)=A(sa,7);
M(10,8,sa)=A(sa,8);
% 00010000
M(11,1,sa)=A(sa,1);
M(11,2,sa)=A(sa,2);
M(11,3,sa)=A(sa,3);
M(11,4,sa)=d;
M(11,5,sa)=A(sa,5);
```

```
    M(11,6,sa)=A(sa,6);
    M(11,7,sa)=A(sa,7);
    M(11,8,sa)=A(sa,8);
    % 00110000
    M(12,1,sa)=A(sa,1);
    M(12,2,sa)=A(sa,2);
    M(12,3,sa)=d;
    M(12,4,sa)=d;
    M(12,5,sa)=A(sa,5);
    M(12,6,sa)=A(sa,6);
    M(12,7,sa)=A(sa,7);
    M(12,8,sa)=A(sa,8);
    % 0001XXX1
    M(13,1,sa)=A(sa,1);
    M(13,2,sa)=A(sa,2);
    M(13,3,sa)=A(sa,3);
    M(13,4,sa)=d;
    M(13,5,sa)=A(sa,1);
    M(13,6,sa)=A(sa,2);
    M(13,7,sa)=A(sa,3);
    M(13,8,sa)=d;

    for sf=1:size(F,1)
    M(sf,:,sa)=chek(M(sf,:,sa));

    %проверка есть ли состояние в A
    ch=0;
    saa=1;
        while saa<=size(A,1)%крутит все состояния
            if M(sf,:,sa)==A(saa,:)
            ch=1;%если 1 раз попадался
            end
        saa=saa+1;
        end
        %добавление нового состояния
        if ch==0%если 0 то ни разу не попадался
        A((size(A,1)+1),:)= M(sf,:,sa);
        end
    end
sa=sa+1;
if sa>1
b(1,1)=M(1,1,sa-1);
b(1,2)=M(1,2,sa-1);
b(1,3)=M(1,3,sa-1);
b(1,4)=M(1,4,sa-1);
b(1,5)=M(1,5,sa-1);
b(1,6)=M(1,6,sa-1);
b(1,7)=M(1,7,sa-1);
b(1,8)=M(1,8,sa-1);
a{1,sa-1}=b;
table{1,sa-1}=b;
b1(1,1)=M(2,1,sa-1);
b1(1,2)=M(2,2,sa-1);
b1(1,3)=M(2,3,sa-1);
b1(1,4)=M(2,4,sa-1);
b1(1,5)=M(2,5,sa-1);
b1(1,6)=M(2,6,sa-1);
b1(1,7)=M(2,7,sa-1);
b1(1,8)=M(2,8,sa-1);
table{2,sa-1}=b1;
b2(1,1)=M(3,1,sa-1);
b2(1,2)=M(3,2,sa-1);
b2(1,3)=M(3,3,sa-1);
b2(1,4)=M(3,4,sa-1);
```

```
b2(1,5)=M(3,5,sa-1);
b2(1,6)=M(3,6,sa-1);
b2(1,7)=M(3,7,sa-1);
b2(1,8)=M(3,8,sa-1);
table{3,sa-1}=b2;
b3(1,1)=M(4,1,sa-1);
b3(1,2)=M(4,2,sa-1);
b3(1,3)=M(4,3,sa-1);
b3(1,4)=M(4,4,sa-1);
b3(1,5)=M(4,5,sa-1);
b3(1,6)=M(4,6,sa-1);
b3(1,7)=M(4,7,sa-1);
b3(1,8)=M(4,8,sa-1);
table{4,sa-1}=b3;
b4(1,1)=M(5,1,sa-1);
b4(1,2)=M(5,2,sa-1);
b4(1,3)=M(5,3,sa-1);
b4(1,4)=M(5,4,sa-1);
b4(1,5)=M(5,5,sa-1);
b4(1,6)=M(5,6,sa-1);
b4(1,7)=M(5,7,sa-1);
b4(1,8)=M(5,8,sa-1);
table{5,sa-1}=b4;
b5(1,1)=M(6,1,sa-1);
b5(1,2)=M(6,2,sa-1);
b5(1,3)=M(6,3,sa-1);
b5(1,4)=M(6,4,sa-1);
b5(1,5)=M(6,5,sa-1);
b5(1,6)=M(6,6,sa-1);
b5(1,7)=M(6,7,sa-1);
b5(1,8)=M(6,8,sa-1);
table{6,sa-1}=b5;
b6(1,1)=M(7,1,sa-1);
b6(1,2)=M(7,2,sa-1);
b6(1,3)=M(7,3,sa-1);
b6(1,4)=M(7,4,sa-1);
b6(1,5)=M(7,5,sa-1);
b6(1,6)=M(7,6,sa-1);
b6(1,7)=M(7,7,sa-1);
b6(1,8)=M(7,8,sa-1);
table{7,sa-1}=b6;
b7(1,1)=M(8,1,sa-1);
b7(1,2)=M(8,2,sa-1);
b7(1,3)=M(8,3,sa-1);
b7(1,4)=M(8,4,sa-1);
b7(1,5)=M(8,5,sa-1);
b7(1,6)=M(8,6,sa-1);
b7(1,7)=M(8,7,sa-1);
b7(1,8)=M(8,8,sa-1);
table{8,sa-1}=b7;
b8(1,1)=M(9,1,sa-1);
b8(1,2)=M(9,2,sa-1);
b8(1,3)=M(9,3,sa-1);
b8(1,4)=M(9,4,sa-1);
b8(1,5)=M(9,5,sa-1);
b8(1,6)=M(9,6,sa-1);
b8(1,7)=M(9,7,sa-1);
b8(1,8)=M(9,8,sa-1);
table{9,sa-1}=b8;
b9(1,1)=M(10,1,sa-1);
b9(1,2)=M(10,2,sa-1);
b9(1,3)=M(10,3,sa-1);
b9(1,4)=M(10,4,sa-1);
b9(1,5)=M(10,5,sa-1);
```

```
b9(1,6)=M(10,6,sa-1);
b9(1,7)=M(10,7,sa-1);
b9(1,8)=M(10,8,sa-1);
table{10,sa-1}=b9;
b10(1,1)=M(11,1,sa-1);
b10(1,2)=M(11,2,sa-1);
b10(1,3)=M(11,3,sa-1);
b10(1,4)=M(11,4,sa-1);
b10(1,5)=M(11,5,sa-1);
b10(1,6)=M(11,6,sa-1);
b10(1,7)=M(11,7,sa-1);
b10(1,8)=M(11,8,sa-1);
table{11,sa-1}=b10;
b11(1,1)=M(12,1,sa-1);
b11(1,2)=M(12,2,sa-1);
b11(1,3)=M(12,3,sa-1);
b11(1,4)=M(12,4,sa-1);
b11(1,5)=M(12,5,sa-1);
b11(1,6)=M(12,6,sa-1);
b11(1,7)=M(12,7,sa-1);
b11(1,8)=M(12,8,sa-1);
table{12,sa-1}=b11;
b12(1,1)=M(13,1,sa-1);
b12(1,2)=M(13,2,sa-1);
b12(1,3)=M(13,3,sa-1);
b12(1,4)=M(13,4,sa-1);
b12(1,5)=M(13,5,sa-1);
b12(1,6)=M(13,6,sa-1);
b12(1,7)=M(13,7,sa-1);
b12(1,8)=M(13,8,sa-1);
table{13,sa-1}=b12;
end
end

%функция поиска вновь полученного состояния в таблице переходов
function X=chek(Y)
A=Y;
    f1=find(A(1,:)==1);
    f2=find(A(1,:)==2);
    f3=find(A(1,:)==3);
    f4=find(A(1,:)==4);
    f5=find(A(1,:)==5);
    f6=find(A(1,:)==6);
    f7=find(A(1,:)==7);
    f8=find(A(1,:)==8);
    f9=find(A(1,:)==9);
    f10=find(A(1,:)==10);
    f11=find(A(1,:)==11);
    f12=find(A(1,:)==12);
    f13=find(A(1,:)==13);
    if f1>0
    A1(1,f1)=1;
    else
f1=f2;f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A
1(1,f1)=1;
    end
    if f2>0
    A1(1,f2)=2;
    else
f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f2
)=2;
    end
    if f3>0
    A1(1,f3)=3;
```

```
    else
f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f3)=3;
    end
    if f4>0
    A1(1,f4)=4;
    else
f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f4)=4;
    end
    if f5>0
    A1(1,f5)=5;
    else f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f5)=5;
    end
    if f6>0
    A1(1,f6)=6;
    else f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f6)=6;
    end
    if f7>0
    A1(1,f7)=7;
    else f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f7)=7;
    end
    if f8>0
    A1(1,f8)=8;
    else f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f8)=8;
    end
    if f9>0
    A1(1,f9)=9;
    else f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f9)=9;
    end
    if f10>0
    A1(1,f10)=10;
    else f10=f11;f11=f12;f12=f13;A1(1,f10)=10;
    end
    if f11>0
    A1(1,f11)=11;
    else f11=f12;f12=f13;A1(1,f11)=11;
    end
    if f12>0
    A1(1,f12)=12;
    else f12=f13;A1(1,f12)=12;
    end

    A=A1;
    f1=find(A(1,:)==1);
    f2=find(A(1,:)==2);
    f3=find(A(1,:)==3);
    f4=find(A(1,:)==4);
    f5=find(A(1,:)==5);
    f6=find(A(1,:)==6);
    f7=find(A(1,:)==7);
    f8=find(A(1,:)==8);
    f9=find(A(1,:)==9);
    f10=find(A(1,:)==10);
    f11=find(A(1,:)==11);
    f12=find(A(1,:)==12);
    f13=find(A(1,:)==13);
    if f1>0
    A1(1,f1)=1;
    else
f1=f2;f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A
1(1,f1)=1;
    end
    if f2>0
    A1(1,f2)=2;
```

```
    else
f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f2
)=2;
    end
    if f3>0
    A1(1,f3)=3;
    else
f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f3)=3;
    end
    if f4>0
    A1(1,f4)=4;
    else
f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f4)=4;
    end
    if f5>0
    A1(1,f5)=5;
    else f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f5)=5;
    end
    if f6>0
    A1(1,f6)=6;
    else f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f6)=6;
    end
    if f7>0
    A1(1,f7)=7;
    else f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f7)=7;
    end
    if f8>0
    A1(1,f8)=8;
    else f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f8)=8;
    end
    if f9>0
    A1(1,f9)=9;
    else f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f9)=9;
    end
    if f10>0
    A1(1,f10)=10;
    else f10=f11;f11=f12;f12=f13;A1(1,f10)=10;
    end
    if f11>0
    A1(1,f11)=11;
    else f11=f12;f12=f13;A1(1,f11)=11;
    end
    if f12>0
    A1(1,f12)=12;
    else f12=f13;A1(1,f12)=12;
    end

    A=A1;
    f1=find(A(1,:)==1);
    f2=find(A(1,:)==2);
    f3=find(A(1,:)==3);
    f4=find(A(1,:)==4);
    f5=find(A(1,:)==5);
    f6=find(A(1,:)==6);
    f7=find(A(1,:)==7);
    f8=find(A(1,:)==8);
    f9=find(A(1,:)==9);
    f10=find(A(1,:)==10);
    f11=find(A(1,:)==11);
    f12=find(A(1,:)==12);
    f13=find(A(1,:)==13);
    if f1>0
    A1(1,f1)=1;
```

```
    else
f1=f2;f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A
1(1,f1)=1;
    end
    if f2>0
    A1(1,f2)=2;
    else
f2=f3;f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f2
)=2;
    end
    if f3>0
    A1(1,f3)=3;
    else
f3=f4;f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f3)=3;
    end
    if f4>0
    A1(1,f4)=4;
    else
f4=f5;f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f4)=4;
    end
    if f5>0
    A1(1,f5)=5;
    else f5=f6;f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f5)=5;
    end
    if f6>0
    A1(1,f6)=6;
    else f6=f7;f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f6)=6;
    end
    if f7>0
    A1(1,f7)=7;
    else f7=f8;f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f7)=7;
    end
    if f8>0
    A1(1,f8)=8;
    else f8=f9;f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f8)=8;
    end
    if f9>0
    A1(1,f9)=9;
    else f9=f10;f10=f11;f11=f12;f12=f13;A1(1,f9)=9;
    end
    if f10>0
    A1(1,f10)=10;
    else f10=f11;f11=f12;f12=f13;A1(1,f10)=10;
    end
    if f11>0
    A1(1,f11)=1;
    else f11=f12;f12=f13;A1(1,f11)=11;
    end
    if f12>0
    A1(1,f12)=12;
    else f12=f13;A1(1,f12)=12;
    end

    A=A1;
    X=A;
```

ПРИЛОЖЕНИЕ Б.
ТАБЛИЦА ПЕРЕХОДОВ МЕЖДУ ВВУ

x(t)	s(t)																									
	s0	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s16	s17	s18	s19	s20	s21	s22	s23	s24	s25	s26
x1	s1	s1	s19	s1	s1	s1	s19	s19	s19	s1	s19	s19	s19	s19	s19	s1	s19	s19	s19	s19	s19	s19	s1	s19	s19	s19	s1
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s3	s3	s3	s3	s3	s34	s3	s40	s40	s3	s3	s53	s57	s40	s40	s3	s3	s3	s72	s3	s78	s3	s3	s34	s40	s40	s3
x4	s4	s4	s4	s22	s4	s35	s4	s41	s49	s22	s22	s54	s58	s41	s49	s22	s4	s22	s73	s4	s79	s22	s22	s35	s41	s95	s22
x5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s5	s5	s5	s5	s42	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s5
x6	s6	s12	s20	s23	s30	s23	s38	s43	s38	s51	s43	s38	s59	s62	s59	s69	s59	s62	s59	s76	s59	s82	s83	s87	s89	s87	s97
x7	s7	s13	s13	s24	s24	s24	s7	s7	s7	s7	s7	s7	s13	s13	s13	s13	s13	s13	s13	s13	s13	s13	s24	s24	s24	s24	s24
x8	s8	s14	s14	s25	s31	s25	s8	s44	s8	s44	s44	s8	s14	s63	s14	s63	s14	s63	s14	s14	s14	s63	s84	s25	s90	s25	s90
x9	s9	s15	s21	s26	s26	s36	s39	s45	s45	s9	s39	s55	s60	s64	s64	s15	s71	s71	s74	s77	s80	s21	s26	s88	s91	s91	s26
x10	s6	s16	s16	s27	s32	s23	s6	s46	s50	s52	s52	s38	s12	s65	s68	s70	s16	s70	s59	s16	s20	s70	s85	s23	s92	s96	s98
x11	s10	s17	s17	s28	s28	s37	s10	s47	s47	s10	s10	s56	s61	s66	s66	s17	s17	s17	s75	s17	s81	s17	s28	s37	s93	s93	s28
x12	s11	s18	s18	s29	s33	s29	s11	s48	s11	s48	s48	s11	s18	s67	s18	s67	s18	s67	s18	s18	s18	s67	s86	s29	s94	s29	s94

x(t)	s(t)																									
	s27	s28	s29	s30	s31	s32	s33	s34	s35	s36	s37	s38	s39	s40	s41	s42	s43	s44	s45	s46	s47	s48	s49	s50	s51	s52	s53
x1	s19	s19	s19	s19	s19	s19	s19	s1	s1	s1	s19	s19	s19	s1	s1	s1	s19	s19	s19	s19	s19	s19	s1	s19	s19	s19	s1
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s3	s3	s100	s104	s40	s3	s110	s34	s34	s34	s34	s53	s3	s40	s40	s127	s132	s40	s40	s40	s40	s140	s40	s40	s145	s3	s53
x4	s22	s22	s101	s105	s49	s4	s111	s114	s35	s114	s114	s54	s22	s122	s41	s128	s133	s95	s122	s41	s122	s141	s49	s49	s146	s22	s149
x5	s5	s5	s5	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s5	s42	s42	s42	s42	s5	s42	s42	s5	s5	s5
x6	s87	s89	s87	s106	s106	s106	s106	s23	s83	s97	s89	s38	s121	s23	s83	s23	s134	s134	s121	s134	s43	s134	s30	s38	s134	s134	s23
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s7	s7	s24	s24	s24	s7	s7	s7	s7	s7	s7	s24	s7	s7	s7	s24
x8	s25	s90	s25	s31	s31	s31	s31	s25	s84	s90	s90	s8	s44	s25	s84	s25	s44	s44	s44	s44	s44	s44	s31	s8	s44	s44	s25
x9	s99	s99	s102	s107	s91	s99	s112	s115	s115	s36	s88	s120	s39	s123	s123	s129	s135	s45	s45	s139	s139	s142	s123	s139	s147	s39	s150
x10	s27	s98	s87	s30	s109	s32	s106	s116	s118	s119	s119	s38	s52	s124	s126	s130	s43	s137	s138	s46	s138	s134	s144	s50	s51	s52	s151
x11	s28	s28	s103	s108	s93	s28	s113	s117	s117	s37	s37	s56	s10	s125	s125	s131	s136	s47	s47	s47	s47	s143	s125	s47	s148	s10	s152

x12	s29	s94	s29	s33	s33	s33	s33	s29	s86	s94	s94	s11	s48	s29	s86	s29	s48	s48	s48	s48	s48	s48	s33	s11	s48	s48	s29
x(t)	s(t)																										
	s54	s55	s56	s57	s58	s59	s60	s61	s62	s63	s64	s65	s66	s67	s68	s69	s70	s71	s72	s73	s74	s75	s76	s77	s78	s79	s80
x1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s53	s53	s53	s57	s57	s72	s57	s57	s163	s40	s40	s40	s40	s171	s40	s175	s3	s3	s72	s72	s72	s72	s185	s3	s78	s78	s78
x4	s54	s149	s149	s155	s58	s73	s155	s155	s164	s95	s122	s41	s122	s172	s49	s176	s22	s22	s179	s73	s179	s179	s186	s22	s190	s79	s190
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s42	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s30	s121	s43	s23	s30	s59	s161	s62	s165	s165	s161	s165	s62	s165	s59	s165	s165	s161	s23	s30	s161	s62	s59	s189	s23	s30	s161
x7	s24	s7	s7	s24	s24	s13	s13	s13	s13	s13	s13	s13	s13	s13	s13	s13	s13	s13	s24	s24	s13	s13	s13	s13	s24	s24	s13
x8	s31	s44	s44	s25	s31	s14	s63	s63	s63	s63	s63	s63	s63	s63	s14	s63	s63	s63	s25	s31	s63	s63	s14	s63	s25	s31	s63
x9	s150	s55	s120	s156	s156	s160	s60	s60	s166	s64	s64	s170	s170	s173	s170	s177	s71	s71	s180	s180	s74	s160	s187	s77	s191	s191	s80
x10	s153	s154	s154	s157	s159	s59	s162	s162	s62	s168	s169	s65	s169	s165	s68	s69	s70	s70	s181	s183	s184	s184	s76	s70	s192	s194	s195
x11	s152	s56	s56	s158	s158	s75	s61	s61	s167	s66	s66	s66	s66	s174	s66	s178	s17	s17	s182	s182	s75	s75	s188	s17	s193	s193	s81
x12	s33	s48	s48	s29	s33	s18	s67	s67	s67	s67	s67	s67	s67	s67	s18	s67	s67	s67	s29	s33	s67	s67	s18	s67	s29	s33	s67

x(t)	s(t)																										
	s81	s82	s83	s84	s85	s86	s87	s88	s89	s90	s91	s92	s93	s94	s95	s96	s97	s98	s99	s100	s101	s102	s103	s104	s105	s106	s107
x1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1	s19	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s78	s196	s200	s40	s3	s206	s100	s34	s212	s40	s40	s40	s40	s220	s40	s40	s225	s3	s3	s100	s100	s100	s100	s104	s104	s110	s104
x4	s190	s197	s201	s95	s22	s207	s101	s114	s213	s95	s122	s41	s122	s221	s95	s95	s226	s22	s22	s229	s101	s229	s229	s235	s105	s111	s235
x5	s5	s5	s5	s42	s5	s5	s5	s5	s5	s42	s42	s42	s42	s5	s42	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s62	s165	s202	s202	s202	s202	s87	s211	s214	s214	s211	s214	s89	s214	s83	s87	s214	s214	s211	s23	s83	s211	s89	s23	s30	s106	s211
x7	s13	s13	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s63	s63	s84	s84	s84	s84	s25	s90	s90	s90	s90	s90	s90	s90	s84	s25	s90	s90	s90	s25	s84	s90	s90	s25	s31	s31	s90
x9	s80	s198	s203	s91	s99	s208	s210	s88	s215	s91	s91	s219	s219	s222	s123	s219	s227	s99	s99	s230	s230	s102	s210	s236	s236	s240	s107
x10	s195	s82	s83	s205	s85	s202	s87	s119	s89	s217	s218	s92	s218	s214	s224	s96	s97	s98	s98	s231	s233	s234	s234	s237	s239	s106	s241
x11	s81	s199	s204	s93	s28	s209	s103	s37	s216	s93	s93	s93	s93	s223	s125	s93	s228	s28	s28	s232	s232	s103	s103	s238	s238	s113	s108
x12	s67	s67	s86	s86	s86	s86	s29	s94	s94	s94	s94	s94	s94	s94	s86	s29	s94	s94	s94	s29	s86	s94	s94	s29	s33	s33	s94

x(t)	s(t)																											
	s108	s109	s110	s111	s112	s113	s114	s115	s116	s117	s118	s119	s120	s121	s122	s123	s124	s125	s126	s127	s128	s129	s130	s131	s132	s133	s134
x1	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s1	s1	s1	s19	s19	s1	s1	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s104	s40	s110	s110	s110	s110	s34	s34	s34	s34	s34	s34	s53	s251	s40	s40	s40	s40	s40	s127	s127	s127	s127	s127	s132	s132	s140
x4	s235	s49	s242	s111	s242	s242	s114	s114	s114	s114	s35	s114	s149	s252	s122	s122	s122	s122	s41	s258	s128	s258	s128	s258	s265	s133	s141
x5	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s42	s42	s42	s42	s42	s42	s42	s5	s5	s5
x6	s89	s106	s23	s30	s211	s89	s83	s97	s87	s89	s202	s214	s121	s134	s83	s97	s87	s89	s202	s23	s83	s97	s87	s89	s23	s83	s134
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s7	s7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s7
x8	s90	s31	s25	s31	s90	s90	s84	s90	s25	s90	s84	s90	s44	s44	s84	s90	s25	s90	s84	s25	s84	s90	s25	s90	s25	s84	s44
x9	s107	s219	s243	s243	s112	s240	s115	s115	s250	s250	s250	s88	s120	s253	s123	s123	s257	s257	s257	s259	s259	s129	s264	s264	s266	s266	s270
x10	s241	s109	s244	s246	s247	s247	s248	s249	s116	s249	s118	s119	s154	s121	s255	s256	s124	s256	s126	s260	s262	s263	s130	s263	s267	s269	s134
x11	s108	s93	s245	s245	s113	s113	s117	s117	s117	s117	s117	s37	s56	s254	s125	s125	s125	s125	s125	s261	s261	s131	s131	s131	s268	s268	s143
x12	s94	s33	s29	s33	s94	s94	s86	s94	s29	s94	s86	s94	s48	s48	s86	s94	s29	s94	s86	s29	s86	s94	s29	s94	s29	s86	s48

x(t)	s(t)																											
	s135	s136	s137	s138	s139	s140	s141	s142	s143	s144	s145	s146	s147	s148	s149	s150	s151	s152	s153	s154	s155	s156	s157	s158	s159	s160	s161
x1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s132	s132	s40	s40	s40	s140	s140	s140	s140	s40	s145	s145	s145	s145	s53	s53	s53	s53	s53	s53	s57	s57	s57	s57	s57	s72	s290
x4	s265	s265	s95	s122	s122	s272	s141	s272	s272	s49	s278	s146	s278	s278	s149	s149	s149	s149	s54	s149	s155	s155	s155	s155	s58	s179	s291
x5	s5	s5	s42	s42	s42	s5	s5	s5	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s121	s43	s134	s134	s121	s23	s83	s121	s43	s106	s23	s83	s121	s43	s83	s97	s87	s89	s106	s134	s83	s97	s87	s89	s106	s161	s165
x7	s7	s7	s7	s7	s7	s24	s24	s7	s7	s24	s24	s24	s7	s7	s24	s24	s24	s24	s24	s7	s24	s24	s24	s24	s24	s13	s13
x8	s44	s44	s44	s44	s44	s25	s84	s44	s44	s31	s25	s84	s44	s44	s84	s90	s25	s90	s31	s44	s84	s90	s25	s90	s31	s63	s63
x9	s135	s135	s139	s139	s139	s273	s273	s142	s270	s257	s279	s279	s147	s147	s150	s150	s286	s286	s286	s120	s156	s156	s289	s289	s289	s160	s292
x10	s271	s271	s137	s138	s138	s274	s276	s277	s277	s144	s280	s282	s283	s283	s284	s285	s151	s285	s153	s154	s287	s288	s157	s288	s159	s184	s161
x11	s136	s136	s47	s47	s47	s275	s275	s143	s143	s125	s281	s281	s148	s148	s152	s152	s152	s152	s152	s56	s158	s158	s158	s158	s158	s75	s293
x12	s48	s48	s48	s48	s48	s29	s86	s48	s48	s33	s29	s86	s48	s48	s86	s94	s29	s94	s33	s48	s86	s94	s29	s94	s33	s67	s67

x(t)	s(t)																											
	s162	s163	s164	s165	s166	s167	s168	s169	s170	s171	s172	s173	s174	s175	s176	s177	s178	s179	s180	s181	s182	s183	s184	s185	s186	s187	s188
x1	s19	s1	s1	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s57	s163	s163	s171	s163	s163	s40	s40	s40	s171	s171	s171	s171	s175	s175	s175	s175	s72	s72	s72	s72	s72	s72	s185	s185	s185	s185
x4	s155	s294	s164	s172	s294	s294	s95	s122	s122	s301	s172	s301	s301	s307	s176	s307	s307	s179	s179	s179	s179	s73	s179	s316	s186	s316	s316
x5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s165	s23	s83	s165	s161	s62	s165	s165	s161	s23	s83	s161	s62	s23	s83	s161	s62	s83	s97	s87	s89	s106	s165	s23	s30	s161	s62
x7	s13	s24	s24	s13	s13	s13	s13	s13	s13	s24	s24	s13	s13	s24	s24	s13	s13	s24	s24	s24	s24	s24	s13	s24	s24	s13	s13
x8	s63	s25	s84	s63	s63	s63	s63	s63	s63	s25	s84	s63	s63	s25	s84	s63	s63	s84	s90	s25	s90	s31	s63	s25	s31	s63	s63
x9	s60	s295	s295	s299	s166	s166	s170	s170	s170	s302	s302	s173	s299	s308	s308	s177	s177	s180	s180	s315	s315	s315	s160	s317	s317	s187	s187
x10	s162	s296	s298	s165	s300	s300	s168	s169	s169	s303	s305	s306	s306	s309	s311	s312	s312	s313	s314	s181	s314	s183	s184	s318	s320	s321	s321
x11	s61	s297	s297	s174	s167	s167	s66	s66	s66	s304	s304	s174	s174	s310	s310	s178	s178	s182	s182	s182	s182	s182	s75	s319	s319	s188	s188
x12	s67	s29	s86	s67	s67	s67	s67	s67	s67	s29	s86	s67	s67	s29	s86	s67	s67	s86	s94	s29	s94	s33	s67	s29	s33	s67	s67

x(t)	s(t)																											
	s189	s190	s191	s192	s193	s194	s195	s196	s197	s198	s199	s200	s201	s202	s203	s204	s205	s206	s207	s208	s209	s210	s211	s212	s213	s214	s215
x1	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s322	s78	s78	s78	s78	s78	s78	s196	s196	s196	s196	s200	s200	s206	s200	s200	s40	s206	s206	s206	s206	s100	s348	s212	s212	s220	s212
x4	s323	s190	s190	s190	s190	s79	s190	s329	s197	s329	s329	s335	s201	s207	s335	s335	s95	s342	s207	s342	s342	s229	s349	s352	s213	s221	s352
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s165	s83	s97	s87	s89	s106	s165	s23	s83	s161	s62	s23	s83	s202	s211	s89	s202	s23	s83	s211	s89	s211	s214	s23	s83	s214	s211
x7	s13	s24	s24	s24	s24	s24	s13	s24	s24	s13	s13	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s63	s84	s90	s25	s90	s31	s63	s25	s84	s63	s63	s25	s84	s84	s90	s90	s84	s25	s84	s90	s90	s90	s90	s25	s84	s90	s90
x9	s324	s191	s191	s328	s328	s328	s80	s330	s330	s198	s198	s336	s336	s340	s203	s203	s219	s343	s343	s208	s340	s210	s350	s353	s353	s357	s215
x10	s189	s326	s327	s192	s327	s194	s195	s331	s333	s334	s334	s337	s339	s202	s341	s341	s205	s344	s346	s347	s347	s234	s211	s354	s356	s214	s358
x11	s325	s193	s193	s193	s193	s193	s81	s332	s332	s199	s199	s338	s338	s209	s204	s204	s93	s345	s345	s209	s209	s103	s351	s355	s355	s223	s216
x12	s67	s86	s94	s29	s94	s33	s67	s29	s86	s67	s67	s29	s86	s86	s94	s94	s86	s29	s86	s94	s94	s94	s94	s29	s86	s94	s94

x(t)	s(t)																											
	s216	s217	s218	s219	s220	s221	s222	s223	s224	s225	s226	s227	s228	s229	s230	s231	s232	s233	s234	s235	s236	s237	s238	s239	s240	s241	s242
x1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s1
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s212	s40	s40	s40	s220	s220	s220	s220	s40	s225	s225	s225	s225	s100	s100	s100	s100	s100	s100	s104	s104	s104	s104	s104	s110	s104	s110
x4	s352	s95	s122	s122	s359	s221	s359	s359	s95	s365	s226	s365	s365	s229	s229	s229	s229	s101	s229	s235	s235	s235	s235	s105	s242	s235	s242
x5	s5	s42	s42	s42	s5	s5	s5	s5	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s89	s214	s214	s211	s23	s83	s211	s89	s202	s23	s83	s211	s89	s83	s97	s87	s89	s202	s214	s83	s97	s87	s89	s106	s211	s214	s83
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s90	s90	s90	s90	s25	s84	s90	s90	s84	s25	s84	s90	s90	s84	s90	s25	s90	s84	s90	s84	s90	s25	s90	s31	s90	s90	s84
x9	s215	s219	s219	s219	s360	s360	s222	s357	s257	s366	s366	s227	s227	s230	s230	s373	s373	s373	s210	s236	s236	s376	s376	s376	s240	s107	s243
x10	s358	s217	s218	s218	s361	s363	s364	s364	s224	s367	s369	s370	s370	s371	s372	s231	s372	s233	s234	s374	s375	s237	s375	s239	s247	s241	s377
x11	s216	s93	s93	s93	s362	s362	s223	s223	s125	s368	s368	s228	s228	s232	s232	s232	s232	s232	s103	s238	s238	s238	s238	s238	s113	s108	s245
x12	s94	s94	s94	s94	s29	s86	s94	s94	s86	s29	s86	s94	s94	s86	s94	s29	s94	s86	s94	s86	s94	s29	s94	s33	s94	s94	s86

x(t)	s(t)																											
	s243	s244	s245	s246	s247	s248	s249	s250	s251	s252	s253	s254	s255	s256	s257	s258	s259	s260	s261	s262	s263	s264	s265	s266	s267	s268	s269
x1	s1	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s110	s110	s110	s110	s110	s34	s34	s34	s251	s251	s251	s251	s40	s40	s40	s127	s127	s127	s127	s127	s127	s127	s132	s132	s132	s132	s132
x4	s242	s242	s242	s111	s242	s114	s114	s114	s380	s252	s380	s380	s122	s122	s122	s258	s258	s258	s258	s128	s258	s258	s265	s265	s265	s265	s133
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s42	s42	s42	s42	s42	s42	s42	s5	s5	s5	s5	s5
x6	s97	s87	s89	s106	s214	s202	s214	s211	s23	s83	s121	s43	s202	s214	s211	s83	s97	s87	s89	s202	s214	s211	s83	s97	s87	s89	s202
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s7	s7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s90	s25	s90	s31	s90	s84	s90	s90	s25	s84	s44	s44	s84	s90	s90	s84	s90	s25	s90	s84	s90	s90	s84	s90	s25	s90	s84
x9	s243	s379	s379	s379	s240	s250	s250	s250	s381	s381	s253	s253	s257	s257	s257	s259	s259	s388	s388	s388	s264	s264	s266	s266	s391	s391	s391
x10	s378	s244	s378	s246	s247	s248	s249	s249	s382	s384	s385	s385	s255	s256	s256	s386	s387	s260	s387	s262	s263	s263	s389	s390	s267	s390	s269
x11	s245	s245	s245	s245	s113	s117	s117	s117	s383	s383	s254	s254	s125	s125	s125	s261	s261	s261	s261	s261	s131	s131	s268	s268	s268	s268	s268
x12	s94	s29	s94	s33	s94	s86	s94	s94	s29	s86	s48	s48	s86	s94	s94	s86	s94	s29	s94	s86	s94	s94	s86	s94	s29	s94	s86

x(t)	s(t)																											
	s270	s271	s272	s273	s274	s275	s276	s277	s278	s279	s280	s281	s282	s283	s284	s285	s286	s287	s288	s289	s290	s291	s292	s293	s294	s295	s296
x1	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s1	s1	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s140	s132	s140	s140	s140	s140	s140	s140	s145	s145	s145	s145	s145	s145	s53	s53	s53	s57	s57	s57	s290	s290	s290	s290	s163	s163	s163
x4	s272	s265	s272	s272	s272	s272	s141	s272	s278	s278	s278	s278	s146	s278	s149	s149	s149	s155	s155	s155	s398	s291	s398	s398	s294	s294	s294
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s121	s134	s83	s97	s87	s89	s202	s134	s83	s97	s87	s89	s202	s134	s202	s214	s211	s202	s214	s211	s23	s83	s161	s62	s83	s97	s87
x7	s7	s7	s24	s24	s24	s24	s24	s7	s24	s24	s24	s24	s24	s7	s24	s24	s24	s24	s24	s24	s24	s24	s13	s13	s24	s24	s24
x8	s44	s44	s84	s90	s25	s90	s84	s44	s84	s90	s25	s90	s84	s44	s84	s90	s90	s84	s90	s90	s25	s84	s63	s63	s84	s90	s25
x9	s270	s135	s273	s273	s394	s394	s394	s270	s279	s279	s397	s397	s397	s147	s286	s286	s286	s289	s289	s289	s399	s399	s292	s292	s295	s295	s406
x10	s277	s271	s392	s393	s274	s393	s276	s277	s395	s396	s280	s396	s282	s283	s284	s285	s285	s287	s288	s288	s400	s402	s403	s403	s404	s405	s296
x11	s143	s136	s275	s275	s275	s275	s275	s143	s281	s281	s281	s281	s281	s148	s152	s152	s152	s158	s158	s158	s401	s401	s293	s293	s297	s297	s297
x12	s48	s48	s86	s94	s29	s94	s86	s48	s86	s94	s29	s94	s86	s48	s86	s94	s94	s86	s94	s94	s29	s86	s67	s67	s86	s94	s29

x(t)	s(t)																											
	s297	s298	s299	s300	s301	s302	s303	s304	s305	s306	s307	s308	s309	s310	s311	s312	s313	s314	s315	s316	s317	s318	s319	s320	s321	s322	s323
x1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s163	s163	s171	s163	s171	s171	s171	s171	s171	s171	s175	s175	s175	s175	s175	s175	s72	s72	s72	s185	s185	s185	s185	s185	s185	s322	s322
x4	s294	s164	s301	s294	s301	s301	s301	s301	s172	s301	s307	s307	s307	s307	s176	s307	s179	s179	s179	s316	s316	s316	s316	s186	s316	s416	s323
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s89	s202	s161	s165	s83	s97	s87	s89	s202	s165	s83	s97	s87	s89	s202	s165	s202	s214	s211	s83	s97	s87	s89	s106	s165	s23	s83
x7	s24	s24	s13	s13	s24	s24	s24	s24	s24	s13	s24	s24	s24	s24	s24	s13	s24	s24	s24	s24	s24	s24	s24	s24	s13	s24	s24
x8	s90	s84	s63	s63	s84	s90	s25	s90	s84	s63	s84	s90	s25	s90	s84	s63	s84	s90	s90	s84	s90	s25	s90	s31	s63	s25	s84
x9	s406	s406	s299	s166	s302	s302	s409	s409	s409	s299	s308	s308	s412	s412	s412	s177	s315	s315	s315	s317	s317	s415	s415	s415	s187	s417	s417
x10	s405	s298	s306	s300	s407	s408	s303	s408	s305	s306	s410	s411	s309	s411	s311	s312	s313	s314	s314	s413	s414	s318	s414	s320	s321	s418	s420
x11	s297	s297	s174	s167	s304	s304	s304	s304	s304	s174	s310	s310	s310	s310	s310	s178	s182	s182	s182	s319	s319	s319	s319	s319	s188	s419	s419
x12	s94	s86	s67	s67	s86	s94	s29	s94	s86	s67	s86	s94	s29	s94	s86	s67	s86	s94	s94	s86	s94	s29	s94	s33	s67	s29	s86

x(t)	s(t)																											
	s324	s325	s326	s327	s328	s329	s330	s331	s332	s333	s334	s335	s336	s337	s338	s339	s340	s341	s342	s343	s344	s345	s346	s347	s348	s349	s350
x1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s322	s322	s78	s78	s78	s196	s196	s196	s196	s196	s196	s200	s200	s200	s200	s200	s206	s200	s206	s206	s206	s206	s206	s206	s348	s348	s348
x4	s416	s416	s190	s190	s190	s329	s329	s329	s329	s197	s329	s335	s335	s335	s335	s201	s342	s335	s342	s342	s342	s342	s207	s342	s431	s349	s431
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s161	s62	s202	s214	s211	s83	s97	s87	s89	s202	s165	s83	s97	s87	s89	s202	s211	s214	s83	s97	s87	s89	s202	s214	s23	s83	s211
x7	s13	s13	s24	s24	s24	s24	s24	s24	s24	s24	s13	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s63	s63	s84	s90	s90	s84	s90	s25	s90	s84	s63	s84	s90	s25	s90	s84	s90	s90	s84	s90	s25	s90	s84	s90	s25	s84	s90
x9	s324	s324	s328	s328	s328	s330	s330	s424	s424	s424	s198	s336	s336	s427	s427	s427	s340	s203	s343	s343	s430	s430	s430	s340	s432	s432	s350
x10	s421	s421	s326	s327	s327	s422	s423	s331	s423	s333	s334	s425	s426	s337	s426	s339	s347	s341	s428	s429	s344	s429	s346	s347	s433	s435	s436
x11	s325	s325	s193	s193	s193	s332	s332	s332	s332	s332	s199	s338	s338	s338	s338	s338	s209	s204	s345	s345	s345	s345	s345	s209	s434	s434	s351
x12	s67	s67	s86	s94	s94	s86	s94	s29	s94	s86	s67	s86	s94	s29	s94	s86	s94	s94	s86	s94	s29	s94	s86	s94	s29	s86	s94

x(t)	s(t)																											
	s351	s352	s353	s354	s355	s356	s357	s358	s359	s360	s361	s362	s363	s364	s365	s366	s367	s368	s369	s370	s371	s372	s373	s374	s375	s376	s377
x1	s19	s1	s1	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s348	s212	s212	s212	s212	s212	s220	s212	s220	s220	s220	s220	s220	s220	s225	s225	s225	s225	s225	s225	s100	s100	s100	s104	s104	s104	s110
x4	s431	s352	s352	s352	s352	s213	s359	s352	s359	s359	s359	s359	s221	s359	s365	s365	s365	s365	s226	s365	s229	s229	s229	s235	s235	s235	s242
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s89	s83	s97	s87	s89	s202	s211	s214	s83	s97	s87	s89	s202	s214	s83	s97	s87	s89	s202	s214	s202	s214	s211	s202	s214	s211	s202
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s90	s84	s90	s25	s90	s84	s90	s90	s84	s90	s25	s90	s84	s90	s84	s90	s25	s90	s84	s90	s84	s90	s90	s84	s90	s90	s84
x9	s350	s353	s353	s439	s439	s439	s357	s215	s360	s360	s442	s442	s442	s357	s366	s366	s445	s445	s445	s227	s373	s373	s373	s376	s376	s376	s379
x10	s436	s437	s438	s354	s438	s356	s364	s358	s440	s441	s361	s441	s363	s364	s443	s444	s367	s444	s369	s370	s371	s372	s372	s374	s375	s375	s377
x11	s351	s355	s355	s355	s355	s355	s223	s216	s362	s362	s362	s362	s362	s223	s368	s368	s368	s368	s368	s228	s232	s232	s232	s238	s238	s238	s245
x12	s94	s86	s94	s29	s94	s86	s94	s94	s86	s94	s29	s94	s86	s94	s86	s94	s29	s94	s86	s94	s86	s94	s94	s86	s94	s94	s86

x(t)	s(t)																											
	s378	s379	s380	s381	s382	s383	s384	s385	s386	s387	s388	s389	s390	s391	s392	s393	s394	s395	s396	s397	s398	s399	s400	s401	s402	s403	s404
x1	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s110	s110	s251	s251	s251	s251	s251	s251	s127	s127	s127	s132	s132	s132	s140	s140	s140	s145	s145	s145	s290	s290	s290	s290	s290	s290	s163
x4	s242	s242	s380	s380	s380	s380	s252	s380	s258	s258	s258	s265	s265	s265	s272	s272	s272	s278	s278	s278	s398	s398	s398	s398	s291	s398	s294
x5	s5	s5	s5	s5	s5	s5	s5	s5	s42	s42	s42	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s214	s211	s83	s97	s87	s89	s202	s134	s202	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211	s83	s97	s87	s89	s202	s165	s202
x7	s24	s24	s24	s24	s24	s24	s24	s7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s13	s24
x8	s90	s90	s84	s90	s25	s90	s84	s44	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s25	s90	s84	s63	s84
x9	s379	s379	s381	s381	s448	s448	s448	s253	s388	s388	s388	s391	s391	s391	s394	s394	s394	s397	s397	s397	s399	s399	s451	s451	s451	s292	s406
x10	s378	s378	s446	s447	s382	s447	s384	s385	s386	s387	s387	s389	s390	s390	s392	s393	s393	s395	s396	s396	s449	s450	s400	s450	s402	s403	s404
x11	s245	s245	s383	s383	s383	s383	s383	s254	s261	s261	s261	s268	s268	s268	s275	s275	s275	s281	s281	s281	s401	s401	s401	s401	s401	s293	s297
x12	s94	s94	s86	s94	s29	s94	s86	s48	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s29	s94	s86	s67	s86

x(t)	s(t)																											
	s405	s406	s407	s408	s409	s410	s411	s412	s413	s414	s415	s416	s417	s418	s419	s420	s421	s422	s423	s424	s425	s426	s427	s428	s429	s430	s431
x1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s1
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2
x3	s163	s163	s171	s171	s171	s175	s175	s175	s185	s185	s185	s322	s322	s322	s322	s322	s322	s196	s196	s196	s200	s200	s200	s206	s206	s206	s348
x4	s294	s294	s301	s301	s301	s307	s307	s307	s316	s316	s316	s416	s416	s416	s416	s323	s416	s329	s329	s329	s335	s335	s335	s342	s342	s342	s431
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5
x6	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211	s83	s97	s87	s89	s202	s165	s202	s214	s211	s202	s214	s211	s202	s214	s211	s83
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s13	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24
x8	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s25	s90	s84	s63	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84
x9	s406	s406	s409	s409	s409	s412	s412	s412	s415	s415	s415	s417	s417	s454	s454	s454	s324	s424	s424	s424	s427	s427	s427	s430	s430	s430	s432
x10	s405	s405	s407	s408	s408	s410	s411	s411	s413	s414	s414	s452	s453	s418	s453	s420	s421	s422	s423	s423	s425	s426	s426	s428	s429	s429	s455
x11	s297	s297	s304	s304	s304	s310	s310	s310	s319	s319	s319	s419	s419	s419	s419	s419	s325	s332	s332	s332	s338	s338	s338	s345	s345	s345	s434
x12	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s29	s94	s86	s67	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86

x(t)	s(t)																											
	s432	s433	s434	s435	s436	s437	s438	s439	s440	s441	s442	s443	s444	s445	s446	s447	s448	s449	s450	s451	s452	s453	s454	s455	s456	s457		
x1	s1	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19	s19		
x2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2	s2		
x3	s348	s348	s348	s348	s348	s212	s212	s212	s220	s220	s220	s225	s225	s225	s251	s251	s251	s290	s290	s290	s322	s322	s322	s348	s348	s348		
x4	s431	s431	s431	s349	s431	s352	s352	s352	s359	s359	s359	s365	s365	s365	s380	s380	s380	s398	s398	s398	s416	s416	s416	s431	s431	s431		
x5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5	s5		
x6	s97	s87	s89	s202	s214	s202	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211	s202	s214	s211		
x7	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24	s24		
x8	s90	s25	s90	s84	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90	s84	s90	s90		
x9	s432	s457	s457	s457	s350	s439	s439	s439	s442	s442	s442	s445	s445	s445	s448	s448	s448	s451	s451	s451	s454	s454	s454	s457	s457	s457		
x10	s456	s433	s456	s435	s436	s437	s438	s438	s440	s441	s441	s443	s444	s444	s446	s447	s447	s449	s450	s450	s452	s453	s453	s455	s456	s456		
x11	s434	s434	s434	s434	s351	s355	s355	s355	s362	s362	s362	s368	s368	s368	s383	s383	s383	s401	s401	s401	s419	s419	s419	s434	s434	s434		
x12	s94	s29	s94	s86	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94	s86	s94	s94		

ПРИЛОЖЕНИЕ В.
ТАБЛИЦА ВОЗМОЖНЫХ ВЫЯВЛЕННЫХ СОСТОЯНИЙ ВО

S	ВВУ
s0	(1,1,1,1,1,1,1,1)
s1	(2,1,2,1,2,2,2,2)
s2	(2,1,2,2,2,2,2,2)
s3	(1,2,2,2,1,1,1,1)
s4	(1,2,1,2,1,1,1,1)
s5	(1,2,2,2,1,2,2,2)
s6	(1,1,1,2,1,1,1,1)
s7	(1,1,2,2,2,2,2,2)
s8	(1,1,1,2,2,2,2,2)
s9	(1,1,2,1,1,1,1,1)
s10	(1,1,2,2,1,1,1,1)
s11	(1,1,1,2,1,1,1,2)
s12	(2,1,2,3,2,1,2,1)
s13	(2,1,3,3,3,3,3,3)
s14	(2,1,2,3,3,3,3,3)
s15	(2,1,3,1,2,2,2,2)
s16	(2,1,2,3,2,2,2,2)
s17	(2,1,3,3,2,2,2,2)
s18	(2,1,2,3,2,1,2,3)
s19	(3,1,3,2,3,3,3,3)
s20	(2,1,2,3,2,1,2,2)
s21	(2,1,3,2,2,2,2,2)

S	ВВУ
s22	(1,3,2,3,1,1,1,1)
s23	(1,2,2,3,1,2,2,2)
s24	(1,2,3,3,3,3,3,3)
s25	(1,2,2,3,3,3,3,3)
s26	(1,2,3,2,1,1,1,1)
s27	(1,2,2,3,1,1,1,1)
s28	(1,2,3,3,1,1,1,1)
s29	(1,2,2,3,1,2,2,3)
s30	(1,2,1,3,1,2,1,2)
s31	(1,2,1,3,3,3,3,3)
s32	(1,2,1,3,1,1,1,1)
s33	(1,2,1,3,1,2,1,3)
s34	(1,3,3,3,1,2,2,2)
s35	(1,3,2,3,1,2,2,2)
s36	(1,2,3,2,1,2,2,2)
s37	(1,2,3,3,1,2,2,2)
s38	(1,1,1,3,1,1,1,2)
s39	(1,1,3,2,1,1,1,1)
s40	(1,3,3,3,2,2,2,2)
s41	(1,3,2,3,2,2,2,2)
s42	(1,3,3,3,2,3,3,3)
s43	(1,1,2,3,1,1,2,2)

S	ВВУ
s44	(1,1,2,3,3,3,3,3)
s45	(1,1,3,2,2,2,2,2)
s46	(1,1,2,3,2,2,2,2)
s47	(1,1,3,3,2,2,2,2)
s48	(1,1,2,3,1,1,2,3)
s49	(1,3,1,3,2,2,2,2)
s50	(1,1,1,3,2,2,2,2)
s51	(1,1,2,3,1,1,2,1)
s52	(1,1,2,3,1,1,1,1)
s53	(1,3,3,3,1,1,1,2)
s54	(1,3,1,3,1,1,1,2)
s55	(1,1,3,2,1,1,1,2)
s56	(1,1,3,3,1,1,1,2)
s57	(2,3,3,3,2,1,2,1)
s58	(2,3,2,3,2,1,2,1)
s59	(2,1,2,4,2,1,2,3)
s60	(2,1,4,3,2,1,2,1)
s61	(2,1,3,3,2,1,2,1)
s62	(2,1,3,4,2,1,3,3)
s63	(2,1,3,4,4,4,4,4)
s64	(2,1,4,3,3,3,3,3)
s65	(2,1,3,4,3,3,3,3)

S	ВВУ
s66	(2,1,4,4,3,3,3,3)
s67	(2,1,3,4,2,1,3,4)
s68	(2,1,2,4,3,3,3,3)
s69	(2,1,3,4,2,1,3,1)
s70	(2,1,3,4,2,2,2,2)
s71	(2,1,4,3,2,2,2,2)
s72	(2,4,4,4,2,1,2,3)
s73	(2,4,2,4,2,1,2,3)
s74	(2,1,4,3,2,1,2,3)
s75	(2,1,4,4,2,1,2,3)
s76	(3,1,3,4,3,1,3,2)
s77	(3,1,4,2,3,3,3,3)
s78	(2,3,3,3,2,1,2,2)
s79	(2,3,2,3,2,1,2,2)
s80	(2,1,4,3,2,1,2,2)
s81	(2,1,3,3,2,1,2,2)
s82	(2,1,3,4,2,1,3,2)
s83	(1,3,2,4,1,3,2,3)
s84	(1,3,2,4,4,4,4,4)
s85	(1,3,2,4,1,1,1,1)
s86	(1,3,2,4,1,3,2,4)
s87	(1,2,2,4,1,2,2,3)

S	ВВУ
s88	(1,2,4,3,1,2,2,2)
s89	(1,2,3,4,1,2,3,3)
s90	(1,2,3,4,4,4,4,4)
s91	(1,2,4,3,3,3,3,3)
s92	(1,2,3,4,3,3,3,3)
s93	(1,2,4,4,3,3,3,3)
s94	(1,2,3,4,1,2,3,4)
s95	(1,4,2,4,3,3,3,3)
s96	(1,2,2,4,3,3,3,3)
s97	(1,2,3,4,1,2,3,2)
s98	(1,2,3,4,1,1,1,1)
s99	(1,2,4,3,1,1,1,1)
s100	(1,4,4,4,1,2,2,3)
s101	(1,4,2,4,1,2,2,3)
s102	(1,2,4,3,1,2,2,3)
s103	(1,2,4,4,1,2,2,3)
s104	(1,3,3,3,1,2,1,2)
s105	(1,3,1,3,1,2,1,2)
s106	(1,2,1,4,1,2,1,3)
s107	(1,2,4,3,1,2,1,2)
s108	(1,2,3,3,1,2,1,2)
s109	(1,2,1,4,3,3,3,3)

S	ВВУ
s110	(1,4,4,4,1,2,1,3)
s111	(1,4,1,4,1,2,1,3)
s112	(1,2,4,3,1,2,1,3)
s113	(1,2,4,4,1,2,1,3)
s114	(1,4,3,4,1,2,2,2)
s115	(1,3,4,3,1,2,2,2)
s116	(1,3,3,4,1,2,2,2)
s117	(1,3,4,4,1,2,2,2)
s118	(1,3,2,4,1,2,2,2)
s119	(1,2,3,4,1,2,2,2)
s120	(1,1,4,3,1,1,1,2)
s121	(1,1,3,4,1,1,3,2)
s122	(1,4,3,4,2,2,2,2)
s123	(1,3,4,3,2,2,2,2)
s124	(1,3,3,4,2,2,2,2)
s125	(1,3,4,4,2,2,2,2)
s126	(1,3,2,4,2,2,2,2)
s127	(1,4,4,4,2,3,3,3)
s128	(1,4,3,4,2,3,3,3)
s129	(1,3,4,3,2,3,3,3)
s130	(1,3,3,4,2,3,3,3)
s131	(1,3,4,4,2,3,3,3)

S	ВВУ
s132	(1,3,3,3,1,1,2,2)
s133	(1,3,2,3,1,1,2,2)
s134	(1,1,2,4,1,1,2,3)
s135	(1,1,4,3,1,1,2,2)
s136	(1,1,3,3,1,1,2,2)
s137	(1,1,2,4,3,3,3,3)
s138	(1,1,3,4,2,2,2,2)
s139	(1,1,4,3,2,2,2,2)
s140	(1,4,4,4,1,1,2,3)
s141	(1,4,2,4,1,1,2,3)
s142	(1,1,4,3,1,1,2,3)
s143	(1,1,4,4,1,1,2,3)
s144	(1,3,1,4,2,2,2,2)
s145	(1,3,3,3,1,1,2,1)
s146	(1,3,2,3,1,1,2,1)
s147	(1,1,4,3,1,1,2,1)
s148	(1,1,3,3,1,1,2,1)
s149	(1,4,3,4,1,1,1,2)
s150	(1,3,4,3,1,1,1,2)
s151	(1,3,3,4,1,1,1,2)
s152	(1,3,4,4,1,1,1,2)
s153	(1,3,1,4,1,1,1,2)
s154	(1,1,3,4,1,1,1,2)
s155	(2,4,3,4,2,1,2,1)
s156	(2,3,4,3,2,1,2,1)
s157	(2,3,3,4,2,1,2,1)
s158	(2,3,4,4,2,1,2,1)
s159	(2,3,2,4,2,1,2,1)

S	ВВУ
s160	(2,1,5,4,2,1,2,3)
s161	(2,1,4,5,2,1,4,3)
s162	(2,1,3,4,2,1,2,1)
s163	(2,4,4,4,2,1,3,3)
s164	(2,4,3,4,2,1,3,3)
s165	(2,1,3,5,2,1,3,4)
s166	(2,1,5,4,2,1,3,3)
s167	(2,1,4,4,2,1,3,3)
s168	(2,1,3,5,4,4,4,4)
s169	(2,1,4,5,3,3,3,3)
s170	(2,1,5,4,3,3,3,3)
s171	(2,5,5,5,2,1,3,4)
s172	(2,5,3,5,2,1,3,4)
s173	(2,1,5,4,2,1,3,4)
s174	(2,1,5,5,2,1,3,4)
s175	(2,4,4,4,2,1,3,1)
s176	(2,4,3,4,2,1,3,1)
s177	(2,1,5,4,2,1,3,1)
s178	(2,1,4,4,2,1,3,1)
s179	(2,5,4,5,2,1,2,3)
s180	(2,4,5,4,2,1,2,3)
s181	(2,4,4,5,2,1,2,3)
s182	(2,4,5,5,2,1,2,3)
s183	(2,4,2,5,2,1,2,3)
s184	(2,1,4,5,2,1,2,3)
s185	(3,4,4,4,3,1,3,2)
s186	(3,4,3,4,3,1,3,2)
s187	(3,1,5,4,3,1,3,2)

S	ВВУ
s188	(3,1,4,4,3,1,3,2)
s189	(3,1,4,5,3,1,4,2)
s190	(2,4,3,4,2,1,2,2)
s191	(2,3,4,3,2,1,2,2)
s192	(2,3,3,4,2,1,2,2)
s193	(2,3,4,4,2,1,2,2)
s194	(2,3,2,4,2,1,2,2)
s195	(2,1,3,4,2,1,2,2)
s196	(2,4,4,4,2,1,3,2)
s197	(2,4,3,4,2,1,3,2)
s198	(2,1,5,4,2,1,3,2)
s199	(2,1,4,4,2,1,3,2)
s200	(1,4,4,4,1,3,2,3)
s201	(1,4,2,4,1,3,2,3)
s202	(1,3,2,5,1,3,2,4)
s203	(1,3,5,4,1,3,2,3)
s204	(1,3,4,4,1,3,2,3)
s205	(1,3,2,5,4,4,4,4)
s206	(1,5,5,5,1,3,2,4)
s207	(1,5,2,5,1,3,2,4)
s208	(1,3,5,4,1,3,2,4)
s209	(1,3,5,5,1,3,2,4)
s210	(1,2,5,4,1,2,2,3)
s211	(1,2,4,5,1,2,4,3)
s212	(1,4,4,4,1,2,3,3)
s213	(1,4,3,4,1,2,3,3)
s214	(1,2,3,5,1,2,3,4)
s215	(1,2,5,4,1,2,3,3)

S	ВВУ
s216	(1,2,4,4,1,2,3,3)
s217	(1,2,3,5,4,4,4,4)
s218	(1,2,4,5,3,3,3,3)
s219	(1,2,5,4,3,3,3,3)
s220	(1,5,5,5,1,2,3,4)
s221	(1,5,3,5,1,2,3,4)
s222	(1,2,5,4,1,2,3,4)
s223	(1,2,5,5,1,2,3,4)
s224	(1,4,2,5,3,3,3,3)
s225	(1,4,4,4,1,2,3,2)
s226	(1,4,3,4,1,2,3,2)
s227	(1,2,5,4,1,2,3,2)
s228	(1,2,4,4,1,2,3,2)
s229	(1,5,4,5,1,2,2,3)
s230	(1,4,5,4,1,2,2,3)
s231	(1,4,4,5,1,2,2,3)
s232	(1,4,5,5,1,2,2,3)
s233	(1,4,2,5,1,2,2,3)
s234	(1,2,4,5,1,2,2,3)
s235	(1,4,3,4,1,2,1,2)
s236	(1,3,4,3,1,2,1,2)
s237	(1,3,3,4,1,2,1,2)
s238	(1,3,4,4,1,2,1,2)
s239	(1,3,1,4,1,2,1,2)
s240	(1,2,5,4,1,2,1,3)
s241	(1,2,3,4,1,2,1,2)
s242	(1,5,4,5,1,2,1,3)
s243	(1,4,5,4,1,2,1,3)

S	ВВУ
s244	(1,4,4,5,1,2,1,3)
s245	(1,4,5,5,1,2,1,3)
s246	(1,4,1,5,1,2,1,3)
s247	(1,2,4,5,1,2,1,3)
s248	(1,4,3,5,1,2,2,2)
s249	(1,3,4,5,1,2,2,2)
s250	(1,3,5,4,1,2,2,2)
s251	(1,4,4,4,1,1,3,2)
s252	(1,4,3,4,1,1,3,2)
s253	(1,1,5,4,1,1,3,2)
s254	(1,1,4,4,1,1,3,2)
s255	(1,4,3,5,2,2,2,2)
s256	(1,3,4,5,2,2,2,2)
s257	(1,3,5,4,2,2,2,2)
s258	(1,5,4,5,2,3,3,3)
s259	(1,4,5,4,2,3,3,3)
s260	(1,4,4,5,2,3,3,3)
s261	(1,4,5,5,2,3,3,3)
s262	(1,4,3,5,2,3,3,3)
s263	(1,3,4,5,2,3,3,3)
s264	(1,3,5,4,2,3,3,3)
s265	(1,4,3,4,1,1,2,2)
s266	(1,3,4,3,1,1,2,2)
s267	(1,3,3,4,1,1,2,2)
s268	(1,3,4,4,1,1,2,2)
s269	(1,3,2,4,1,1,2,2)
s270	(1,1,5,4,1,1,2,3)
s271	(1,1,3,4,1,1,2,2)

S	ВВУ
s272	(1,5,4,5,1,1,2,3)
s273	(1,4,5,4,1,1,2,3)
s274	(1,4,4,5,1,1,2,3)
s275	(1,4,5,5,1,1,2,3)
s276	(1,4,2,5,1,1,2,3)
s277	(1,1,4,5,1,1,2,3)
s278	(1,4,3,4,1,1,2,1)
s279	(1,3,4,3,1,1,2,1)
s280	(1,3,3,4,1,1,2,1)
s281	(1,3,4,4,1,1,2,1)
s282	(1,3,2,4,1,1,2,1)
s283	(1,1,3,4,1,1,2,1)
s284	(1,4,3,5,1,1,1,2)
s285	(1,3,4,5,1,1,1,2)
s286	(1,3,5,4,1,1,1,2)
s287	(2,4,3,5,2,1,2,1)
s288	(2,3,4,5,2,1,2,1)
s289	(2,3,5,4,2,1,2,1)
s290	(2,5,5,5,2,1,4,3)
s291	(2,5,4,5,2,1,4,3)
s292	(2,1,6,5,2,1,4,3)
s293	(2,1,5,5,2,1,4,3)
s294	(2,5,4,5,2,1,3,3)
s295	(2,4,5,4,2,1,3,3)
s296	(2,4,4,5,2,1,3,3)
s297	(2,4,5,5,2,1,3,3)
s298	(2,4,3,5,2,1,3,3)

S	ВВУ
s299	(2,1,6,5,2,1,3,4)
s300	(2,1,4,5,2,1,3,3)
s301	(2,6,5,6,2,1,3,4)
s302	(2,5,6,5,2,1,3,4)
s303	(2,5,5,6,2,1,3,4)
s304	(2,5,6,6,2,1,3,4)
s305	(2,5,3,6,2,1,3,4)
s306	(2,1,5,6,2,1,3,4)
s307	(2,5,4,5,2,1,3,1)
s308	(2,4,5,4,2,1,3,1)
s309	(2,4,4,5,2,1,3,1)
s310	(2,4,5,5,2,1,3,1)
s311	(2,4,3,5,2,1,3,1)
s312	(2,1,4,5,2,1,3,1)
s313	(2,5,4,6,2,1,2,3)
s314	(2,4,5,6,2,1,2,3)
s315	(2,4,6,5,2,1,2,3)
s316	(3,5,4,5,3,1,3,2)
s317	(3,4,5,4,3,1,3,2)
s318	(3,4,4,5,3,1,3,2)
s319	(3,4,5,5,3,1,3,2)
s320	(3,4,3,5,3,1,3,2)
s321	(3,1,4,5,3,1,3,2)
s322	(3,5,5,5,3,1,4,2)
s323	(3,5,4,5,3,1,4,2)
s324	(3,1,6,5,3,1,4,2)
s325	(3,1,5,5,3,1,4,2)

S	ВВУ
s326	(2,4,3,5,2,1,2,2)
s327	(2,3,4,5,2,1,2,2)
s328	(2,3,5,4,2,1,2,2)
s329	(2,5,4,5,2,1,3,2)
s330	(2,4,5,4,2,1,3,2)
s331	(2,4,4,5,2,1,3,2)
s332	(2,4,5,5,2,1,3,2)
s333	(2,4,3,5,2,1,3,2)
s334	(2,1,4,5,2,1,3,2)
s335	(1,5,4,5,1,3,2,3)
s336	(1,4,5,4,1,3,2,3)
s337	(1,4,4,5,1,3,2,3)
s338	(1,4,5,5,1,3,2,3)
s339	(1,4,2,5,1,3,2,3)
s340	(1,3,6,5,1,3,2,4)
s341	(1,3,4,5,1,3,2,3)
s342	(1,6,5,6,1,3,2,4)
s343	(1,5,6,5,1,3,2,4)
s344	(1,5,5,6,1,3,2,4)
s345	(1,5,6,6,1,3,2,4)
s346	(1,5,2,6,1,3,2,4)
s347	(1,3,5,6,1,3,2,4)
s348	(1,5,5,5,1,2,4,3)
s349	(1,5,4,5,1,2,4,3)
s350	(1,2,6,5,1,2,4,3)
s351	(1,2,5,5,1,2,4,3)
s352	(1,5,4,5,1,2,3,3)

S	ВВУ
s353	(1,4,5,4,1,2,3,3)
s354	(1,4,4,5,1,2,3,3)
s355	(1,4,5,5,1,2,3,3)
s356	(1,4,3,5,1,2,3,3)
s357	(1,2,6,5,1,2,3,4)
s358	(1,2,4,5,1,2,3,3)
s359	(1,6,5,6,1,2,3,4)
s360	(1,5,6,5,1,2,3,4)
s361	(1,5,5,6,1,2,3,4)
s362	(1,5,6,6,1,2,3,4)
s363	(1,5,3,6,1,2,3,4)
s364	(1,2,5,6,1,2,3,4)
s365	(1,5,4,5,1,2,3,2)
s366	(1,4,5,4,1,2,3,2)
s367	(1,4,4,5,1,2,3,2)
s368	(1,4,5,5,1,2,3,2)
s369	(1,4,3,5,1,2,3,2)
s370	(1,2,4,5,1,2,3,2)
s371	(1,5,4,6,1,2,2,3)
s372	(1,4,5,6,1,2,2,3)
s373	(1,4,6,5,1,2,2,3)
s374	(1,4,3,5,1,2,1,2)
s375	(1,3,4,5,1,2,1,2)
s376	(1,3,5,4,1,2,1,2)
s377	(1,5,4,6,1,2,1,3)
s378	(1,4,5,6,1,2,1,3)
s379	(1,4,6,5,1,2,1,3)

S	ВВУ
s380	(1,5,4,5,1,1,3,2)
s381	(1,4,5,4,1,1,3,2)
s382	(1,4,4,5,1,1,3,2)
s383	(1,4,5,5,1,1,3,2)
s384	(1,4,3,5,1,1,3,2)
s385	(1,1,4,5,1,1,3,2)
s386	(1,5,4,6,2,3,3,3)
s387	(1,4,5,6,2,3,3,3)
s388	(1,4,6,5,2,3,3,3)
s389	(1,4,3,5,1,1,2,2)
s390	(1,3,4,5,1,1,2,2)
s391	(1,3,5,4,1,1,2,2)
s392	(1,5,4,6,1,1,2,3)
s393	(1,4,5,6,1,1,2,3)
s394	(1,4,6,5,1,1,2,3)
s395	(1,4,3,5,1,1,2,1)
s396	(1,3,4,5,1,1,2,1)
s397	(1,3,5,4,1,1,2,1)
s398	(2,6,5,6,2,1,4,3)
s399	(2,5,6,5,2,1,4,3)
s400	(2,5,5,6,2,1,4,3)
s401	(2,5,6,6,2,1,4,3)
s402	(2,5,4,6,2,1,4,3)
s403	(2,1,5,6,2,1,4,3)
s404	(2,5,4,6,2,1,3,3)
s405	(2,4,5,6,2,1,3,3)
s406	(2,4,6,5,2,1,3,3)

S	ВВУ
s407	(2,6,5,7,2,1,3,4)
s408	(2,5,6,7,2,1,3,4)
s409	(2,5,7,6,2,1,3,4)
s410	(2,5,4,6,2,1,3,1)
s411	(2,4,5,6,2,1,3,1)
s412	(2,4,6,5,2,1,3,1)
s413	(3,5,4,6,3,1,3,2)
s414	(3,4,5,6,3,1,3,2)
s415	(3,4,6,5,3,1,3,2)
s416	(3,6,5,6,3,1,4,2)
s417	(3,5,6,5,3,1,4,2)
s418	(3,5,5,6,3,1,4,2)
s419	(3,5,6,6,3,1,4,2)
s420	(3,5,4,6,3,1,4,2)
s421	(3,1,5,6,3,1,4,2)
s422	(2,5,4,6,2,1,3,2)
s423	(2,4,5,6,2,1,3,2)
s424	(2,4,6,5,2,1,3,2)
s425	(1,5,4,6,1,3,2,3)
s426	(1,4,5,6,1,3,2,3)
s427	(1,4,6,5,1,3,2,3)
s428	(1,6,5,7,1,3,2,4)
s429	(1,5,6,7,1,3,2,4)
s430	(1,5,7,6,1,3,2,4)
s431	(1,6,5,6,1,2,4,3)
s432	(1,5,6,5,1,2,4,3)
s433	(1,5,5,6,1,2,4,3)

S	ВВУ
s434	(1,5,6,6,1,2,4,3)
s435	(1,5,4,6,1,2,4,3)
s436	(1,2,5,6,1,2,4,3)
s437	(1,5,4,6,1,2,3,3)
s438	(1,4,5,6,1,2,3,3)
s439	(1,4,6,5,1,2,3,3)
s440	(1,6,5,7,1,2,3,4)
s441	(1,5,6,7,1,2,3,4)
s442	(1,5,7,6,1,2,3,4)
s443	(1,5,4,6,1,2,3,2)
s444	(1,4,5,6,1,2,3,2)
s445	(1,4,6,5,1,2,3,2)
s446	(1,5,4,6,1,1,3,2)
s447	(1,4,5,6,1,1,3,2)
s448	(1,4,6,5,1,1,3,2)
s449	(2,6,5,7,2,1,4,3)
s450	(2,5,6,7,2,1,4,3)
s451	(2,5,7,6,2,1,4,3)
s452	(3,6,5,7,3,1,4,2)
s453	(3,5,6,7,3,1,4,2)
s454	(3,5,7,6,3,1,4,2)
s455	(1,6,5,7,1,2,4,3)
s456	(1,5,6,7,1,2,4,3)
s457	(1,5,7,6,1,2,4,3)

ПРИЛОЖЕНИЕ Г. ПОДСЧЕТ КОЛИЧЕСТВА И ПРОЦЕНТА ИЗВЕСТНЫХ ВВУ ДЛЯ ИССЛЕДУЕМЫХ ПЭВМ

Форматы файлов	ПЭВМ № 1			ПЭВМ № 2			ПЭВМ № 3			ПЭВМ № 4			ПЭВМ № 5		
	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ
Документы (txt, odt, pdf, rtf, djvu, djv, dot, doc(x), xls(x), ppt(x))	2 855	2 838	**99,40 %**	4 605	4 581	**99,47 %**	2 606	2 512	**96,39 %**	2 919	2 901	**99,38 %**	2 196	2 173	**98,95 %**
Исполняемые файлы (exe, dll, com, sys)	88	88	**100 %**	3 833	3 821	**99,68 %**	1 575	1 572	**99,81 %**	292	292	**100 %**	143	141	**98,60 %**
Изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff)	2 393	2 393	**100 %**	40 786	40 616	**99,58 %**	16 787	16 708	**99,53 %**	1 752	1 752	**100 %**	1 041	1 041	**100 %**
Видеофайлы (mov, avi, mpeg, mpg, mkv, mp4)	3	3	**100 %**	736	733	**99,59 %**	46	46	**100 %**	67	67	**100 %**	12	12	**100 %**
Звуковые файлы (mp3, wav)	30	30	**100 %**	1 193	1 180	**98.91 %**	768	766	**99,73 %**	724	724	**100 %**	187	187	**100 %**
Архивы (zip, rar, 7z)	105	103	**98,09 %**	371	368	**99,19 %**	145	142	**97,93 %**	279	277	**99,28 %**	199	193	**96,98 %**
Интернет-файлы (url, eml)	4	4	**100 %**	10	10	**100 %**	36	36	**100 %**	24	24	**100 %**	13	13	**100 %**
Итого:	5 478	5 459	**99,65 %**	51 534	51 309	**99,56 %**	21 963	21 782	**99,18 %**	6 057	6 037	**99,66 %**	3 791	3 760	**99,18 %**

Форматы файлов	ПЭВМ № 6			ПЭВМ № 7			ПЭВМ № 8			ПЭВМ № 9			ПЭВМ № 10		
	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ
Документы (txt, odt, pdf, rtf, djvu, djv, dot, doc(x), xls(x), ppt(x))	2 688	2 665	**99,14 %**	2 485	2 444	**98,35 %**	1 541	1 532	**99,42 %**	2 574	2 568	**99,76 %**	3 618	3 613	**99,86 %**
Исполняемые файлы (exe, dll, com, sys)	244	238	**97,54 %**	138	138	**100 %**	56	56	**100 %**	70	70	**100 %**	99	99	**100 %**
Изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff)	1 092	1 080	**98,90 %**	14 883	14 856	**99,82 %**	1 911	1 908	**99,84 %**	2 274	2 263	**99,51 %**	461	461	**100 %**
Видеофайлы (mov, avi, mpeg, mpg, mkv, mp4)	199	199	**100 %**	239	239	**100 %**	88	88	**100 %**	214	214	**100 %**	22	22	**100 %**
Звуковые файлы (mp3, wav)	510	510	**100 %**	1 374	1 374	**100 %**	195	195	**100 %**	1 183	1 181	**99,83 %**	12	12	**100 %**
Архивы (zip, rar, 7z)	267	261	**97,75 %**	437	425	**97,25 %**	15	15	**100 %**	284	280	**98,59 %**	158	157	**98,36 %**
Интернет-файлы (url, eml)	27	27	**100 %**	32	32	**100 %**	9	9	**100 %**	18	18	**100 %**	1	1	**100 %**
Итого:	5 027	4 980	**99,06 %**	19 588	19 508	**99,59 %**	3 815	3 803	**99,68 %**	6 617	6 594	**99,65 %**	4 371	4 365	**99,86 %**

Форматы файлов	ПЭВМ № 11			ПЭВМ № 12			ПЭВМ № 13			ПЭВМ № 14			ПЭВМ № 15		
	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ
Документы (txt, odt, pdf, rtf, djvu, djv, dot, doc(x), xls(x), ppt(x))	4 848	4 842	**99,87 %**	2 341	2 330	**99,53 %**	2 385	2 376	**99,62 %**	3 184	3 174	**99,69 %**	1 459	1 450	**99,38 %**
Исполняемые файлы (exe, dll, com, sys)	162	162	**100 %**	80	80	**100 %**	115	107	**93,04 %**	84	84	**100 %**	23	23	**100 %**
Изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff)	66	66	**100 %**	1 167	1 167	**100 %**	7 223	7 199	**99,66 %**	2 788	2 765	**99,18 %**	384	384	**100 %**
Видеофайлы (mov, avi, mpeg, mpg, mkv, mp4)	5	5	**100 %**	185	185	**100 %**	274	269	**98,18 %**	228	222	**97,37 %**	90	90	**100 %**
Звуковые файлы (mp3, wav)	3	3	**100 %**	348	340	**97,70 %**	176	172	**97,72 %**	913	900	**98,58 %**	62	62	**100 %**
Архивы (zip, rar, 7z)	154	152	**98,70 %**	87	87	**100 %**	73	73	**100 %**	22	22	**100 %**	10	10	**100 %**
Интернет-файлы (url, eml)	13	13	**100 %**	10	10	**100 %**	46	46	**100 %**	27	27	**100 %**	8	8	**100 %**
Итого:	5 248	5 240	**99,85 %**	4 218	4 199	**99,55 %**	10 292	10 242	**99,51 %**	7 246	7 194	**99,28 %**	2 036	2 027	**99,56 %**

Форматы файлов	ПЭВМ № 16			ПЭВМ № 17			ПЭВМ № 18			ПЭВМ № 19			ПЭВМ № 20		
	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ	Количество экспериментально полученным ВВУ	Количество известных ВВУ	Процент известных ВВУ
Документы (txt, odt, pdf, rtf, djvu, djv, dot, doc(x), xls(x), ppt(x))	2 437	2 437	**100 %**	2 872	2 825	**98,36%**	1 727	1 722	**99,71%**	1 996	1 898	**99,64%**	2 985	2 980	**99,83%**
Исполняемые файлы (exe, dll, com, sys)	40	40	**100 %**	66	66	**100 %**	19	19	**100 %**	1 442	1 435	**99,51 %**	5	5	**100 %**
Изображения (jpg, jpeg, gif, bmp, png, ico, tif, tiff)	131	111	**84,73 %**	7 570	7 532	**99,50 %**	215	215	**100 %**	4 731	4 715	**99,66 %**	8 691	8 685	**99,93 %**
Видеофайлы (mov, avi, mpeg, mpg, mkv, mp4)	19	19	**100 %**	45	41	**91,11 %**	12	11	**91,66 %**	136	136	**100 %**	257	257	**100 %**
Звуковые файлы (mp3, wav)	36	36	**100 %**	299	296	**98,99 %**	83	79	**95,18 %**	249	249	**100 %**	592	592	**100 %**
Архивы (zip, rar, 7z)	35	35	**100 %**	95	95	**100 %**	48	44	**91,66 %**	106	103	**97,16 %**	55	55	**100 %**
Интернет-файлы (url, eml)	3	3	**100 %**	10	10	**100 %**	8	7	**87,5 %**	37	37	**100 %**	36	36	**100 %**
Итого:	2 701	2 681	**99,25 %**	10 957	10 865	**99,16 %**	2 112	2 097	**99,29 %**	8 697	8 573	**98,57 %**	12 621	12 610	**99,91 %**

ПРИЛОЖЕНИЕ Д. ЛИСТИНГ ФУНКЦИИ TRANSITION_DIGRAPH.M

```
clear
A = importdata('A.mat'); %таблица смежности
B = importdata('B.mat'); %список состояний
G2=digraph(A,B);
for z=1:457
    x=["s"+z];
    t = bfsearch(G2, x, {'edgetonew','edgetodiscovered',
'edgetofinished'});
    M(1,1)=1;
    for i=1:length(t.EdgeIndex)
        M(i,2)=G2.Edges{t.EdgeIndex(i),2};
        t{i,5}=M(i,2);
    end
    EdgeTable = table([t.Edge],t{:,5},t{:,4},t{:,1},'VariableNames',
{'EndNodes' 'Code' 'EdgeIndex' 'Event'});
    G3=digraph(EdgeTable);
    [e,edge_indices] = dfsearch(G3, x, 'edgetodiscovered', 'Restart', true);
    if length(edge_indices)==1
        G = G3;
    else
        G = rmedge(G3,e(:,1),e(:,2));
    end
    p=plot(G,'NodeLabel',G.Nodes.Name,'EdgeLabel',G.Edges.Code);
    layout(p,'force','Iterations',20);
    layout(p,'layered','Direction','right','Sources',[1]);
    [paths,edgepaths] = allpaths(G,x,'s0','MaxNumPaths',10);
    l=length(edgepaths);
    for i=1:l
        l2=length(edgepaths{i,1});
        for j=1:l2
            fop=edgepaths{i,1}(1,j);
            SS{z,1}(i,j)=G.Edges.Code(fop);
        end
    end
end
```

ПРИЛОЖЕНИЕ Е. ЛИСТИНГ ФУНКЦИИ RETRO.M

```
clear
SS = importdata('SS.mat');
C = importdata('C.mat');
% исходные условия: файловая запись
% открытие файла
[FileName,PathName] = uigetfile('*.*','Выберете файл для открытия');
file = fopen([PathName,FileName],'r');
dec(1,:)=fread(file);
fclose(file);
str=char(dec(1,:));
ind=strfind(str,'FILE0');
file = fopen([PathName,FileName],'r');
clear FileName PathName ans

% определение ВО
for k=1:length(ind)
    clear T
    attr=dec(1,20+ind(1,k));
    SI=dec(1,attr+ind(1,k));
    if SI==16
        fseek(file, attr+24+ind(1,k)-1, 'bof');
        T(1,:)=fread(file, 'uint64', 'l');
        j=1;
        for i=1:4
        t(k,i)=T(1,j);
        j=j+1;
        end
    end
    clear T
    lattr=dec(1,attr+ind(1,k)+4);
    FN=dec(1,attr+ind(1,k)+lattr);
    if FN==48
        fseek(file, attr+ind(1,k)+lattr+32-1, 'bof');
        T(1,:)=fread(file, 'uint64', 'l');
        j=1;
        for i=5:8
        t(k,i)=T(1,j);
        j=j+1;
        end
    end
    % чтение имени
    lname=dec(1,attr+ind(1,k)+lattr+88);
    fseek(file, attr+ind(1,k)+lattr+90-1, 'bof');
    name(1,:)=fread(file, lname, 'ubit16');
    namestr{k,:}=char(name(1,:));
    clear name

    lattr2=dec(1,attr+ind(1,k)+lattr+4);
    DATA=dec(1,attr+ind(1,k)+lattr+lattr2);

    lattr3=0;
    if DATA~=128
        lattr3=dec(1,attr+ind(1,k)+lattr+4+lattr2);
        DATA=dec(1,attr+ind(1,k)+lattr+lattr2+lattr3);
    end

    % чтение размера файла
    fseek(file, attr+ind(1,k)+lattr+lattr2+lattr3+48-1, 'bof');
    size(1,:)=fread(file, 4, 'uint32');
```

```
    sizestr{k,:}=size(1,1);
    clear size
end
fclose(file);

clear T ans dec file i ind j k lname str FN SI lattr attr

%проверка округления
for i=1:length(t(:,1))
    tround(i,8)=0;
    for j=1:length(t(i,:))
        if (t(i,j)-10000*round(t(i,j)/10000)==0)
            tround(i,j)=1;
        end
    end
end
clear i j

%определение ВВУ
for k=1:length(t(:,1))
    j=1;
    for i=1:8
        date=datevec((t(k,i)/10000000/60/60/24)+584755);

Date{k,i}=datestr([date(1,1),date(1,2),date(1,3),date(1,4),date(1,5),date(1,6)],
'dd.mm.yyyy HH:MM:SS');
    end
    tmin=min(t(k,:));
    while tmin>1
        for i=1:8
            t(k,i)=t(k,i)-tmin;
            if t(k,i)<10000000
                K(k,i)=j;
                t(k,i)=nan;
                D{k,j}=Date{k,i};
            end
        end
        tmin=min(t(k,:));
        j=j+1;
    end

VVU=sprintf('(%d,%d,%d,%d,%d,%d,%d,%d)',K(k,1),K(k,2),K(k,4),K(k,3),K(k,5),K(k,6
),K(k,8),K(k,7));
    i=1;
     S(k,1)=0;
    while strcmpi(VVU,C{i,1})==0
        i=i+1;
        if i==458
            S(k,1)=-1;
            break
        end
        S(k,1)=i-1;
    end
end
clear date i j t tmin

%уточняющие вопросы
choice = questdlg('Версия ОС на исследуемом носителе', 'Уточнение версии ОС',
'Windows XP', 'Windows 7/8/10', 'Неизвестно', 'Неизвестно');
switch choice
    case 'Windows XP'
        OS = 1;
    case 'Windows 7/8/10'
```

```
        OS = 2;
    case 'Неизвестно'
        OS = 0;
end

if OS==2
    choice = questdlg('Версия ОС на исследуемом носителе', 'Уточнение версии
ОС', 'Windows 7', 'Windows 8', 'Windows 10', 'Windows 10');
    switch choice
        case 'Windows 7'
            OS = 2;
        case 'Windows 8'
            OS = 3;
        case 'Windows 10'
            OS = 4;
    end
end

choice = questdlg('Временная отметка A', 'Состояние временной отметки A', 'вкл.
A', 'выкл. A', 'Неизвестно', 'Неизвестно');
switch choice
    case 'вкл. A'
        SIA = 2;
    case 'выкл. A'
        SIA = 1;
    case 'Неизвестно'
        SIA = 0;
end

clear choice

%Обработка ВВУ
for i=1:k
    if S(i,1)>0
        V=length(SS{S(i,1),1}(:,1));
        if SIA==2
            t=1;
            for j=1:V
                for m=1:length(SS{S(i,1),1}(1,:))
                    if SS{S(i,1),1}(j,m)==4
                        Md(1,t)=j;
                        t=t+1;
                    end
                end
            end
            Md = unique(Md);
            for j=1:length(Md)
                SS{S(i,1),1}(Md(length(Md)-j+1),:)=[];
            end
        end
        if SIA==1
            t=1;
            for j=1:V
                for m=1:length(SS{S(i,1),1}(1,:))
                    if SS{S(i,1),1}(j,m)==3 || SS{S(i,1),1}(j,m)==9 ||
SS{S(i,1),1}(j,m)==11
                        Md(1,t)=j;
                        t=t+1;
                    end
                end
            end
            Md = unique(Md);
            for j=1:length(Md)
```

```
                SS{S(i,1),1}(Md(length(Md)-j+1),:)=[];
            end
        end
        clear j m Md
        V=length(SS{S(i,1),1}(:,1));
        var=1;
        STR{1,1}=sprintf('%s %s', 'Имя файла:', namestr{k,1}(1,:));
        STR{1,2}='Файловая операция';
        STR{1,3}='Временная отметка';
        STR{2,1}=sprintf('%s %d %s', 'Размер файла:', sizestr{k,1}(1,:),
'байт');
        STR{3,1}=sprintf('%s%s', 'ВВУ = ', VVU);
        SS{S(i,1),1}(1,length(SS{S(i,1),1}(1,:))+1)=0;
        for j=1:V
            kmax=length(D);
            var=var+1;
            STR{var,2}=sprintf('%s %d', 'Вариант', j);
            m=1;
            x4=0; x5=0; x6=0;
            while SS{S(i,1),1}(j,m)>0
                x2=0; x3=0;
                if SS{S(i,1),1}(j,m)==1
                   x='файл был создан методом копирования';
                   x2=1; x4=1;
                end
                if SS{S(i,1),1}(j,m)==2
                   x='файл был создан методом копирования или извлечен из
архива';
                   x2=1; x4=1; x5=1; x6=1;
                end
                if SS{S(i,1),1}(j,m)==3
                   x='файл редактировался (вкл. A)';
                   x4=1; x5=1; x6=1;
                end
                if SS{S(i,1),1}(j,m)==4
                   x='файл редактировался (выкл. A)';
                   x5=1;
                end
                if SS{S(i,1),1}(j,m)==5
                   x='файл редактировался в пакете Microsoft Office';
                   x4=1; x5=1; x6=1;
                end
                if SS{S(i,1),1}(j,m)==6
                   x='файл был перемещен/переименован';
                   x3=1;
                end
                if SS{S(i,1),1}(j,m)==7
                   x='файл был перемещен/переименован из файловой системы FAT';
                   x4=1; x5=1;
                end
                if SS{S(i,1),1}(j,m)==8
                   x='файл был извлечен из архива'; x5=1;

                end
                if SS{S(i,1),1}(j,m)==9
                   x='файл копировался или был разархивирован или был запу-
щен/открыт';
                   if x4==1
                       x4=2;
                   end
                end
                if SS{S(i,1),1}(j,m)==10
                   x='у файла были изменены атрибуты или файл запускал-
ся/открывался';
```

```
                    if x5==1
                        x5=2;
                    end
                end
                if SS{S(i,1),1}(j,m)==11
                    x='файл запускался/открывался (вкл. A)';
                    x4=1; x5=1;
                    if x6==1
                        x6=2;
                    end
                end
                if SS{S(i,1),1}(j,m)==12
                    x='файл был перемещен/переименован';
                    x3=1;
                end
                var=var+1;
                STR{var,2}=[x];
                m=m+1;
                if (kmax>1 && x3==0 && x4~=2 && x5~=2 && x6~=2) || (kmax==1 &&
x2==1)
                    STR{var,3}=[D{1,kmax}];
                    kmax=kmax-1;
                else
                    STR{var,3}='н/д';
                end
                if kmax==length(D) && x3==1
                    STR{var,3}=[D{1,kmax}];
                    kmax=kmax-1;
                end
            end
            if K(1,2)==1 || K(1,6)==1
                x='исходный файл редактировался';
                var=var+1;
                STR{var,2}=[x];
                STR{var,3}=[D{1,1}];
            end
            if K(1,1)==1 && x2==0
                x='создана файловая запись';
                var=var+1;
                STR{var,2}=[x];
                STR{var,3}=[D{1,1}];
            end

        end
    elseif S(i,1)==0
        STR{1,2}=sprintf('%s%s%s%s', 'Файловая запись файла: "', '6.doc', '"
создана ', D{1,1});
        var=1;
    else
        STR{1,2}=sprintf('%s%s%s', 'Для файла "', namestr{k,1}(1,:), '"
неизвестный вектор временных уровней');
        STR{2,2}=VVU;
        var=1;
    end

    %Вопросы по внутренним ВО
        str2=sprintf('%s%s%s', 'Имеются ли внутренние временные отметки для
файла "', namestr{k,1}(1,:), '" ?');
        choice = questdlg(str2, 'Внутренние временные отметки', 'Да', 'Нет',
'Нет');
        switch choice
            case 'Да'
                Dop = 1;
```

```
            case 'Нет'
                Dop = 2;
        end
        if Dop==1
            prompt={'Временная отметка создания содержимого','Временная отметка
изменения','Временная отметка печати'};
            name='Введите значения внутренних временных отметок в формате
ДД.ММ.ГГГГ ЧЧ:ММ';
            answer=inputdlg(prompt,name,[1 100; 1 100; 1 100]);
            t1 = datetime(Date{1,1});
            t2 = datetime(Date{1,2});
            t3 = datetime(answer{1,1});
            t4 = datetime(answer{2,1});
            t5 = datetime(answer{3,1});
            T0=minus(t1,t3);
            T1=minus(t2,t4);
            T2=minus(t3,t5);
            T3=minus(t3,t4);
            x1=''; x2=''; x3=''; x4=''; x5='';
            if T0<-60
                x1='У файла изменялась временная отметка создания с помощью
специальных утилит';
            end
            if T0>60
                x2='Файл является копией';
            end
            if T1>60
                x3='У файла изменялась временная отметка изменения с помощью
специальных утилит';
            end
            if T2>60
                x4='Файл является копией';
            end
            if T3==0
                x5='Файл был сохранен с помощью команды «Сохранить как…»';
            end
            var=var+2;
            STR{var,2}=sprintf('%s', 'Анализ внутренних временных отметок
выявил:');
            var=var+1;
            if length(x1)>1
                STR{var,2}=sprintf('! %s', x1);
                var=var+1;
            end
            if length(x2)>1
                STR{var,2}=sprintf('! %s', x2);
                var=var+1;
                STR{var,2}=sprintf('Время создания содержимого:');
                STR{var,3}=datestr(t3, 'dd.mm.yyyy HH:MM:SS');
            end
            if length(x3)>1
                STR{var,2}=sprintf('! %s', x3);
                var=var+1;
            end
            if length(x4)>1
                STR{var,2}=sprintf('! %s', x4);
                var=var+1;
                STR{var,2}=sprintf('Время создания содержимого:');
                STR{var,3}=datestr(t5, 'dd.mm.yyyy HH:MM:SS');
            end
            if length(x5)>1
                STR{var,2}=sprintf('! %s', x5);
                var=var+1;
            end
```

```
            clear T0 T1 T2 T3 x1 x2 x3 x4 x5
        end
        if tround(1,1)==1
            var=var+2;
            STR{var,2}=sprintf('%s', '! Округлена временная отметка создания');
        end
        if tround(1,2)==1
            var=var+1;
            STR{var,2}=sprintf('%s', '! Округлена временная отметка изменения');
        end
        var=var+2;
        STR{var,2}=sprintf('%s', 'Не забудьте проверить наличие ярлыков и
записей в системном реестре');

%Запись результатов анализа в файл

    T=cell2table(STR);
    str={int2str(i),'.csv'};
    filename=strjoin(str,'');
    writetable(T,filename,'Delimiter',',')
    clear STR T V i m str x
end
```

ПРИЛОЖЕНИЕ Ж. КОПИИ АКТОВ ОБ ИСПОЛЬЗОВАНИИ РЕЗУЛЬТАТОВ ДИССЕРТАЦИОННОГО ИССЛЕДОВАНИЯ

МИНИСТЕРСТВО ПРОМЫШЛЕННОСТИ И ТОРГОВЛИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ УНИТАРНОЕ ПРЕДПРИЯТИЕ
«НАУЧНО – ПРОИЗВОДСТВЕННОЕ ПРЕДПРИЯТИЕ
«ГАММА»
ЕКАТЕРИНБУРГСКИЙ НАУЧНО-ТЕХНИЧЕСКИЙ ЦЕНТР

620078, г. Екатеринбург, ул. Студенческая, д. 51
620078, г. Екатеринбург, а/я 65

Тел / факс: (343) 362-40-04
E-mail: info@gammaural.ru,

«10» сентября 2021 г. № 884

АКТ
Об использовании результатов диссертационного исследования
Князевой Наталии Сергеевны

Мы, представители Екатеринбургского научно-технического центра ФГУП «НПП «Гамма» первый заместитель директора Аникин Д.В. и начальник отдела информационных технологий Елаков Д.О. составили настоящий акт об использовании результатов диссертационного исследования Князевой Н.С. при проведении мониторинга информационной безопасности объектов критической информационной инфраструктуры, в части касающейся восстановления последовательности операций, совершенных пользователем над файлами.

Разработанная Князевой Н.С. методика восстановления последовательности файловых операций при проведении расследования компьютерных инцидентов позволила выявить факты распространения файлов и ознакомления с их содержимым.

Первый заместитель директора Д.В. Аникин

Начальник отдела информационных технологий Д.О. Елаков

Уральский Центр Систем Безопасности
Технологии защиты бизнеса.
Аудит. Проектирование.
Внедрение. Сопровождение.

620100
г. Екатеринбург
ул. Ткачей, д. 6

тел.: +7(343) 379-98-34
факс: +7(343) 382-05-63

info@ussc.ru
www.USSC.ru

УТВЕРЖДАЮ
Генеральный директор ООО «УЦСБ»
к.т.н. В.В. Богданов
17 сентября 2021 г.

АКТ

об использовании результатов диссертационного исследования
Князевой Наталии Сергеевны

г. Екатеринбург 17 сентября 2021 г.

Мы, нижеподписавшиеся, представители ООО «Уральский центр систем безопасности» Директор Ёркин Антон Борисовичи и Заместитель генерального директора по научно-технической работе Домуховский Николай Анатольевич составили настоящий акт в том, что методика восстановления последовательности файловых операций, разработанная в рамках диссертационного исследования Князевой Н.С., используется при проведении исследования компьютерных инцидентов. Функция, реализующая методику, позволяет значительно сокращать время, затрачиваемое специалистами при проведении анализа временных отметок файлов.

Директор А.Б. Ёркин

Зам. ген. директора по научно-технической работе Н.А. Домуховский

Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования «Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина» (УрФУ)

ул. Мира, 19, Екатеринбург, 620002,
факс: +7 (343) 375-97-78; тел.: +7 (343) 374-38-84
контакт-центр: +7 (343) 375-44-44, 8-800-100-50-44 (звонок бесплатный)
e-mail: rector@urfu.ru, www.urfu.ru
ОКПО 02069208, ОГРН 1026604939855, ИНН/КПП 6660003190/667001001

_____ № _____

На № _____ от _____

УТВЕРЖДАЮ
Проректор по науке
Германенко А.В.
«__» _____ 2021 года

АКТ
Об использовании результатов диссертационного исследования Князевой Наталии Сергеевны

Мы, нижеподписавшиеся, представители Уральского Федерального университета имени первого Президента России Б.Н. Ельцина (УрФУ) директор Института Радиоэлектроники и Информационных технологий-РТФ (ИТИТ-РтФ) Илья Николаевич Обабков и директор учебно-научного центра «Информационная безопасность» (УНЦ ИБ) Сергей Владимирович Поршнев составили настоящий акт в том, что результаты диссертационного исследования Князевой Н.С. используются при реализации учебного процесса по дисциплине «Элементы компьютерной криминалистики» в том числе:

1. Алгоритм анализа изменения временных отметок файлов используется в лекционных занятиях при изучении механизмов следообразования в операционной системе семейства Windows.
2. Методика восстановления последовательности файловых операций используется в лабораторных работах для закрепления навыков проведения компьютерных исследований.

Директор ИРИТ-РтФ _____ И.Н. Обабков

Директор УНЦ ИБ _____ С.В. Поршнев

196499