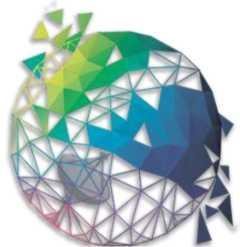
 Министерство образования Красноярского края

Краевое государственное бюджетное профессиональное образовательное

учреждение «Красноярский колледж радиоэлектроники и информационных технологий»



**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ**

По дисциплине Операционные системы и среды

Курс \_\_\_\_

Для специальности (код и наименование)

09.02.11 "Разработка и управление программным обеспечением”

Красноярск, 2025

Методические рекомендации составлены:

Преподавателем КГБПОУ СПО «ККРИТ» К.Н. Татарникова

Преподавателем высшей категории КГБПОУ СПО «ККРИТ» Татарников А.В.

РАССМОТРЕНО

на заседании цикловой комиссии преподавателей

Информатика и вычислительная техника №2

Протокол №\_\_\_ от «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2025 г.

Председатель ЦК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Татарников

Ответственный редактор: зам. директора по учебной работе М.А. Полютова

Одобрено Методическим советом КГБПОУ СПО «ККРИТ»

протокол № \_\_ от «\_\_\_» \_\_\_\_\_\_ 2025 г.

Председатель методического совета

Зам. директора по УР М.А. Полютова

# ПРАКТИЧЕСКИЕ РАБОТЫ

**Практическая работа № 1 Структура операционной системы Цель работы:** различные модели структуры операционных систем.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ПК-1.3 ПК- 3.3

# Теоретический материал

**Понятие, функции и структура операционной системы**

*Операционная система* (operating *system*) – комплекс программ, предостав- ляющий пользователю удобную среду для работы с компьютерным оборудова- нием.

*Операционная система* позволяет запускать пользовательские программы; управляет всеми ресурсами компьютерной системы – процессором (процессора- ми), оперативной памятью, устройствами ввода-вывода; обеспечивает долговре- менное *хранение данных* в виде файлов на устройствах внешней памяти; предо- ставляет *доступ* к компьютерным сетям, как показано на рисунке 1 .

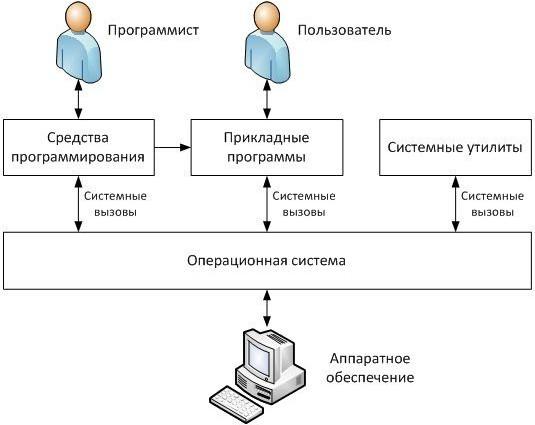


Рисунок 1 - Схема нахождения ОС

Взаимодействие всех программ с операционной системой осуществляется при помощи **системных вызовов** (*system* calls) – запросов программ на выпол- нение операционной системой необходимых действий. Набор системных вызо-

вов образует ***API*** – *Application Programming Interface* (*интерфейс* прикладного программирования).

# Функции операционной системы

К основным функциям, выполняемым операционными системами, можно отнести:

* обеспечение выполнения программ – загрузка программ в память, предоставление программам процессорного времени, обработка системных вы- зовов;
* управление оперативной памятью – эффективное выделение памяти программам, учет свободной и занятой памяти;
* управление внешней памятью – поддержка различных файловых си-

стем;

* управление вводом-выводом – обеспечение работы с различными пе-

риферийными устройствами;

* предоставление пользовательского интерфейса;
* обеспечение безопасности – защита информации и других ресурсов системы от несанкционированного использования;
* организация сетевого взаимодействия.

# Структура операционных систем

Все компоненты операционной системы можно разделить на две группы – работающие в привилегированном режиме и работающие в пользовательском режиме, причем состав этих групп меняется от системы к системе.

Основным компонентом операционной системы является ядро (kernel). Функции ядра могут существенно отличаться в разных системах; но во всех си- стемах ядро работает в привилегированном режиме (который часто называется режим ядра, kernel mode).

Существует два основных вида ядер – **монолитные ядра** (monolithic kernel) и **микроядра** (microkernel). В монолитном ядре реализуются все основные функции операционной системы, и оно является, по сути, единой программой, представляющей собой совокупность процедур. В микроядре остается лишь ми- нимум функций, который должен быть реализован в привилегированном режи- ме: планирование потоков, обработка прерываний, межпроцессное взаимодей- ствие.

Ядра, которые занимают промежуточные положения между монолитными и микроядрами, называют **гибридными** (hybrid kernel).

В состав операционной системы также входят:

* **системные библиотеки** (system DLL – Dynamic Link Library, динами- чески подключаемая библиотека), преобразующие системные вызовы приложе- ний в системные вызовы ядра;
* **пользовательские оболочки** (shell), предоставляющие пользователю интерфейс – удобный способ работы с операционной системой.
* Пользовательские оболочки реализуют один из двух основных видов пользовательского интерфейса:
* **текстовый интерфейс** (Text User Interface, TUI), другие названия – консольный интерфейс (Console User Interface, CUI), интерфейс командной стро- ки (Command Line Interface, CLI);
* **графический интерфейс** (Graphic User Interface, GUI).

# 1 Монолитная система

**Монолитное ядро** — классическая и, на сегодняшний день, наиболее рас- пространённая архитектура ядер операционных систем. Монолитные ядра предоставляют богатый набор абстракций оборудования. Все части монолитного ядра работают в одном адресном пространстве.

Вместе с тем, монолитность ядер усложняет их отладку, понимание кода ядра, добавление новых функций и возможностей, удаление «мёртвого», ненуж- ного, унаследованного от предыдущих версий кода. «Разбухание» кода моно- литных ядер также повышает требования к объёму оперативной памяти, требуе- мому для функционирования ядра ОС. Это делает монолитные ядерные архитек- туры малопригодными к эксплуатации в системах, сильно ограниченных по объ- ёму ОЗУ, например, встраиваемых системах, производственных микроконтрол- лерах и т. д.

Альтернативой монолитным ядрам считаются архитектуры, основанные на микроядрах.

Структура системы :

1. Главная программа, которая вызывает требуемые сервисные процедуры.
2. Набор сервисных процедур, реализующих системные вызовы.
3. Набор утилит, обслуживающих сервисные процедуры.

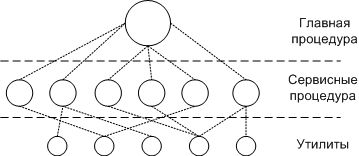


Рисунок 1 - Простая модель монолитной системы

В этой модели для каждого системного вызова имеется одна сервисная про- цедура (например, читать из файла). Утилиты выполняют функции, которые нужны нескольким сервисным процедурам (например, для чтения и записи фай- ла необходима утилита работы с диском).

Этапы обработки вызова:

* Принимается вызов
* Выполняется переход из режима пользователя в режим ядра
* ОС проверяет параметры вызова для того, чтобы определить, какой си- стемный вызов должен быть выполнен
* После этого ОС обращается к таблице, содержащей ссылки на процеду- ры, и вызывает соответствующую процедуру.

# 2 Многоуровневая структура ОС

**Загружаемый модуль ядра** (*loadable kernel module, LKM*) — в информати- ке — объектный файл, содержащий код, расширяющий возможности ядра опе- рационной системы. Модули используются, чтобы добавить поддержку нового оборудования или файловых систем или для добавления новых системных вызо- вов. Большинство современных Unix систем, и Windows, поддерживают загру- жаемые модули ядра, хотя они могут использовать для них разные названия.

Без загружаемых модулей ядра, операционные системы должны были бы включать всю возможную функциональность в базовом ядре.

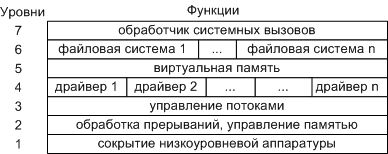


Рисунок 2- Пример структуры многоуровневой системы

Преимущества:

* Высокая производительность Недостатки:
* Большой код ядра, и как следствие большое содержание ошибок
* Ядро плохо защищено от вспомогательных процессов

Пример реализации многоуровневой модели Windows, показан на рисунке 3

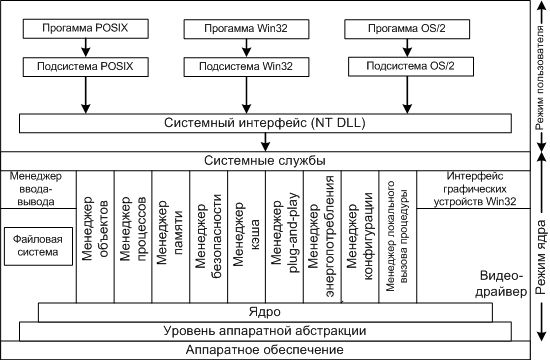


Рисунок 3- Структура Windows 2000

# Требования к операционным системам

Основное требование, предъявляемое к современным операционным систе- мам – выполнение функций, перечисленных выше в параграфе "Функции опера-

ционных систем". Кроме этого очевидного требования существуют другие, часто не менее важные:

* расширяемость – возможность приобретения системой новых функций в процессе эволюции; часто реализуется за счет добавления новых модулей;
* переносимость – возможность переноса операционной системы на другую аппаратную платформу с минимальными изменениями;
* совместимость – способность совместной работы; может иметь место совместимость новой версии операционной системы с приложениями, написан- ными для старой версии, или совместимость разных операционных систем в том смысле, что приложения для одной из этих систем можно запускать на другой и наоборот;
* надежность – вероятность безотказной работы системы;
* производительность – способность обеспечивать приемлемые время решения задач и время реакции системы.

# Практическая часть Ответить письменно на контрольные вопросы:

1. Дайте определение понятию "операционная система".
2. Назовите примеры прикладного, инструментального и системного про- граммного обеспечения.
3. Дайте определение понятий "системный вызов", "API", "ядро".
4. Какие виды ядер вы знаете? К каким видам относятся ядра известных вам операционных систем?
5. Чем ядро отличается от операционной системы?
6. Приведите несколько способов классификации операционных систем.
7. Назовите требования к современным операционным системам и объяс- ните, что они означают.

# Содержание отчета:

* 1. Ознакомиться с теоретическим материалом.
  2. Ответить письменно на поставленные вопросы.
  3. Сделать выводы по работе

# Практическая работа № 2 Функциональные компоненты операционной системы

**Цель работы:** изучить функциональные компоненты операционной си- стемы и их назначение..

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ПК-1.3 ПК- 3.3

# Теоретическая часть

**Операционная система** – это комплект программного обеспечения, предназначенный для управления компьютером и обеспечивающий поддержку хранения, исполнения и разработки прикладных программ.

Состав операционной системы и связь ее базовых подсистем с решаемы- ми задачами, схематично показаны на рисунке 1. Ряд подсистем операционной системы на рисунке объединены в общий блок, названный ядро операционной системы.

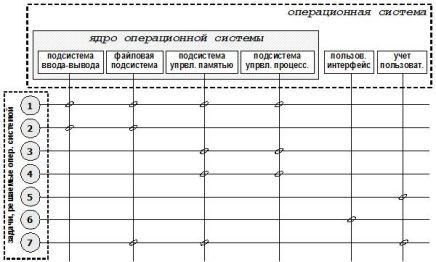


Рисунок 1 - Состав и назначение основных подсистем операционной системы

Ядро объединяет в себе базовые компоненты операционной системы, во многом определяющие специфику и основную область применения данной опе- рационной системы. Важнейшим свойством ядра, отличающим его от любых других системных или пользовательских программ, является то, что код ядра ис- полняется с максимально высоким уровнем привилегий – ему доступны все ко- манды процессора и разрешен прямой доступ к любой аппаратуре компьютера.

**Практическая часть**

**Подсистема ввода-вывода**

Подсистема ввода-вывода принимает запросы на передачу или прием дан- ных от всех выполняющихся программ, как прикладных, так и системных, упо- рядочивает их и переадресует внешним устройствам с учетом их аппаратных особенностей.

Можно выделить две основные функции подсистемы ввода-вывода:

* изоляция всех других компонентов операционной системы и прикладных программ от особенностей аппаратной реализации внешних устройств компью- тера;
* организация бесконфликтного и эффективного использования внешних устройств всеми исполняющимися программами.

Фактически, задачу изоляции от аппаратных особенностей внешних устройств решают драйверы – специальные подпрограммы в составе подсистемы ввода-вывода, которые непосредственно взаимодействуют с внешними устрой- ствами, при этом для каждого внешнего устройства необходим собственный драйвер.

Для персональных компьютеров операции ввода-вывода могут выполнять- ся тремя способами.

1. **С помощью программируемого ввода-вывода**. В этом случае, когда процессору встречается команда, связанная с вводом-выводом, он выполняет ее, посылая соответствующие команды контроллеру ввода-вывода. Таким образом, процессор непосредственно управляет операциями ввода-вывода, включая опо- знание состояния устройства, пересылку команд чтения-записи и передачу дан- ных. Процессор посылает необходимые команды контроллеру ввода-вывода и переводит текущий процесс в состояние ожидания завершения операции ввода- вывода. Недостатки такого метода – большие потери процессорного времени, связанные с управлением вводом-выводом.
2. **Ввод-вывод, управляемый прерываниями**. Процессор посылает необ- ходимые команды контроллеру ввода-вывода и продолжает выполнять текущий процесс, если нет необходимости в ожидании выполнения операции ввода- вывода. В противном случае текущий процесс приостанавливается до получения сигнала прерывания о завершении ввода-вывода, а процессор переключается на выполнение другого процесса. Наличие прерываний процессор проверяет в кон- це каждого цикла выполняемых команд. Такой ввод-вывод намного эффектив- нее, чем программируемый ввод-вывод, так как при этом исключается ненужное ожидание с бесполезным простоем процессора.
3. **Прямой доступ к памяти (direct memory access – DMA)**. В этом случае специальный модуль прямого доступа к памяти управляет обменом данных между основной памятью и контроллером ввода-вывода. Процессор посылает запрос на передачу блока данных модулю прямого доступа к памяти, а прерыва- ние происходит только после передачи всего блока данных.

DMA-контроллер имеет доступ к системной шине независимо от цен- трального процессора

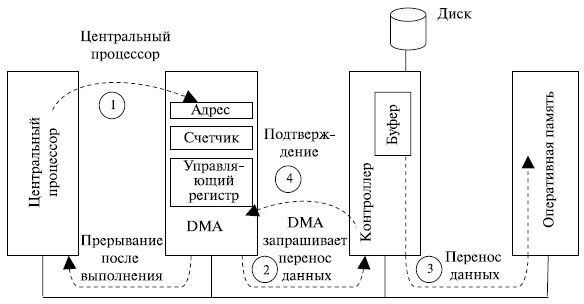


Рисунок 2 - Работа DMA

**Файловая подсистема**

Файловая подсистема обеспечивает пользователя удобным, аппаратно не- зависимым интерфейсом для упорядоченного хранения пользовательских про- грамм и данных на внешних устройствах долговременной памяти.

Файловая система распределяет дисковую память, поддерживает именова- ние файлов, отображает имена файлов в соответствующие адреса во внешней памяти, обеспечивает доступ к данным, поддерживает разделение, защиту и вос- становление данных.

Таким образом, файловая система играет роль промежуточного слоя, экра- низирующего все сложности физической организации долговременного храни- лища данных и создающего для программ более простую логическую модель этого хранилища, а затем предоставляет им набор удобных в использовании ко- манд для манипулирования файлами.

Классическая схема организации программного обеспечения файловой си- стемы представлена на рисунке 4.

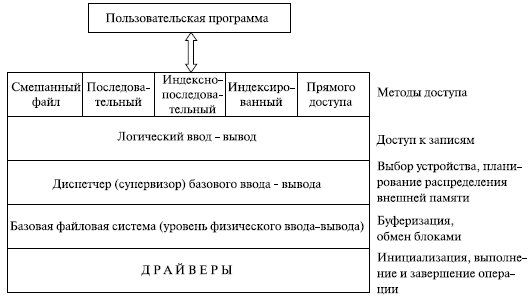


Рисунок 4 -Организация программного обеспечения файловой системы

**Подсистема управления памятью**

Подсистема управления памятью в современной операционной системе решает следующие основные задачи:

* бесконфликтное распределение памяти между исполняющимися про- граммами;
* защита памяти от несанкционированного доступа.

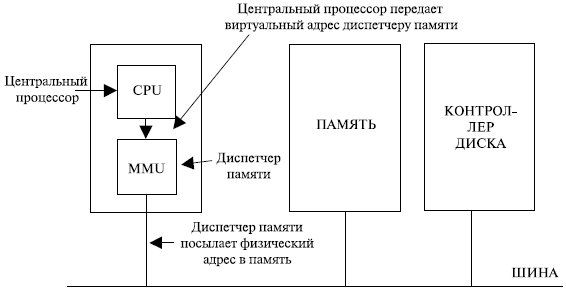
Большинство систем виртуальной памяти используют технику, называе- мую страничной организацией памяти. Любой процесс, реализуемый в компью- тере, может обратиться к множеству адресов в памяти. Адреса могут формиро- ваться с применением индексации, базовых регистров, сегментных регистров и другими путями. Эти программно формируемые адреса, называемые *виртуаль- ными адресами*, формируют *виртуальное адресное пространство*. Виртуальные адреса передаются диспетчеру памяти (MMU – Memory Management Unit), кото- рый отображает виртуальные адреса на физические адреса памяти, как показано на [рисунке](http://www.intuit.ru/studies/courses/631/487/lecture/11057?page=3&image.6.9) 5.

Рисунок 5 - Диспетчер памяти

**Подсистема управления процессами**

Подсистема управления процессами в современной операционной системе решает довольно обширный набор задач, которые удобно представить в виде диаграммы:

Таким образом, подсистема управления процессами решает два класса за- дач: создание и удаление процессов, и управление существующими процессами.

**Пользовательский интерфейс**

**Интерфейс** - совокупность технических, программных и методических (протоколов, правил, соглашений) средств сопряжения в вычислительной систе- ме пользователей с устройствами и программами, а также устройств с другими устройствами и программами.

**Пользовательский интерфейс** - это совокупность программных и аппа- ратных средств, обеспечивающих взаимодействие пользователя с компьютером. Основу такого взаимодействия составляют диалоги.

**Подсистема учета пользователей**

Подсистема учета пользователей необходима в операционной системе для защиты данных пользователей и сохранения их индивидуальных настроек.

Первоначально, подсистема учета пользователей ориентировалась исклю- чительно на решение задачи разграничения прав доступа, но в последствии, по мере развития графических интерфейсов пользователя, появилась еще и необхо- димость сохранять для каждого пользователя его индивидуальные настройки (цветовая схема, состав и расположение пунктов меню и иконок и т.п.).

# Содержание отчета:

1. Изучить приведенный выше материал по компонентам ОС и их основ- ным функциям.
2. Подготовить отчет о практической работе

**Задания для самостоятельной проверки:** Какими способами можно вво- дить формулу? С чего начинается формула? Какие данные можно вводить в формулу? Что используют для создания формул с функциями?

**Практическая работа № 3 Утилиты операционной системы Цель работы:** изучить работу утилит операционной системы Windows.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Теоретическая часть Утилита «Конфигурация системы»

Программа **«Конфигурация системы»** - это утилита операционной систе- мы Windows, предназначенная для управления автоматически запускаемыми программами и загрузкой системы, а также определения проблем, которые могут помешать запуску операционной системы в обычном режиме..Утилита вызыва- ется файлом MSConfig.exe, который расположен в папке System32 раздела с установленной операционной системой.

Для того чтобы открыть эту утилиту выполните любое из следующих дей- ствий:

* Нажмите на кнопку **«Пуск»** для открытия меню, в поле поиска введите

*msconfig* и в найденных результатах откройте приложение;

* Воспользуйтесь комбинацией клавиш **Win**+ **R** для открытия диалога

**«Выполнить»**. В диалоговом окне **«Выполнить»**, в поле **«Открыть»** введите

***msconfig*** и нажмите на кнопку**«ОК»**.

На рисунке 1 отображена утилита **«Конфигурация системы»**:

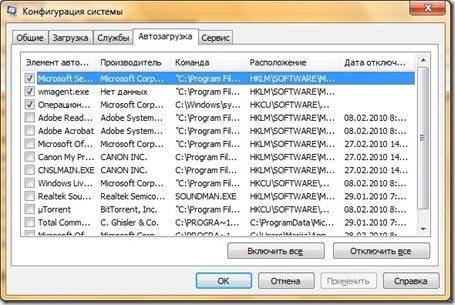


Рисунок1 - Утилита **«Конфигурация системы»**:

В текущей утилите существует пять вкладок:

* **Общие**. На этой вкладке вы можете выбрать вариант загрузки: **«Обыч- ный запуск»** – операционная система запускается обычным способом, **«Диагно- стический запуск»** - система загружается только с использованием основных служб и драйверов, а также **«Выборочный запуск»** - помимо основных служб и драйверов, с операционной системой еще загружаются выбранные службы и ав- томатически загружаемые программы.
* **Загрузка**. вы можете найти параметры загрузки операционной системы, а также дополнительные параметры отладки, такие как **«Без GUI»** - при загрузке не отображается экран приветствия, **«Информация об ОС»** - в процессе загруз- ки операционной системы отображаются загружаемые драйвера и прочее.
* **Службы**. Эта вкладка содержит список только тех служб, которые за- пускаются автоматически вместе с операционной системой, а также текущее со- стояние каждой службы..
* **Автозагрузка**. отвечает за загрузку приложений, а так же определён- ных служебных утилит, загружаемых не через службы..
* **Сервис**. можете найти список диагностических средств, позволяющих следить за работоспособностью вашей системы. Для запуска любого средства, отображенного в этой вкладке, выделите его и нажмите на кнопку **«Запуск»**.

# Утилита «Восстановление системы»

**Утилита Windows XP «Восстановление системы»** позволяет возвращать последнюю удачную конфигурацию системы в случае возникновения различных проблем. Утилита хранит в своей памяти множество контрольных точек, что позволяет пользователю выбирать вариант любой сохранённой ранее конфигу- рации.

# Функциональные возможностей программы «Восстановление систе- мы» в среде Windows XP:

**Файлы данных и восстановление системы**

Поскольку «Восстановление системы» представляется как утилита для возврата более раннего состояния системы, многие пользователи ошибочно по-

лагают, что в результате её действий они могут потерять файлы, созданные ими до формирования контрольной точки. Однако программа не проверяет содержи- мое папки «Мои документы», любые файлы с известными типами расширений (к примеру, .doc или .xls), журналы электронных писем, журналы интернет- браузеров и даже файлы с паролями. Вся эта информация останется нетронутой при восстановлении.

С другой стороны, директория «Рабочий стол» не защищена, а это значит, что некоторые файлы могут быть утеряны в процессе восстановления. Следова- тельно, перед началом операции нужно переместить всё необходимое содержи- мое из папки «Рабочий стол» в каталог «Мои документы».

# Утилита Дефрагментация диска

В процессе работы, когда вы удаляете ненужные файлы, на жестком диске образуются свободные области, окруженные пространством, занятым другими файлами. Если при записи новый файл не будет расположен в свободной обла- сти, его содержимое окажется разбитым на части, расположенные в разных об- ластях жесткого диска. В результате при чтении такого файла головкам жесткого диска придется несколько раз перемещаться по диску, повышая тем самым вре- мя считывания файла (и, соответственно, снижая скорость чтения).

**Дефрагментация жесткого диска** –Для дефрагментации можно использо- вать как встроенные возможности Windows - утилиту дефрагментации, так и от- дельные программы. Чтобы запустить утилиту дефрагментации Windows, выбе- рите команду *Пуск > Все программы > Стандартные > Служебные > Дефраг- ментация диска*. Выберите в верхней области окна жесткий диск для дефрагмен- тации и щелкните на кнопке *Анализ*.

Если диск фрагментирован более чем на 15% процентов, операционная си- стема предложит выполнить дефрагментацию. На этом этапе можно выбрать де- фрагментацию, просмотреть отчет по дефрагментации, или просто закрыть окно.

# Утилита Очистка Диска

Утилита *Очистка Диска* (для ее вызова выберите команду *Пуск > Все про- граммы > Стандартные > Служебные > Очистка диска*) позволяет очистить ваш диск от различных временных файлов. Непосредственно после запуска вам будет предложено выбрать диск для очистки, а затем утилита проведет поиск ненужных файлов по выбранному диску, представив результат в следующем окне.

# Утилита Планировщик заданий

С помощью планировщика заданий можно составить расписание запуска командных файлов, документов, обычных приложений или различных утилит для обслуживания системы. Программы могут запускаться однократно, еже- дневно, еженедельно или ежемесячно в заданные дни, при загрузке системы или регистрации в ней, при возникновении определенных событий, а также при без- действии системы (idle state). Планировщик позволяет задавать достаточно сложное расписание для выполнения заданий, в котором определяются продол-

жительность задания, время его окончания, количество повторов, зависимость от состояния источника питания (работа от сети или от батарей) и т. п.

# Утилита «Сведения о системе».

Эта программа предоставляет достаточно много информации о Вашей си- стеме. Для ее запуска открываем ***«Пуск»*** → ***Все программы*** (или ***Программы***) → ***Стандартные*** → ***Служебные*** →***Сведения о системе***, в итоге выйдет окно про- граммы, разделенное на две части – слева приведен **список категорий**, которые можно раскрыть, нажав на значок «+», справа отображается **информация**, для выбранной категории.

# Практическая часть

1. Изучить приведенный выше материал по некоторым утилитам, входя- щим в состав операционной системы Windows.
2. Рассмотреть имеющиеся в ОС утилиты, открыв их перечень с помощью

«Пуск» - «Программы» - «Стандартные» - «Служебные».

1. Изучить диалоговые окна утилит, сделать их скриншоты для отчета о практической работе.

# Содержание отчета:

1. Подготовить отчет о практической работе по теме «Утилиты операци- онной системы».
2. Сделать выводы по работе.

# Практическая работа № 4 Операционная система Windows. Интерфейс пользователя

**Цель работы:** отработка основных действий пользователя в среде опера- ционной системы MS Windows: работа с объектами интерфейса.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Теоретическая часть

**Основные понятия**

**Рабочий стол** (РС) — исходное состояние диалоговой среды MS Windows. РС раскрывается на экране после запуска MS Windows. На «поверхности» РС располагаются ярлыки наиболее часто используемых приложений, документов, папок, устройств.

**Объект** — любой элемент в среде Windows, в том числе: рабочий стол, ок- но, папка, документ (файл), устройство, приложение (программа). Объект обла- дает определенными свойствами, над ним могут быть произведены определен- ные действия.

**Контекстное меню** — меню, связанное с объектом. Контекстное меню раскрывается щелчком правой кнопки мыши, если указатель мыши установлен на объекте. Через контекстное меню можно просмотреть свойства объекта (в не- которых случаях их можно изменить), а также выполнить допустимые действия над объектом.

**Панель задач** – обычно располагается в нижней части рабочего стола (может быть перемещена к любому краю). Содержит кнопки активных про- грамм, документов. Щелчок мышкой по кнопке раскрывает окно соответствую- щего приложения. На панели задач располагается кнопка «Пуск».

**Пуск** — кнопка открытия Главного меню.

## Специальные папки Рабочего стола

**Мой компьютер** – специальная папка, которая позволяет просматривать содержимое дисков компьютера и выполнять различные операции с файлами и папками (запуск программ, копирование, перемещение, удаление файлов, созда- ние папок и др.).

**Корзина** – специальная папка, содержащая список всех удалённых файлов и папок.

**Мои документы** – системная папка, которая по умолчанию используется для хранения документов, созданных пользователем. Реально она находится на жёстком диске, в разделе с операционной системой, в каталоге «Documents and Settings».

**Сетевое окружение** – специальная папка, которая используется для про- смотра содержимого дисков компьютеров, подключенных к локальной сети, и выполнения различных операций на них.

**Контекстное меню** содержит основные команды по управлению объек- том. Для вызова контекстного меню нужно щелкнуть на объекте правой клави- шей мыши. Например, если щелкнуть правой клавишей мыши на панели задач, появится контекстное меню панели задач, которое содержит команды для управ- ления окнами активных приложений. Пункты меню выбираются, как обычно, нажатием левой кнопки мыши.

**Панели инструментов –** после выбора этого пункта откроется подменю, с помощью которого можно выводить и убирать панели инструментов. Если ря- дом с именем панели стоит метка √, то панель выведена на экран.

**Окна каскадом** – упорядочение окон активных приложений каскадом, за исключением свернутых на панель задач.

**Окна сверху вниз** – упорядочение окон активных приложений горизон- тальными полосами.

**Окна слева направо** – упорядочение окон активных приложений верти- кальными полосами.

**Свернуть все окна** – свертывание всех активных приложений на панель задач.

**Свойства** – вызов диалогового окна для настройки панели задач.

**Основные виды окон** - диалоговое окно, окно папки, окно справочной си- стемы, окно программы, окно документа.

**Диалоговое окно** - окно, появляющееся на экране при вводе команды, вы- полнение которой требует от пользователя ввести дополнительные данные, не- обходимые для дальнейшей работы программы (например, Оk или Yes («Гото- во», «Принять», «Да»»и т.п.) и Cancel или No («Отменить», «Отказаться»,

«Нет»)).

# Диалоговые окна содержат следующие элементы управления:

* **Вкладки (закладки)** - предназначены для выполнения некоторых функций или команд в окне;
* **Кнопка** - элемент управления в интерфейсе пользователя, который предназначен для выполнения команд. По форме кнопка может быть прямо- угольником с надписью или значком с рисунком. Поскольку надпись на кнопке может быть очень краткой, а рисунок - символическим и не сразу понятным, то во многих приложениях используются подсказки (примечания). Подсказка появ- ляется в виде текста в рамке, если на кнопку навести указатель мыши;
* **Надпись** со статическим текстом обычно используется для вывода за- головков. Часто надпись размещается рядом с элементом управления, который не имеет собственного заголовка. К числу таких элементов, например, относятся объекты **поле** и **счетчик**;
* **Поле** - элемент управления, предназначенный для ввода и редактиро- вания данных. Вводимый текст может быть длиннее стороны прямоугольника, ограничивающего поле, т.е. может быть похож на бегущую строку;
* **Счетчик** - элемент управления, предназначенный для изменения чис- лового значения, выводимого в поле. Чаще всего счетчик размещается рядом с полем. Счетчик, по существу, состоит из двух кнопок - для увеличения или уменьшения;
* **Поле со списком** - позволяет выбрать элемент из списка или ввести данные вручную. Текущее значение отображается в поле, а список возможных значений раскрывается при нажатии кнопки со стрелкой;
* **Переключатели (радиокнопки)** - используются для предоставления возможности выбора одного варианта из нескольких (многих). В одной группе переключателей можно выбрать только один;
* **Флажок** - используется для выбора одной или нескольких позиций из предложенного списка. Представляет из себя квадратик, который пользователь может пометить галочкой. Для отмены действия достаточно повторно щелкнуть мышью в квадратике. Заголовок - это название флажка, поясняющее его смысл;
* **Регулятор** - устанавливает одну из позиций на шкале перемещением движка (больше, меньше).
* **Окно справочной системы** - окно, которое выводит справочную ин- формацию о том объекте, с которым работает пользователь. Обычно появляется при нажатии на клавишу F1.

# Практическая часть

1. В практической части вставить в нужных местах приведенного выше текста скриншоты, иллюстрирующие описываемые объекты (окна папок, напри- мер, Мои документы; диалоговые окна, окна справочной системы и т.д.).
2. Обозначить стрелками и надписями элементы окна, например, как пока- зано на приведенном ниже рисунке 1. Для этого можно воспользоваться Пане- лью рисования – Автофигуры – Выноски (в версии Word 2003) или закладкой Вставка – Фигуры – Выноски (в версии Word 2007).

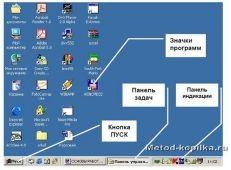
 

Рисунок 1 – Рабочий стол

1. Открыть 3 окна: Word, Мои документы, Мой компьютер. По очереди располагать окна следующими способами: **Окна каскадом, Окна сверху вниз, Окна слева направо, Свернуть все окна**. Вставить в отчет скриншоты, иллю- стрирующие эти действия, и подписать их.
2. Выполнить задание в приведенной таблице 1.

Таблица 1 – Соответствие копок заданию

|  |  |  |
| --- | --- | --- |
| **№ п⁄п** | **Задание** | **Кнопки** |
| 1 | Обведите ту кнопку, на которой необходимо щелкнуть, чтобы увеличить  окно программы. |  |
| 2 | Обведите ту кнопку, на которой необходимо щелкнуть, чтобы окно про-  граммы не занимало весь экран. |  |
| 3 | Обведите ту кнопку, на которой необходимо щелкнуть, чтобы закрыть окно  программы. |  |
| 4 | Обведите ту кнопку, на которой необходимо щелкнуть, чтобы свернуть окно  программы. |  |

# Содержание отчета:

1. . В теоретической части раскрыть понятия «интерфейс», «интерфейс пользователя», указать, какие виды интерфейса существуют.
2. Вставить скриншоты, иллюстрирующие описываемые объекты
3. Оформить отчет о практической работе.
4. Сделать выводы по работе.

**Задания для самостоятельной проверки:** 1. Какие папки вы знаете? 2. Перечислите системные папки? 3 Для чего используется флажок? 3. какими спо- собами можно расположить окна? 4. Что можно изменять с помощью переклю- чателя?

# Практическая работа № 5 Операционная система Windows.

# Работа с файлами и каталогами

**Цель работы:** ознакомиться с файловой структурой ОС Windows и сер- висными возможностями программы Проводник, научиться выполнять стан- дартные операции с папками: создавать, выделять, копировать, перемещать и удалять, переименовывать, менять свойства, скрывать.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Теоретическая часть

**Операции с папками:**

Большинство задач Windows включают в себя работу с файлами и папками. Папки используются Windows для создания системы хранения файлов на ком- пьютере аналогично тому, как картонные папки используются для систематиза- ции данных в картотеке. Папки могут содержать файлы различных типов — до- кументы, музыкальные клипы, изображения, видео, программы и др. Можно со- здавать новые папки, копировать или перемещать в них файлы из других мест — из других папок, с других компьютеров или из Интернета. В папках можно со- здавать подпапки.

1. **Создание папок:** команда *Создать папку* в контекстном меню или в верхнем меню *Файл/Создать папку* (или по-другому). Появится папка с именем *Новая папка*, написать новое название и нажать Enter для подтверждения. Мож- но переименовать папку, щёлкну на ней правой кнопкой мыши и выбрав в Кон- текстном меню команду *Переименовать*.
2. **Копирование папок:** выделить одну или несколько папок, которые нужно скопировать, выбрать команду *Копировать* в контекстном меню, в меню *Правка* верхнего меню или на панели инструментов, открыть папку, в которую нужно вставить копии и выбрать команду *Вставить* любым вышеуказанным способом.
3. **Перемещение папок:** выполняется аналогично копированию, но вместо команды *Копировать* выбирается команда *Вырезать*. Операция переме- щения отличается от операции копирования тем, что в результате копирования создаются копии папок, а исходные папки остаются на месте, в результате пере- мещения исходные папки перемещаются в другое место (там, где они были, их нет).
4. **Выделение папок:** чтобы выделить несколько подряд идущих фай- лов или папок, выберите первый объект и, удерживая нажатой клавишу SHIFT, выберите последний объект. Для выбора разрозненных файлов или папок щелк- ните поочередно каждый объект, удерживая нажатой клавишу CTRL.
5. **Удаление папок:** выделить одну или несколько папок, которые нужно удалить, выбрать команду *Удалить* в контекстном меню, в слове *Файл* верхнего меню или на панели инструментов, или перетащить выделенные папки на значок *Корзины* при нажатой левой кнопке мыши.
6. **Скрытие папки:** щелкните на значке папки правой кнопкой мыши и выберите команду *Свойства*. На вкладке *Общие* установите флажок *Скрытый*. Для просмотра скрытых папок нужно выбрать команду *Свойства папки* в меню *Сервис*. На вкладке *Вид* в группе *Дополнительные параметры* выбрать вариант *Показывать скрытые файлы и папки.*
7. **Изменение свойств папки**: команда *Свойства папки* меню *Сервис*. Можно изменить параметры щелчка мыши для выделения и открытия папки, установить дополнительные параметры для папок и т.д.

# Сервисные возможности программы Проводник.

Все папки и файлы образуют на дисках *иерархическую файловую структу- ру.* Все файлы находятся в папках, которые вложены в другие папки, более вы- сокого уровня. Папка самого высокого уровня называется *корневой*. Назначение файловой структуры – обеспечить однозначное отыскание файла, если известно его имя и путь поиска. Создание и обслуживание файловой структуры – одна из основных функций операционной системы.

Для розыска файлов нужны специальные *навигационные средства.* Они позволяют просмотреть файловую структуру, найти нужную папку или файл и выполнить с ними необходимые действия. *Проводник* – это служебная програм-

ма, специально предназначенная для просмотра файловой структуры и её обслу- живания. Окно этой программы состоит из двух панелей. на левой в виде дерева представлена вся файловая структура компьютера. Плюс рядом с папкой означа- ет, что в ней есть вложенные папки. С помощью левой панели очень удобно про- сматривать содержимое всех вложенных папок. На правой панели отображается содержимое папки, открытой в данный момент на левой панели. Между панеля- ми очень легко взаимодействовать: копировать папки, перемещать их, удалять в корзину.

# Практическая часть Методы запуска программы Проводник.

Изучить самостоятельно во время выполнения практической работы и запи- сать в отчёте!

# Порядок выполнения:

1. Откройте окно *Мой компьютер*/диск D, создайте папку и назовите ее своей фамилией*.*
2. Создайте в своей папке следующую структуру папок: три папки *Кино*, *Музы- ка*, *Литература*; в каждой из них ещё по три папки (назовите их по-своему: фамилии актёров, фильмы, музыкальные группы, книги, авторы и т.д.).
3. Скопируйте папку *Кино* со всем её содержимым в папку Литература.
4. Выделите несколько папок, стоящих рядом, затем несколько папок, стоящих вразброс.
5. Папку Литература переименуйте и назовите *Книги*.
6. Переместите содержимое папки *Музыка* в папку *Книги*.
7. Запустите программу *Проводник* с помощью *Главного меню* (Пуск/Все про- граммы/Стандартные/Проводник). Обратите внимание на то, какая папка от- крыта на левой панели Проводника в момент запуска. Это должна быть папка Мои документы.
8. Разыщите на левой панели папку своей группы и откройте ее одним щелчком на значке папки. Её содержимое должно появиться на правой панели Провод- ника.
9. На правой панели раскройте в своей папке все плюсы, чтобы раскрылось со- зданное Вами «дерево». Оно должно выглядеть как на рисунке 1:

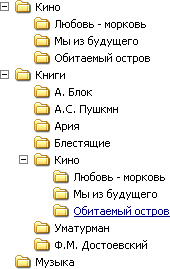


Рисунок 1 - Дерево каталога

1. Раскройте папку Музыка на правой панели и создайте в ней какую-нибудь папку.
2. Убедитесь, что на левой панели рядом с папкой Музыка появится плюс.
3. На левой панели перенесите только что созданную папку на значок Корзи- ны.
4. Откройте Корзину и убедитесь, что эта папка там присутствует.
5. Остальные папки удалите разными способами во время ответа преподавате- лю!
6. Выполните исследовательскую работу, результаты которой запишите в от- чёт.

# Содержание отчета:

* 1. В теоретической части показать операции с папками, сервисные воз- можности программы Проводник..
  2. Вставить скриншоты, иллюстрирующие описываемые объекты
  3. Оформить отчет о практической работе.
  4. Сделать выводы по работе.

**Задания для самостоятельной проверки: 1.** Какими способами можно создать папки? 2 Как по другому можно назвать папку?. 3 Что такое проводник? 4 Каким способом можно вызвать проводник?. 5 Как скрыть папки?6 Как выде- лить смежно- расположенные папки?

# Практическая работа № 6 Операционная система Windows. Работа с папка- ми, дисками

**Цель работы**: изучить основные приемы работы с файлами и папками. возможности управления интерфейсом и отображением информации в окне про- граммы Проводник. Изучить основные приемы работы с файлами и папками.

Освоить программу Поиск и ее основные функциональные возможности..

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Практическая часть

**Задание 1.** Создайте папку **Фамилия** в имеющейся папке **D:\Мои доку- менты\ Группа\**. Создайте текстовый файл ***Информация*** в папке**Фамилия.**

1. В папке **Мои документы** создайте (откройте) папку **Группа**.
2. В папке **Группа** создайте папку **Фамилия**:
   * перед созданием нужно войти в папку **Группа.**
3. В папке **Фамилия** создайте текстовый файл ***Информация***, в который введите дату занятия, фамилию, имя, отчество, факультет, группу, № аудитории и № компьютера:
   * в контекстном меню выбрать команду **Создать/ Текстовый файл** (или

# Документ Microsoft Word);

* ввести имя ***Информация***, [**Enter**];
* открыть файл двойным щелчком мыши;
* ввести содержимое файла;
* сохранить файл командой **Файл/ Сохранить** или кнопкой панели ин- струментов
* закрыть текстовый редактор **Файл/ Выход**.

**Задание 2.** Дополните папку **Группа** структурой папок в соответствии со схемой 1, где во втором уровне папок необходимо создать папки с именами 4-х имеющихся факультетов, начиная с того, который соответствует номеру Вашего компьютера в перечне факультетов на сайте университета **bseu.by**, а в третьем уровне – папки с названиями всех кафедр выбранных факультетов.

# III. Программа Поиск. Работа с найденными файлами. Создание яр- лыков

**Задание 3.** Найдите загрузочный файл программы **Калькулятор** и создай- те к нему ярлык в папке **Группа.**

1. Найдите на винчестере загрузочный файл **calc.exe** программы **Кальку- лятор**:
   * открыть окно поиска командой **Пуск/Найти/Файлы и папки**;
   * в поле **Где искать** выбрать **Мой компьютер** или **Локальные диски С:,**

# D:;

вать!

- в поле **Имя** ввести ***calc.exe***. Нажать [**Найти**]. Окно **Найти** не закры-

1. В папке **Группа** создайте ярлык для загрузочного файла **Calc.exe**:
   * выделить щелчком мыши файл **calc.exe** в окне **Найти**;
   * скопировать файл в буфер командой **Правка/ Копировать**
   * открыть в программе **Проводник** папку **D:\ Мои документы\ Группа**;

# лык .

* + вызвать на правой панели контекстное меню и команду **Вставить яр-**

1. Проверьте работоспособность ярлыка программы **Калькулятор**:
   * двойным щелчком мыши по ярлыку открыть программу **Калькулятор**;
   * закрыть программу Калькулятор.

**Задание 4.** С помощью программы **Поиск** найдите файлы по заданным

критериям поиска и скопируйте их в папку **Фамилия** или **Группа** (по заданию)**.**

* + 1. Найдите в папке **D:\Мои документы** все файлы, созданные в приложе- нии **Excel** (с расширением ***.xls)*** с 5-го по 25-е число предыдущего месяца. Мини- мальный из них скопируйте в свою папку **Фамилия**:
       - в окне **Найти** в поле **Папка** выбрать **D:\Мои документы**;
       - в поле **Имя** ввести имя группы файлов с любыми именами (\*) и задан- ным расширением: ***\*.xls***;
       - во вкладке **Дата** установить диапазон поиска по дате;
       - найденные файлы упорядочить по размеру;
       - выделить файл минимального размера щелчком мыши;
       - скопировать его в буфер командой **Правка/ Копировать**;
       - в окне **Проводника** открыть папку **Фамилия**;
       - вставить файл из буфера командой **Правка/ Вставить**.
    2. Найти на винчестере загрузочный файл программы **Блокнот**

**NotePad.exe** и создать для него ярлык в папке **Группа** (см п. 9, 10).

* + 1. Найти в папке **Мои документы** соседнего компьютера все файлы, со- зданные в редакторе **Word** (расширение ***.doc***) за последнюю неделю. Два мини- мальных из них скопировать в свою папку **Фамилия**.
    2. Найти на винчестере все файлы, имеющие в имени не более 3-х симво- лов. Скопировать самый ранний из них в папку **Группа** и переименовать его в файл **Пример**.

**Задание 5.** Создайте ярлык к собственной папке **Группа** на **Рабочем сто-**

**ле**.

# Содержание отчета:

1. В теоретической части показать работу с папками, дисками, вставить скриншоты, иллюстрирующие описываемые объекты
2. Оформить отчет о практической работе.
3. Сделать выводы по работе.

# Практическая работа № 7 Операционная система Windows.

# Конфигурирование системы

**Цель работы:** изучить возможности и получить навыки работы с Настройкой систем на базе Windows7, XP.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 2.3 ПК-3.3

# Теоретический материал

**Настройка системы** – это диагностический инструмент, созданный для настройки параметров запуска Windows 7, в целях выявления причин неполадок в работе компьютера и операционной системы. С помощью программы «Конфи- гурация системы» можно выявить драйверы, программы и компоненты, из-за не- корректной работы которых возникают ошибки во время запуска и функциони- рования Windows 7.

# Настройка системы Windows 7

**Запуск программы Настройка системы**

Чтобы запустить программу ***«Настройка системы»,*** откройте меню Пуск, введите в поисковую строку **msconfig** и нажмите ***Ввод***, как показано на рисунке 1***.***



Рисунок 1 – Запуск программы

Также можно воспользоваться сочетанием клавиш Windows+R, ввести

msconfig и нажать OK, как показано на рисунке 2***.*** .

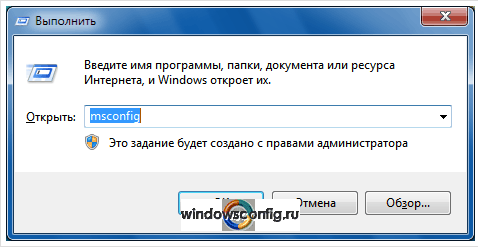


Рисунок 2 – Запуск программы

Ознакомьтесь с внешним видом окна утилиты **Настройка системы**. **Вкладка «Общие»**

На первой вкладке, **Общие**, можно выбрать вариант запуска операционной системы.

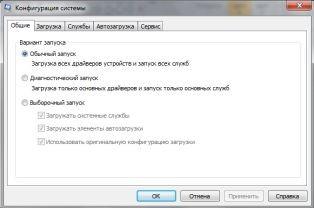


Рисунок 3 – Настройка системы/Общие

На вкладке ***Общие*** можно выбрать один из трёх вариантов запуска опера- ционной системы:

## Обычный запуск

В этом режиме Windows 7 запускается обычным способом. «Обычный за- пуск» используется, когда нет проблем с загрузкой ОС или после устранения неполадок.

## Диагностический запуск

В режиме диагностического запуска вместе с Windows запускаются только основные службы и драйверы, необходимые для функционирования операцион- ной системы и компьютера.

## Выборочный запуск

В этом режиме запуск Windows 7 производится с использованием основ- ных служб и драйверов, а также других служб и автоматически загружаемых программ, выбранных пользователем. Доступны три дополнительных параметра:

* ***Загружать системные службы*** – если этот параметр включен, то операционная система загружается со стандартным набором служб, необходи- мых для её работы.
* ***Загружать элементы автозагрузки*** – если этот параметр включен, то вместе с операционной системой запускаются программы, отмеченные флажка- ми на вкладке Автозагрузка.
* ***Использовать оригинальную конфигурацию загрузки*** – этот параметр по умолчанию включен и затенен (отображается серым цветом). Данный пара- метр восстанавливает изначальные настройки запуска Windows 7 в случае внесе- ния изменений на вкладке ***Загрузка.***

# Вкладка «Загрузка»

На вкладке ***Загрузка*** находятся детальные настройки параметров запуска

Windows 7, как показано на рисунке 4.

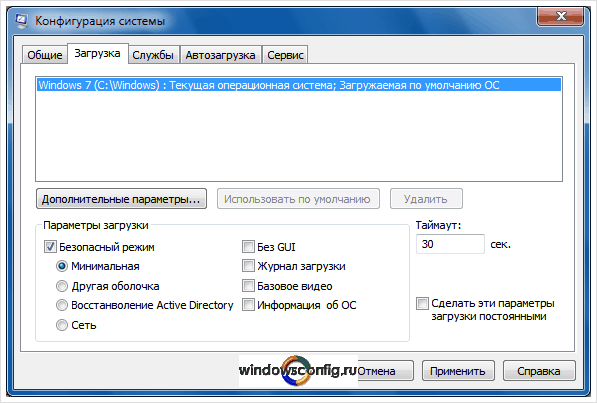


Рисунок 4 – Настройка системы/Загрузка

# Операционная система по умолчанию

Если на компьютере установлено несколько операционных систем, можно назначить любую из них загружаемой по умолчанию. Чтобы сделать это, выде- лите нужную операционную систему и нажмите кнопку ***Использовать по умол- чанию.***

# Безопасный режим

Безопасный режим – это режим работы операционной системы с ограни- ченным набором служб, устройств и драйверов, необходимых для функциониро- вания компьютера.

Службы Windows, запускающиеся в безопасном режиме:

* «Журнал событий Windows»;
* «Поддержка самонастраивающихся устройств Plug and Play»;
* «Удалённый вызов процедур (RPC – Remote Procedure Call)»;
* «Службы криптографии»;
* «Защитник Windows»;
* «Инструментарий управления Windows (WMI – Windows Management Instrumentation)».

Устройства и драйверы, запускающиеся в безопасном режиме:

* «Внутренние жёсткие диски (ATA – Advanced Technology Attachment, SATA – Serial ATA, SCSI – Small Computer System Interface)»;
* «Внешние жёсткие диски (USB – Universal Serial Bus)»;
* «Дисководы гибких дисков (внутренние и USB)»;
* «Внутренние дисководы для компакт-дисков и DVD-дисков (ATA, SCSI)»;
* «Внешние USB-дисководы для компакт-дисков и DVD-дисков»;
* «Клавиатуры и мыши (USB, последовательный порт)»;
* «Видеокарты VGA – Video Graphics Array)».

**Дополнительные параметры загрузки** показаны на рисунке 5

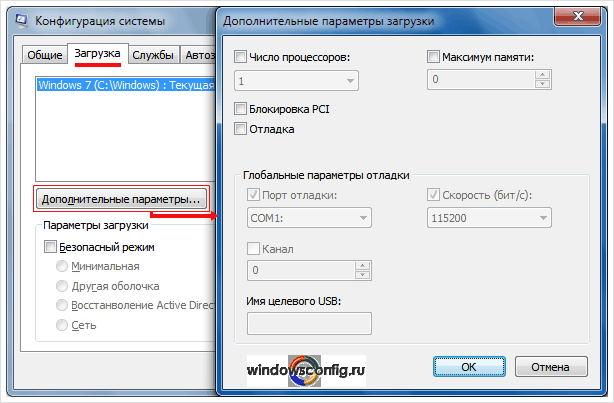


Рисунок 5 – Дополнительные параметры

Чтобы настроить дополнительные параметры загрузки Windows 7, на вкладке ***Загрузка*** нажмите кнопку ***Дополнительные параметры****.* К **Дополни- тельным параметрам** относятся:

* «Число процессоров». С помощью этого параметра можно ограничить количество как реальных, так и виртуальных процессоров, используемых в си- стеме.
* «Максимальный объём памяти». С помощью этого параметра можно ограничить объём физической оперативной памяти, используемый операцион- ной системой.
* «Блокировка PCI (Peripheral component interconnect)». Если этот пара- метр включен, то операционная система не распределяет ресурсы ввода-вывода и прерывания на шине PCI. При этом ресурсы ввода-вывода и памяти, заданные в BIOS, сохраняются.
* «Отладка». Если этот параметр включен, можно задать глобальные па- раметры отладки в режиме ядра для разработчиков драйверов устройств.

# Вкладка «Службы»

Перейдите на вкладку **Службы**. Здесь представлен список всех служб, установленных в системе. Каждая служба представляет собой некое приложение, работающее в фоновом режиме.

# Вкладка «Автозагрузка»

Перейдите к последней вкладке, **Автозагрузка**, и убедитесь, что в списке приложений, автоматически запускаемых при загрузке системы, есть **Блокнот**.

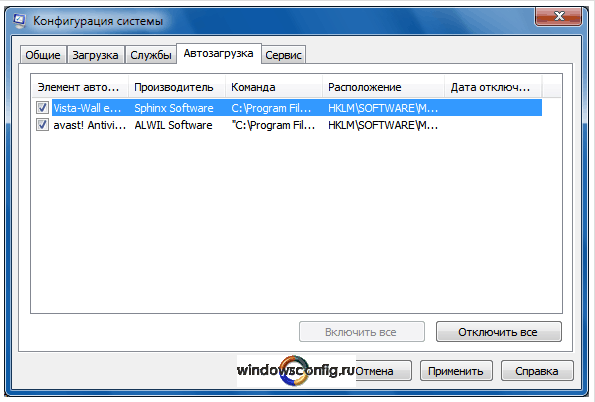


Рисунок 7– Настройка системы/Автозагрузка

Вкладка ***Автозагрузка*** содержит список программ, автоматически запус- кающихся вместе с Windows 7:

* В столбце ***Элемент автозагрузки*** отображается название программы;
* В столбце ***Производитель*** – разработчик программы;
* В столбце ***Команда*** указан исполняемый файл, запускающийся вместе с ОС, а также местоположение этого файла;
* В столбце ***Расположение*** отображается ключ реестра, отвечающий за автоматический запуск программы вместе с Windows 7;
* В столбце ***Дата отключения*** указана дата отключения элементов авто- загрузки, не запускающихся автоматически вместе с операционной системой.

# Вкладка «сервис»

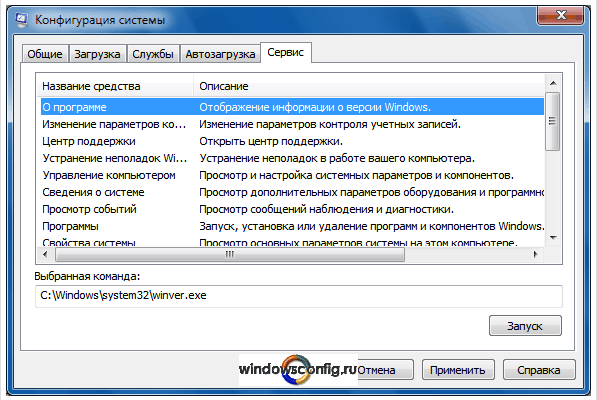


Рисунок 8 – Настройка системы/Сервис

Вкладка ***Сервис*** позволяет быстро запустить средства настройки, админи- стрирования и диагностики Windows. Выделите нужное средство и нажмите кнопку ***Запуск.***

# Практическая часть

Кратко описать каждую вкладку Конфигурирования системы

# Содержание отчета:

1. В теоретической части показать как осуществляется настройка си- стемы Windows 7, вставить скриншоты, иллюстрирующие описываемые объекты
2. Оформить отчет о практической работе.
3. Сделать выводы по работе.

# Задания для самостоятельной проверки

1. Что позволяет сделать Вкладка ***Сервис***?
2. Как осуществляется Запуск программы **Настройка системы?**
3. Для чего предназначен редактор реестра?
4. Для чего предназначен Диспетчер задач?
5. Что содержит Вкладка ***Автозагрузка***?

# Практическая работа № 8 Операционная система Windows.

# Управление виртуальной памятью, настройка файла подкачки

**Цель работы**: научиться осуществлять настройку файла подкачки.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-3 ОК-4 ОК-5 ОК-6 ОК-7 ПК- 3.1 ПК-3.

# Теоретическая часть

**ОС Windows использует не только оперативную память для своей ра- боты. Чтобы немного** разгрузить ресурсы оперативной памяти, на жестком диске создается специальный файл, в котором ОС также хранит текущие дан- ные. Называется он **swap - файлом,** или **файлом подкачки,** а также виртуаль- ной памятью компьютера. **Оперативную память** называют физической, т.к. она создана из конкретного материала, т.е. ее можно взять в руки и рассмотреть. **Виртуальная память** - ненастоящая, это область дискового пространства, кото- рую ОС тоже считает памятью. Слово «виртуальный» в данном случае подразу- мевает «созданный на компьютере»

**Файл подкачки** автоматически создается ОС в корневой папке того дис- ка, где расположена сама система. Его размер определяется исходя из объема физической (перативной) памяти компьютера. По умолчанию минимальный размер файла подкачки соответствует полутора размерам физической памяти, а максимальный размер обычно превышает ее в 3 раза.

Размер файла подкачки и его расположение можно изменять. Для этого откройте вкладку *Дополнительно* (ПКМ Мой компьютер-дополнительно- быстродействие-параметры-виртуальная память )в окне *параметры быстродей-*

*ствия* и в области *Виртуальная память* нажмите кнопку *изменить.* Откроется окно *Виртуальная память.*

Текущая информация о размере файла подкачки отображается в нижней части данного окна в области Общий объем файла подкачки на всех дисках по- казан на рисунке 1.

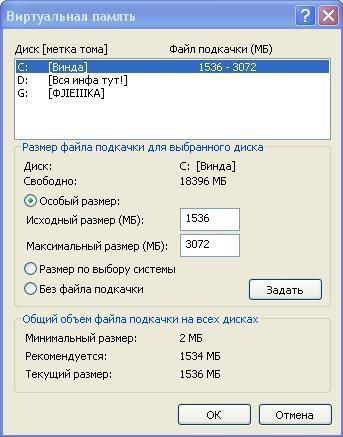


Рисунок 1 – Виртуальная память

*Обратите внимание на то, что*:

1. Если у вас установлено несколько жестких дисков (не логических разделов, именно жестких дисков), логично установить файл подкачки на самый быстрый из них. При этом лучше, чтобы ОС находилась на другом диске.
2. Можно разместить файл подкачки на нескольких жестких дисках одновременно.
3. Если у вас только один жесткий диск, разбитый на разделы, лучше всего установить файл подкачки на самом первом из них, физически располо- женном ближе к краю диска.
4. Не нужно устанавливать файл подкачки сразу на нескольких логиче- ских дисков одного жесткого. Это только замедлит работу системы, т.к. магнит- ным головкам жесткого диска придется постоянно перепрыгивать с места на ме- сто.
5. Если у вас много оперативной памяти (512 Мбайт и выше), то разме- ры файла подкачки по умолчанию (от 1 Гбайт до 1,5 Гбайт) будут неоправданно занимать пространство жесткого диска. Однако, как вы знаете, полностью от- ключать файл подкачки нельзя. Поэтому поступите так: установите переключа- тель в положение *особый размер.* Далее в зависимости от ресурсов вашего жест- кого диска возможно несколько вариантов:
   * если объем жесткого диска достаточно большой, то в поле *Исходный размер* введите значение 512 Мбайт. *Максимальный размер* зависит от приложе- ний, с которыми вы работаете (не более 1 Гбайт) o если свободного места на диске маловато, то установите *исходный размер* 2 Мбайт. *Максимальный размер*

при этом должен быть не меньше, чем размер физической памяти. Установите его равным ей.

* + если оперативной памяти не так уж много (менее 256 Мбайт), то нужно оставить параметры, принятые по умолчанию, либо выбрать режим Размер по выбору системы, при котором Windows, собирая информацию об использовании файла подкачки в процессе работы, при необходимости сможет корректировать (увеличивать) его размеры.

После внесения изменений в настройки файла подкачки компьютер нужно перезагрузить (при уменьшении его размеров либо при создании нового файла подкачки система сама предложит вам это). Затем для оптимизации файла под- качки следует запустить программу дефрагментации.

# Распределение работы памяти и процессора.

Windows позволяет настроить некоторые дополнительные параметры управления быстродействием системы.

Для знакомства с ним перейдите на вкладку *Дополнительно* окна *пара- метры быстродействия.*

Обратите внимание на область *Распределение времени процессор.* По умолчанию переключатель *Оптимизировать работу* находится в положении *программ.* Однако если вы работаете с большим количеством программ одно- временно, то для увеличения стабильности работы специалисты рекомендуют установить переключатель в положении *служб работающих в фоновом режиме.* Однако, при этом стабильность работы повысится, скорость выполнения теку- щего активного приложения немного снизится.

В области *Использование памяти по умолчанию* ресурсы памяти также оп- тимизируют работу программ. Если вами программы работают с файлами боль- ших объемов, то стоит установить переключатель оптимизировать работу в по- ложение системного КЭШа.

Уменьшение дискового пространства, занимаемого Windows. Для умень- шения дискового пространства нужно удалить все неиспользуемые программные компоненты Windows (MSN Explorer, Paint)

*Очистка дисков*. Для очистки дисков в Windows встроены специальные служебные программы, найти которые можно в программной группе *пуск- все программы - стандартные - служебные - очистка диска.* После запуска про- грамма *очистка диска* оценит, сколько мусора поднакопилось в ваших закромах. Затем выдаст результаты проверки.

# Диспетчер задач.

Для каждого запущенного приложения ОС отводит определенный объем оперативной памяти, или другими словами, задачу. Каждая задача, в свою оче- редь, создает один или несколько *процессов* - отдельных процедур выполняю- щих конкретную функцию задачи и занимающих строго определенное место в адресном пространстве памяти. Для управления задачами и процессами в Win- dows предназначена специальная служебная программа - Диспетчер задач (рис.

2). Вызвать его можно либо с помощью контекстного меню панели задач, либо нажав комбинации клавиш Ctrl-Shift-Esc.

В строке состояния располагается информация об общем количестве про- цессов в памяти, загрузке процессора и общем количестве физической и вирту- альной памяти, используемой этими процессами.

Диспетчер задач является отличным инструментом борьбы с зависшими приложениями. Если в поле состояние напротив той или иной задачи появилось значение не отвечает, то нужно только выделить такую задачу и нажать кнопку снять задачу, после чего спокойно продолжить работу. чтобы переключить на какую-либо задачу, нужно выделить мышью ее название в поле Задача и нажать кнопку *Переключиться.* Для запуска новой задачи - кнопка новая задача и т.д.

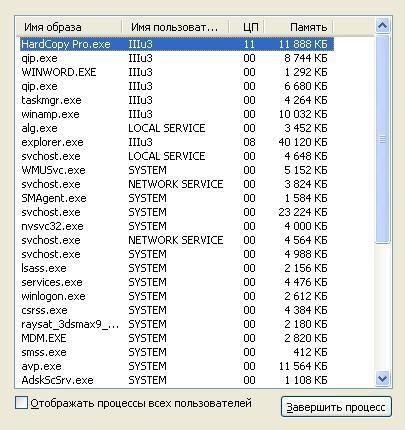


Рисунок 2 - Диспетчер задач

# Практическая часть

**Задание 1.** Быстродействие системы

1. Вызовите окно *Свойства системы* и перейти в нем на вкладку *До- полнительно*, как показано на рисунке 2

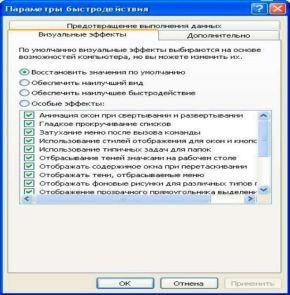


Рисунок 2 – Параметры быстродействия памяти

1. Нажмите в области быстродействие на кнопку *параметры.*
2. В открывшемся окне *Параметры быстродействия* установите по- ложение *Обеспечить наилучшее быстродействие,* которое сделает картинку намного скромнее, зато увеличит производительность системы.
3. Поэкспериментируйте с меню *особые эффекты* для изменения визу- альных эффектов. Запишите полученные результаты.
4. Верните состояние операционной системы в прежние настройки, для этого установите переключатель в положение *Восстановить значения по умол- чанию* (рис. 4).

**Задание 2.** Очистка дисков.

1. Вызовите диалоговое окно свойств диска и прейдите на вкладку *сер- сис*, на которой находятся кнопки запуска двух полезных команд:
   1. проверка диска на наличие ошибок
   2. программа дефрагментации (способствует повышению производитель- ности системы). Чтобы найти все фрагментированные файлы и собрать их части в единое целое необходима дефрагментация.
2. Выполните дефрагментацию переносного устройства (флэшки). За- пишите увиденные результаты.
3. Оформите отчет по проделанной работе.

**Задание 3.** Файл подкачки.

1. Выберите нужный диск в верхней части окна. Если на данном диске вам не нужен файл подкачки, то установите переключатель в области *Размер файла подкачки для выбранного диска* в положение *Без файла подкачки.* Если компьютер оснащен большим количеством оперативной памяти, то может воз- никнуть соблазн убрать файлы подкачки со всех дисков.
2. Выберите параметр *размер по выбору системы,* который включает динамическое (т.е. изменяемое со временем) управление размером файла под- качки ОС.
3. Установив переключатель в положение *Особый размер,* вручную установите значения файла подкачки.
4. После внесения всех изменений нажмите кнопку *Задать* (рис. 5).

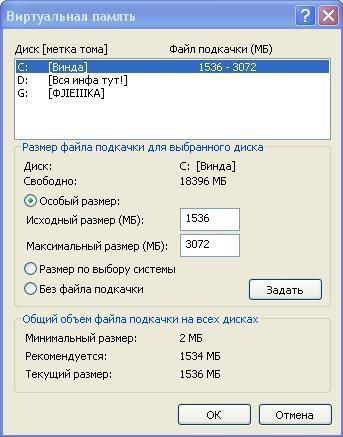


Рисунок 5 - Виртуальная память - задание файла подкачки

# Содержание отчета:

1. Титульный лист.
2. Название и цель работы.
3. Краткий конспект теоретической части.
4. Выводы к работе.

# Практическая работа № 9 Операционная система Windows.

# Управление периферийными устройствами

**Цель работы:** изучить работу различных периферийных устройств.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Теоретическая часть

Периферийное устройство (англ. peripheral) — аппаратура, которая позво- ляет вводить информацию в компьютер или выводить её из него.

Периферийные устройства являются не обязательными для работы систе- мы и могут быть отключены от компьютера. Однако большинство компьютеров используются вместе с теми или иными периферийными устройствами.

Периферийные устройства делят на три типа:

* устройства ввода — устройства, использующиеся для ввода информа- ции в компьютер: мышь, клавиатура, тачпад, сенсорный экран, микрофон, ска- нер, веб- камера, устройство захвата видео, ТВ[-тюнер;](https://ru.wikipedia.org/wiki/%D0%A2%D0%92-%D1%82%D1%8E%D0%BD%D0%B5%D1%80)
* устройства вывода — устройства, служащие для вывода информации из компьютера: видеокарта, монитор, принтер, [акустическая система;](https://ru.wikipedia.org/wiki/%D0%90%D0%BA%D1%83%D1%81%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0)
* устройства хранения(ввода/вывода) — устройства, служащие для накопления информации, обрабатываемой компьютером: накопитель на жёстких магнитных дисках (НЖМД), накопитель на гибких магнитных дисках (НГМД), ленточный накопитель, USB- [флеш-накопитель.](https://ru.wikipedia.org/wiki/USB-%D1%84%D0%BB%D0%B5%D1%88-%D0%BD%D0%B0%D0%BA%D0%BE%D0%BF%D0%B8%D1%82%D0%B5%D0%BB%D1%8C)

Иногда одно периферийное устройство относится сразу к нескольким ти- пам. Например, устройство ввода-вывода, звуковая карта, сетевая [плата.](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D0%B0)

Устройства компьютера разделили на два вида:

-внутренние (процессор, ОЗУ);

-внешние (периферийные).

Внутренние устройства реализуют определённую архитектуру, формируют аппаратную платформу компьютера. Внешние устройства не зависят от архитек- туры компьютера, расширяют возможности компьютера.

# Практическая часть

1. Составить сравнительный обзор периферийных устройств. Выяснить назначение и применение; описать основные характеристики; приложить скриншоты.

# Сделать выводы по работе

**Задания для самостоятельной проверки:**

1. Дайте определения периферийным устройствам
2. Назовите основные типы и виды периферийных устройств

# Практическая работа № 10 Средства управления локальными ресурсами

**Цель работы**: изучить и освоить основные понятия: управление процес- сами , состояние процессов, алгоритмы планирования процессов.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.3

# Теоретическая часть Управление локальными ресурсами.

Важнейшей функцией операционной системы является организация раци- онального использования всех аппаратных и программных ресурсов системы.

К основным ресурсам могут быть отнесены: *процессоры, память, внешние устройства, данные и программы.* Располагающая одними и теми же ресурсами, но управляемая различными ОС, вычислительная система может работать с раз- ной степенью эффективности. Поэтому знание внутренних механизмов операци- онной системы позволяет косвенно судить о ее эксплуатационных возможностях и характеристиках.

# Управление процессами

**Процесс** (или по-другому, задача) – абстракция, описывающая выполняю- щуюся программу. Для операционной системы процесс представляет собой еди- ницу работы, заявку на потребление системных ресурсов. Подсистема управле- ния процессами планирует выполнение процессов, то есть распределяет процес- сорное время между несколькими одновременно существующими в системе процессами, а также занимается созданием и уничтожением процессов, обеспе- чивает процессы необходимыми системными ресурсами, поддерживает взаимо- действие между процессами.

# Состояние процессов

В многозадачной (многопроцессной) системе процесс может находиться в одном из трех основных состояний:

**Выполнение** – активное состояние процесса, во время которого процесс обладает всеми необходимыми ресурсами и непосредственно выполняется про- цессором;

**Ожидание** – пассивное состояние процесса, процесс заблокирован, он не может выполняться по своим внутренним причинам, он ждет осуществления не- которого события, например, завершения операции ввода-вывода, получения со- общения от другого процесса, освобождения какого-либо необходимого ему ре- сурса;

**Готовность** – также пассивное состояние процесса, но в этом случае про- цесс заблокирован в связи с внешними по отношению к нему обстоятельствами: процесс имеет все требуемые для него ресурсы, он готов выполняться, однако процессор занят выполнением другого процесса.

В ходе жизненного цикла каждый процесс переходит из одного состояния в другое в соответствии с алгоритмом планирования процессов, реализуемым в данной операционной системе.

# Контекст и дескриптор процесса

На протяжении существования процесса его выполнение может быть мно- гократно прервано и продолжено. Для того, чтобы возобновить выполнение про- цесса, необходимо восстановить состояние его операционной среды.

Состояние операционной среды отображается состоянием регистров и программного счетчика, режимом работы процессора, указателями на открытые файлы, информацией о незавершенных операциях ввода-вывода, кодами ошибок выполняемых данным процессом системных вызовов и т.д.

Эта информация называется контекстом процесса.

**Дескриптор процесса** по сравнению с контекстом содержит более опера- тивную информацию, которая должна быть легко доступна подсистеме планиро- вания процессов.

Контекст процесса содержит менее актуальную информацию и использу- ется операционной системой только после того, как принято решение о возоб- новлении прерванного процесса..

**Создать процесс** – это значит: создать информационные структуры, опи- сывающие данный процесс, то есть его дескриптор и контекст; включить де- скриптор нового процесса в очередь готовых процессов; загрузить кодовый сег- мент процесса в оперативную память или в область свопинга.

# Алгоритмы планирования процессов

Планирование процессов включает в себя решение следующих задач:

* + определение момента времени для смены выполняемого процесса;
  + выбор процесса на выполнение из очереди готовых процессов;
  + переключение контекстов “старого” и “нового” процессов.

Первые две задачи решаются программными средствами, а последняя в значительной степени аппаратно. Существует множество различных алгоритмов планирования процессов, по разному решающих вышеперечисленные задачи, преследующих различные цели и обеспечивающих различное качество мульти- программирования.

# Вытесняющие и невытесняющие алгоритмы планирования

Существует два основных типа процедур планирования процессов – вы- тесняющие (preemptive) и невытесняющие (non-preemptive).

Non-preemptive multitasking – невытесняющая многозадачность – это спо- соб планирования процессов, при котором активный процесс выполняется до тех пор, пока он сам, по собственной инициативе, не отдаст управление планиров- щику операционной системы для того, чтобы тот выбрал из очереди другой, го- товый к выполнению процесс.

Preemptive multitasking – вытесняющая многозадачность – это такой спо- соб, при котором решение о переключении процессора с выполнения одного процесса на выполнение другого процесса принимается планировщиком опера- ционной системы, а не самой активной задачей.

# Управление реестром

**Работа с реестром из командной строки**

Работа с реестром возможна не только через утилиту **regedit.exe**, но и напрямую из командной строки с помощью утилиты **REG**, она поддерживает все возможности, которые имеет программа regedit.exe.

# Практическая часть

Письменно ответить на вопросы:

1. Что может быть отнесено к основным ресурсам ?
2. Что подразумевается под процессом?
3. Что выполняет Подсистема управления процессами?
4. Назовите три состояния процесса
5. Что такое Контекст и дескриптор процесса
6. Назовите Алгоритмы планирования процессов

**Сделать выводы по работе**

# Практическая работа № 11 Операционные системы Unix и Windows.

# Организация контроля доступа к файлам

**Цель работы:** получил навыки анализа данных

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 2.3 ПК-3.3

# Теоретическая часть

**Файловые системы Windows XP**

*Файловая система* – это набор спецификаций и соответствующее им про- граммное обеспечение, которое отвечает за создание, удаление, организацию, чтение, запись, модификацию и перемещение файлов информации, а также за управление доступом к файлам и за управление ресурсами, которые используют- ся файлами. *Файловая система* определяет способ организации данных на диске и принципы хранения данных на физическом носителе. Формат хранения дан- ных определяет основные характеристики файловой системы.

Информация на магнитных дисках размещается и передается блоками. Каждый блок называется *сектором* и располагается на концентрических дорож- ках поверхности диска. Группа дорожек одного радиуса, расположенных на по- верхностях магнитных дисков, образуют *цилиндры*. Каждый сектор состоит из поля данных и поля служебной информации, ограничивающей и идентифициру- ющей его. Размер сектора (объем поля данных) устанавливается контроллером или драйвером. Физический адрес сектора на диске определяется с помощью трех «координат»:

* номер цилиндра;
* номер рабочей поверхности диска;
* номер сектора на дорожке.

Обмен информацией между оперативно запоминающим устройством и дисками физически осуществляется только секторами. Диск может быть разбит на несколько разделов, которые могут использоваться как одной операционной системой, так и несколькими. На каждом разделе может быть организована своя файловая система. Для организации хотя бы одной файловой системы должен быть определен, по крайней мере, один раздел. Разделы могут быть двух типов:

* первичный раздел;
* расширенный раздел.

Максимальное число первичных разделов – четыре, но обязательно должен быть хотя бы один. Если первичных разделов больше одного, то один должен быть активным, в нем находится загрузчик операционной системы. На одном диске может быть только один расширенный раздел, который в свою очередь может содержать большое количество подразделов – *логических дисков*.

Операционная система *Windows XP* поддерживает работу со следующими файловыми системами:

* *FAT* (*File Allocation Table*) – файловая система, разработанная для *MS-DOS* и являющаяся основной для *Windows 3.x* и *9x*. *Windows XP* и *Windows Server 2003* поддерживают три разновидности *FAT*: *FAT12*, *FAT16 и FAT32*. Пер- вые две обеспечивают совместимость со старыми операционными системами *Microsoft*. Кроме того, *FAT12* используется как формат хранения данных на гиб- ких дисках. *FAT 32* – модифицированная версия *FAT*, используемая в *Windows 95 OSR2*, *Windows 98* и *Windows Millennium*.
* *NTFS* (*Windows NT file system*) – файловая система, разработанная специально для *Windows NT* и унаследованная *Windows 2000*, *Windows XP*, *Windows 2003*.
* *CDFS* (*Compact Disk File System*) – файловая система компакт- дисков.
* *UDF* (*Universal Disk Format*) – универсальный формат дисков, ис- пользуемый современными магнитооптическими накопителями и технологией *DVD*.
* *DFS* (*Distributed File System*) – распределенная файловая система.

## Файловая система FAT

Аббревиатура *FAT* (*File Allocation Table*) означает «таблица размещения файлов». Этот термин относится к линейной табличной структуре со сведениями о файлах – именами файлов, их атрибутами и другими данными, определяющи- ми местоположение файлов или их фрагментов в среде *FAТ*. Элемент *FAТ* опре- деляет фактическую область диска, в котором хранится начало физического файла. В файловой системе *FAT* логическое дисковое пространство любого ло- гического диска состоит из двух областей:

* + *системная область* – создается при форматировании диска и обнов- ляется при манипулировании файловой структурой;
  + *область данных* – содержит файлы и каталоги, подчиненные корне- вому каталогу, доступна через пользовательский интерфейс.

*Системная область* состоит из следующих компонентов:

* + загрузочной записи;
  + зарезервированных секторов;
  + таблицы размещения файлов (*FAT*);
  + корневого каталога.

Таблица размещения файлов представляет собой карту (образ) области данных, в которой описывается состояние каждого участка области данных. Об- ласть данных разбивается на *кластеры*. Один или несколько смежных секторов в логическом дисковом адресном пространстве (только в области данных) объеди- няются в единый дисковый блок – кластер.

*Кластер* – минимальная адресуемая единица дисковой памяти, выделяемая файлу или некорневому каталогу. Например, в *FAT16* размер кластера составля- ет 32 кбайт. Файл или каталог занимает целое число кластеров. Последний кла- стер при этом может быть задействован не полностью, что приведет к заметной потере дискового пространства при большом размере кластера.

В таблице *FAT* кластеры, принадлежащие одному файлу (некорневому ка- талогу), связываются в цепочки. Для указания номера кластера в системе управ- ления файлами *FAT16* используется 16-битовое слово, следовательно, можно хранить информацию максимум о 65536 кластерах. Так как *FAT* используется при доступе к диску очень интенсивно, она загружается в оперативную память и находится там максимально долго.

Корневой каталог отличается от обычного каталога тем, что он размещается в фиксированном месте логического диска и имеет фиксированное число элементов. Структура системы файлов является иерархической. Файлам присваиваются первые доступные адреса кластеров в томе. Номер начального кластера файла представляет собой адрес первого кластера, занятого файлом, в таблице размещения файлов. Каж- дый кластер содержит указатель на следующий кластер, использованный файлом, или индикатор (*OxFFFF*), указывающий, что данный кластер является последним кластером файла.

Файлы на дисках имеют 4 атрибута, которые могут сбрасываться и устанавли- ваться пользователем: *Archive* (архивный), *System* (системный), *Hidden* (скрытый) и *Read-only* (только чтение).

Таблица 2 - Размеры кластеров в *FAT32* по умолчанию (в *Windows*)

|  |  |  |
| --- | --- | --- |
| **№**  **п/п** | **Размер тома (Мб)** | **Размер кластера** |
| 1 | От 32 Мб до 8 Гб | 4 |
| 2 | 8-16 Гб | 8 |
| 3 | 16-32 Гб | 16 |
| 4 | 32Гб | 32 |

Файловая система *FAT* не обеспечивает функций защиты данных и автома- тического восстановления. Поэтому она используется, только если альтернатив- ной системой на компьютере является *MS-DOS* или *Windows 95/98*, а также для передачи данных на гибких дисках.

## Файловая система NTFS

Аббревиатура *NTFS* (*New Technology File System*) означает новая техноло- гия файловой системы. *NTFS* является наиболее надежной системой специально разработанной для *Windows NT* и усовершенствованной в более поздних версиях *Windows*. Она обладает характеристиками защищенности, поддерживая контроль доступа к данным и привилегии владельца, играющие важную роль в обеспече- нии целостности конфиденциальных данных. Папки и файлы *NTFS* могут иметь назначенные им права доступа вне зависимости от того, являются они общими или нет.

*NTFS* использует 64-разрядные индексы кластеров, но *Windows XP* ограни- чивает размеры томов *NTFS* до значений, при которых возможна адресация 32- разрядными кластерами, то есть до *128 Тб* (с использованием кластеров по 64 Кб).

Таблица 3 - Размеры кластеров на томах *NTFS*

|  |  |  |
| --- | --- | --- |
| №  п/п | Размер тома (Мб) | Размер кластера |
| 1 | 512 Мб и менее | 512 байт |
| 2 | 513 - 1024 Мб | 1 Кб |
| 3 | 1025 -2048 Мб | 2 Кб |
| 4 | более 2048 Мб (2 Гб) | 4 Кб |

Одно из важнейших свойств *NTFS* – самовосстановление. При неожидан- ном сбое системы информация о структуре папок и файлов на томе *FAT* может быть утеряна. *NTFS* протоколирует все вносимые изменения, что позволяет из- бежать разрушения данных о структуре тома (в некоторых случаях данные фай- лов могут быть утеряны).

Способность самовосстановления и поддержка целостности реализуется за счет использования протокола выполняемых действий и ряда других механиз- мов. *NTFS* рассматривает каждую операцию, модифицирующую системные фай- лы на *NTFS*-томах, как *транзакцию* и сохраняет информацию о такой транзак- ции в протоколе. Начатая транзакция может быть либо полностью завершена (*commit*), либо откатывается (*rollback*). В последнем случае *NTFS*-том возвраща- ется в состояние, предшествующее началу транзакции. Для того чтобы управ- лять транзакциями, *NTFS* записывает все операции, входящие в транзакцию, в файл протокола, перед тем как осуществить запись на диск. После того как тран- закция завершена, все операции выполняются. Таким образом, под управлением *NTFS* не может быть незавершенных операций. В случае дисковых сбоев неза- вершенные операции отменяются.

Под управлением *NTFS* также выполняются операции, позволяющие опре- делять дефектные кластеры и отводить новые кластеры для файловых операций. Этот механизм называется *cluster remapping*. *NTFS*, по сравнению с *FAT*, под- держивает ряд дополнительных возможностей, основные из них:

* защита файлов и каталогов;
* сжатие файлов;
* поддержка многопоточных файлов;
* отслеживание связей;
* дисковые квоты;
* шифрование;
* точки повторной обработки;
* точки соединения;
* теневые копии.

# Файловая система операционной системы UNIX

Операционная система *UNIX* представляет собой ядро многопользователь- ской операционной системы с разделением времени. Она дает пользователям возможность запускать свои программы, управляет периферийными устройства- ми и обеспечивает работу файловой системы.

Работу операционной системы *UNIX* можно представить в виде функцио- нирования множества взаимосвязанных процессов. При загрузке системы снача- ла запускается ядро (процесс 0), которое в свою очередь запускает командный интерпретатор *shell* (процесс 1). Взаимодействие пользователя с системой *UNIX* происходит в интерактивном режиме посредством командного языка. Оболочка операционной системы *shell* интерпретирует вводимые команды, запускает соот- ветствующие программы (процессы), формирует и выводит ответные сообще- ния.

Важной составной частью *UNIX* является файловая система, которая явля- ется сложной многопользовательской системой, так как в основе этой системы лежала операционная система *MULTICS*. Файловая система имеет иерархиче- скую структуру, образующую дерево каталогов и файлов. Дерево начинается в корневом каталоге, с добавлением связей, формирующих направленный ацикли- ческий граф. Имена файлов могут содержать до 14 символов, включающих в се- бя любые символы *ASCII*, кроме косой черты (использовавшейся в качестве раз- делителя компонентов пути) и символа *NUL* (использовавшегося для дополнения имен короче 14 символов). Символ NUL обозначается байтом 0.

Корневой каталог обозначается символом «/», путь по дереву каталогов со- стоит из имен каталогов, разделенных символом «/», например: ***/****usrlinclude****/****sys*. В каждый момент времени с любым пользователем связан текущий каталог, то есть местоположение пользователя в иерархической файловой системе. Каталог *UNIX* содержит по одной записи для каждого файла этого каталога. Каждая ка- талоговая запись проста, так как в системе *UNIX* используется схема *i-узлов* (рис.5.). Каталоговая запись состоит всего из двух полей: имени файла (14 байт) и номера *i-узла* для этого файла (рис.6.).

Каждый файл операционной системы *UNIX* может быть однозначно опре- делен некоторой структурой данных, называемой *описателем файла* (*дескрип- тором*). Он содержит всю информацию о файле: тип файла, режим доступа, идентификатор владельца, размер, адрес файла, даты последнего доступа и по- следней модификации, дату создания и пр.

Обращение к файлу происходит по имени. Путь к файлу от корневого ка- талога называется *полным именем файла*. Если обращение к файлу начинается с символа «/», то считается, что указано полное имя файла и его поиск начинается с корневого каталога, в любом другом случае поиск файла начинается с текуще- го каталога. У любого файла может быть несколько имен. Фактически имя файла является ссылкой на файл, специфицированный номером описателя.

# Практическая часть

Ответить письменно на вопросы:

1. Дайте определение термину «файловая система».
2. Определите назначение файловой системы.
3. Как определяется Физический адрес сектора на диске.
4. Дайте определение термину «сектор» и поясните из каких элементов он состоит.
5. Как определяется физический адрес сектора?
6. С какими файловыми системами поддерживает работу операционная си- стема Windows XP?

10. Что собой представляет FAT?

12. Что собой представляет NTFL?

13 Определите тип и назначение операционной системы UNIX.

1. Какую файловую структуру имеет операционная система UNIX?
2. Что фактически представляет имя файла в операционной системе

UNIX?

**Сделать выводы по работе**

**Практическая работа № 12 Файловые системы FAT 32, HFS , NTFS , HPFS.**

**Сравнительная характеристика**

**Цель работы:** изучить характеристику файловых систем и на основании имеющихся данных заполнить сравнительную таблицу.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 3.2 ПК-3.3

.

# Практическая часть

Найти данные о файловых системах FAT 32, HFS , NTFS , HPFS

Кратко описать каждую файловую систему, проанализировать данные и заполнить таблицу1.

Таблица 1 – Сравнительная характеристика файловых систем FAT 32, HFS , NTFS , HPFS

|  |  |  |  |
| --- | --- | --- | --- |
| **Файловая система** | **Создатель** | **Дата представле- ния** | **Родная операци-**  **онная система или платформа** |
| FAT 32 |  |  |  |
| HFS (Hierarchical  File System) |  |  |  |
| [NTFS (New Tech-](https://ru.bmstu.wiki/NTFS_(New_Technology_File_System))  [nology File System)](https://ru.bmstu.wiki/NTFS_(New_Technology_File_System)) |  |  |  |
| [HPFS (High Perfor-](https://ru.bmstu.wiki/HPFS_(High_Performance_File_System))  [mance File System)](https://ru.bmstu.wiki/HPFS_(High_Performance_File_System)) |  |  |  |

# Сделать выводы по работе Эталон

|  |  |  |  |
| --- | --- | --- | --- |
| **Файловая система** | **Создатель** | **Дата представле- ния** | **Родная операци- онная система или**  **платформа** |
| FAT 32 | [Microsoft](https://ru.bmstu.wiki/Microsoft_Corporation)  [Corporation](https://ru.bmstu.wiki/Microsoft_Corporation) | 1996 | Windows 95[[i 1]](https://ru.bmstu.wiki/index.php?title=%D0%A1%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D1%84%D0%B0%D0%B9%D0%BB%D0%BE%D0%B2%D1%8B%D1%85_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC&mobileaction=toggle_view_mobile&cite_note-ref10-2) |
| HFS (Hierarchical  File System) | [Apple Inc.](https://ru.bmstu.wiki/Apple_Inc) | 1998 | Mac OS |
| [NTFS (New Tech-](https://ru.bmstu.wiki/NTFS_(New_Technology_File_System)) [nology File System)](https://ru.bmstu.wiki/NTFS_(New_Technology_File_System)) | [Microsoft Corpora-](https://ru.bmstu.wiki/Microsoft_Corporation)  [tion,](https://ru.bmstu.wiki/Microsoft_Corporation) Gary Kimura, Tom Miller | 1993 | Windows NT |
| [HPFS (High Perfor-](https://ru.bmstu.wiki/HPFS_(High_Performance_File_System))  [mance File System)](https://ru.bmstu.wiki/HPFS_(High_Performance_File_System)) | [Apple Inc.](https://ru.bmstu.wiki/Apple_Inc) | 1998 | Mac OS |

# Практическая работа № 13 Сетевые операционные системы, функции и характеристики, структура сетевой операционной системы

**Цель работы**: изучить функции и характеристики сетевых ОС, структуру сетевой ОС.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции:** ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 2.3 ПК-3.3

# Теоретическая часть Что такое сетевое программное обеспечение?

Сетевое программное обеспечение предназначено для организации сов- местной работы группы пользователей на разных компьютерах. Позволяет орга- низовать общую файловую структуру, общие базы данных, доступные каждому члену группы. Обеспечивает возможность передачи сообщений и работы над общими проектами, возможность разделения ресурсов.

**Сетевые операционные системы**

(Network Operating System – NOS) – это комплекс программ, обеспечива- ющих обработку, хранение и передачу данных в сети.

Сетевая операционная система выполняет функции прикладной платфор- мы, предоставляет разнообразные виды сетевых служб и поддерживает работу прикладных процессов, выполняемых в абонентских системах. Сетевые опера- ционные системы используют клиент-серверную, либо одноранговую архитек- туру. Компоненты NOS располагаются на всех рабочих станциях, включенных в сеть.

NOS определяет взаимосвязанную группу протоколов верхних уровней, обеспечивающих выполнение основных функций сети. К ним, в первую очередь, относятся:

1. адресация объектов сети;
2. функционирование сетевых служб;
3. обеспечение безопасности данных;
4. управление сетью.

При выборе NOS необходимо рассматривать множество факторов. Среди

них:

* + набор сетевых служб, которые предоставляет сеть;
  + возможность наращивания имен, определяющих хранимые данные и

прикладные программы;

* + механизм рассредоточения ресурсов по сети;
  + способ модификации сети и сетевых служб;
  + надежность функционирования и быстродействие сети;
  + используемые или выбираемые физические средства соединения;
  + типы компьютеров, объединяемых в сеть, их операционные системы;
  + предлагаемые системы, обеспечивающие управление сетью;
  + используемые средства защиты данных;
  + совместимость с уже созданными прикладными процессами;
  + число серверов, которое может работать в сети;
  + перечень ретрансляционных систем, обеспечивающих сопряжение ло- кальных сетей с различными территориальными сетями;
  + способ документирования работы сети, организация подсказок и под- держек.

**Функции и характеристики сетевых операционных систем (ОС).**

Различают ОС со встроенными сетевыми функциями и оболочки над ло- кальными ОС. По другому признаку классификации различают сетевые ОС од- норанговые и функционально несимметричные (для систем “клиент/сервер”).

Основные функции сетевой ОС:

1. управление каталогами и файлами;
2. управление ресурсами;
3. коммуникационные функции;
4. защита от несанкционированного доступа;
5. обеспечение отказоустойчивости;
6. управление сетью.

Управление каталогами и файлами в сетях заключается в обеспечении до- ступа к данным, физически расположенным в других узлах сети. Управление осуществляется с помощью специальной сетевой файловой системы. Файловая система позволяет обращаться к файлам путем применения привычных для ло- кальной работы языковых средств. При обмене файлами должен быть обеспечен необходимый уровень конфиденциальности обмена (секретности данных).

Управление ресурсами включает обслуживание запросов на предоставле- ние ресурсов, доступных по сети.

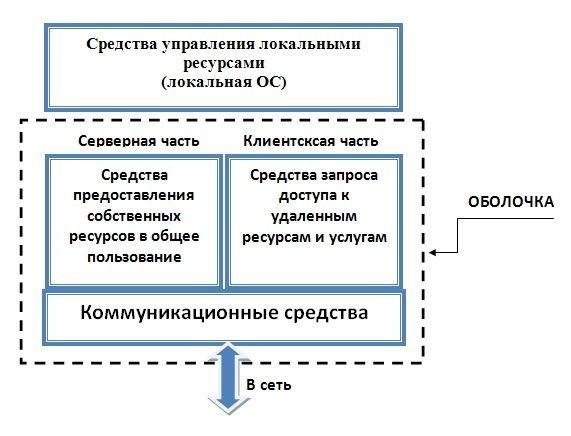
Коммуникационные функции обеспечивают адресацию, буферизацию, вы- бор направления для движения данных в разветвленной сети (маршрутизацию), управление потоками данных и др. Защита от несанкционированного доступа — важная функция, способствующая поддержанию целостности данных и их кон- фиденциальности. Средства защиты могут разрешать доступ к определенным данным только с некоторых терминалов, в оговоренное время, определенное число раз и т.п. У каждого пользователя в корпоративной сети могут быть свои права доступа с ограничением совокупности доступных директорий или списка возможных действий, например, может быть запрещено изменение содержимого некоторых файлов.

Отказоустойчивость характеризуется сохранением работоспособности си- стемы при воздействии дестабилизирующих факторов. Отказоустойчивость обеспечивается применением для серверов автономных источников питания, отображением или дублированием информации в дисковых накопителях. Под отображением обычно понимают наличие в системе двух копий данных с их расположением на разных дисках, но подключенных к одному контроллеру. Дублирование отличается тем, что для каждого из дисков с копиями использу- ются разные контроллеры. Очевидно, что дублирование более надежно. Даль- нейшее повышение отказоустойчивости связано с дублированием серверов, что однако требует дополнительных затрат на приобретение оборудования.

Управление сетью связано с применением соответствующих протоколов управления. Программное обеспечение управления сетью обычно состоит из ме- неджеров и агентов. Менеджером называется программа, вырабатывающая сете- вые команды. Агенты представляют собой программы, расположенные в раз- личных узлах сети. Они выполняют команды менеджеров, следят за состоянием узлов, собирают информацию о параметрах их функционирования, сигнализи- руют о происходящих событиях, фиксируют аномалии, следят за трафиком, осуществляют защиту от вирусов. Агенты с достаточной степенью интеллекту-

альности могут участвовать в восстановлении информации после сбоев, в кор- ректировке параметров

**Структура сетевой операционной системы**

Сетевая операционная система составляет основу любой вычислительной сети. Каждый компьютер в сети автономен, поэтому под сетевой операционной системой в широком смысле понимается совокупность операционных систем от- дельных компьютеров, взаимодействующих с целью обмена сообщениями и раз- деления ресурсов по единым правилам – протоколам. В узком смысле сетевая ОС – это операционная система отдельного компьютера, обеспечивающая ему возможность работать в сети.

[Рисунок 1 - Структура сетевой ОС](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)

В [соответствии со структурой, приведенной на рис. 1, в сетевой операци](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)- онной [системе отдельной машины можно выделить несколько частей.](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)

1. [Средства управления локальными ресурсами компьютера: функци](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)и распределения [оперативной памяти между процессами, планирования и диспет](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)- черизации [процессов, управления процессорами, управления периферийн](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)ыми устройствами [и другие функции управления ресурсами локальных ОС.](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)
2. [Средства предоставления собственных ресурсов и услуг в обще](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)е пользование [– серверная часть ОС (сервер). Эти средства обеспечивают, напри](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)- мер, блокировку [файлов и записей, ведение справочников имен сетевых ресу](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)р- сов; обработк[у запросов удаленного доступа к собственной файловой системе](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg) и базе данных[; управление очередями запросов удаленных пользователей к с](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)воим периферийным [устройствам.](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)
3. [Средства запроса доступа к удаленным ресурсам и услугам – клиент](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)- ская часть [ОС (редиректор). Эта часть выполняет распознавание и перенапра](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)в- ление в [сеть запросов к удаленным ресурсам от приложений и пользователей](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg). Клиентская [часть также осуществляет прием ответов от серверов и преобразова](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg)- ние их в [локальный формат, так что для приложения выполнение локальных](http://teacherbox.ru/wp-content/uploads/2015/11/pm02-1.jpg) и удаленных запросов неразличимо.
4. Коммуникационные средства ОС, с помощью которых происходит обмен сообщениями в сети. Эта часть обеспечивает адресацию и буферизацию сообщений, выбор маршрута передачи сообщения по сети, надежность передачи и т.п., т. е. является средством транспортировки сообщений.

# Практическая часть

Ответить письменно на вопросы:

* 1. Для чего предназначено сетевое программное обеспечение?
  2. (Network Operating System – NOS) что это такое?
  3. Какие функции выполняет сетевая операционная Система?
  4. Какую группу протоколов определяет NOS?
  5. Назовите основные функции сетевой ОС? Опишите кратко каждую функцию.
  6. Зарисовать структуру сетевой ОС
  7. В соответствии с рисунком описать отдельные части структуры ОС

# Сделать выводы по работе

**Практическая работа № 14 Использование редактора реестра операционной системы Windows. Оптимизация работы Windows.**

**Цель работы:** научится использовать реестр, для просмотра и настройки безопасности системы, ознакомиться с операциями, направленными на оптими- зацию работы операционной системы.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции**: ОК-1 ОК-2 ОК-4 ОК-5 ОК-6 ОК-7 ПК-1.3 ПК- 2.3 ПК-3.3

# Теоретическая часть Работа с реестром в Windows 10

Реестр - особая часть операционной системы Windows, которая представ- ляет собой базу данных из всех параметров или настроек ОС. Все опции [Панели](http://windowsten.ru/naxodim-panel-upravleniya-v-windows-10/) [управления и](http://windowsten.ru/naxodim-panel-upravleniya-v-windows-10/) других мест системы, где есть возможность изменения каких-либо параметров, зафиксированы в реестре. Там же хранятся данные о путях к фай- лам, о расположении установленных программ и других моментах, связанных с функционированием Windows. Настраивать и оптимизировать ОС можно также и с помощью сторонних программ.

Физически реестр состоит из группы файлов, которые хранятся в папке System32\config. При загрузке ОС из этих файлов собирается база данных с те- кущими параметрами, которые и применяются в работе Windows. Эта база со- стоит из пяти главных веток. Редактировать файлы из указанного каталога напрямую не рекомендуется, поскольку для того, чтобы войти в реестр Windows 10, существует специальный инструмент, зовущийся regedit. , как показано на рисунке 1 .

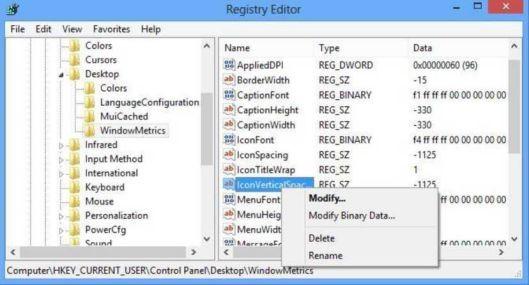


Рисунок 1 – Registry Editor

Запуск редактора значений реестра осуществляется следующим образом:

1. Щелкнуть по поисковой строке на панели задач или воспользоваться сочетанием клавиш Win + S.
2. Вписать в текстовое поле команду «regedit».
3. При появлении результатов поиска нажать на кнопку «выполнить команду».

Процесс внесения правок в содержимое реестра представляет собой поиск нужной строки в определенном каталоге и указание нового значения для неё. Поиск нужной строки проще всего осуществить через соответствующую опцию, которая вызывается сочетанием клавиш Ctrl + F или кнопкой F3.

Редактировать реестр в Windows 10 нужно крайне осторожно, поскольку неверное значение в каком-либо параметре может привести к ошибке при сле- дующей загрузке системы и выведению её из строя. Прежде чем вносить изме- нения в какой-либо параметр, следует поискать в интернете

Совет! Перед редактированием реестра рекомендуется создавать резервную ко- пию через меню «Файл» -> «Экспорт».

Для очистки реестра Windows 10 от ненужных строк и ошибочных значе- ний пользователи устанавливают стороннее программное обеспечение, посколь- ку ОС не обладает в своём составе необходимыми функциями. Ошибки в реестре могут появляться в результате сбоев установленных программ, а пустые строки - из-за не до конца удаленных приложений.

Например, почистить реестр на Windows 10 можно утилитой CCleaner по- казанной на рисунке 2, которая распространяется бесплатно на сайте компании- разработчика. Скачав и установив данный продукт, можно приступить к проце- дуре поиска неисправностей и их устранения. Для этого нужно:

1. Запустить CCleaner.
2. Перейти на вкладку «Реестр».
3. Нажать на кнопку «Поиск проблем».
4. Дождаться завершения процедуры поиска.
5. Нажать на кнопку «Исправить».
6. Подтвердить создание резервной копии и указать файл для сохране- ния текущей версии значений реестра.
7. Щелкнуть по кнопке «Исправить отмеченные».
8. Закрыть окно.

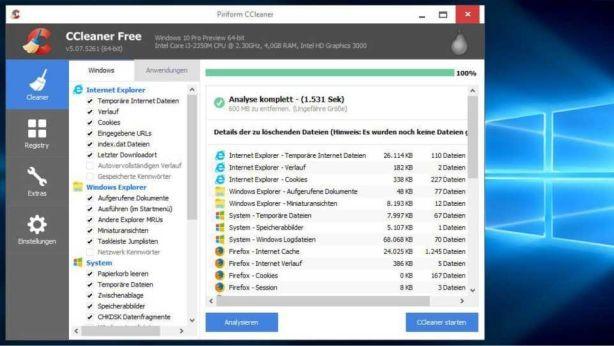


Рисунок 2 - Программа для очистки CCleaner Free

В том случае, если после ручного редактирования значений реестра или автоматической чистки с применением стороннего программного обеспечения наблюдаются проблемы в работе операционной системы, нужно восстановить реестр Windows 10 до прежнего состояния. Делается это путём импортирования созданного на этапе редактирования файла.

Для того, чтобы записать значения из файла в реестр, потребуется:

1. С помощью Проводника Windows найти папку, хранящую ^reg-файл с резервной копией.
2. Дважды кликнуть по файлу.
3. Подтвердить импорт значений.

Реестр Windows - мощный инструмент для управления ОС, предоставля- ющий доступ ко многим параметрам, не реализованным в стандартном визуаль- ном интерфейсе Windows. Для доступа к таким параметрам часто применяют программы-твикеры, которые могут настроить размер оконных рамок проводни- ка, время проигрывания анимаций и много других параметров. В любом случае, изменять параметры реестра не рекомендуется без особой причины, а перед ис- правлением ошибочных значений всегда следует создавать резервную копию.

**Практическая часть**

мы.

1. Изучите основные моменты оптимизации работы операционной систе-
2. Проведите оптимизацию операционной системы и для этого выполните

задания.

**Задание 1.** Отключить службу индексирования

* 1. Откройте окно *Мой компьютер*
  2. Вызовите окно свойств жесткого (логического) диска
  3. *Снимите флажок* Разрешить индексирование диска для быстрого поиска
  4. Нажмите кнопку *Применить* и в новом окне установите переключа- тель в положение *Применить ко всем вложенным файлам и папкам*
  5. Дождитесь завершение процесса применения новых атрибутов ко всем вложенным файлам и папкам. Будьте готовы, что он может занять некото- рое время.
  6. Повторите эти же действия для всех остальных дисков.
  7. Подготовьте отчет о проделанной работе.

**Задание 2.** Отключить визуальных эффектов

1. Вызовите окно *Свойства системы* и перейти в нем на вкладку *Допол- нительно.* Здесь нажмите в области быстродействие на кнопку *параметры.* От- кроется окно *Параметры быстродействия.*
2. Установите положение *Обеспечить наилучшее быстродействие* сдела- ет картинку намного скромнее, его производительность системы при этом резко возрастет.
3. С помощью меню *особые эффекты* в индивидуальном порядке порабо- тайте с различными типами визуального эффекта.
4. Верните состояние системы в исходное положение, установив пере- ключатель в положение *Восстановить значения по умолчанию.*
5. Подготовьте отчет о проделанной работе.

# Сделать выводы по работе

# Практическая работа № 15 Система Kerberos

**Цель работы:** изучить аутентификацию пользователей с помощью прото- кола Kerberos.

**Оснащение**: OS Windows, MS Office.

**Формируемые компетенции**: ОК-1 ОК-2 ОК-5 ПК- 1.3 ПК-2.3 ПК-3.2 ПК- 3.3

# Теоретическая часть

В операционной системе Microsoft® Windows® 2000 аутентификация пользователей производится по умолчанию с помощью протокола Kerberos. Ис- пользование этого стандарта создает надежную основу для взаимодействия меж- ду различными платформами и при этом значительно повышает безопасность сетевой аутентификации.

Протокол Kerberos был создан в Массачусетском технологическом инсти- туте в рамках проекта Athena. Однако общедоступным этот протокол стал лишь после появления версии 4. После того, как специалисты отрасли изучили новый протокол, его авторы разработали и предложили пользователям очередную вер- сию – Kerberos 5, которая и была принята в качестве стандарта IETF.

# Расширения протокола Kerberos

Обзор протокола Kerberos

Протокол аутентификации Kerberos предлагает механизм взаимной иден- тификации клиента и сервера (или двух серверов) перед установлением связи между ними. В протоколе учтено, что начальный обмен информацией между клиентами и серверами происходит в открытой среде, а пакеты, передаваемые по каналам связи, могут быть перехвачены и модифицированы. Другими словами, протокол предназначен для работы в среде, которая очень напоминает сего- дняшний Интернет. Здесь злоумышленник легко может имитировать запросы клиента или сервера, перехватывать связь между легитимными клиентами и сер- верами, даже искажать передаваемую информацию.

# Основные концепции

Протокол Kerberos активно использует технологии аутентификации, опи- рающиеся на «секреты для двоих». Основная концепция довольно проста: если есть секрет, известный только двоим, любой из его хранителей может легко удо- стовериться, что имеет дело именно со своим напарником. Для этого ему доста- точно каким-либо способом проверить, знает ли собеседник их общий секрет.

В протоколе Kerberos эта проблема решается средствами криптографии с секретным ключом. Вместо того чтобы сообщать друг другу пароль, участники сеанса обмениваются криптографическим ключом, знание которого подтвержда- ет личность собеседника. Один из участников использует такой ключ для шиф- рования блока информации, а другой с его помощью извлекает эту информацию.

# Аутентификаторы

Простой протокол аутентификации с секретным ключом вступает в дей- ствие, когда кто-либо стучится в сетевую дверь и просит впустить его. Чтобы доказать свое право на вход, пользователь предъявляет **аутентифика- тор**(authenticator) в виде блока информации, зашифрованного с помощью сек- ретного ключа. Содержание этого блока должно меняться при каждом последу- ющем сеансе, в противном случае злоумышленник вполне может проникнуть в систему, воспользовавшись перехваченным сообщением. Получив аутентифика- тор, привратник расшифровывает его и проверяет полученную информацию, чтобы убедиться в успешности расшифрования. Если все прошло нормально, страж может быть уверен, что человеку, предъявившему аутентификатор, изве- стен секретный ключ. А ведь этот ключ знают всего двое, причем один из них – сам привратник. Таким образом, он делает вывод, что пришелец действительно тот человек, за которого себя выдает.

Но может случиться и так, что субъект, постучавшийся в дверь, захочет провести взаимную аутентификацию. В этом случае используется тот же самый протокол, но в обратном порядке и с некоторыми изменениями. Теперь приврат- ник извлекает из исходного аутентификатора часть информации, а затем шифру- ет ее, превращая в новый аутентификатор, и в таком виде возвращает пришель- цу. Тот, в свою очередь, расшифровывает полученную информацию и сравнива- ет ее с исходной. Если все совпадает, пришелец может быть уверен: раз при- вратник расшифровал оригинал, значит, он знает секретный ключ.

Само название протокола Kerberos говорит о том, как здесь решена про- блема управления ключами. Кербер (или *Цербер*) – персонаж классической гре- ческой мифологии. Этот свирепый пес о трех головах, по поверьям греков, охра- няет врата подземного царства мертвых. Трем головам Кербера в протоколе Kerberos соответствуют три участника безопасной связи: клиент, сервер и дове- ренный посредник между ними. Роль посредника здесь играет так называемый центр распределения ключей **Key Distribution Center** или, сокращенно, KDC.

KDC представляет собой службу, работающую на физически защищенном сервере. Она ведет базу данных с информацией об учетных записях всех главных абонентов безопасности (security principal) своей *области* (области Kerberos в се- тях Windows 2000 соответствует домен, поэтому в дальнейшем мы будем приме- нять именно этот привычный термин). Вместе с информацией о каждом абонен- те безопасности в базе данных KDC сохраняется криптографический ключ, из- вестный только этому абоненту и службе KDC. Данный ключ, который называ- ют **долговременным**, используется для связи пользователя системы безопасно- сти с центром распределения ключей. В большинстве практических реализаций протокола Kerberos долговременные ключи создаются на основе пароля пользо- вателя.

Когда клиенту нужно обратиться к серверу, он, прежде всего, направляет запрос в центр KDC, который в ответ направляет каждому участнику предстоя- щего сеанса копии уникального **сеансового ключа** (session key), действующие в

течение короткого времени. Назначение этих ключей – проведение аутентифи- кации клиента и сервера. Копия сеансового ключа, пересылаемая на сервер, шифруется с помощью долговременного ключа этого сервера, а направляемая клиенту – долговременного ключа данного клиента.

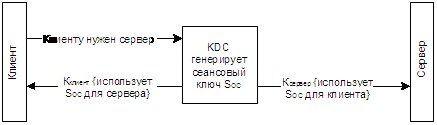


Рисунок 2 - Управление ключами (в теории)

Теоретически, для выполнения функций доверенного посредника центру KDC достаточно направить сеансовые ключи непосредственно абонентам без- опасности, как показано выше. Однако на практике реализовать такую схему чрезвычайно сложно. Прежде всего, серверу пришлось бы сохранять свою ко- пию сеансового ключа в памяти до тех пор, пока клиент не свяжется с ним. А ведь сервер обслуживает не одного клиента, поэтому ему нужно хранить пароли всех клиентов, которые могут потребовать его внимания. В таких условиях управление ключами требует значительной затраты серверных ресурсов, что ограничивает масштабируемость системы. Нельзя забывать и о превратностях сетевого трафика. Они могут привести к тому, что запрос от клиента, уже полу- чившего сеансовый пароль, поступит на сервер раньше, чем сообщение KDC с этим паролем. В результате серверу придется повременить с ответом до тех пор, пока он не получит свою копию сеансового пароля. То есть, нужно будет сохра- нить состояния сервера, а это накладывает на его ресурсы еще одно тяжкое бре- мя. Поэтому на практике применяется другая схема управления паролями, кото- рая делает протокол Kerberos гораздо более эффективным. Ее описание приво- дится ниже.

# Сеансовые билеты

В ответ на запрос клиента, который намерен подключиться к серверу, служба KDC направляет обе копии сеансового ключа клиенту, как показано на рис. 3. Сообщение, предназначенное клиенту, шифруется посредством долго- временного ключа клиента, а сеансовый ключ для сервера вместе с информацией о клиенте вкладывается в блок данных, получивший название **сеансового биле- та** (session ticket). Затем сеансовый билет целиком шифруется с помощью долго- временного ключа сервера, который знают только служба KDC и данный сервер. После этого вся ответственность за обработку билета, несущего в себе зашифро- ванный сеансовый ключ, возлагается на клиента, который должен доставить его на сервер.

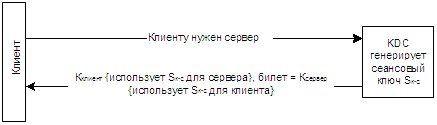


Рисунок 3 - Управление ключами (на практике)

Обратите внимание, что в данном случае функции службы KDC ограничи- ваются выдачей билета. Ей больше не нужно следить за тем, все ли отправлен- ные сообщения доставлены соответствующим адресатам. Даже если какое- нибудь из них попадет не туда, – ничего страшного не случится. Расшифровать клиентскую копию сеансового ключа может только тот, кто знает секретный долговременный ключ данного клиента, а чтобы прочесть содержимое сеансово- го билета, нужен долговременный секретный ключ сервера.

Получив ответ KDC, клиент извлекает из него сеансовый билет и свою ко- пию сеансового ключа, которые помещает в безопасное хранилище (оно распо- лагается не на диске, а в оперативной памяти). Когда возникает необходимость связаться с сервером, клиент посылает ему сообщение, состоящее из билета, ко- торый по-прежнему зашифрован с применением долговременного ключа этого сервера, и собственного аутентификатора, зашифрованного посредством сеансо- вого ключа. Этот билет в комбинации с аутентификатором как раз и составляет удостоверение, по которому сервер определяет «личность» клиента.

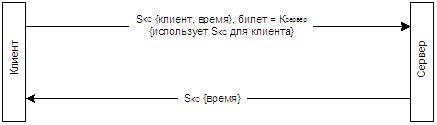


Рисунок 4 - Взаимная аутентификация (клиент-сервер)

Сервер, получив «удостоверение личности» клиента, с помощью своего секретного ключа расшифровывает сеансовый билет и извлекает из него сеансо- вый ключ, который затем использует для расшифрования аутентификатора кли- ента. Если все проходит нормально, делается заключение, что удостоверение клиента выдано доверенным посредником, то есть, службой KDC. Клиент может потребовать у сервера проведения взаимной аутентификации. В этом случае сер- вер с помощью своей копии сеансового ключа шифрует метку времени из аутен- тификатора клиента и в таком виде пересылает ее клиенту в качестве собствен- ного аутентификатора.

Одно из достоинств применения сеансовых билетов состоит в том, что серверу не нужно хранить сеансовые ключи для связи с клиентами. Они сохра- няются в кэш-памяти удостоверений (credentials cache) клиента, который

направляет билет на сервер каждый раз, когда хочет связаться с ним. Сервер, со своей стороны, получив от клиента билет, расшифровывает его и извлекает сеан- совый ключ. Когда надобность в этом ключе исчезает, сервер может просто сте- реть его из своей памяти.

Такой метод дает и еще одно преимущество: у клиента исчезает необходи- мость обращаться к центру KDC перед каждым сеансом связи с конкретным сер- вером. Сеансовые билеты можно использовать многократно. На случай же их хищения устанавливается срок годности билета, который KDC указывает в са- мой структуре данных. Это время определяется политикой Kerberos для кон- кретного домена. Обычно срок годности билетов не превышает восьми часов, то есть, стандартной продолжительности одного сеанса работы в сети. Когда поль- зователь отключается от нее, кэш-память удостоверений обнуляется, и все сеан- совые билеты вместе с сеансовыми ключами уничтожаются.

# Практическая часть

Ответить письменно на вопросы:

1. С помощью какого протокола по умолчанию производится аутентифи- кация пользователей?
2. Описать кратко механизм работы протокола Kerberos.
3. Основные концепции протокола Kerberos.
4. Что такое Аутентификатор и как он работает?
5. История возникновения название протокола Kerberos.
6. Что такое сеансовые билеты и кА кони работают.

# Сделать выводы по работе

# СПИСОК ОСНОВНЫХ ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ

1. Федеральный государственный образовательный стандарт среднего профессионального образования по специальности 09.02.03 Программирование в компьютерных системах (Приказ Министерства образования и науки РФ от 28 июля 2014. № 804 " Об утверждении федерального государственного образова- тельного стандарта среднего профессионального образования по специальности

09.02.03 Программирование в компьютерных системах).

**БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

**Перечень ресурсов информационно-телекоммуникационной сети**

**"Интернет"**

1. Партыка Т.Л. Операционные системы, среды и оболочки : учебное пособие / Т.Л. Партыка, И.И. Попов. — 5-е изд., перераб. и доп. — М. : ФОРУМ

: ИНФРА-М, 2017. — 560 с. : ил. — (Профессиональное образование). <http://znanium.com/bookread2.php?book=552493>

1. Рудаков А.В. Операционные системы и среды : учебник // Рудаков А.В. — М.: КУРС: ИНФРА-М, 2018. — 304 с. — (Среднее профессиональное образование). <http://znanium.com/bookread2.php?book> =946818

# ПРИЛОЖЕНИЕ А

(обязательное)

# ФОРМА ТИТУЛЬНОГО ЛИСТА

**Форма титульного листа**



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ИНСТИТУТ СФЕРЫ ОБСЛУЖИВАНИЯ И ПРЕДПРИНИМАТЕЛЬСТВА (ФИЛИАЛ) ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ» В Г. ШАХТЫ РОСТОВСКОЙ ОБЛАСТИ**

**(ИСОиП (филиал) ДГТУ в г. Шахты)**

КОЛЛЕДЖ ЭКОНОМИКИ И СЕРВИСА

***Ж ур нал***

***пр акти че ски х р або т***

***по дисциплине "Операционные системы" .***

Выполнил

(подпись)

Паршина Т.П. группа КВ 9-212

(инициалы, фамилия, группа)

Проверил

(подпись)

преподаватель Е.Н. Семеренко .

ученая степень, зание, инициалы, фамилия)

# ПРИЛОЖЕНИЕ Б

(обязательное)

# ПРИМЕР ОФОРМЛЕНИЯ ПЕРВОЙ СТРАНИЦЫ ОТЧЕТА

**Пример оформления первой страницы отчета**

