МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ
Директор Морского колледжа
_____ М.М. Майстришин
"10" 02 2026 г.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКИХ РАБОТ

ОП.02 Операционные системы и среды

специальность 09.02.11 Разработка и управление программным обеспечением

уровень среднего профессионального образования Специалист среднего звена

квалификация Программист

год приема 2026

Практическая работа № 1

Работа с интерпретатором командной строки

Цель работы: *изучение команд среды командной строки windows и приобретение практических навыков их использования.*

1. Основные теоретические сведения

Командная строка Windows является стандартным средством диагностики, настройки и управления компонентами операционной системы и прикладным программным обеспечением.

Командная строка Windows может быть запущена как и любое другое стандартное приложение - через Главное меню Windows, с использованием диалога Выполнить (комбинации клавиш Win+R, Win+X), а также с использованием открытия Проводником (по двойному щелчку) ярлыка или исполняемого файла C:\WINDOWS\System32\cmd.exe.

В результате запуска откроется окно командной строки с приглашением к вводу команд. Приложение cmd.exe часто называют командным процессором или интерпретатором команд, а его основное окно – консолью Windows.

Результаты выполнения команд отображаются в окне консоли Windows и нередко зависят от наличия достаточных прав у текущего пользователя. Для выполнения команд в контексте учетной записи Администратора в операционных системах Windows Vista - Windows 10 необходимо использовать режим Запуск от имени Администратора ПКМ.

Вся информация в компьютере хранится в файлах. Файл это логически связанная совокупность данных (программ, текстов, изображений и т.д.) определенной длины, имеющая имя. Каждый файл имеет обозначение, которое состоит из двух частей: имени и расширения. Хотя расширение файла является необязательным, его использование удобно для классификации файлов по типу, например:

.com, .exe программы, которые могут быть выполнены;

.bat – пакетные командные файлы;

.bak - резервные копии;

.txt - текстовые файлы.

В среде есть зарезервированные имена устройств, которые нельзя использовать в качестве имени файла:

PRN - принтер; NUL - «пустое» устройство;

LPT1 - LPT3 - устройства, присоединяемые к параллельным портам;

COM1 - COM4 - устройства, связанные с последовательными асинхронными портами;

CON - при вводе информации - клавиатура, при выводе - экран;

Каталог это специальное место на диске, в котором хранятся имена файлов, сведения о размерах, времени последнего редактирования, атрибуты и т.д. Один и тот же файл на диске может быть зарегистрирован только в одном каталоге. Текущим называется каталог, с которым в настоящий момент производится работа. По умолчанию команды DOSищут нужные

файлы в текущем каталоге. Если используется файл не из текущего каталога, необходимо указать путь к файлу, например:

c:\program files\far\far.exe - файл far.exe в подкаталоге far каталога program files.

Для указания группы файлов из одного каталога можно употреблять символы «*» (любое число любых символов) и «?» (один произвольный символ), например:

***.txt -все файлы с расширением .txt;**

d*.e* -все файлы с именем, начинающимся на d, и расширением, начинающимся на букву e;

a??.* -файлы любого расширения с длиной имени в три символа и начинающихся на букву a.

Подробную информацию по любой команде можно получить одним из следующих способов:

help*[имя_команды]*

***имя_команды* /?**.

Команды состоят из имени и параметров, разделенных пробелами. Далее при записи формата команд будет принято, что параметры, заключенные в квадратные скобки, не являются обязательными.

2. Основная часть.

Задание

1. Открыть редактор командной строки используя сочетание клавиш "Win+R" для вызова окна "Выполнить" и ввести в пустое поле значение "cmd".
1. В своей папке D:/P121 создать каталог с именем COMANDS.
2. В этом каталоге создать каталоги: с именем KATALOGS, с именем DISCS, с именем FILES.
3. В каталоге DISCS создать файлы: с именем disk.txt, в который записать: свою фамилию, имя, отчество, группу, определение <Диск - это:>; с именем com_disk.txt, в котором описать команды для работы с дисками.
4. В каталоге KATALOGS создать файл opr.txt, в котором написать определение каталога, и каталога: с именем SOZDAN_K, с именем UDALEN_K, с именем PROSM_K.
5. В каталоге SOZDAN_K создать файл с именем sozd.txt, в котором описать формат команды md и её назначение.
6. В каталоге UDALEN_K создать файл с именем udal.txt, в котором описать формат команды удаления каталога и условия удаления.
7. В папке PROSM_K создать файл с именем soder.txt, в котором описать все известные ключи команды DIR и их назначение.
8. В каталоге FILES создать файлы: с именем com_file.txt, в котором описать команду удаления файлов; с именем opr.txt, в котором написать определение файла <Файл - это:>; с именем kop.txt, в котором написать формат команды копирования файлов.
9. Объединить файлы kop.txt, opr.txt, com_file.txt папки FILES в один файл с именем obed.txt
10. Содержимое каталога FILES перенаправить в файл cod_file.txt
11. Переименовать файл sozd.txt в файл с именем proba.txt.
12. Скопировать из каталога FILES все файлы с расширением txt в каталог DISCI.
13. Установить текущим каталогом COMANDS и построить в нем дерево каталогов.
14. Удалить каталог PROSM_K.
15. Сделать рабочим каталогом COMANDS и построить в него дерево каталогов.
16. Удалить в каталоге FILES все файлы, имя которых начинается на букву C.

17. Добавить в файл obed.txt из каталога DISCS текущее время.
18. В строке приглашения среды вывести системную дату.
19. Изменить приглашение системы (выбор самостоятельно)
20. Написать пакетный файл с именем start для запуска любой программы с диска C: (предварительно очистив экран, выдав на экран сообщение о назначении запускающего файла, изменить цвет фона и символов) и сохранить в папке COMANDS. В пакетном файле должна быть указана фамилия автора.
21. Показать результат преподавателю вместе с электронным отчетом, после получения оценки удалить все созданное на диске.
22. Очистить экран от служебных записей.

Отчет должен содержать:

- титульный лист;
- цель работы;
- ход работы - протокол диалога пользователя с ЭВМ при выполнении задания.
- выводы по работе

3. Заключительная часть.

Проверьте свои знания, ответив на контрольные вопросы:

1. Понятие операционной системы, ее назначение.
2. Как воспользоваться справкой по командам среды?
3. Какие команды операционной системы называются внутренними, а какие внешними?
4. Какой дисковод называется текущим?
5. Как обозначаются различные устройства в командной строке?
6. Как перейти с одного диска на другой?
7. Как образуются имена файлов и каталогов в файловой системе?
8. Что называется путем к файлу?
9. Какие команды работы с файлами вы можете назвать?
10. Какие команды работы с каталогами вы знаете?
11. Как удалить пустой и непустой каталоги?
12. Какие существуют команды общесистемного назначения?
13. Каковы команды для работы с экраном?

Практическая работа № 2

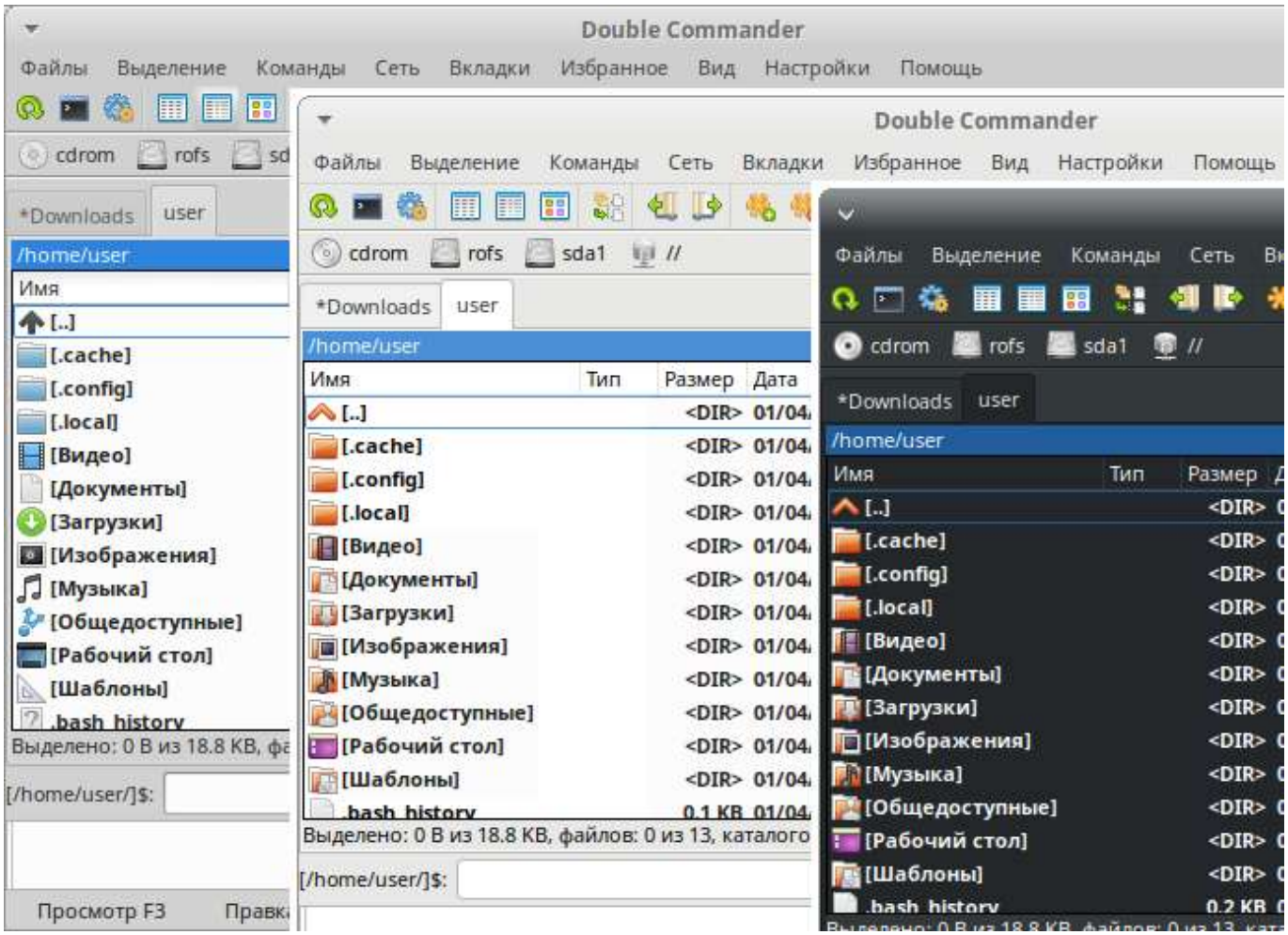
Работа с операционной оболочкой Double Commander

Цель работы: *получить основные навыки работы с программой Double Commander: освоить различные способы выполнения операций с файлами и папками.*

1. Основные теоретические сведения

Операционная система (operating system) – комплекс взаимосвязанных программ, процедур, правил, документации, обеспечивающих взаимодействие пользователя с вычислительной системой, а также управления её ресурсами.

Операционная оболочка (shell) — это программное средство, которое создаёт интерфейс для взаимодействия пользователя с операционной системой (ОС). Название «shell» в переводе на русский («оболочка», «скорлупа») отражает суть: оболочка окружает ядро ОС, предоставляя пользователю доступ к сервисам и функциям системы через текстовые команды.



2. Основная часть.

Задание

Работа с файлами и папками в панелях

1. Найдите на рабочем столе ярлык и запустите программу *Double Commander*.
2. Найдите левую и правую панели, просмотрите содержимое корневого каталога. Для этого нажмите кнопку с символом «\» в окне выбора дисков.
3. Запустите программу Блокнот, в блокноте наберите текст: «Я сейчас изучаю возможности программы *Double Commander*». Сохраните файл с именем Double в папке 2 курс.
4. Для просмотра содержимого этого текстового файла в левой панели окна Double Commander перейдите в папку 2 курс, выделите файл Double.txt. Среди кнопок функциональных клавиш найдите кнопку «Просмотр» или нажмите F3. Закройте окно просмотра.
5. В правой панели откройте папку 2 курс. Для создания каталога найдите кнопку «Каталог» или нажмите F7. В появившемся окне укажите имя ЭКСПЕРИМЕНТ и нажмите ok. Откройте эту папку двойным щелчком левой кнопкой мыши.
6. Для того чтобы переместить файл Double.txt. из папки 2 курс в папку ЭКСПЕРИМЕНТ, перейдите в левую панель, щёлкнув левой кнопкой мыши или нажав клавишу Tab. Выделив файл Double.txt, нажмите кнопку «Перемещение» или F6.
7. Щелчком правой кнопкой мыши по файлу Double.txt вызовите контекстное меню. Выполните команды – создать ярлык и добавить в архив Double.rar.
8. Закройте папку ЭКСПЕРИМЕНТ, для этого выполните двойной щелчок по значку .

Меню Файл и Вид

1. Откройте пункт меню «Файл» и выберите команду «Свойства файла/каталога». Просмотрите размер вашей папки, тип, дату создания, атрибуты.
2. Заархивируйте папку ЭКСПЕРИМЕНТ, поместив архив в папку Р22Х. Для этого:

- в одной из панелей выберите и откройте папку Р22Х (эта папка - получатель);
- в другой панели выделите щелчком левой кнопкой мыши папку ЭКСПЕРИМЕНТ (источник);
- в пункте меню Файл выберите упаковать;
- в появившемся окне необходимо выбрать архиватор (например, zip);
- нажать ok.

1. Разархивируйте полученный архив в папку 3 курс, предварительно открыв эту папку на одной из панелей (получатель) и примените команду распаковать для выделенного архива (источника).
2. Примечание. Можно создать многотомный архив (в случае, если архив более 1.44 Mb). Тогда устанавливают напротив этого пункта флажок.
3. Поиск файла (например, Double.txt) выполняют с помощью команды Поиск файлов в пункте меню КОМАНДЫ. В окне поиска необходимо ввести Double.txt. В окне результата поиска можно не только сразу же перейти к искомому файлу (или папке), можно вынести найденное на панель, а также просмотреть найденный файл.
4. Вызовите окно Сведения о системе. (Выберите пункт КОМАНДЫ и сведения о системе).
5. Откройте пункт меню ВИД. Примените различные способы отображения файлов и папок в панелях Double Commander. Организуйте отбор файлов по шаблону Text Files с помощью команды ФИЛЬТР, убедитесь в том, что в панелях видны только текстовые файлы.

Меню Сеть

1. Создайте копию каталога ЭКСПЕРИМЕНТ и назовите его СЕТЕВОЙ ЭКСПЕРИМЕНТ.
2. Откройте меню Сеть, выберите команду Открыть общий доступ к каталогу и настройте общий доступ к каталогу СЕТЕВОЙ ЭКСПЕРИМЕНТ.
3. Создайте сетевой диск с именем Z из каталога СЕТЕВОЙ ЭКСПЕРИМЕНТ.

Оформить электронный отчет по практическому занятию.

Отчет должен содержать:

- титульный лист;
- цель работы;
- ход работы - протокол диалога пользователя с ЭВМ при выполнении задания.
- выводы по работе

3. Заключительная часть.

Проверьте свои знания, ответив на контрольные вопросы:

1. Дайте определение «Операционная система», «Операционная среда».
2. Состав операционной системы.
3. Какие программы
4. Опишите структуру окна Double Commander.
5. Укажите сходство Double Commander и программы Проводник.
6. Какие способы выделения в панелях Double Commander вы знаете?
7. Как осуществить операции копирования, перемещения и удаления?
8. Как узнать свойства файла?
9. Продемонстрируйте, как заархивировать папку.
10. Как разбить и собрать файл? Когда используются эти команды?
11. Как найти папку и поместить её на панель?
12. Как отобразить файлы и папки в виде дерева на одной из панелей Double Commander?
13. Как выполнить сортировку по дате создания?

Практическое занятие №3

Исследование работы планировщиков процессов в мультипрограммных операционных системах

ЦЕЛЬ РАБОТЫ:

1) знакомство с программой имитирующей модель управления работой планировщика процессов в операционной системе;

2) сравнение методов планирования по различным критериям.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ:

Операционные системы различаются

- особенностями реализации внутренних алгоритмов управления основными ресурсами компьютера (процессорами, памятью, устройствами),
- особенностями использованных методов проектирования,
- типами аппаратных платформ,
- критериями эффективности,
- особенностями реализации сетевых решений
- и многими другими свойствами.

Мультипрограммирование, или многозадачность, — это способ организации вычислительного процесса, при котором на одном процессоре выполняются сразу несколько программ (создается видимость одновременного выполнения нескольких программ).

Мультипрограммирование применяется для повышения эффективности вычислительной системы.

Эффективность же может пониматься как

- общая пропускная способность вычислительной системы;
- удобство работы пользователей, например, возможность интерактивной работы для нескольких пользователей или возможность работы одного пользователя с несколькими приложениями на одной машине;
- реактивность системы – то есть способность системы выдерживать заранее заданные (возможно, очень короткие) интервалы времени между запуском программы и получением результата.

В зависимости от выбранного критерия эффективности операционные системы делятся на:

- системы пакетной обработки (например, ОС ЕС),
- системы разделения времени (UNIX, VMS),
- системы реального времени (QNX, RT/11).

Системы пакетной обработки служат для решения задач в основном вычислительного характера, не требующих быстрого получения результатов.

Главной целью и критерием эффективности таких систем является максимальная пропускная способность, то есть решение максимального числа задач в единицу времени.

Схема функционирования систем пакетной обработки данных:

- в начале работы формируется пакет заданий, каждое задание содержит требование к системным ресурсам;

- из этого пакета заданий формируется мультипрограммная смесь, то есть множество одновременно выполняемых задач. Для одновременного выполнения выбираются задачи, предъявляющие отличающиеся требования к ресурсам, так, чтобы обеспечивалась сбалансированная загрузка всех устройств вычислительной машины; (в мультипрограммной смеси желательно одновременное присутствие вычислительных задач и задач с интенсивным вводом-выводом).

Таким образом, выбор нового задания из пакета заданий зависит от внутренней ситуации, складывающейся в системе, то есть выбирается "выгодное" задание.

В таких ОС невозможно гарантировать выполнение того или иного задания в течение определенного периода времени.

В системах пакетной обработки переключение процессора с выполнения одной задачи на выполнение другой происходит только в случае, если активная задача сама отказывается от процессора, например, из-за необходимости выполнить операцию ввода-вывода. Поэтому одна задача может надолго занять процессор, что делает невозможным выполнение интерактивных задач.

Взаимодействие пользователя с вычислительной машиной, на которой установлена система пакетной обработки, сводится к тому, что он приносит задание, отдает его диспетчеру-оператору, а в конце дня после выполнения всего пакета заданий получает результат. Очевидно, что такой порядок снижает эффективность работы пользователя.

Основной недостаток систем пакетной обработки - изоляция пользователя-программиста от процесса выполнения его задач.

Системы разделения времени (интерактивные) призваны исправить недостаток систем пакетной обработки данных.

Каждому пользователю системы разделения времени предоставляется терминал, с которого он может вести диалог со своей программой.

Каждой задаче выделяется только квант процессорного времени,

Þ ни одна задача не занимает процессор надолго, и время ответа оказывается приемлемым.

Если квант выбран достаточно небольшим, то у всех пользователей, одновременно работающих на одной и той же машине, складывается впечатление, что каждый из них единолично использует машину.

Системы разделения времени обладают меньшей пропускной способностью, чем системы пакетной обработки, так как

- на выполнение принимается каждая запущенная пользователем задача, а не та, которая "выгодна" системе,

- увеличивается время работы, так как выполняется более частое переключение процессора с задачи на задачу.

Таким образом, критерием эффективности систем разделения времени является не максимальная пропускная способность, а удобство и эффективность работы пользователя.

Системы реального времени применяются для управления различными техническими объектами, такими, например, как станок, спутник, научная экспериментальная установка или технологическими процессами, такими, как гальваническая линия, доменный процесс и т.п.

Существует предельно допустимое время, в течение которого должна быть выполнена та или иная программа, управляющая объектом, в противном случае может произойти авария: спутник выйдет из зоны видимости, экспериментальные данные, поступающие с датчиков, будут потеряны, толщина гальванического покрытия не будет соответствовать норме.

Таким образом, критерием эффективности для систем реального времени является их способность выдерживать заранее заданные интервалы времени между запуском программы и получением результата (управляющего воздействия).

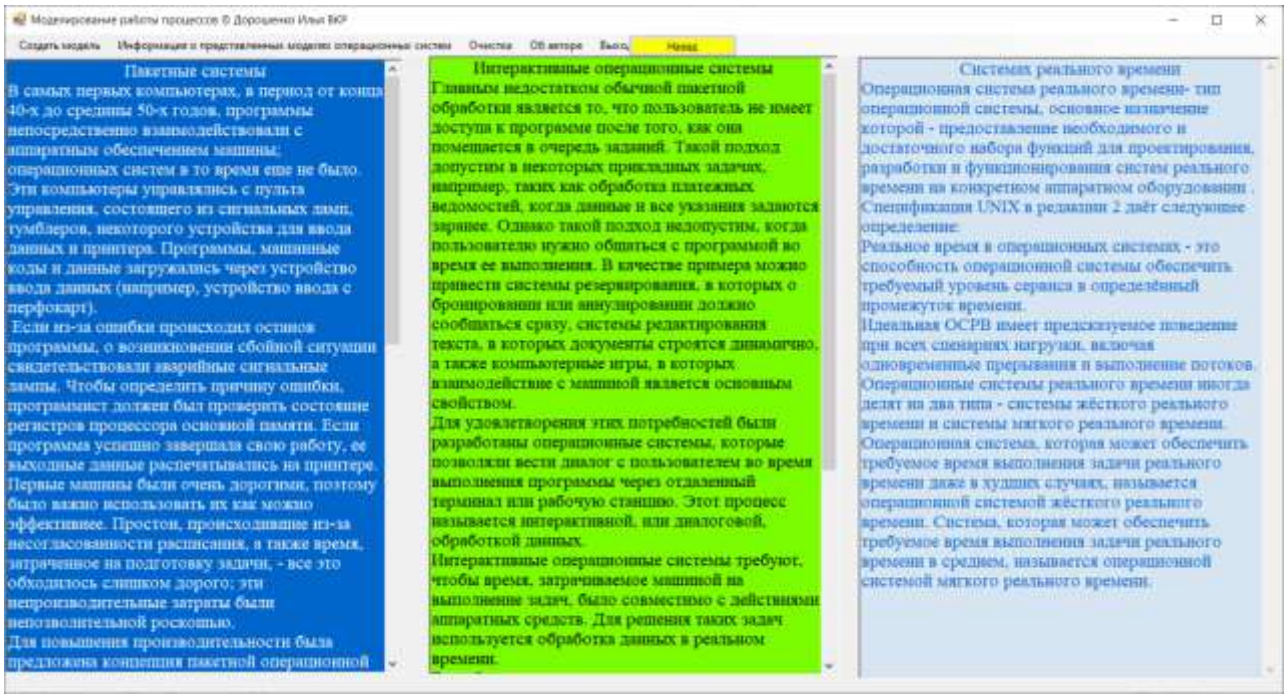
Это время называется временем реакции системы, а соответствующее свойство системы - реактивностью.

Для этих систем мультипрограммная смесь представляет собой фиксированный набор заранее разработанных программ, а выбор программы на выполнение осуществляется исходя из текущего состояния объекта или в соответствии с расписанием плановых работ.

Замечание. Некоторые операционные системы могут совмещать в себе свойства систем разных типов, например, часть задач может выполняться в режиме пакетной обработки, а часть - в режиме реального времени или в режиме разделения времени. В таких случаях режим пакетной обработки часто называют фоновым режимом.

ХОД РАБОТЫ:

1. Скачать с диска приложение planprocess для исследования алгоритмов планирования процессов;
2. Выбрать из таблицы вариантов свой вариант по списку алгоритм по каждому виду систем;
3. Ввести для каждого алгоритма данные о пяти процессах в системе;
4. Сравнить эффективность работы алгоритмов и выявить процесс, который выполнится самым первым (почему?), последним.



СОСТАВ ОТЧЕТА:

Титульный лист

Цель работы

Ход работы: сведения о типе системы по критериям эффективности, сведения об алгоритме планирования процессов, скриншоты введенных данных о процессах, скриншоты диаграммы/таблицы выполнения процессов

Выводы.

Отчет предъявить преподавателю для оценивания

Таблица вариантов

№	Алгоритм систем пакетной обработки	Алгоритм систем разделения времени	Алгоритм систем реального времени
1.	Неприоритетные алгоритмы FIFO	Циклическое	статические
2.	Неприоритетные алгоритмы SJF	Приоритетное	динамические
3.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Циклическое	динамические
4.	Неприоритетные алгоритмы FIFO	Приоритетное	статические
5.	Неприоритетные алгоритмы SJF	Циклическое	статические
6.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	динамические
7.	Неприоритетные алгоритмы SJF	Циклическое	динамические
8.	Неприоритетные алгоритмы FIFO	Приоритетное	динамические
9.	Неприоритетные алгоритмы SJF	Циклическое	статические
10.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	статические
11.	Неприоритетные алгоритмы FIFO	Циклическое	динамические
12.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	динамические
13.	Неприоритетные алгоритмы FIFO	Циклическое	динамические
14.	Неприоритетные алгоритмы SJF	Приоритетное	статические
15.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Циклическое	статические
16.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	динамические
17.	Неприоритетные алгоритмы SJF	Циклическое	статические
18.	Неприоритетные алгоритмы FIFO	Приоритетное	статические
19.	Неприоритетные алгоритмы SJF	Циклическое	динамические
20.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	статические
21.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Циклическое	статические
22.	Неприоритетные алгоритмы SJF	Приоритетное	статические
23.	Неприоритетные алгоритмы FIFO	Циклическое	динамические
24.	Приоритетные алгоритмы – по наименьшему алгоритму планирования	Приоритетное	динамические
25.	Неприоритетные алгоритмы FIFO	Циклическое	динамические

КОНТРОЛЬНЫЕ ВОПРОСЫ:

1. Дайте определение «процесс», «планировщик процесса».
2. В каких состояниях может находится процесс?
3. Какую информацию о процессе хранит операционная система?
4. Какие очереди процессов формирует система? Каков принцип обслуживания очереди?
5. Дайте определение мультипрограммирование.
6. По каким критериям классифицируются операционные системы?
7. Какие приоритеты может получать процесс в системе?
8. Какие приоритетные алгоритмы планирования вам известны? Неприоритетные?

Практическая работа № 4

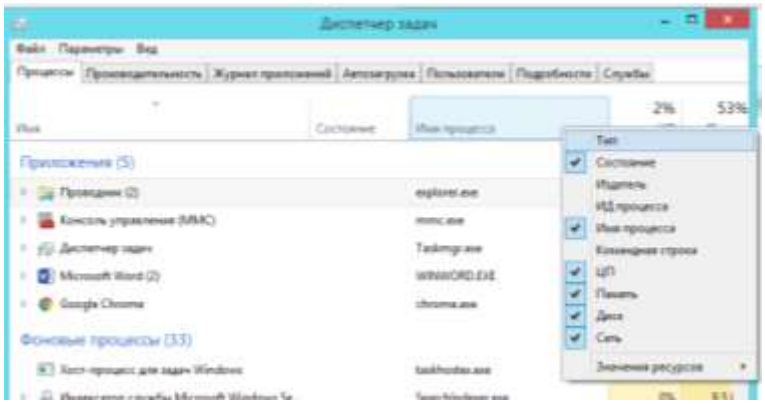
МОНИТОРИНГ И ОПТИМИЗАЦИЯ СИСТЕМЫ

Цель работы: *получить основные навыки работы с оснастками мониторинга и оптимизации.*

Для доступа к оснасткам нажимаем Win + R и в открывшемся окошке «Выполнить» набираем название оснастки, после чего нажимаем OK или Enter.

Задание

Вызовите Диспетчер задач. (Ctrl-Alt-Del)

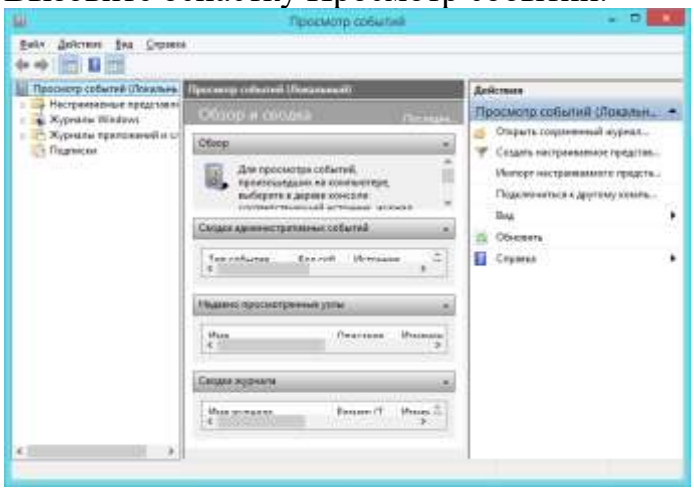


1. Просмотрите все запущенные приложения.
2. Какие процессы запущены в системе? Почему их больше, чем приложений?
3. Для каждого процесса покажите в окне следующие счетчики:
    - Имя процесса
    - Состояние
    - ЦП
    - Память
    - Базовый приоритет
    - ИД потоков

   Сравните процессы по этим показателям
4. Как изменить приоритет некоторого процесса? На что это влияет? Какие процессы имеют высокие приоритеты? Почему?
5. Посмотрите на вкладке Быстродействие общую картину потребления ресурсов вашего компьютера. Запустите три-четыре офисных приложения. Проверьте, изменилась ли картина.

6. Как можно убрать свернутое окно Диспетчера задач с панели задач, чтобы не занимать место на ней? Как тогда вызвать Диспетчер задач?

Вызовите оснастку Просмотр событий.



7. Какие типы основных журналов можно просматривать в этой оснастке?
8. Какие существуют типы событий?
9. Какие параметры можно увидеть для каждого события? (Просмотрите их через окно свойств события.)
10. Отсортируйте события в окнах журналов: журнал Приложений – по типу событий (Ошибки, Предупреждения, Уведомления); журнал системы – по дате возникновения событий)
11. В окне журнала событий системы оставьте только столбцы: Тип, Дата, Время, Категория, Источник.
12. В журнале Безопасности проведите фильтрацию событий: оставьте только аудит отказов за последние 2 недели.
13. Поиск последних критических ошибок системы

Откройте «Просмотр событий» (выполните eventvwr.msc от имени администратора). Перейдите в раздел Журналы Windows → Система.

Отфильтруйте события, оставив только ошибки (Error) и критические (Critical) за последние 7 дней. Найдите хотя бы одно событие с уровнем Критический.

Запишите:

Идентификатор события (Event ID)
Источник события (Source)
Дату и время
Краткое описание проблемы (если доступно)

Какой компонент системы, по вашему мнению, вызвал это событие?

14. Анализ успешных и неудачных попыток входа в систему

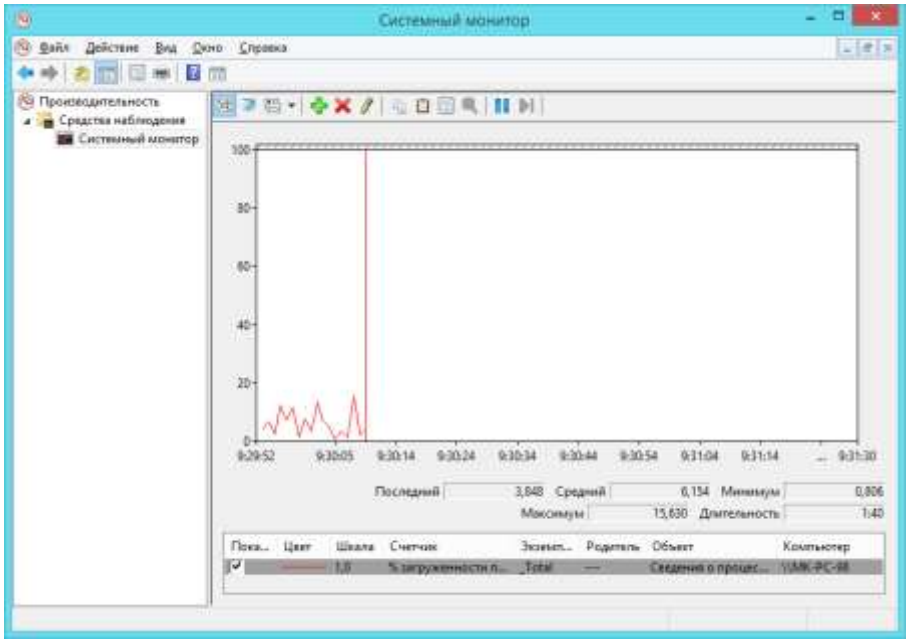
Убедитесь, что включено аудирование входа в систему: Откройте Локальная политика безопасности (secpol.msc), Перейдите в Локальные политики → Политика аудита, Убедитесь, что включено Аудит входа в систему (успешные и неудачные попытки).

В «Просмотре событий» откройте Журналы Windows → Безопасность.
Найдите:
Событие с ID 4624 (успешный вход)
Событие с ID 4625 (неудачная попытка входа)
Для каждого события укажите:
Имя пользователя
Тип входа (например, интерактивный, по сети)
IP-адрес (если применимо, особенно для сетевых попыток)
Сколько неудачных попыток входа было за день, неделю, месяц?

15. Создание пользовательского представления для отслеживания перезагрузок

В «Просмотре событий» щёлкните правой кнопкой по Пользовательские представления → Создать пользовательское представление.
Настройте фильтр по следующим критериям:
Журнал: Система
Источник события: Kernel-General
Идентификаторы событий: 12, 13 (события включения и выключения Windows)
Назовите представление, например: Перезагрузки и выключения.
Сохраните и откройте созданное представление.
Укажите даты и время последних трёх выключений системы.

Запустите оснастку Производительность (работа с Системным монитором).



16. Просмотрите, какие в системе существуют объекты производительности.
17. Просмотрите основные счетчики одного из объектов. Как получить разъяснение, что отображает этот счетчик?
18. Откройте объект процесс. Как можно добавить счетчик для конкретного процесса (Запустите, например, WORD и просмотрите для него некоторые счетчики.).

19. Просмотрите, какие в системе есть потоки (например, потоки того же процесса WORD). Добавьте счетчики Текущий приоритет для потоков WORD, и посмотрите, как они изменяются при переходе в окно программы WORD и обратно. Удалите какой-нибудь счетчик (Кнопка Удалить панели инструментов).
20. Внесите в отчет информацию о степени загрузки центрального процессора.
21. Ознакомьтесь с настройкой внешнего вида представления информации в окне вывода программы Системный монитор: введите названия графика, подпись по вертикальной оси, удаления панели инструментов.
22. Рассмотрите возможность представления информации в окне вывода в виде гистограммы, отчета.
23. Для диагностики узких мест: Для процессора проверьте счетчики: Процессор \ %загруженности процессора, Система \ длина очереди к процессору. Запустите несколько приложений, определите, какое из них в большей степени загружает процессор (через счетчики Процесс \ % загрузки процессора).
24. Для проверки использования памяти введите счетчики Память \Доступно байт (не должен быть меньше 4 Мбайт), Память \ Обмен страниц. Затем запустите несколько приложений и проследите, как в динамике изменяются их рабочие множества (счетчик Процесс \ Рабочее множество).
25. Для дисковой памяти: добавьте счетчики объекта Физический диск: Обращений чтения с диска / сек, Текущая длина очереди диска, % активности диска. Сравните, какие они для разных компьютеров в вашей сети. Эти счетчики определяют производительность вашей дисковой системы, если они стали с течением времени заметно увеличиваться, то необходимо дополнительно протестировать дисковую подсистему с использованием счетчиков: Физический диск \ Среднее время обращения к диску (должно быть не более 0,3 сек.), Физический диск \ Средний размер одного обмена с диском (хороший показатель должен быть в районе 20 кбайт).
26. Встройте экраны с наиболее интересными показаниями счетчиков в отчет по практической работе.
27. Посмотрите через Диспетчер устройств, какие устройства установлены на вашем ПК и все ли они работают нормально. Как определить, какой драйвер управляет устройством? Составьте таблицу драйверов для всех устройств ПК.

Оформить электронный отчет по практическому занятию.

Отчет должен содержать:

- ФИО, группа;
- цель работы;
- ход работы - протокол диалога пользователя с ЭВМ при выполнении задания.
- выводы по работе

Практическое занятие №5

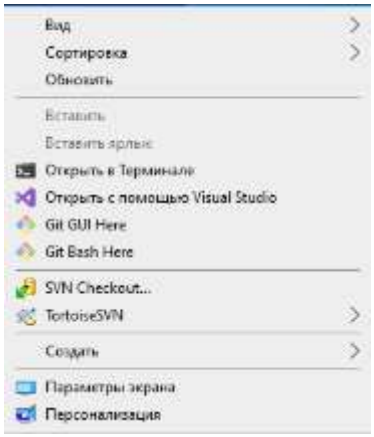
Настройка рабочего стола. Настройка системы с помощью Панели управления

ЦЕЛЬ РАБОТЫ:

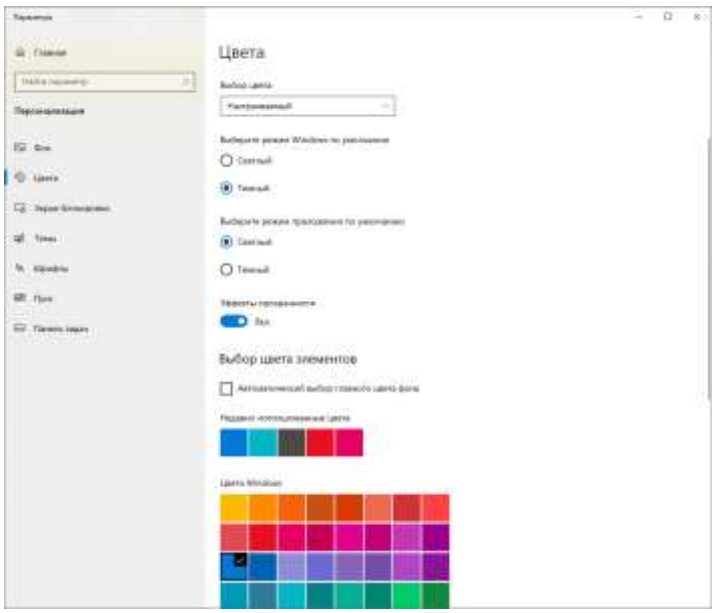
1. знакомство с настройками операционной системы;
2. получение практических навыков настройки интерфейса системы.

ХОД РАБОТЫ:

1. Ознакомьтесь с содержанием раздела «Основы работы в Windows» Руководство по использованию OS Windows 8.1 :: Школа Windows (windows-school.ru)
2. Вызовите контекстное меню на свободном месте Рабочего стола.Выберите команду Персонализация. Внесите в отчет информацию о настройках фона рабочего стола



3. В настройках цвета установите темный режим по умолчанию для windows и светлый для приложений. Включите эффекты прозрачности и смените цвет элементов на любой фиолетовый или зеленый.

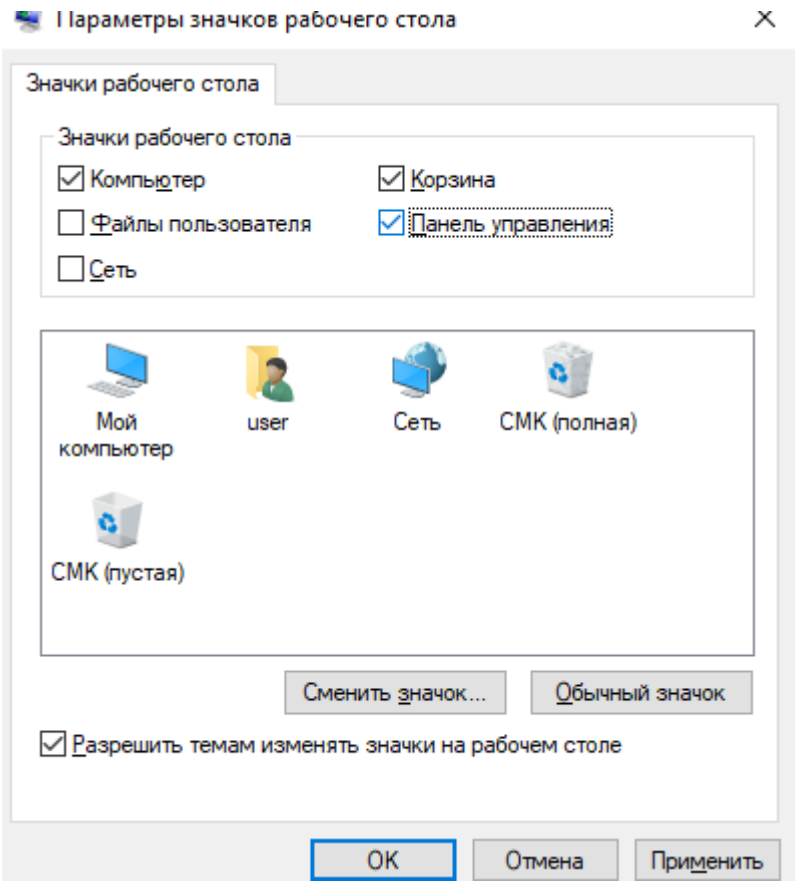


4. Выберите фото для экрана блокировки

Отключите подсказки на экране блокировки

5. Смените курсор мыши в темах на рукописный ввод

6. Настройте параметры значков рабочего стола, добавьте панель управления



7. Установите шрифт Cambria или Segoe UI, отфильтровав предварительно кириллические шрифты.
8. В меню пуск доюавьте отображение наиболее часто используемых приложений.
9. Установить положение панели задач вверху экрана
10. Внести в отчет в виде таблицы информацию о каждой группе настроек панели управления: система, устройства, сеть и интернет, приложения, учетные записи, время и язык. Сами настройки не менять. Сделать выводы по работе.

Практическая работа № 6

Мониторинг процессов

ЦЕЛЬ РАБОТЫ: *изучение работы приложения для мониторинга процессов и приобретение практических навыков его использования.*

ХОД РАБОТЫ:

1. Основные теоретические сведения

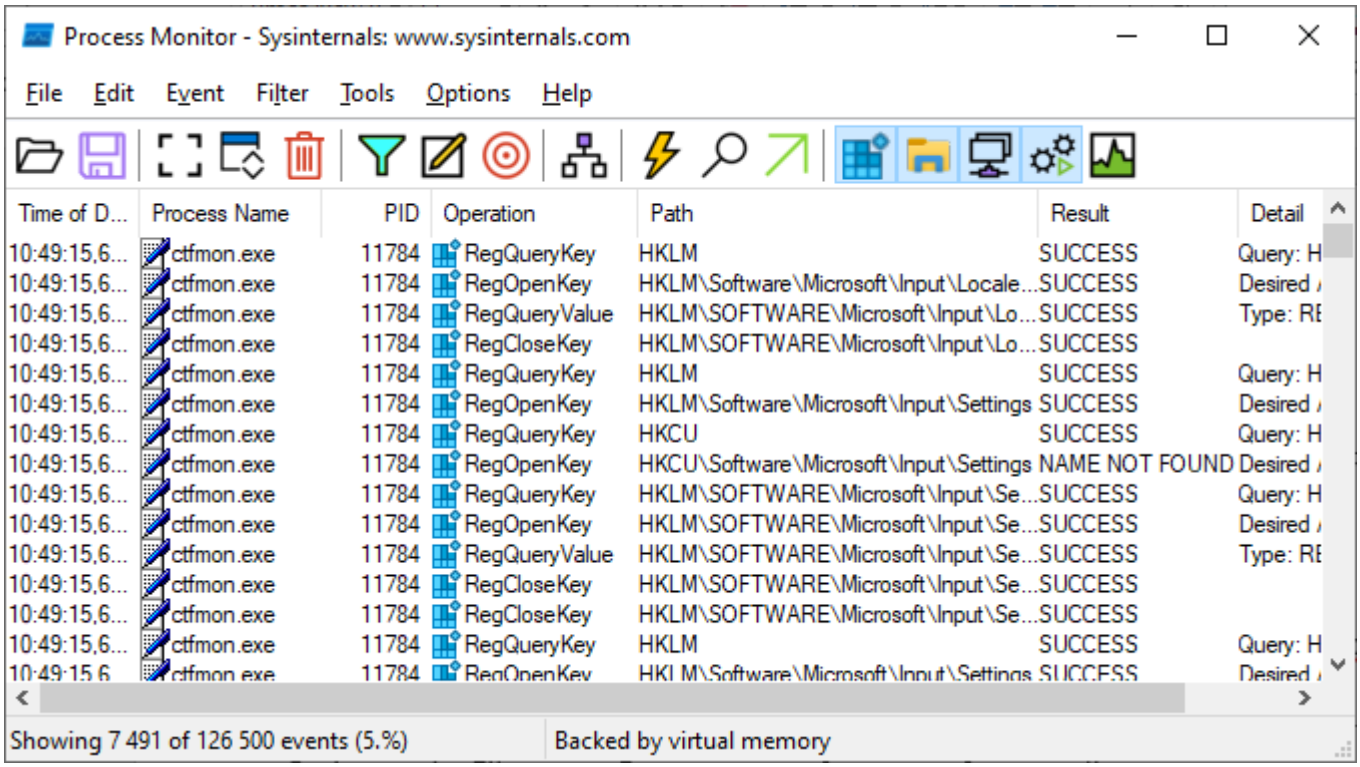
Process Monitor — это расширенный инструмент мониторинга для Windows, который в реальном времени показывает файловую систему, реестр и активность процессов/потоков. Он сочетает в себе функции двух устаревших утилит Sysinternals, *Filemon* и *Regmon* , и добавляет обширный список улучшений, включая богатую и неразрушающую фильтрацию, комплексные свойства событий, такие как идентификаторы сеансов и имена пользователей, надежную информацию о процессах, полные стеки потоков со встроенными символами. поддержка каждой операции, одновременная запись в файл и многое другое. Его уникальные мощные функции сделают Process Monitor основной утилитой в наборе инструментов для устранения неполадок вашей системы и поиска вредоносных программ.

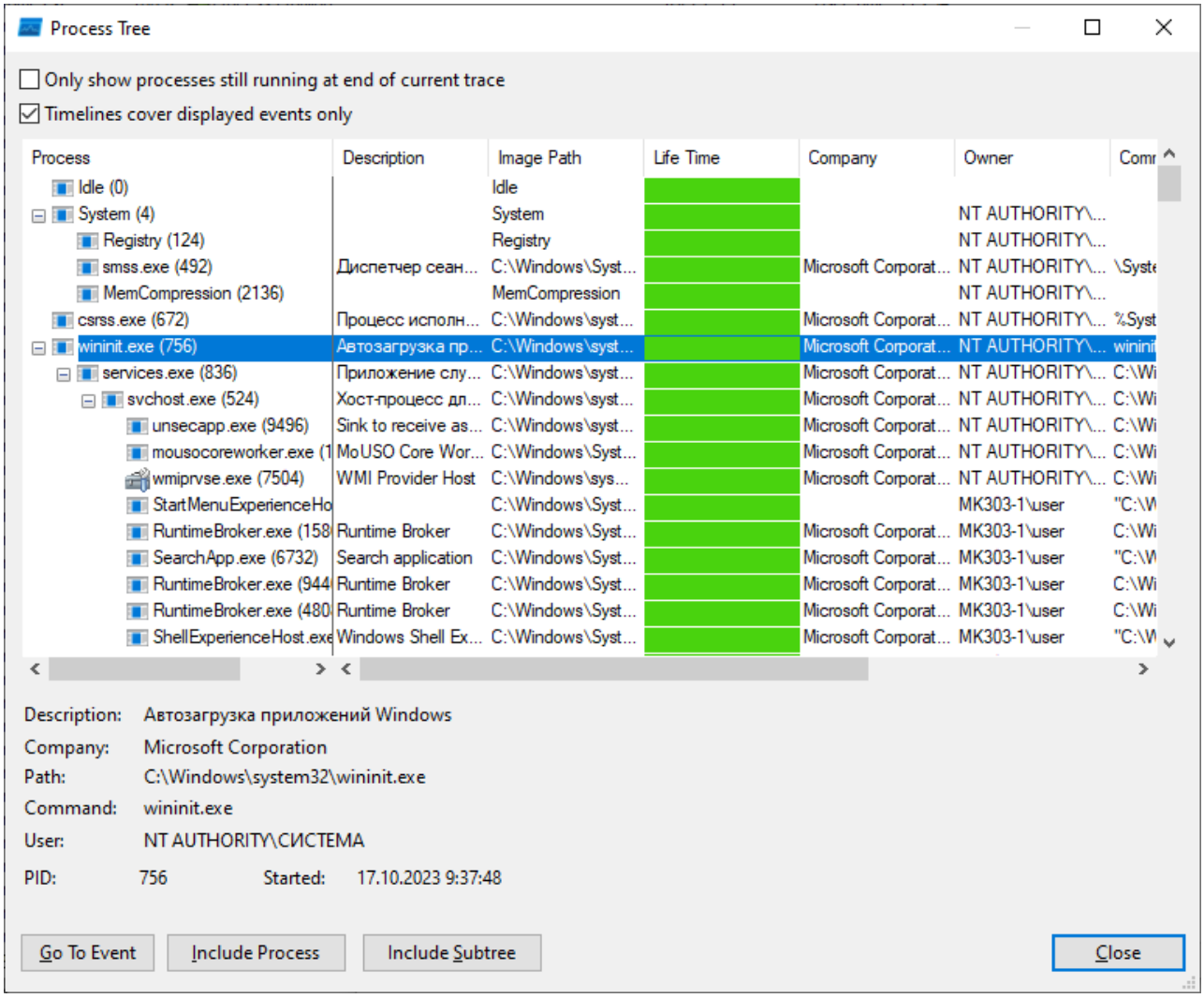
Обзор возможностей монитора процессов

Process Monitor включает в себя мощные возможности мониторинга и фильтрации, в том числе:

- Больше данных, собираемых для входных и выходных параметров операции.
- Неразрушающие фильтры позволяют устанавливать фильтры без потери данных.
- Захват стеков потоков для каждой операции во многих случаях позволяет определить основную причину операции.
- Надежный сбор сведений о процессе, включая путь к образу, командную строку, идентификатор пользователя и сеанса.
- Настраиваемые и перемещаемые столбцы для любого свойства события.
- Фильтры можно установить для любого поля данных, включая поля, не настроенные как столбцы.
- Усовершенствованная архитектура журналирования масштабируется до десятков миллионов зафиксированных событий и гигабайт данных журналов.
- Инструмент дерева процессов показывает взаимосвязь всех процессов, на которые есть ссылки в трассировке.
- Собственный формат журнала сохраняет все данные для загрузки в другой экземпляр Process Monitor.
- Всплывающая подсказка процесса для удобного просмотра информации об изображении процесса.
- Подробная подсказка обеспечивает удобный доступ к отформатированным данным, которые не помещаются в столбец.
- Отменяемый поиск
- Регистрация всех операций во время загрузки

Лучший способ ознакомиться с функциями Process Monitor — прочитать файл справки, а затем просмотреть каждый из его пунктов меню и опций в работающей системе.





2. Основная часть.

Задание

1. Скачать монитор процессов и установить на ПК https://download.sysinternals.com/files/ProcessMonitor.zip
2. Выполнить команды на панели инструментов Capture-Autoscrol-Clear- Capture. Какие команды они выполняют? Внести информацию в отчет.
3. Выполнить фильтрацию процессов:
   - - процессы с 32-разрядной архитектурой
   - - процессы НЕ с 32-разрядной архитектурой
   - - процессы содержат работу с сетью Network
   - - процессы разработчика Yandex LLC
   - - процессы других разработчиков кроме компании Yandex LLC
4. Вывести дерево процессов.
5. Вывести процессы, включая процессы windows.
6. Вывести только процессы, выполнявшие записи в реестр (registry activity)
7. Вывести только процессы file system activity
8. Вывести только процессы network activity
9. Вывести только процессы process and thread activity
10. Вывести только процессы profiling events
11. Через меню tools вывести системную информацию, суммарную активность процессов

Отчет должен содержать:

- титульный лист;
- цель работы;
- ход работы - протокол диалога пользователя с ЭВМ при выполнении задания.
- выводы по работе

3. Заключительная часть.

Проверьте свои знания, ответив на контрольные вопросы:

1. Понятие операционной системы, ее назначение.
2. Что такое процесс? Поток?
3. Какие состояния процессов вам известны?
4. Что такое ID процесса? PID?
5. Перечислите возможности программы Process Monitor.

Практическое занятие №7

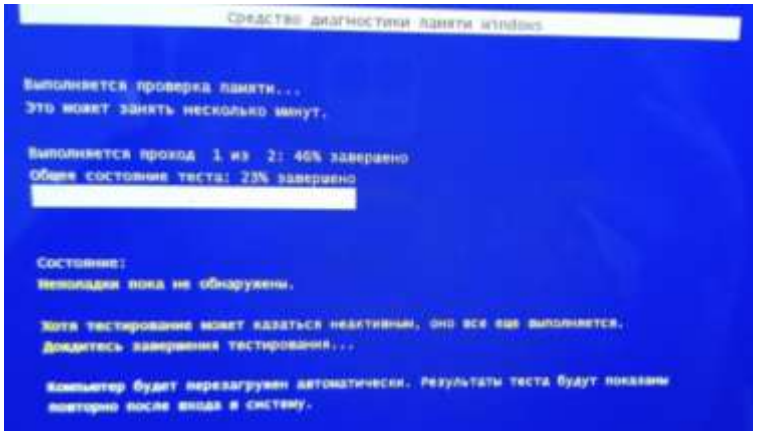
РАСПРЕДЕЛЕНИЕ ПАМЯТИ ПК

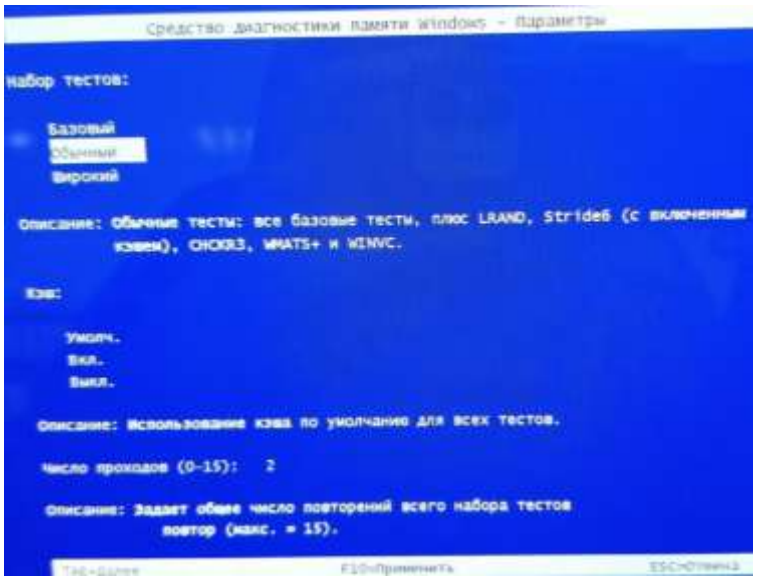
ЦЕЛЬ РАБОТЫ: знакомство с функциями и структурами данных, используемыми для управления памятью.

ХОД РАБОТЫ:

1. Вывести сведения об оперативной памяти из панели управления.
2. Вывести сведения о распределении памяти на диске.
3. Вывести сведения о диске из диспетчера устройств.
4. Средство диагностики памяти Windows (теоретически или на домашнем ПК)

   mdsched.exe (или просто mdsched в диалоге Выполнить Win+R): Запускает "Средство диагностики памяти Windows". Выбрать перезагрузку для теста — проверит на ошибки, объём, скорость. После перезагрузки результаты в Журнале событий (eventvwr.msc > Журналы Windows > Система, MemoryDiagnostics-Results). Можно настроить тип теста (F1: базовый/обычный/широкий, с/без кэша, проходы).



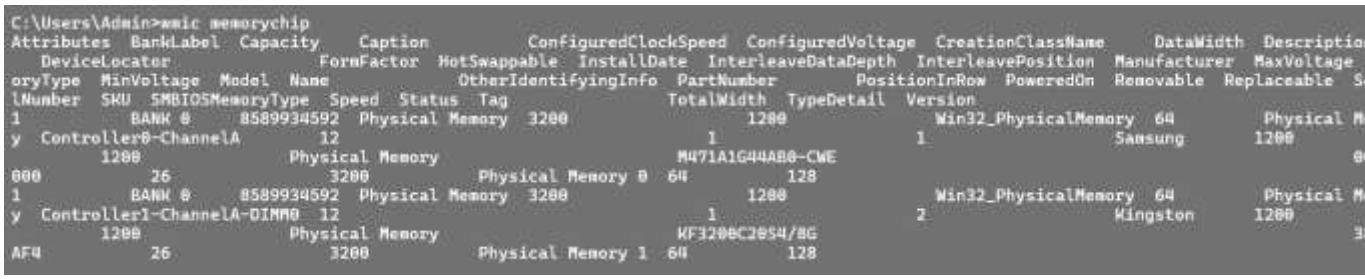


5. Характеристики модулей RAM в cmd:

   wmic MEMCASHE

   wmic MEMPHYSICAL

   wmic MEMORYCHIP



6. Проверка кэша и производительности

   1) perfmon (Win+R): Открывает "Монитор ресурсов" — в разделе "Производительность" добавить счётчики (Memory > Cache Bytes для кэша, Pages/sec для подкачки). Это покажет текущие настройки кэша (размер, использование).

   2) resmon (Win+R): "Диспетчер ресурсов" — вкладка "Память" отобразит резидентную память, standby (кэш), частые сбои. Идеально для мониторинга в реальном времени.

   3) dxdiag (Win+R): Диагностика DirectX (скриншоты по системе, экрану, звуку, вводу).

7. Диагностика BIOS/ПЗУ

   1) msinfo32 (Win+R > msinfo32): В "Сведения о системе" отобразить (скриншот) "Версия BIOS" и "Режим BIOS" (Legacy/UEFI).

   Компоненты > Аппаратное обеспечение > Память

   2) systeminfo | find "BIOS" (в CMD): Покажет версию и дату.

   3) wmic bios get * /format:list (CMD): Детали BIOS — производитель, версия, характеристики.

8. Установить утилиту VMMap, вывести распределение памяти для выбранного процесса.

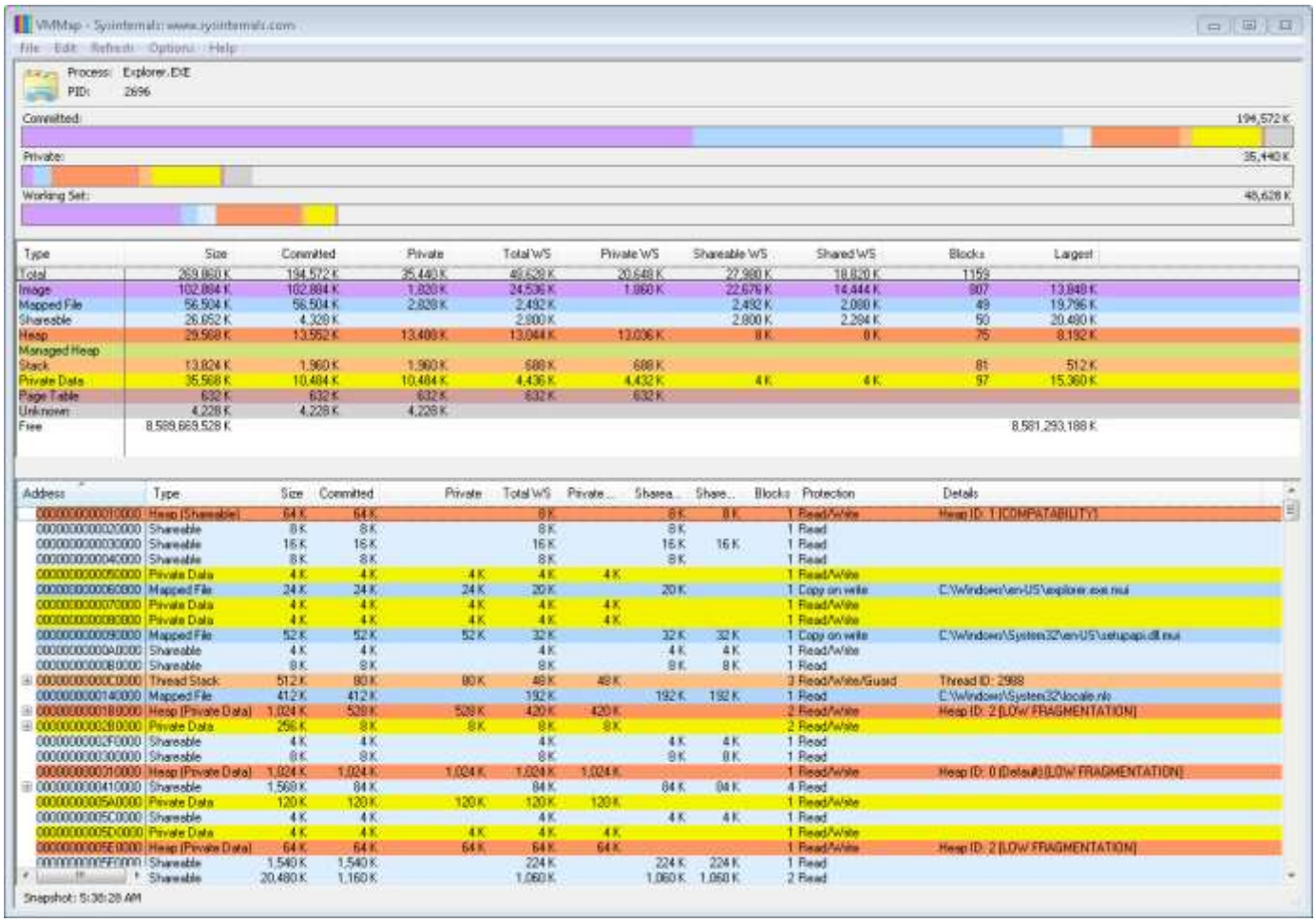
VMMap — это утилита для анализа виртуальной и физической памяти процессов. Он показывает разбивку типов виртуальной памяти, выделенных процессом, а также объем физической памяти (рабочий набор), назначенный операционной системой этим типам. Помимо графического представления использования памяти, VMMap также показывает сводную информацию и подробную карту памяти процесса. Мощные возможности фильтрации и обновления позволяют определить источники использования памяти процессом и стоимость памяти для функций приложения.

Помимо гибких представлений для анализа текущих процессов, VMMap поддерживает экспорт данных в нескольких формах, включая собственный формат, который со-

храняет всю информацию, чтобы вы могли загрузить ее обратно. Он также включает параметры командной строки, которые позволяют создавать сценарии.

VMMap — идеальный инструмент для разработчиков, желающих понять и оптимизировать использование ресурсов памяти своего приложения.

https://docs.microsoft.com/en-us/sysinternals/downloads/vmmap



Отчет должен содержать:

- титульный лист;
- цель работы;
- ход работы - протокол диалога пользователя с ЭВМ при выполнении задания.
- выводы по работе

Ответьте на контрольные вопросы:

1. Понятие операционной системы, ее назначение.
2. Какие виды памяти в вычислительных системах вам известны?
3. Какие единицы измерения емкости носителей информации используются в вычислительных системах?
4. Какие функции выполняет модуль управления памятью в операционных системах?
5. Как выполняется преобразование адресов в операционных системах?
6. Что такое свопинг?
7. Дайте определение виртуальной памяти.
8. Какие характеристики виртуальной памяти для заданного процесса предоставляет утилита vmmap?
9. Какие средства мониторинга и диагностики памяти вы исследовали в работе?

Практическое занятие №8

Монитор ресурсов памяти

ЦЕЛЬ РАБОТЫ: знакомство с функциями и структурами данных, используемыми для управления памятью.

ХОД РАБОТЫ:

Теоретические сведения:

Память является важнейшим ресурсом, требующим тщательного управления со стороны мультипрограммной операционной системы. Особая роль памяти объясняется тем, что процессор может выполнять инструкции протравы только в том случае, если они находятся в памяти. Память распределяется как между модулями прикладных программ, так и между модулями самой операционной системы.

Откачка и подкачка – механизм сброса образов неактивных процессов на диск и, при их активизации, считывания обратно в основную память. Файл откачки – пространство на диске, где система хранит образы откачанных процессов.

Смежное распределение памяти – отведение памяти для всех процессов из одной смежной области памяти. Регистр перемещения используется для преобразования логического адреса в физический (суммируется с логическим адресом).

При распределении памяти в общем случае ОС хранит список свободных и занятых областей. Возникает общая задача распределения памяти – разработка оптимального алгоритма удовлетворения запроса на выделение области памяти заданной длины. Три стратегии решения этой задачи – методы первого подходящего, наиболее подходящего и наименее подходящего. Первая и вторая эффективнее, третья позволяет уменьшить фрагментацию.

Фрагментация – дробление свободной памяти на мелкие несмежные области. Внешняя фрагментация – ситуация, при которой имеется большая область свободной памяти, но она не является непрерывной. Внутренняя фрагментация – выделение "лишних" участков свободной памяти вследствие применения стратегии выделения памяти с точностью до страницы. Внешняя фрагментация может быть ликвидирована с помощью компактировки – перемешивания или сдвига свободной памяти с целью преобразования ее в одну смежную область.

Страничная организация – метод управления памятью, при котором логическая и физическая память делится на страницы одинаковой длины (степень двойки). Физические страницы (фреймы), выделяемые для логических страниц процесса, могут располагаться произвольным образом. Для трансляции логических адресов в физические используется таблица страниц процесса. Логический адрес состоит из номера страницы и смещения внутри страницы. ОС хранит список свободных фреймов в основной памяти.

Сегментная организация памяти – страгетия распределения памяти сегментами переменной длины, каждый из которых выполняет определенную логическую функцию в программе как модуль программы или данных – стек, массив, подпрограмма и т.д.

При сегментной организации логический адрес имеет вид: (номер сегмента, смещение внутри сегмента). Организуется системная таблица сегментов, каждый элемент которой содержит базовый адрес сегмента, его длину и признаки защиты – validation-бит, определяющий корректность номера сегмента, биты защиты от записи, чтения и исполнения. Базовый регистр таблицы сегментов содержит начальный адрес таблицы сегментов, регистр длины таблицы сегментов содержит ее длину.

Перемещение при сегментной организации осуществляется динамически, во время выполнения программы.

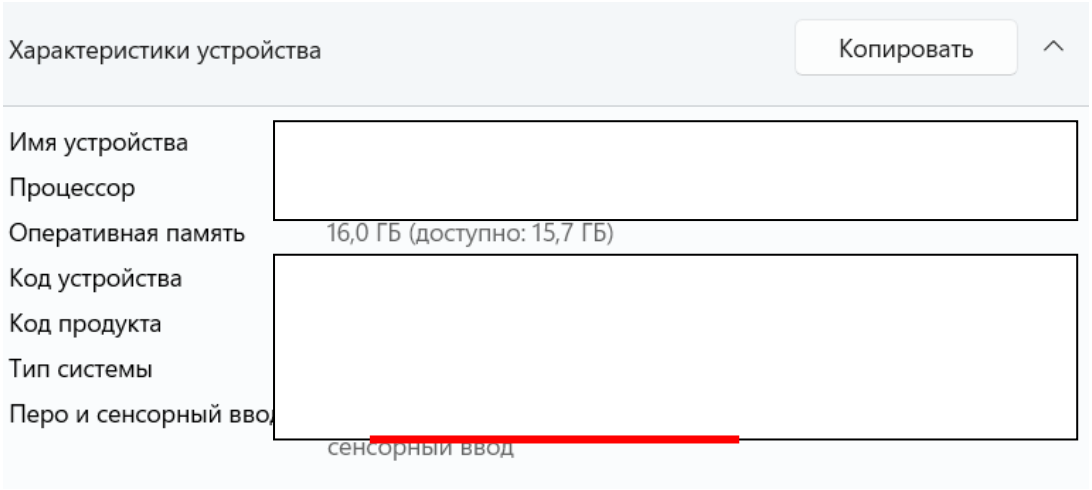
Возможен общий доступ нескольких процессов к одному и тому же сегменту с одним логическим номером (разделяемые сегменты).

Стратегии распределения памяти при сегментной организации – методы первого подходящего и наиболее подходящего. Возможна внешняя фрагментация.

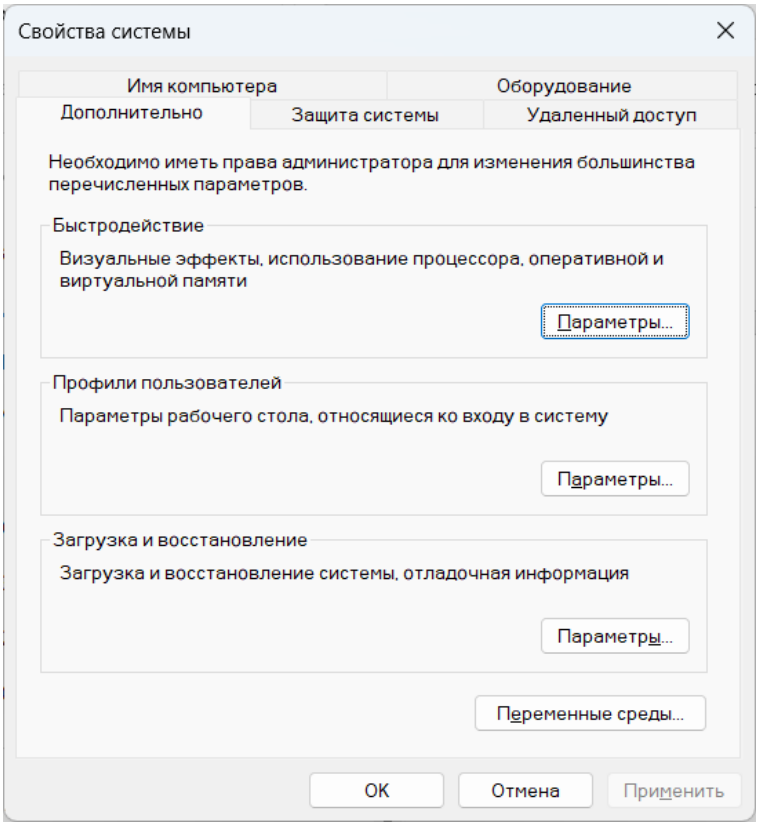
Для борьбы с фрагментацией в некоторых системах применяется смешанная, сегментно-страничная организация памяти, при которой для каждого сегмента организуется собственная таблица страниц.

Ход работы:

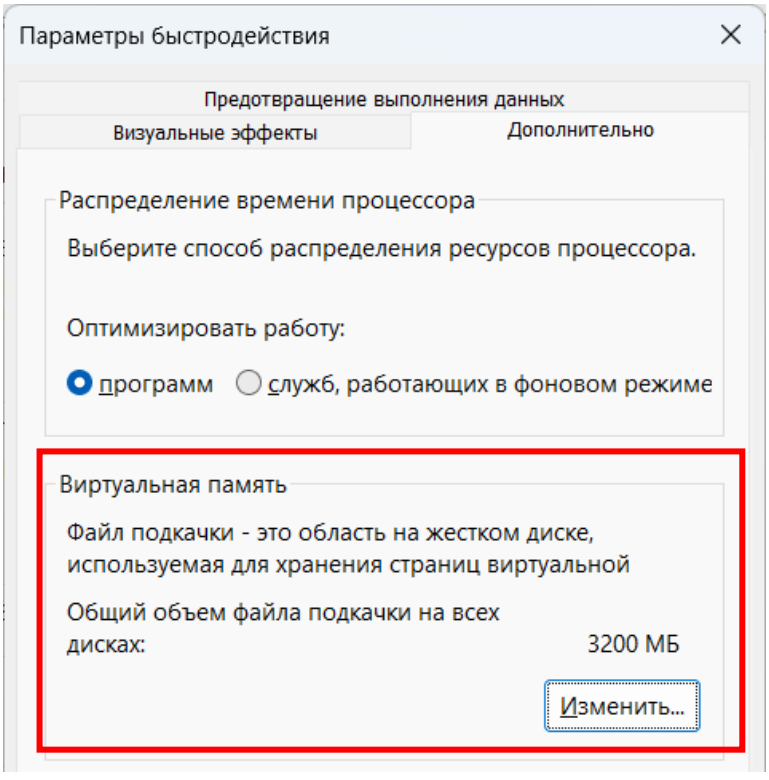
9. Вывести сведения из панели управления о системе и выделить сведения об оперативной памяти.



10. Вызовите окно выполнения и вызовите оснастку конфигурация системы msconfig. Внесите в отчет все вкладки окна конфигурации.
11. На вкладке загрузка вызовите окно дополнительных параметров, убедитесь, что НЕ задействован режим «Максимум памяти» (в противном случае на нужды системы будет выделен максимальный объем). Внесите информацию в отчет.
12. Вызовите дополнительные параметры системы.



13. На вкладке дополнительно в разделе быстродействие выбираем параметры быстродействия. Внесите в отчет все вкладки параметров быстродействия системы.



14. Опишите параметры, влияющие на быстродействие системы. Сделайте вывод как влияет изменение быстродействия на потребление памяти.

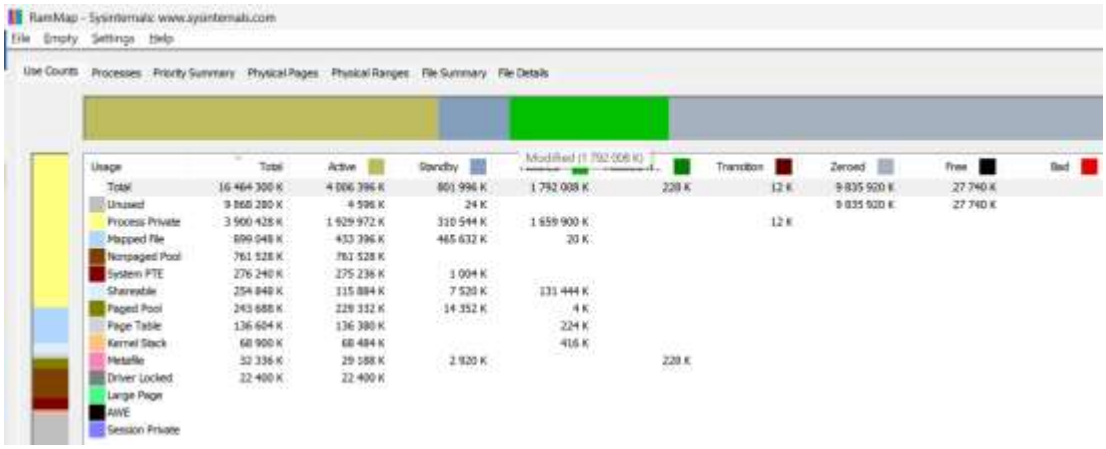
15. Опишите минимальный, текущий, максимальный доступный размер файла подкачки для вашей системы. Не меняйте его значение!!!
16. Используя сведения о системе в интерпретаторе командной строки, выведите сведения:
    - Полный объем физической памяти:
    - Доступная физическая память:
    - Виртуальная память: Макс. размер:
    - Виртуальная память: Доступна:
    - Виртуальная память: Используется:
    - Расположение файла подкачки:

17. Выведите сведения о памяти из диспетчера задач системы.
18. Установить утилиту RAMMap, запустить следующие приложения: проводник, браузер, текстовый редактор, procmon.exe, vmmap.exe, rammap.exe. Запустите утилиту RAMMap. Внесите в отчет информацию:
    - категории используемых показателей памяти
    - использование памяти отдельными процессами
    - распределение памяти на основе уровней приоритета, назначенных операционной системой
    - распределении физических страниц
    - список кэшированных файлов, хранящихся в оперативной памяти
    - информацию о физических страницах, которые занимают файлы, отнесённые системой в память.

Заполните таблицу

Имя процесса	PID	Private	StandBy	Modified	Page Table	Total
explorer.exe						
_____.exe браузер						
_____.exe текстовый редактор						
procmon.exe						
vmmap.exe						
rammap.exe						

На вкладке process отсортируйте процессы по полю page table, по полю total и сделайте вывод: какие процессы потребляют больше всего памяти? какие меньше?



Столбцы в этой утилите показывают состояния страниц, а строки — чем эти страницы используются:

- **Process Private** — страницы используются процессами;
- **Mapped File** — страницы выделены для файлов, проецируемых в память;
- **Shareable** — общие страницы памяти;
- **Page Table** — страницы в которых хранятся таблицы страниц;
- **Paged Pool** — выгружаемый пул;
- **Nonpages Pool** — невыгружаемый пул;
- **System PTE** — пул системных страниц. Таких как пространство ввода-вывода, стеки ядра и списки дескрипторов памяти, также тут могут находиться страницы виртуальных машин;
- **Session Private** — память, которая является частной для определенного сеанса, вошедшего в систему. Она будет выше на серверах RDS;
- **Metafile** — часть системного кеша который содержит метаданные NTFS;
- **AWE (Address Windowing Extensions)** — позволяет приложению отображать различные представления физической памяти в свое адресное пространство, обычно это используется SQL или другими приложениями баз данных;
- **Driver Locked** — страницы, заблокированные драйверами;
- **Kernel Stack** — страницы, используемые стеками потоков ядра;
- **Unused** — неиспользуемые страницы;
- **Large Page** — большие страницы (> 2 MB).

Обратите внимание — все обнуленные страницы не используются (**Unused**). Некоторые приватные страницы процессов находятся в переходном состоянии (**transition**). Страницы в состоянии **modified no-write** это метафайлы файловой системы (ожидают окончания транзакций).

Проецируемые в память файлы (memory-mapped files) — это способ работы с файлами, при котором всему файлу или некоторой непрерывной его части ставится в соответствие определённый участок памяти (диапазон адресов оперативной памяти).

Особенности такого подхода:

- Чтение данных из этих адресов приводит к чтению данных из отображённого файла, а запись данных по этим адресам — к записи этих данных в файл.
- Операционная система не загружает в память сразу весь файл, а делает это по мере необходимости, блоками размером со страницу памяти (как правило, 4 кб ~ 64 кб).

- Файл, проецируемый в память, удобен тем, что можно достаточно легко менять его размер и при этом (после переотображения) получать в своё распоряжение непрерывный кусок памяти нужного размера.

https://learn.microsoft.com/en-us/sysinternals/downloads/rammap

11: Диагностика "утечки" памяти и освобождение кэша

Ситуация: Ваш компьютер работает медленно, в Диспетчере задач показано высокое использование ОЗУ (например, 90–95%), но сумма памяти по процессам не объясняет это полностью. Подозрение на большой кэш файлов (Standby List).

1. Скачайте и запустите RAMMap от имени администратора.
2. Перейдите на вкладку **Use Counts**.
3. Проанализируйте столбцы: обратите внимание на большие значения в **Metafile** (кэш системных файлов, часто на серверах) или **Mapped File** (кэшированные файлы).
4. Перейдите на вкладку **File Summary** или **File Details**, чтобы увидеть, какие именно файлы (например, большие базы данных или временные файлы) занимают кэш.
5. В меню **Empty** выберите **Empty Standby List** (или **Empty Modified List**, если нужно) для принудительного освобождения кэша.
6. Обновите данные (F5) и сравните использование памяти в Диспетчере задач до и после.

Ожидаемый результат: Освободится несколько ГБ ОЗУ, система станет отзывчивее. Это полезно перед запуском ресурсоёмких программ (игры, редакторы видео).

12: Анализ использования памяти конкретным процессом

Ситуация: Одно приложение (например, браузер или виртуальная машина) потребляет много памяти, и вы хотите понять, сколько именно приватной (Private) памяти оно использует, а сколько — shared или cached.

1. Запустите RAMMap от имени администратора.
2. Перейдите на вкладку **Processes**.
3. Отсортируйте таблицу по столбцу **Private** (приватная память процесса) или **Total** (общий рабочий набор).
4. Найдите нужный процесс (например, chrome.exe или vmware.exe) и отметьте его вклад в память.
5. Для детализации перейдите на вкладку **Priority Summary**, чтобы увидеть распределение по приоритетам (высокий приоритет — активная память).
6. Сохраните снимок состояния (File → Save) для сравнения позже, после закрытия приложения.

Ожидаемый результат: Вы точно увидите, сколько памяти "съедает" процесс (в отличие от Диспетчера задач, где cached память может искажать картину). Полезно для выявления "прожорливых" приложений.

13: Проверка драйверов и ядра на утечки памяти

Ситуация: Система стабильна, но со временем память заполняется, и перезагрузка помогает. Подозрение на драйверы устройств (видеокарта, сеть) или kernel-модули.

1. Запустите RAMMap.
2. На вкладке **Use Counts** посмотрите на **Driver** (память драйверов) и **Kernel Stack** / **Paged Pool** / **Nonpaged Pool**.
3. Если значения высокие, перейдите на вкладку **Physical Pages** или **Physical Ranges** для детального просмотра по адресам.
4. В меню **Empty** используйте **Empty Working Sets** для очистки рабочих наборов (без вреда для системы).
5. Сравните данные до и после нагрузки (например, после интенсивной работы с устройством).

Ожидаемый результат: Выявление проблемного драйвера (например, старый драйвер принтера занимает сотни МБ в Nonpaged Pool). Это помогает решить проблему обновлением драйверов или удалением ненужных.

Ответьте на контрольные вопросы:

1. Какие функции выполняет модуль ядра управления памятью в операционных системах?
2. Определение виртуальная память.
3. Как реализована сегментная организация памяти?
4. Как реализована страничная организация памяти? Каким может быть размер одной страницы виртуальной памяти?
5. Как реализована сегментно-страничная организация памяти?
6. Для чего выполняется подкачка на диск?
7. Какие алгоритмы замещения страниц вам известны?
8. Как в операционной системе выполняется преобразование адресов данных?
9. Чем кэш-память CPU отличается от кэш-памяти RAM?

Практическое занятие №9

Работа с программой «Проводник». Работа с файловыми системами и дисками.

ЦЕЛЬ РАБОТЫ: ознакомление с файловыми системами операционной системы windows, получение практических навыков работы с объектами файловой системы.

ХОД РАБОТЫ:
Теоретические сведения:

Файловая система (file system) — это способ организации, хранения и именования данных на носителях информации в компьютерах. Файловые системы используются также в другом электронном оборудовании: в цифровых фотоаппаратах и диктофонах, в мобильных телефонах, и т. п.

Файловые системы хранятся на дисках. Большинство дисков может быть разбито на один или несколько разделов, на каждом из которых будет независимая файловая система. Сектор 0 на диске называется главной загрузочной записью (Master Boot Record (MBR)) и используется для загрузки компьютера. В конце MBR содержится таблица разделов. Из этой таблицы берутся начальные и конечные адреса каждого раздела. Один из разделов в этой таблице помечается как активный. При загрузке компьютера BIOS (базовая система ввода-вывода) считывает и выполняет MBR. Первое, что делает программа MBR, — находит расположение активного раздела, считывает его первый блок, который называется загрузочным, и выполняет его. Программа в загрузочном блоке загружает операционную систему, содержащуюся в этом разделе. Для достижения единообразия каждый раздел начинается с загрузочного блока, даже если он не содержит загружаемой операционной системы. Кроме того, в будущем он может содержать какую-нибудь операционную систему.

Во всем остальном, кроме того, что раздел начинается с загрузочного блока, строение дискового раздела значительно различается от системы к системе. Наиболее часто используемое представление файловой системы представлены на рисунке.

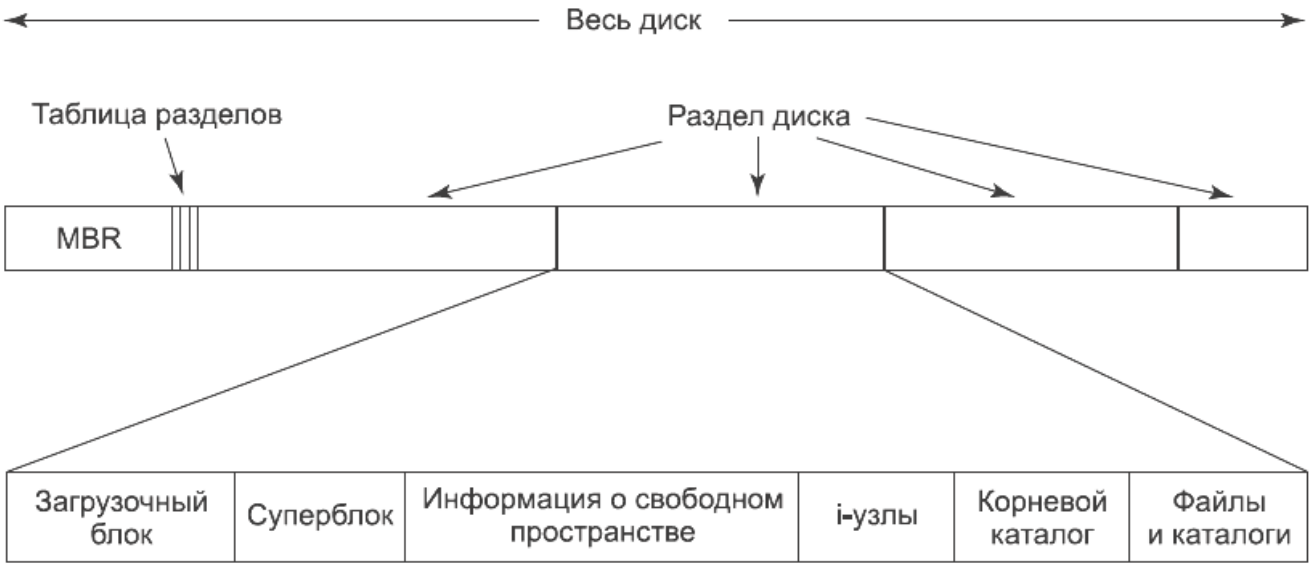


Рисунок 1 – Структура диска и дискового раздела MBR

Первым элементом является суперблок. В нем содержатся все ключевые параметры файловой системы, которые считываются в память при загрузке компьютера или при первом обращении к файловой системе. Обычно в информацию суперблока включаются «магическое» число, позволяющее идентифицировать тип файловой системы, количество блоков в файловой системе, а также другая важная административная информация. Далее может находиться информация о свободных блоках файловой системы, к примеру в виде битового массива или списка указателей. За ней могут следовать i-узлы, массив структур данных — на каждый файл по одной структуре, в которой содержится вся информация о файле (свойства и атрибуты файла, а также дисковые адреса его блоков). Затем может размещаться корневой каталог, содержащий вершину дерева файловой системы. И наконец, оставшаяся часть диска содержит все остальные каталоги и файлы.

MBR и, соответственно, поддержка таблицы разделов, впервые появились на компьютерах IBM PC в 1983 году для поддержки довольно емких на то время жестких дисков в 10 Мбайт в PC XT. С тех пор емкость дисков существенно возросла. Поскольку записи MBR в большинстве систем ограничены 32 битами, максимальная емкость диска, имеющего 512-байтные секторы, который мог быть поддержан, составляет 2 Тбайт. Поэтому в настоящее время большинством операционных систем поддерживаются новые таблицы GPT (GUID Partition Table — таблица разделов, использующая глобально уникальный идентификатор), с помощью которых поддерживаются диски емкостью до 9,4 Збайт.

Выглядит GUID как последовательность из 32 шестнадцатеричных цифр, разделенных на группы. 024DEE41-33E7-11D3-9D69-0008C781F39F — пример того, как выглядит GUID раздела с MBR.

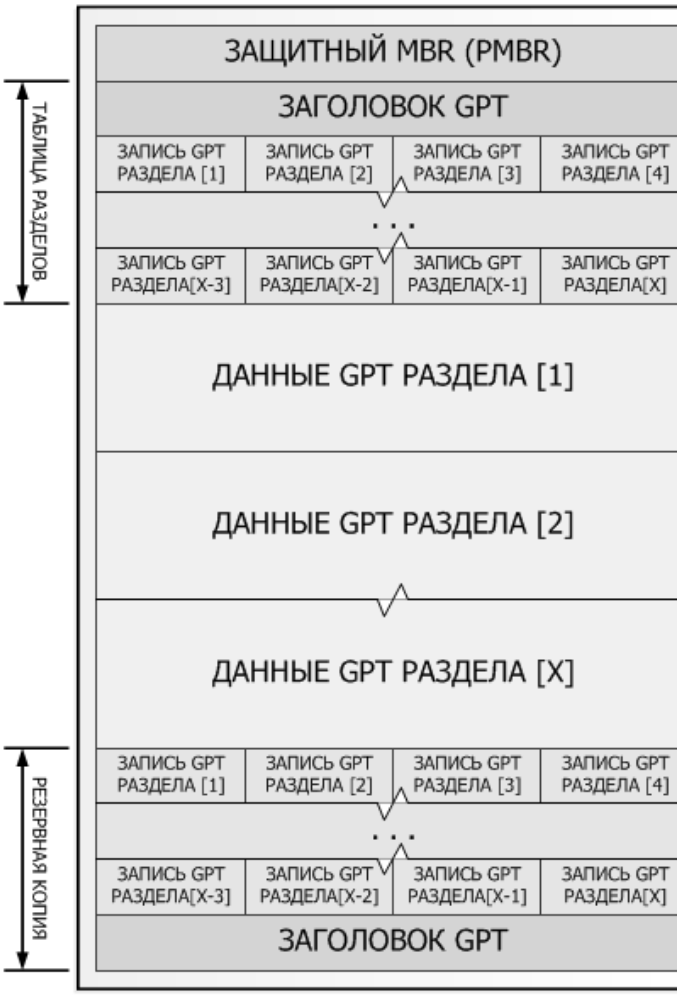


Рисунок 2 – Структура диска и дискового раздела GPT

LBA-адреса для диска (Logical Block Addressing) — это уникальные номера блоков (секторов) на блочном устройстве (жёстком или оптическом диске, твердотельном накопителе). Каждому адресуемому блоку назначается уникальный номер — целое число, начиная с нуля: первый блок — LBA = 0, второй — LBA = 1 и так далее. Преимущество LBA: системному контроллеру не нужно учитывать специфику накопителя (например, геометрию жёсткого диска — количество цилиндров, головок, секторов на дорожке). Он распознает только логические блоки (кластеры).

Структура GPT включает:

- Защитный MBR — первые 512 байт, нужна для обратной совместимости с системами, которые не поддерживают новую схему записи.

- Основной заголовок GPT — в нём содержатся данные о местоположении таблицы разделов, а также контрольные суммы заголовка и таблицы разделов.

- Массив записей о разделах — начинается с блока LBA2, каждая запись — 128 байт. В каждой записи указывается GUID раздела, тип раздела, стартовый и конечный блоки (LBA) раздела.

GPT использует простые и прямые 128 байтные записи для описания разделов. Первые 16 байт обозначают тип раздела - глобально уникальный идентификатор (GUID). Например, для системного раздела EFI {C12A7328-F81F-11D2-BA4B-00A0C93EC93B}. Вторые 16 байтов содержат GUID уникальный для раздела. Затем следуют 64-разрядные (8 байт) LBA адреса начального и конечного логического блока раздела, далее (8 байт) атрибуты раздела (флаги атрибутов), далее имя раздела (72 байта).

Файл — это именованная область на носителе внешней памяти. Файлы идентифицируются именами. Пользователи дают файлам символьные имена, при этом учитываются определенные ограничения ОС. Разные файлы могут иметь одинаковые символьные имена, в этом случае файл однозначно идентифицируется так называемым составным именем (путем), представляющем собой последовательность символьных имен каталогов.

Каталог – это группа файлов, объединенных пользователем исходя из некоторых соображений, содержащая системную информацию о их содержимом. В каталоге содержится список файлов, входящих в него, и устанавливается соответствие между файлами и их характеристиками (атрибутами).

Каталоги могут непосредственно содержать значения характеристик файлов или ссылаться на таблицы, содержащие эти характеристики. Каталоги могут образовывать иерархическую структуру за счет того, что каталог более низкого уровня может входить в каталог более высокого уровня.

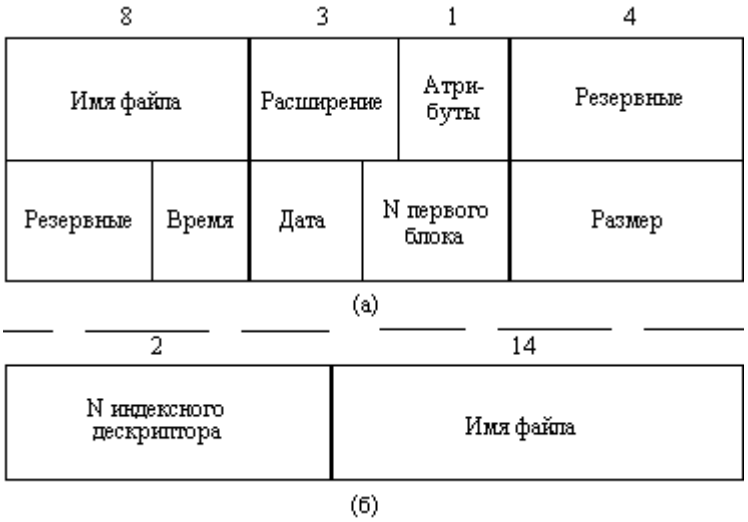


Рисунок 2 - Структура каталогов:
а - структура записи каталога windows (32 байта);
б - структура записи каталога linux

Иерархия каталогов может быть деревом или сетью. Каталоги образуют дерево, если файлу разрешено входить только в один каталог, и сеть - если файл может входить сразу в несколько каталогов. В windows каталоги образуют древовидную структуру, а в linux - сетевую. Как и любой другой файл, каталог имеет символьное имя и однозначно идентифицируется составным именем, содержащим цепочку символьных имен всех каталогов, через которые проходит путь от корня до данного каталога.

Общая модель файловой системы

Функционирование любой файловой системы можно представить многоуровневой моделью, в которой каждый уровень предоставляет некоторый интерфейс (набор функций) вышележащему уровню, а сам, в свою очередь, для выполнения своей работы использует интерфейс (обращается с набором запросов) нижележащего уровня.

Рисунок 3 - Модель функционирования файловой системы

Задачей символьного уровня является определение по символьному имени файла его уникального имени. В файловых системах, в которых каждый файл может иметь только одно символьное имя (например, MS-DOS и MS Windows), этот уровень отсутствует, так как символьное имя, присвоенное файлу пользователем, является одновременно уникальным и может быть использовано операционной системой. В других файловых системах, в которых один и тот же файл может иметь несколько символьных имен, на данном уровне просматривается цепочка каталогов для определения уникального имени файла. В файловой системе linux, например, уникальным именем является номер индексного дескриптора файла (i-node).

На следующем, базовом уровне по уникальному имени файла определяются его характеристики: права доступа, адрес, размер и другие.

Следующим этапом реализации запроса к файлу является проверка прав доступа к нему. Если запрашиваемый вид доступа разрешен, то выполнение запроса продолжается, если нет, то выдается сообщение о нарушении прав доступа.

На логическом уровне определяются координаты запрашиваемой логической записи в файле, то есть требуется определить, на каком расстоянии (в байтах) от начала файла находится требуемая логическая запись.

На физическом уровне файловая система определяет номер физического блока, который содержит требуемую логическую запись, и смещение логической записи в физическом блоке.

После определения номера физического блока, файловая система обращается к системе ввода-вывода для выполнения операции обмена с внешним устройством. В ответ на этот запрос в буфер файловой системы будет передан нужный блок, в котором на основании полученного при работе физического уровня смещения выбирается требуемая логическая запись.

Основными функциями файловой системы являются:

- размещение и упорядочивание на носителе данных в виде файлов;
- определение максимально поддерживаемого объема данных на носителе информации;
- создание, чтение и удаление файлов;
- назначение и изменение атрибутов файлов (размер, время создания и изменения, владелец и создатель файла, доступен только для чтения, скрытый файл, временный файл, архивный, исполняемый, максимальная длина имени файла и т.п.);
- определение структуры файла;
- поиск файлов;
- организация каталогов для логической организации файлов;
- защита файлов при системном сбое;
- защита файлов от несанкционированного доступа и изменения их содержимого.

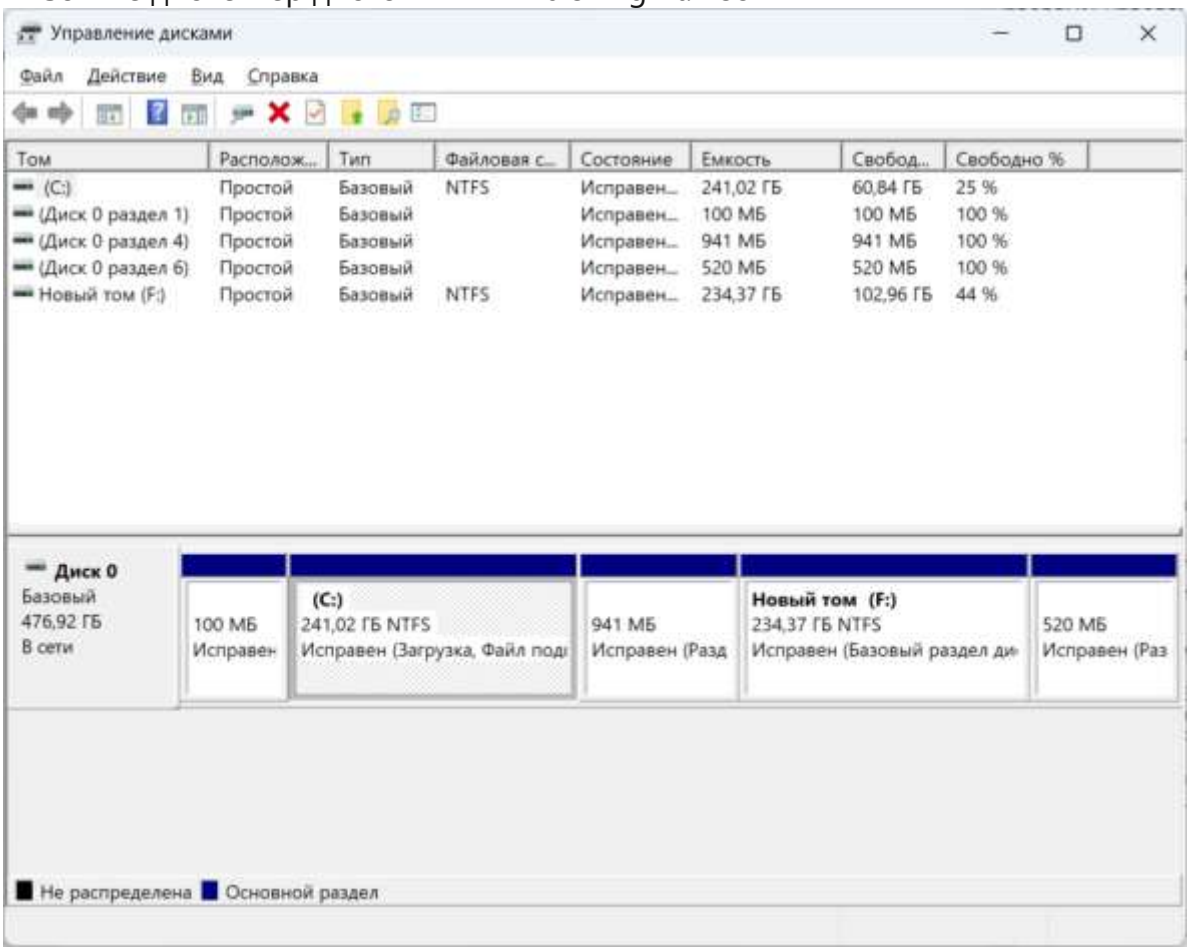
ЗАДАНИЕ:

1. Изучите свойства и атрибуты файла и каталога в проводнике. Внесите информацию в отчет в виде таблицы:

Отличия свойств файлов и папок	Схожие свойства файлов и папок
…	…

2. Создайте папку FileSystem в каталоге вашей учетной записи пользователя. Скопируйте в эту папку все имеющиеся у вас файлы с отчетами по практическим работам по дисциплине, архив с программой process monitor, архив vmmap, архив rammap.

3. Отсортируйте файлы по возрастанию: по имени, по типу, по дате создания; по убыванию: по дате изменения, по размеру, по автору. Внесите информацию в отчет.
4. Выполните отображение содержимого папки проводник в ВИДе: огромных значков, крупных значков, средних значков, мелких значков, списка, таблицы, плитки, содержимого. Внесите информацию в отчет.
5. Выведите свойства локального диска c:. Внесите в отчет скриншоты всех вкладок свойств диска. Какие проверки можно выполнять для диска в ОС windows?
6. Опишите (без выполнения) использование команд cmd для диска C:
    - dir (в корне диска)
    - chkdsk C: /f /r:
    - diskpart
7. wmic diskdrive get Name, Manufacturer, Model, DeviceID, Partitions, Status, InterfaceType, SerialNumber – расшифруйте команду, переведите параметры с английского на русский язык, внесите в отчет скриншот выполнения команды.
8. Вызовите диспетчер дисков win+R - diskmgmt.msc



Внесите информацию в отчет какие разделы отображены в утилите «Управление дисками»? Какую информацию они содержат? Какую файловую систему поддерживают?

9. Выполнение команд в оболочке powershell:
PowerShell — это мощная кросс-платформенная среда автоматизации и управления конфигурациями, разработанная корпорацией Microsoft. Она объединяет в себе *командную оболочку* и *язык сценариев*, что позволяет выполнять

административные задачи, автоматизировать процессы и управлять системой Windows. Некоторые особенности PowerShell:

- Объектно-ориентированный подход. В отличие от традиционных командных оболочек, которые работают с текстом, PowerShell работает с объектами. Каждая команда возвращает объект, который содержит свойства и методы.
- Возможность передачи вывода одной команды (объектов) непосредственно в другую. Благодаря этому можно строить цепочки команд, где каждая команда обрабатывает или преобразует объекты, которые передаются дальше по цепочке.
- Поддержка готовых скриптов. В PowerShell можно использовать готовые скрипты для выполнения целого набора команд одним кликом.
- Создание собственных скриптов. Пользователи могут создавать свои скрипты с расширением .ps1, которые содержат команды для автоматизации действий.

9.1 Get-ChildItem (gci) C:\ -Recurse -Force: Полный список файлов/папок (включая скрытые). Фильтр: gci C:\ -Recurse | Where-Object { $_.Length -gt 1GB } — большие файлы.
9.2 Get-Volume: Список дисков/разделов (ФС, размер, статус). Пример: Get-Volume | Select DriveLetter, FileSystem, Size — таблица с типом ФС (NTFS/exFAT).
9.3 Get-Partition -DiskNumber 0: Разделы на диске 0.
9.4 Get-PSDrive: Диски в сессии (C:). Для ФС: Get-ItemProperty C:\ -Name * — свойства корня.
9.5 Test-Path C:\file.txt: Проверка существования. Для ошибок: Get-EventLog -LogName System | Where { $_.Message -like "*disk*" } — логи ФС.
9.6 Экспорт: gci C:\ | Export-Csv files.csv — в файл для анализа.

Ответьте на контрольные вопросы:

1. Что такое файловая система?
2. Какие файловые системы используются в Windows? Linux?
3. Что такое файл? Каталог? Диск?
4. Опишите структуру диска в виде тома(ов) с использованием MBR.
5. Опишите структуру диска в виде тома(ов) с использованием GPT.
6. Опишите модель функционирования файловой системы.
7. Чем операционная оболочка power shell отличается от cmd?
8. Какие функции выполняет модуль ядра управления вводом-выводом в операционных системах?
9. Какие устройства называют блочными? Символьными?
10. Что такое драйвер устройства? Как просмотреть драйвер диска? Все установленные драйвера?

ЛИТЕРАТУРА (рекомендуемая)

1. Лекции по курсу «Операцилонные системы и среды»
2. *Таненбаум Э., Бос Х.* Современные операционные системы. 4-е изд. — СПб.: Питер, 2015. — 1120 с.: ил. — (Серия «Классика computer science»). ISBN 978-5-496-01395-6— Текст : электронный // Образовательные ресурсы Интернета [сайт]. — URL: https://www.ss-20.ru/
3. https://docs.microsoft.com/en-us/sysinternals/downloads/vmmap
4. https://learn.microsoft.com/ru-ru/windows/win32/cimwin32prov/win32-operatingsystem
5. https://learn.microsoft.com/ru-ru/visualstudio/profiling/memory-usage?view=vs-2022

Приложение 1

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
МОРСКОЙ КОЛЛЕДЖ

Отчет по практической работе №1

Работа с интерпретатором командной строки

Выполнил: студент

Группы Пр/к-26-Х

ФИО

Проверил: Сушкова Ю.А.

Севастополь

202__