

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ
КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ им.И.АРАБАЕВА
ОСПО ИНСТИТУТА НОВЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ



УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС

по дисциплине: **Операционные системы и среды**

для студентов специальности: **ПОВТАС**

форма обучения: **Очная**

Курс: **1, 2** Семестр: **1**

Часов: всего: **18**, лекций: **10**, практ. Зан: **8**

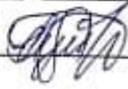
СРС: **12**

Учебно-методический комплекс разработал(а): **Анарбеков Табылды Райымбекович**

Рассмотрена и утверждена на заседании ОСПО ИНИТ КГУ им.И. Арабаева
Протокол № 1 от «07» 09 2023 г.

Зав. ОСПО ИНИТ: **Н.С.Сейткадиева** 

Одобрено учебно-методическим советом ОСПО ИНИТ КГУ им.И. Арабаева
Протокол № 1 от «08» 09 2023 г.

Председатель УМС: 

Бишкек 2023г.

“ОПЕРАЦИОННЫЕ СИСТЕМЫ И СРЕДЫ”

1. ЦЕЛИ И ЗАДАЧИ ПРЕПОДАВАНИЯ ДИСЦИПЛИНЫ

- должен знать назначение, функции, классификацию, структуру и архитектуру операционных систем (ОС); особенности управления процессором; функции ядра ОС; структуру контекста и дескриптора процесса; иерархию процессов; алгоритмы планирования процессов; средства коммуникации процессов; типы прерываний; средства обработки сигналов.

- должен владеть методами управления, совместного использования и защиты памяти; механизмами виртуализации памяти; основами диспетчеризации и синхронизации процессов; способами реализации режима мультипрограммирования; стратегиями подкачки страниц; принципами защиты ОС от сбоев и несанкционированного доступа.

Учебная дисциплина “Операционные системы” является дисциплиной компонента и входит в цикл общепрофессиональных дисциплин.

Целью дисциплины является обучение студентов концепциям построения современных ОС; особенностям планирования процессов в многозадачных ОС; средствам межпроцессного взаимодействия; методам распределения памяти; способам реализации защиты в ОС; а также приобретение навыков работы со стандартными служебными программами современных операционных систем и разработка собственных приложений системного назначения.

2. УЧЕБНЫЙ ПЛАН ДИСЦИПЛИНЫ

№	Наименование тем лекций	Часы
1	Лекция № 1 Основные понятия и определения.	1
2	Лекция № 2 Структура ОС	1
3	Лекция № 3, 4 Эволюция вычислительных систем	2
4	Лекция № 5 Назначение и основные функции операционных систем. Состав операционной системы.	2
5	Лекция № 6, 7 Состав операционной системы	2
6	Лекция № 8, 9 Архитектура операционной системы	2
	Всего лекций	10
Практические занятия		
1.	Освоить порядок загрузки и завершения работы ОС <i>Windows</i> .	1
2.	Изучить назначение <i>Рабочего стола</i> и его элементов.	1
3.	Освоить приемы работы с <i>мышью</i> .	1
4.	Изучить назначение <i>Главного меню</i> , его структуру и доступ к нему.	1
5.	Изучить назначение, вызов и порядок работы <i>контекстного меню</i> .	1
6.	Изучить назначение, возможности и порядок работы <i>Справочной системы Windows 7</i> .	1
7.	Освоить назначение, способы загрузки и завершения работы <i>Стандартных программ Windows 7</i> .	1
8.	Изучить понятие интерфейса и его элементов.	1
9.	Всего	8
10.	Итого	18 ч.

Содержание лекций

Лекция № 1 Основные понятия и определения.

1.1. Назначение и функции операционных систем.

Операционная система — это программа, контролирующая работу пользовательской программы и систем приложений и исполняемая роль интерфейса между приложениями и аппаратным обеспечением компьютера. Её предназначения можно разделить на три основные составляющие:

- удобство: операционная система делает исполнение компьютера простым и удобным
- эффективность: операционная система позволяет эффективно использовать ресурсы компьютерной системы
- возможность развития: операционная система должна допускать разработку тестирования новых приложений и системных функций без нарушения нормального функционирования вычислительной системы.

Согласно многолетней традиции, при рассмотрении основ функционирования ОС принято выделять четыре основных группы функций, выполняемых системой. Управление устройствами. Имеются в виду все периферийные устройства, подключаемые к компьютеру, — клавиатура, монитор, принтеры, диски и т.п. Управление данными. Под этим старинным термином сейчас понимается работа с файлами, хотя были времена, когда обращение к данным на магнитных носителях выполнялось путем указания адреса размещения данных на устройстве, а понятия файла не существовало. Управление процессами. Эта сторона работы ОС связана с запуском и завершением работы программ, обработкой ошибок, обеспечением параллельной работы нескольких программ на одном компьютере. Управление памятью. Оперативная память компьютера — это такой ресурс, которого всегда не хватает. В этих условиях разумное планирование использования памяти является важнейшим фактором эффективной работы. Имеется еще несколько важных обязанностей, лежащих на ОС, которые трудно втиснуть в рамки традиционной классификации функций. К ним, прежде всего, относятся следующие. Организация интерфейса с пользователем. Формы интерфейса могут быть разнообразными, в зависимости от типа и назначения ОС: язык управления пакетами заданий, набор диалоговых команд, средства графического интерфейса. Защита данных. Как только система перестает быть достоянием одного изолированного от внешнего мира пользователя, вопросы защиты данных от несанкционированного доступа приобретают первостепенную важность. ОС, обеспечивающая работу в сети или в системе разделения времени, должна соответствовать имеющимся стандартам безопасности. Ведение статистики. В ходе работы ОС должна собираться, храниться и анализироваться разнообразная информация: о количестве времени, затраченном различными программами и пользователями, об интенсивности использования ресурсов, о попытках некорректных действий пользователей, о сбоях оборудования и т.п. Собранная информация хранится в системных журналах и в учетных записях пользователей.

Лекция № 2 Структура ОС

Традиционным представлением структуры вычислительной системы является пирамида (Рис. 1). Каждый из уровней пирамиды определяет свой уровень абстракции свойств вычислительной системы. Основанием является аппаратный уровень вычислительной системы — это основа всей иерархии, так как многие характеристики и функциональные возможности последующих программных уровней существенно определяются свойствами аппаратуры компьютера, находящегося в основе вычислительной системы.

Представление о возможностях и свойствах конкретной вычислительной системы формируются с позиций каждого из уровней структурной организации. Так вычислительная система представляется пользователю прикладной системы, работающей на компьютере, в виде совокупности возможностей этой прикладной системы. Примером может служить игровой автомат, являющийся компьютером, на котором функционирует операционная система, а также игровая программа, которая в данном случае является прикладной системой. Пользователю данной специализированной вычислительной системы ее свойства представляются в виде функциональных возможностей игровой программы и итоговой производительностью системы (если процессор компьютера маломощный, то динамика игры может быть недостаточной).

Другой пример — компьютер, используемый для обучения школьников языку программирования. Это означает, что для школьника или его учителя свойства данной вычислительной системы будут представляться с позиций уровня системы программирования, построенной на основе транслятора языка программирования, на котором идет обучение. Представление свойств вычислительной системы в данном случае будет формироваться из пользовательского интерфейса системы программирования в сочетании со свойствами и производительностью аппаратных компонентов компьютера.



Рис. 1. Структура вычислительной системы.

Взаимосвязи уровней иерархической структуры вычислительной системы, их характеристики могут проявляться как в виде непосредственных межуровневых интерфейсов, определенных однозначным набором правил использования объектов одного уровня другим, так и косвенным влиянием одного уровня на другой. Примером подобного косвенного взаимодействия может служить влияние, оказываемое на характеристики функционирования всей вычислительной системы в целом, производительности или емкости аппаратных компонентов компьютера (внешних устройств, процессора, оперативной памяти, линий связи и пр.). В качестве иллюстрации рассмотрим вычислительную систему, имеющую канал связи, обеспечивающий доступ в Интернет со скоростью 64 Kbps. Данная система сможет обеспечить достаточно комфортные условия для интенсивной работы в Интернете одного–двух пользователей. Если количество пользователей возрастет до 10, будут возникать задержки при обработке запросов, что снизит качество работы пользователей. При росте числа пользователей до 100, организовать их интенсивную работу в Интернете на данной вычислительной системе не представляется возможным, т.к. пропускная способность канала связи не справится с потоком запросов, поступающих от пользователей. Таким образом, проявляется косвенное влияние пропускной способности канала связи на эксплуатационные характеристики (или качества) вычислительной системы.

Рассмотрим основные характеристики и суть взаимосвязи уровней пирамиды, представляющей структуру вычислительной системы.

Лекция № 3, 4 Эволюция вычислительных систем

Существуют различные классификации ВС. Наиболее часто они классифицируются по элементной базе. В соответствии с этой классификацией в эволюции ВС выделяются 4 этапа:

1. Первый период (1945-1955). Ламповые машины. Операционных систем нет.
2. Второй период (1955-начало 60-х). Компьютеры на основе транзисторов. Пакетные ОС.
3. Третий период (начало 60-х-1980). Компьютеры на основе интегральных микросхем. Первые многозадачные ОС,
4. Четвертый период (1980-по настоящее время). Персональные компьютеры. Классические, сетевые и распределенные системы.

9. Общая характеристика пакетных операционных систем (с использованием перфокарт и магнитных лент), функции «монитора» в многозадачном режиме, spooling

1. Как правило, работа на ЭВМ в первом периоде носила научно-исследовательский характер. По стоимости ЭВМ были доступны только крупным исследовательским центрам. Машины занимали значительное пространство, имели большой вес (содержали до нескольких тысяч электровакуумных ламп), требовалась значительная энергия, была большая теплоотдача (водяные системы охлаждения). За машиной работал специалист, который досконально знал структуру машины, поскольку при работе ему требовалось знать на физическом уровне, в какие ячейки памяти вносить данные, команды. Вся работа велась в двоичных кодах. Адресация и данные вводились с помощью ряда тумблеров на пульте управления. Введенная информация контролировалась индикаторами (лампочками), также расположенными на пульте управления. На нем также находились кнопки пуска программы, принудительного останова, а также кнопка обнуления всех регистров памяти (по завершению работы).

В этот период времени одни и те же группы людей участвовали в проектировании, создании, программировании и эксплуатации ЭВМ.

Этот период можно назвать «ремесленным», поскольку разделения труда практически не было. Программирование велось на машинном языке. Не было никакого системного ПО, кроме библиотек математических и служебных подпрограмм.

При этом ВС выполняла одновременно только одну из 3 операций: ввод, вычисление, вывод. Все задачи организации вычислительного процесса решались вручную каждым программистом с пульта управления.

Основное противоречие: большие затраты времени на ввод данных и команд с одной стороны, и быстрый расчет с другой стороны. То есть процессор использовался крайне неэффективно. Необходимо было сократить время ввода данных и команд. Для этой цели стали использовать перфокарты. С их использованием параллельно начались изменения и в технологии организации работы с ЭВМ. Появилось разделение труда.

Постепенно это привело к тому, что выделилась группа специалистов, которая разрабатывала аппаратуру ЭВМ и периферийные устройства, появились специалисты, в задачу которых входит эксплуатация ЭВМ. Выделился класс специалистов, которые занимались только разработкой ПО.

Это означает, что происходит специализация предметной области. Основная причина этого явления состоит в том, что предметная область расширяется и углубляется и отдельный специалист не способен решать возникающие проблемы.

Также меняется и сама организация работы. Если на первых этапах программист непосредственно работал за машиной, то с использованием перфокарт он практически был «отлучен» от машины. Появился класс операторов: оператор перфокарт, оператор машинного зала, которые соответственно формировали колоду перфокарт, получали колоду от программиста, вводили информацию в ЭВМ, запускали программу на исполнение, распечатывали эти результаты и передавали и программисту.

В целом, первый период характеризуется крайне высокой стоимостью ВС, их малым количеством и низкой эффективностью использования. Поскольку при отладке программ машина часто простаивала, то появилась идея загружать не по одному заданию, а несколько. В том случае, когда первое задание по какой-то причине не работает, управление передается другому заданию. Происходит это в автоматическом режиме, за счет разработанного языка управления заданиями.

Лекция № 5 Назначение и основные функции операционных систем. Состав операционной системы.

Операционная система (ОС) – это комплекс взаимосвязанных системных программ для организации взаимодействия пользователя с компьютером и выполнения всех других программ. ОС относятся к составу системного программного обеспечения и являются основной его частью. Операционные системы: MS DOS 7.0, Windows Vista Business, Windows 2008 Server, OS/2, UNIX, Linux.

Основные функции ОС:

управление устройствами компьютера (ресурсами), т.е. согласованная работа всех аппаратных средств ПК: стандартизированный доступ к периферийным устройствам, управление оперативной памятью и др.

управление процессами, т.е. выполнение программ и их взаимодействие с устройствами компьютера.

управление доступом к данным на энергонезависимых носителях (таких как жесткий диск, компакт-диск и т.д.), как правило, с помощью файловой системы.

ведение файловой структуры.

пользовательский интерфейс, т.е. диалог с пользователем.

Дополнительные функции:

параллельное или псевдопараллельное выполнение задач (многозадачность).

взаимодействие между процессами: обмен данными, взаимная синхронизация.

защита самой системы, а также пользовательских данных и программ от злонамеренных действий пользователей или приложений.

разграничение прав доступа и многопользовательский режим работы (аутентификация, авторизация).

Лекция № 6, 7 Состав операционной системы

В общем случае в состав ОС входят следующие модули:

Программный модуль, управляющий файловой системой.

Командный процессор, выполняющий команды пользователя.

Драйверы устройств.

Программные модули, обеспечивающие графический пользовательский интерфейс.

Сервисные программы.

Справочная система.

Драйвер устройства (device driver) – специальная программа, обеспечивающая управление работой устройств и согласование информационного обмена с другими устройствами.

Командный процессор (command processor) – специальная программа, которая запрашивает у пользователя команды и выполняет их (интерпретатор программ).

Интерпретатор команд отвечает за загрузку приложений и управление информационным потоком между приложениями.

Для упрощения работы пользователя в состав современных ОС входят программные модули, обеспечивающие графический пользовательский интерфейс.

Процесс работы компьютера в определенном смысле сводится к обмену файлами между устройствами. В ОС имеется программный модуль, управляющий файловой системой.

Сервисные программы позволяют обслуживать диски (проверять, сжимать, дефрагментировать и др.), выполнять операции с файлами (копирование, переименование и др.), работать в компьютерных сетях.

Для удобства пользователя в состав ОС входит справочная система, позволяющая оперативно получить необходимую информацию о функционировании как ОС в целом, так и о работе ее отдельных модулей.

Состав модулей ОС, а также их количество зависит от семейства и вида ОС. Так, например, в ОС MS DOS отсутствует модуль, обеспечивающий графический пользовательский интерфейс.

Наиболее общим подходом к структуризации операционной системы является разделение всех ее модулей на две группы:

Ядро – это модули, выполняющие основные функции ОС.

Вспомогательные модули, выполняющие вспомогательные функции ОС. Одним из определяющих свойств ядра является работа в привилегированном режиме.

Модули ядра выполняют следующие базовые функции ОС: Управление процессами, Управление системой прерываний, Управление памятью, управление устройствами ввода-вывода, Функции, решающие внутрисистемные задачи организации вычислительного процесса: переключение контекстов, загрузка/выгрузка страниц, обработка прерываний. Эти функции недоступны для приложений. Функции, служащие для поддержки приложений, создавая для них так называемую прикладную программную среду.

Приложения могут обращаться к ядру с запросами – системными вызовами – для выполнения тех или иных действий: для открытия и чтения файла, вывода графической информации на дисплей, получения системного времени и т.д. Функции ядра, которые могут вызываться приложениями, образуют интерфейс прикладного программирования – API (Application programming interface).

Пример.

Базовый код API Win32 содержится в трех библиотеках динамической загрузки (Dynamic Link Library, DLL): USER32, GDI32 и KERNEL32.

Kernel - модуль Windows, который поддерживает низкоуровневые функции по работе с файлами и управлению памятью и процессами. Этот модуль обеспечивает сервис для 16- и 32-разрядных приложений.

GDI (Graphics Device Interface) - модуль Windows, обеспечивающий реализацию графических функций по работе с цветом, шрифтами и графическими примитивами для дисплея и принтеров.

User - модуль Windows, который является диспетчером окон и занимается созданием и управлением отображаемыми на экране окнами, диалоговыми окнами, кнопками и другими элементами пользовательского интерфейса.

Ядро является движущей силой всех вычислительных процессов в компьютерной системе, и крах ядра равносителен краху всей системы, без него ОС является полностью неработоспособной и не сможет выполнить ни одну из своих функций. Поэтому разработчики операционной системы уделяют особое внимание надежности кодов ядра, в результате процесс их отладки может растягиваться на многие месяцы.

Обычно ядро оформляется в виде программного модуля некоторого специального формата, отличающегося от формата пользовательских приложений.

Вспомогательные модули ОС выполняют вспомогательные функции ОС (полезные, но менее обязательные чем функции ядра).

Примеры вспомогательных модулей:

Программа архивирования данных.

Программа дефрагментации диска.

Текстовый редактор.

Вспомогательные модули ОС оформляются либо в виде приложений, либо в виде библиотек процедур. Вспомогательные модули ОС подразделяются на следующие группы:

утилиты – программы, решающие задачи управления и сопровождения компьютерной системы: обслуживание дисков и файлов.

системные обрабатывающие программы – текстовые или графические редакторы, компиляторы, компоновщики, отладчики.

программы предоставления пользователю дополнительных услуг пользовательского интерфейса (калькулятор, игры).

библиотеки процедур различного назначения, упрощающие разработку приложений (библиотека математических функций, функций ввода-вывода).

Как и обычные приложения, для выполнения своих задач утилиты, обрабатывающие программы и библиотеки ОС, обращаются к функциям ядра посредством системных вызовов.

Функции, выполняемые модулями ядра, являются наиболее часто используемыми функциями операционной системы, поэтому скорость их выполнения определяет производительность всей системы в целом. Для обеспечения высокой скорости работы ОС все модули ядра или большая их часть постоянно находятся в оперативной памяти, то есть являются резидентными.

Вспомогательные модули обычно загружаются в оперативную память только на время выполнения своих функций, то есть являются транзитными. Такая организация ОС экономит оперативную память компьютера.

Примечание

Разделение операционной системы на ядро и вспомогательные модули обеспечивает легкую расширяемость ОС. Чтобы добавить новую высокоуровневую функцию, достаточно разработать новое приложение, и при этом не требуется модифицировать основные функции, образующие ядро системы.

Объектами ядра ОС являются:

Процессы (рассмотрено в теме 2.3).

Файлы.

События.

Потоки (рассмотрено в теме 2.3).

Семафоры – объекты, позволяющие войти в заданный участок кода не более чем n потокам.

Мьютексы – одноместные семафоры, служащие в программировании для синхронизации одновременно выполняющихся потоков.

Файлы, проецируемые в память.

Лекция № 8, 9 Архитектура операционной системы

Под архитектурой операционной системы понимают структурную и функциональную организацию ОС на основе некоторой совокупности программных модулей. В состав ОС входят исполняемые и объектные модули стандартных для данной ОС форматов, программные модули специального формата (например, *загрузчик ОС, драйверы ввода-вывода*), конфигурационные файлы, файлы документации, модули справочной системы и т.д.

Принципиально важными универсальными подходами к разработке архитектуры ОС являются:

- модульная организация;
- функциональная избыточность;
- функциональная избирательность;
- параметрическая универсальность;
- концепция многоуровневой иерархической вычислительной системы, по которой ОС представляется многослойной структурой;

- разделение модулей на две группы по функциям: ядро – модули, выполняющие основные функции ОС, и модули, выполняющие вспомогательные функции ОС;
- разделение модулей ОС на две группы по размещению в памяти вычислительной системы: резидентные, постоянно находящиеся в оперативной памяти, и транзитные, загружаемые в оперативную память только на время выполнения своих функций;
- реализация двух режимов работы вычислительной системы: привилегированного режима (режима ядра – *Kernel mode*), или режима супервизора (*supervisor mode*), и пользовательского режима (*user mode*), или режима задачи (*task mode*);
- ограничение функций ядра (а следовательно, и количества модулей ядра) до минимального количества необходимых самых важных функций.

Первые ОС разрабатывались как монолитные системы без четко выраженной структуры).

Для построения монолитной системы необходимо скомпилировать все отдельные процедуры, а затем связать их вместе в единый объектный *файл* с помощью компоновщика (примерами могут служить ранние версии ядра *UNIX* или *Novell NetWare*). Каждая процедура видит любую другую процедуру (в отличие от структуры, содержащей модули, в которой большая часть информации является локальной для модуля, и процедуры модуля можно вызвать только через специально определенные точки входа).

Однако даже такие монолитные системы могут быть немного структурированными. При обращении к системным вызовам, поддерживаемым ОС, параметры помещаются в строго определенные места, такие как регистры или *стек*, а затем выполняется специальная команда прерывания, известная как вызов ядра или вызов супервизора. Эта команда переключает машину из режима пользователя в режим ядра, называемый также режимом супервизора, и передает управление ОС. Затем ОС проверяет параметры вызова, для того чтобы определить, какой *системный вызов* должен быть выполнен. После этого ОС индексирует таблицу, содержащую ссылки на процедуры, и вызывает соответствующую процедуру.

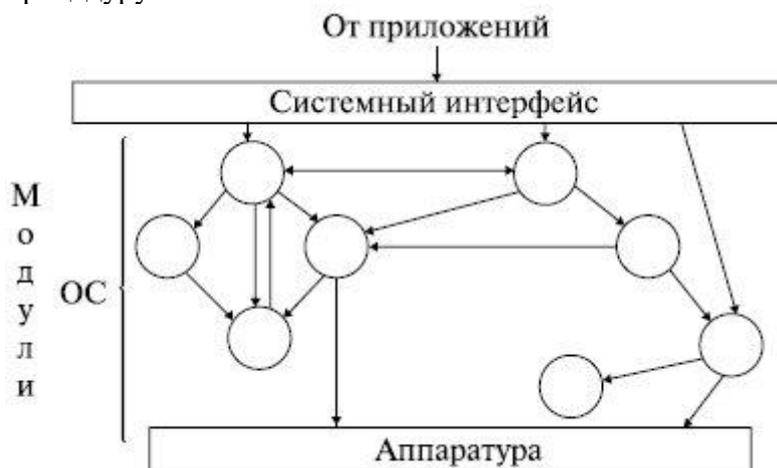


Рис. 1.2. Монолитная архитектура

Такая организация ОС предполагает следующую структуру [13]:

- главная программа, которая вызывает требуемые сервисные процедуры;
- набор сервисных процедур, реализующих системные вызовы;
- набор утилит, обслуживающих сервисные процедуры.

В этой модели для каждого системного вызова имеется одна сервисная процедура. Утилиты выполняют функции, которые нужны нескольким сервисным процедурам. Это *деление* процедур на три слоя показано на рис. 1.3.

Классической считается *архитектура* ОС, основанная на концепции иерархической многоуровневой машины, привилегированном ядре и пользовательском режиме работы транзитных модулей. Модули ядра выполняют базовые функции ОС: *управление процессами*,

памятью, устройствами ввода-вывода и т.п. *Ядро* составляет сердцевину ОС, без которой она является полностью неработоспособной и не может выполнить ни одну из своих функций. В ядре решаются внутрисистемные задачи организации вычислительного процесса, недоступные для приложения.

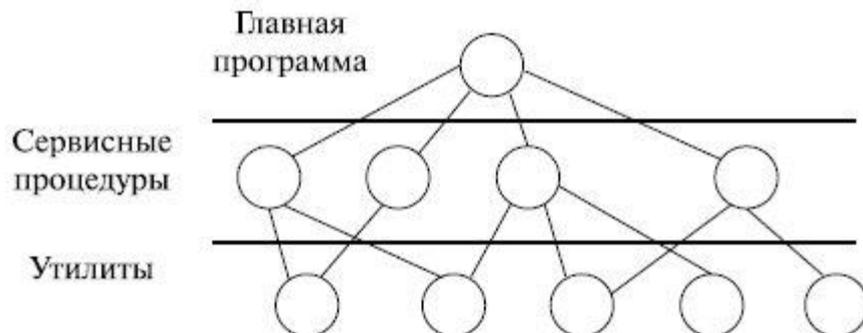


Рис. 1.3. Структурированная архитектура

Особый *класс* функций ядра служит для поддержки приложений, создавая для них так называемую прикладную программную среду. Приложения могут обращаться к ядру с запросами – системными вызовами – для выполнения тех или иных действий, например, открытие и *чтение файла*, получение системного времени, вывода информации на *дисплей* и т.д. Функции ядра, которые могут вызываться приложениями, образуют *интерфейс* прикладного программирования – *API (Application Programming Interface)*. Для обеспечения высокой скорости работы ОС модули ядра (*по крайней мере, большая их часть*) являются резидентными и работают в привилегированном режиме (*Kernel mode*). Этот режим, во-первых, должен обезопасить работу самой ОС от вмешательства приложений, и, во-вторых, должен обеспечить возможность работы модулей ядра с полным набором машинных инструкций, позволяющих собственно ядру выполнять *управление ресурсами* компьютера, в частности, переключение процессора с задачи на задачу, управлением устройствами ввода-вывода, распределением и защитой памяти и др.

Остальные модули ОС выполняют не столь важные функции, как *ядро*, и являются транзитными. Например, это могут быть программы архивирования данных, *дефрагментации диска*, сжатия дисков, *очистки дисков* и т.п.

Вспомогательные модули обычно подразделяются на группы:

- утилиты – программы, выполняющие отдельные задачи управления и сопровождения вычислительной системы;
- системные обрабатывающие программы – текстовые и графические редакторы (Paint, Imaging в Windows 2000), компиляторы и др.;
- программы предоставления пользователю дополнительных услуг (специальный вариант пользовательского интерфейса, калькулятор, игры, средства мультимедиа Windows 2000);
- библиотеки процедур различного назначения, упрощения разработки приложений, например, библиотека функций ввода-вывода, библиотека математических функций и т.п.

Эти модули ОС оформляются как обычные приложения, обращаются к функциям ядра посредством системных вызовов и выполняются в пользовательском режиме (*user mode*). В этом режиме запрещается выполнение некоторых команд, которые связаны с функциями ядра ОС (*управление ресурсами, распределение и защита памяти* и т.п.).

В концепции многоуровневой (многослойной) иерархической машины структура ОС также представляется рядом слоев. При такой организации каждый слой обслуживает вышележащий слой, выполняя для него некоторый набор функций, которые образуют межслойный *интерфейс*. На основе этих функций следующий верхний *по* иерархии слой строит свои функции – более сложные и более мощные и т.д. Такая организация системы существенно упрощает ее разработку, т.к. позволяет сначала "сверху вниз" определить функции слоев и межслойные интерфейсы, а при детальной реализации, двигаясь "снизу вверх", –

наращивать *мощность* функции слоев. Кроме того, модули каждого слоя можно изменять без необходимости изменений в других слоях (но не меняя межслойных интерфейсов!). Многослойная структура ядра ОС может быть представлена, например, вариантом, показанным на рис. 1.4.

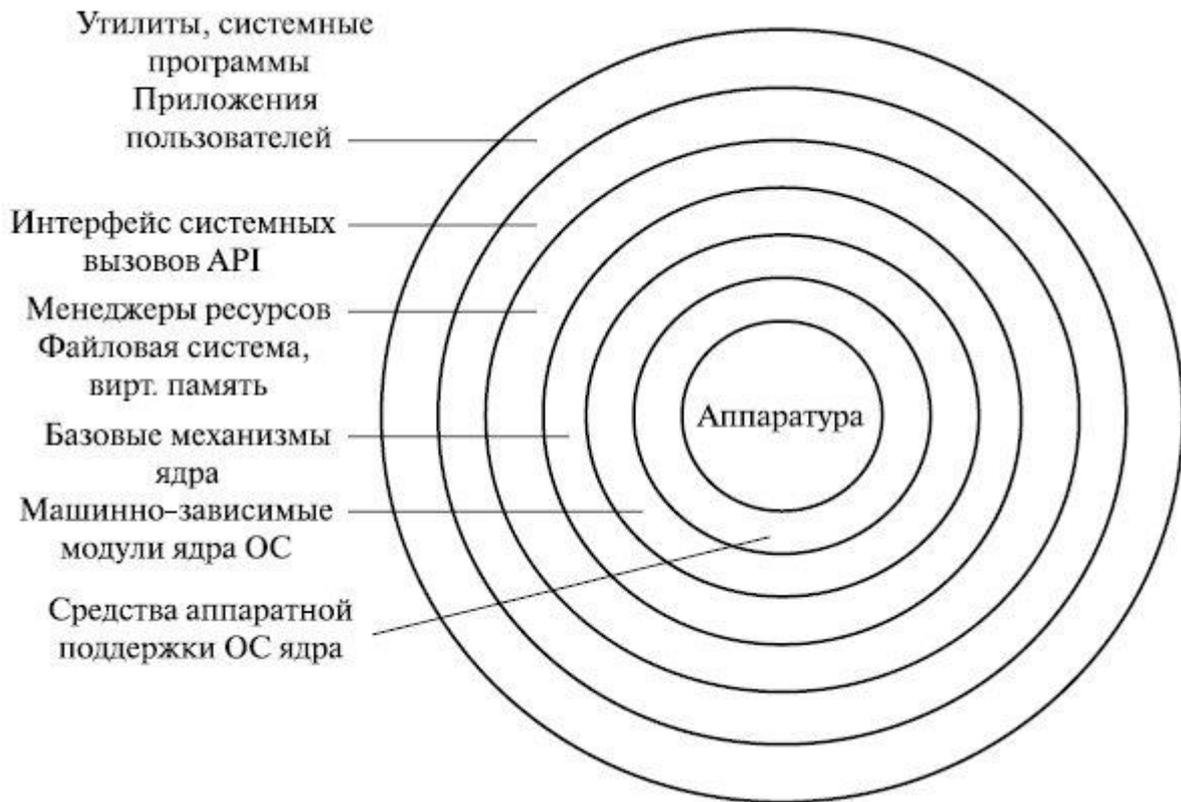


Рис. 1.4. Многослойная структура ОС

Практическая работа по информатике

Тема: «Операционная система.»

«Операционная система.»

Цели:

- закрепить навыки работы с операционной системой Windows;
- отработать навыки работы с файлами и папками в ОС Windows;
- научиться выполнять навигацию с помощью левой панели программы ПРОВОДНИК;
- изучить приемы копирования и перемещения объектов методом перетаскивания между панелями.

Содержание работы:

Задание №1. Заполнить таблицу:

Выполняемое действие	Применяемая команда
1. После загрузки ОС Windows указать, какие кнопки расположены на Панели задач.	<p>Кнопка «Пуск», яндекс, часы, регулятор громкости, значки подключения к локальной сети или интернету, иконка языка, антивирус.</p> 
2. Перечислить, сколько и какие объекты (папки, документы, ярлыки, прикладные программы) расположены на рабочем столе.	<p>11 папок, 4 ярлыка, 2 документа, 5 прикладных программ.</p> 

Задание №2. Заполнить таблицу:

Выполняемое действие	Применяемая команда
----------------------	---------------------

1.Открыть Главное меню. Указать команду.



Щелкнув левой кнопкой мыши на иконку «Пуск» на панели задач или можно нажать на клавишу WIN(кнопка с изображением флага логотипа Microsoft).

2.Перечислить пункты обязательного раздела Главного меню.



3.Перечислить пункты произвольного раздела Главного меню.



Задание №3. Заполнить таблицу:

Выполняемое действие	Применяемая команда
----------------------	---------------------

1.Открыть Контекстное меню. Указать команду.



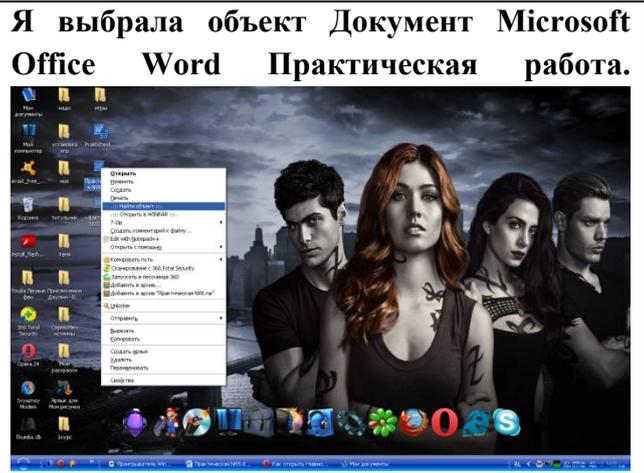
В нижней части клавиатуры, между клавишей "ALT" и клавишей "CTRL" имеется специальная кнопка. На ней обычно нарисована табличка и указатель мыши. Эта кнопка и вызывает контекстное меню.

2.Перечислить пункты Контекстного меню, не выделяя объекты.



Панель инструментов, показать рабочий стол, диспетчер задач, закрепить панель задач, свойства.

3.Перечислить пункты Контекстного меню, выделив какой-либо из объектов. Указать, какой объект выделили.



Я выбрала объект Документ Microsoft Office Word Практическая работа.

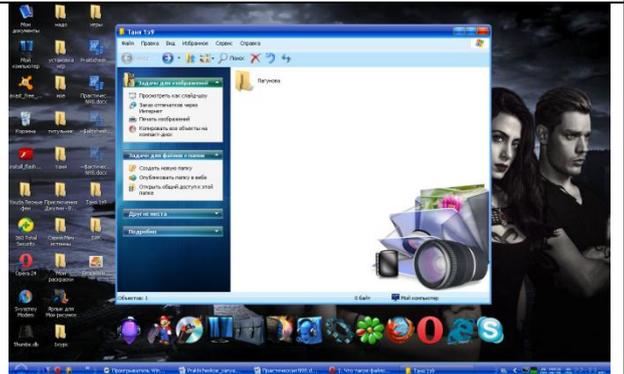
Задание №4. Заполнить таблицу:

Выполняемое действие	Команда
----------------------	---------

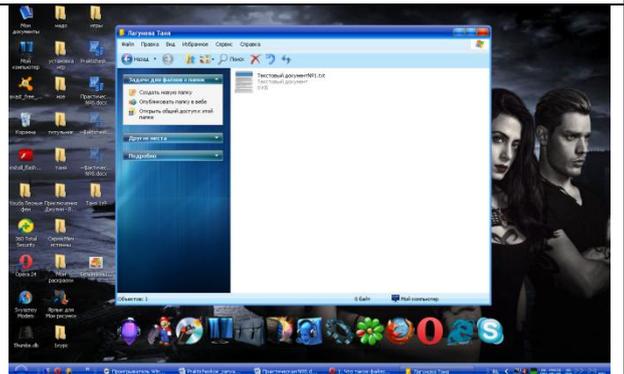
1. Создать на рабочем столе папку с именем – номер группы.



2. В созданной папке создать папку с именем – своя фамилия.



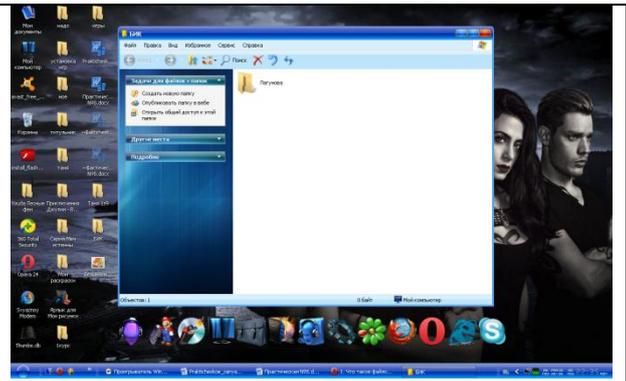
3. В папке с именем – своя фамилия создать текстовый документ. Сохранить его под любым именем.



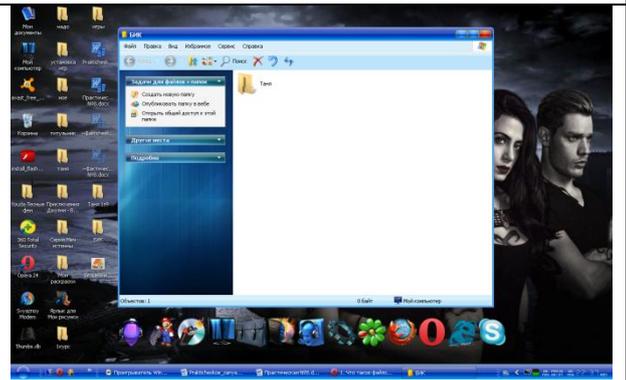
4. Создать на рабочем столе еще одну папку с именем БИК.



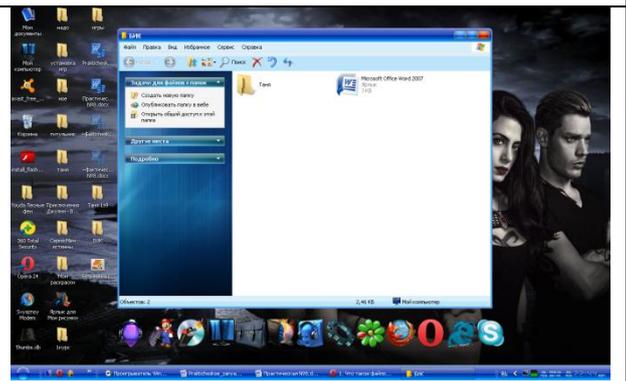
5. Скопировать папку – своя фамилия в папку БИК.



6. Переименовать папку – своя фамилия и дать название – свое имя.



7. Создать в папке БИК ярлык на приложение Word.



8. Удалить с рабочего стола папку – номер группы.



9. Удалить с рабочего стола папку БИК.



10. Открыть папку Мои документы.



11. Упорядочить объекты папки Мои документы по дате.

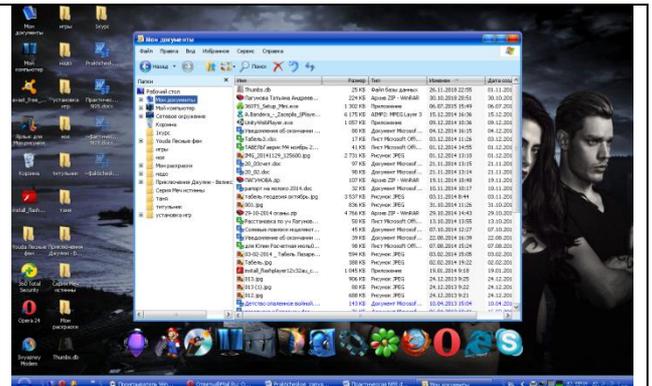


12. Представить объекты папки Мои документы в виде таблицы.

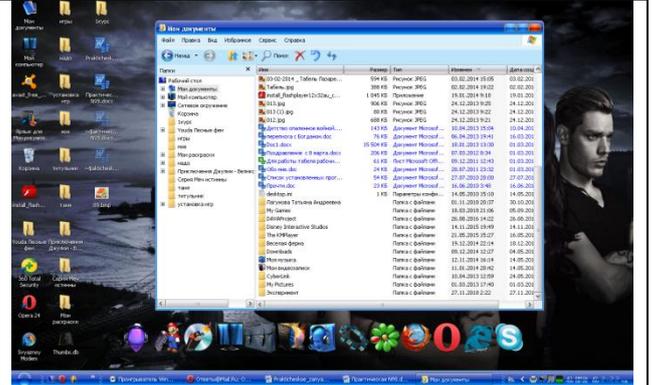


Задание №5. Заполнить таблицу:

- Запустить программу ПРОВОДНИК с помощью главного меню. Указать, какая папка открыта на левой панели ПРОВОДНИКА.



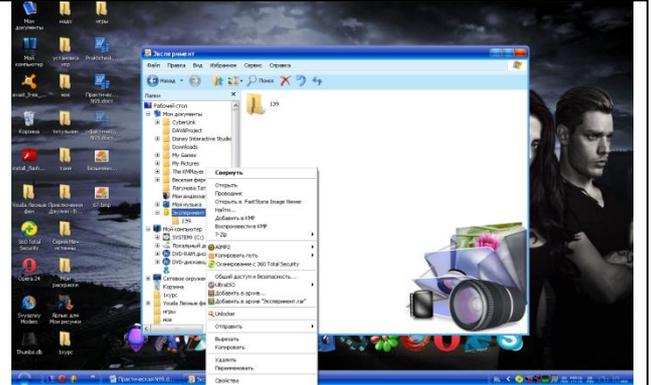
- На правой панели ПРОВОДНИКА создать папку Эксперимент.



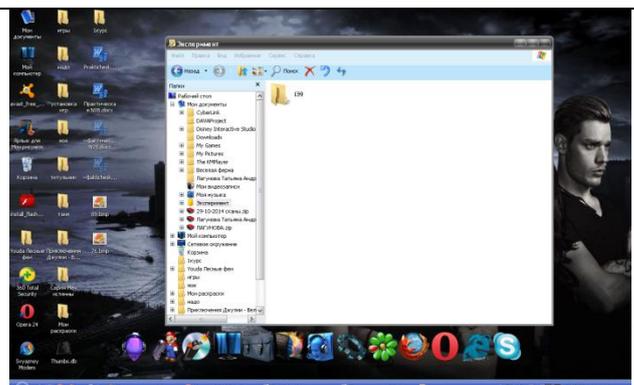
- На левой панели развернуть папку Мои документы щелчком на значке «+». Убедиться в том, что на левой панели в папке Мои документы образовалась вложенная папка Эксперимент.



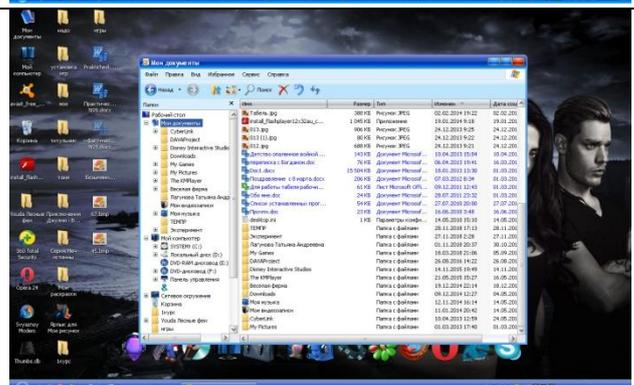
- Открыть папку Эксперимент. Указать содержимое правой панели ПРОВОДНИКА.



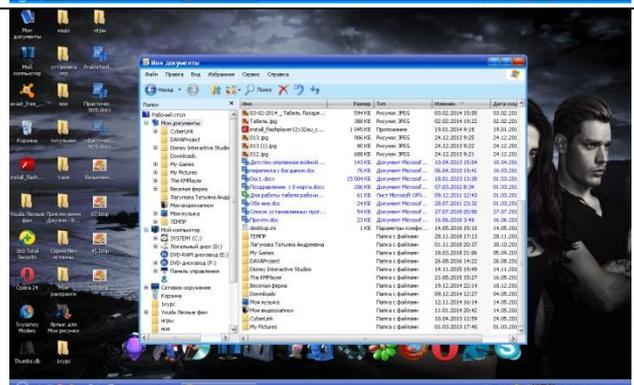
- Создать на правой панели ПРОВОДНИКА новую папку **НОМЕР ГРУППЫ** внутри папки Эксперимент. На левой панели убедиться в том, что рядом со значком папки Эксперимент образовался узел «+». О чем он свидетельствует?



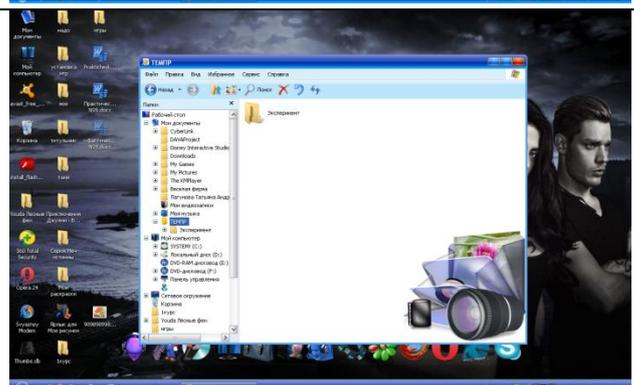
- На левой панели ПРОВОДНИКА разыскать папку **ТЕМП**, но не раскрывать её.



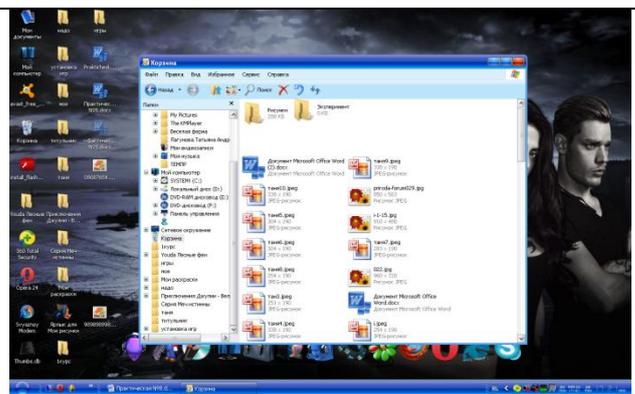
- Методом перетаскивания переместить папку Эксперимент с правой панели ПРОВОДНИКА на левую - в папку **ТЕМП**.



- На левой панели ПРОВОДНИКА открыть папку **ТЕМП**. На правой панели убедиться в наличии в ней папки Эксперимент.



- Разыскать на левой панели ПРОВОДНИКА Корзину и перетащить папку Эксперимент на её значок.

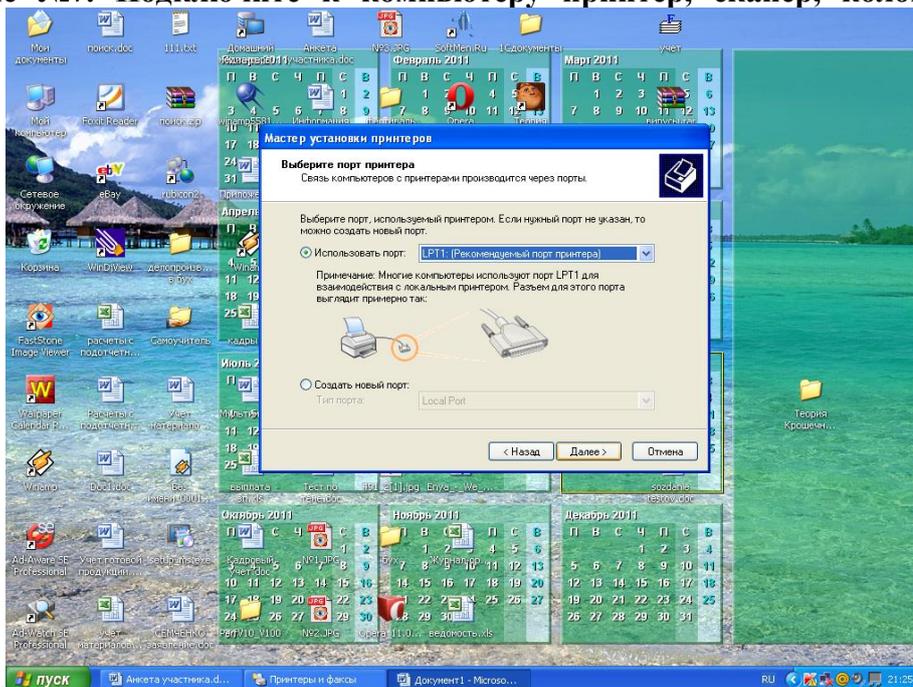


Задание №6. Ответить на вопросы:

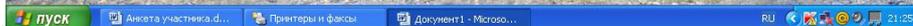
<p>1. Что такое файловая структура компьютера?</p>	<p>Файловая структура компьютера- это порядок хранения и организации файлов на диске.</p>
<p>2. Для чего предназначен ПРОВОДНИК?</p>	<p>Предназначен для навигации по файловой структуре компьютера.</p>
<p>3. Что отображается на левой панели ПРОВОДНИКА?</p>	<p>В левой панели отображается содержание открытых папок (т.е. перечень вложенных папок).</p>
<p>4. Что отображается на правой панели ПРОВОДНИКА?</p>	<p>В правой панели отображается полное содержимое (и файлы, и папки) той папки, которая выделена (открыта) в левой панели.</p>
<p>5. Для чего предназначено Главное меню?</p>	<p>Оно является центральной отправной точкой для запуска программ, а также открытия недавних документов и доступа к свойствам системы.</p>
<p>6. Как открывается контекстное меню?</p>	<p>Существуют разные способы того, как открыть контекстное меню:</p> <ul style="list-style-type: none"> -В нижней части клавиатуры, между клавишей "ALT" и клавишей "CTRL" имеется специальная кнопка. Она показывает добавочные функции и действия, доступные в отношении данного файла. На ней обычно нарисована табличка и указатель мыши. Эта кнопка и вызывает контекстное меню. -Правая кнопка мыши на клавиатуре также с успехом заменяется этой кнопкой. - Наведя мышью на нужный файл, выделяем его щелчком левой кнопки. Щелчок правой кнопкой по выделенной области откроет контекстное меню. При выделении нескольких элементов, в контекстном меню отобразятся действия, доступные для выделенной группы файлов. -Контекстное меню вызывается нажатием на выделенную область правой кнопкой.

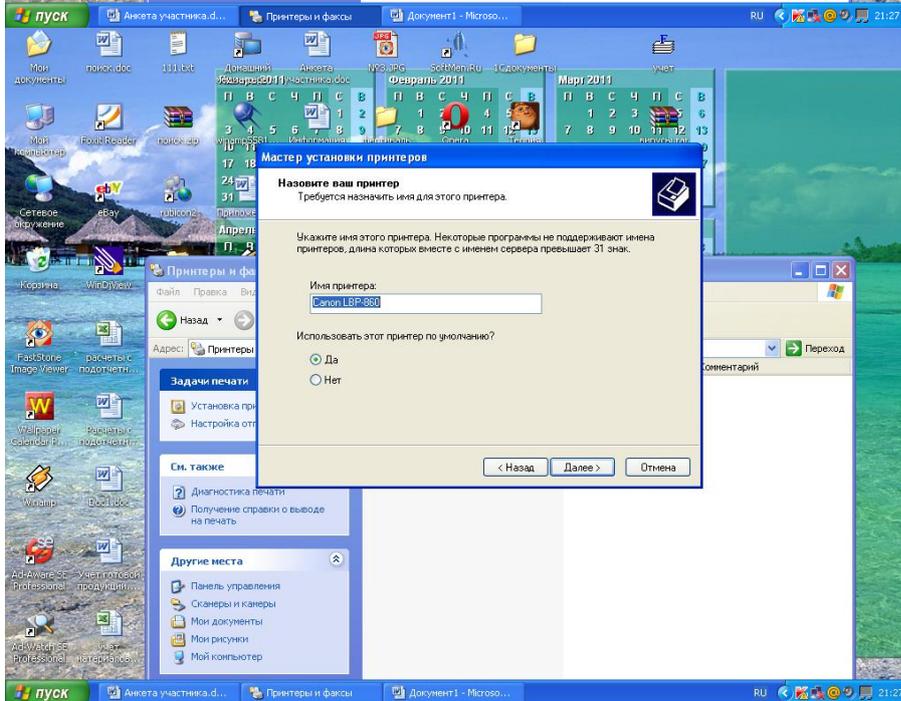
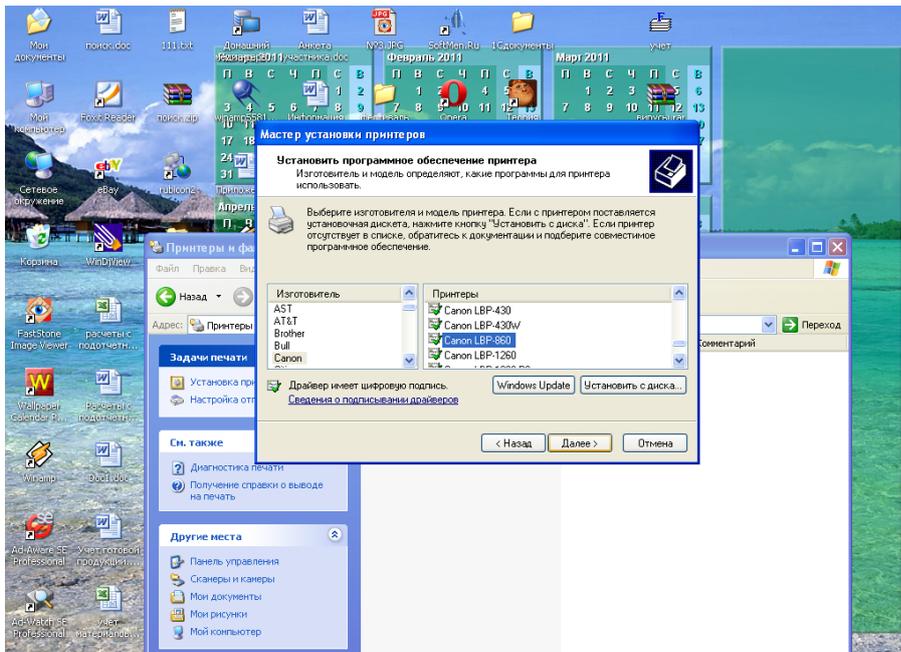
<p>7. В чем особенности ОС Windows?</p>	<p>-Наличие удобных, гибких и простых средств в освоении взаимодействия пользователя со средой – эти средства называются пользовательским интерфейсом. Интерфейс включает окна, меню, ярлыки файлов и приложений.</p> <p>-Многозадачность, т.е. возможность ПЭВМ одновременно работать с несколькими программами.</p> <p>-Возможность переносить данные из одной программы в другую: перенести рисунок и таблицу, создание соответственно графическим редактором и табличным процессором, в окно текстового редактора и создать в нем законченный документ.</p> <p>-Наличие системы настройки новых периферийных устройств при подключении их к ПК.</p>
<p>8. Что является средствами управления ОС Windows?</p>	<p>Средствами управления ОС Windows являются курсор мыши и кнопки мыши, клавиатура.</p>
<p>9. Перечислите основные элементы управления ОС Windows?</p>	<p>Основные элементы управления ОС Windows в первую очередь это панель управления и меню «Пуск», а потом контекстное меню, панель задач, ярлыки, справочная система и т.д.</p>
<p>10. Для чего предназначена Корзина?</p>	<p>Корзина предназначена для того, что бы туда перемещать ненужные файлы, программы, документы.</p>
<p>11. Перечислите основные типы представления объектов.</p>	<p>Существует четыре типа представления объектов:</p> <ul style="list-style-type: none"> • Плитка; • Значки; • Список; • Таблица.
<p>12. Перечислите методы сортировки объектов.</p>	<p>Сортировка в линейных структурах: вставка(простая, бинарная),выбор, обмен.</p> <p>Сортировка в нелинейных структурах: турнирная, пирамидальная.</p>

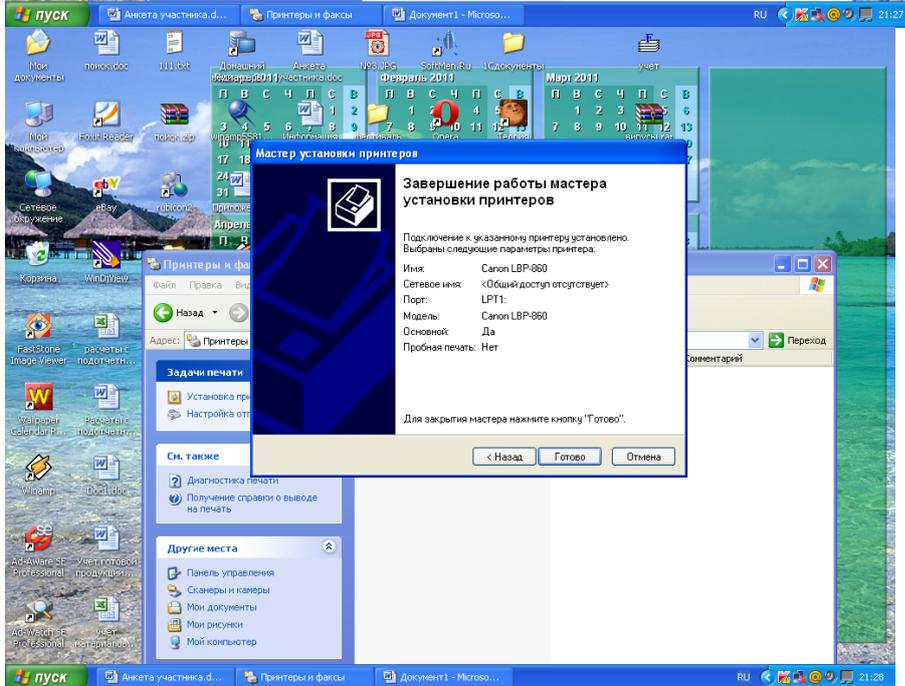
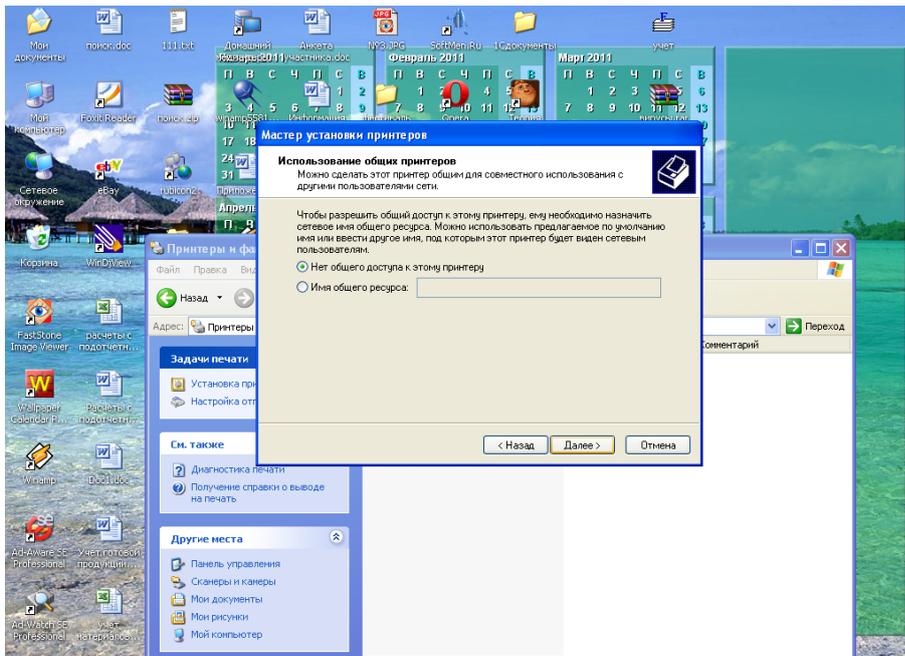
Задание №7. Подключите к компьютеру принтер, сканер, колонки и настройте их

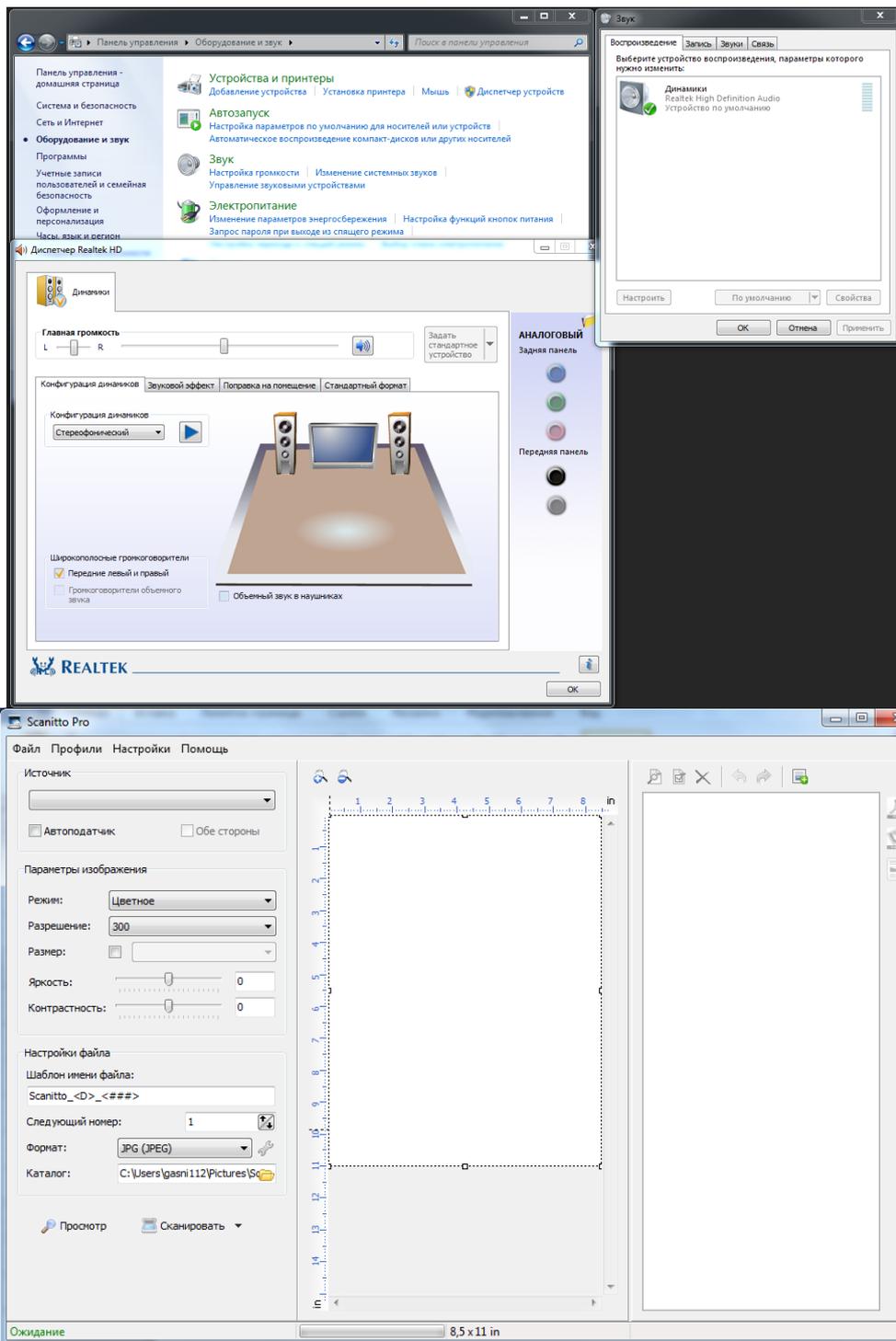


работу.









Задание №8. Сделать вывод о проделанной практической работе:

Я закрепила навыки работы с операционной системой Windows. Отработала навыки работы с файлами и папками в ОС Windows. Научилась выполнять навигацию с помощью левой панели программы ПРОВОДНИК. Изучила приемы копирования и перемещения объектов методом перетаскивания между панелями.

3.3 САМОСТОЯТЕЛЬНАЯ РАБОТА

В результате самостоятельной работы студентом должны быть изучены разделы, решены задачи и даны ответы на следующие вопросы:

Самостоятельная работа студентов

№ п/п	Наименование раздела
1	Структура вычислительной системы. Слои программного обеспечения. Краткая история и этапы эволюции вычислительных систем. Определение операционной системы. Основные понятия и концепции ОС. Требования к современным ОС. Концепция мультипрограммирования: аппаратная и программная поддержка.
2	Функции операционных систем. Схемы классификации ОС. Реализация многозадачности. Поддержка многопользовательского режима. Поддержка многопотоковости. Многопроцессорная обработка.
3	Особенности методов построения ОС. Архитектурные особенности ОС. Ядро и вспомогательные модули ОС. Привилегированный режим и режим пользователя. Монолитное ядро. Слоеные системы; слои ядра ОС. Виртуальные машины. Микроядерная архитектура. Смешанные системы. Преимущества и недостатки различных типов архитектур.
4	Аппаратная зависимость и переносимость ОС. Типовые средства аппаратной поддержки ОС. Средства поддержки привилегированного режима. Средства трансляции адресов. Средства переключения процессов. Система прерываний. Системный таймер. Средства защиты областей памяти. Машинно-зависимые компоненты ОС и роль их локализации при переносимости ОС.
5	Функции ОС по управлению памятью в многозадачной системе. Управление оперативной памятью в ОС. Методы управления физической памятью без использования дискового пространства: фиксированные разделы, динамические разделы и перемещаемые разделы.
6	Управление оперативной памятью в ОС. Методы управления физической памятью с использованием дискового пространства. Метод страничного распределения. Метод сегментного распределения. Метод сегментно-страничного распределения: этапы преобразования виртуального адреса в физический.
7	Понятие файловой системы. Основные функции файловой системы. Способы выделения дискового пространства. Управление свободным и занятым дисковым пространством. Надежность, целостность и производительность файловой системы. Современные архитектуры файловых систем.

Вопросы к экзамену

1. Определение операционной системы. Назначение и функции ОС.
2. Классификация операционных систем по назначению. Требования к современным ОС.
3. Понятие процесса. Состояния процесса в многозадачной системе.
4. Свойства и классификация процессов. Понятие трассы.
5. Понятие ресурса. Свойства и классификация ресурсов. Действия над ресурсами.
6. Структура блока управления процессом (PCB). Контекст процесса.

7. Критерии планирования процессов и требования к алгоритмам планирования.
8. Планирование процессов. Концепция квантования.
9. Планирование процессов. Концепция приоритетности.
10. Концепция вытесняющей многозадачности в ОС.
11. Концепция невытесняющей многозадачности в ОС.
12. Концепция многонитиевой обработки в многозадачных ОС.
13. Планирование процессов и нитей в ОС Windows NT. Классы и уровни приоритетов.
14. Алгоритмы планирования процессов FCFS, RR, SJF. Примеры.
15. Алгоритмы вытесняющего и невытесняющего приоритетного планирования. Примеры.
16. Цели и средства синхронизации. Необходимость синхронизации. Гонки.
17. Задачи синхронизации («Читатели-писатели», «Обедающие философы», «Потребители-производители»).
18. Тупики: условия возникновения и основные способы борьбы.
19. Способы восстановления системы после тупиков.
20. Критическая секция. Использование блокирующих переменных.
21. Задача взаимного исключения. Реализация с помощью системных функций в ОС Windows NT.
22. Синхронизирующие объекты ОС. Семафоры. Мьютексы. Сигналы. Сообщения.
23. Причины кооперации процессов. Категории средств обмена информацией.
24. Способы адресации при установлении связи между процессами.
25. Передача информации по линиям связи. Буферизация.
26. Передача информации по линиям связи. Модели передачи данных.
27. Обеспечение надежности средств связи при кооперации процессов. Завершение связи.
28. Архитектура ОС. Ядро и вспомогательные модули ОС.
29. Привилегированный режим и режим пользователя.
30. Многослойная архитектура ОС. Слои ядра.
31. Микроядерная архитектура ОС.
32. Аппаратная зависимость и переносимость ОС. Типовые средства аппаратной поддержки ОС.
33. Машинно-зависимые компоненты ОС, роль их локализации при переносимости ОС.
34. Функции ОС по управлению памятью. Типы адресов.
35. Классификация методов распределения физической памяти. Метод фиксированных разделов.
36. Классификация методов распределения физической памяти. Метод динамических разделов.
37. Классификация методов распределения физической памяти. Метод перемещаемых разделов.
38. Механизм и реализация виртуальной памяти. Архитектурные средства поддержки виртуальной памяти в ОС. Свопинг.
39. Страничное распределение памяти. Преобразование виртуального адреса в физический.
40. Сегментное распределение памяти. Преобразование виртуального адреса в физический.
41. Сегментно-страничное распределение памяти. Преобразование виртуального адреса в физический.
42. Принцип кэширования данных. Алгоритм выполнения запроса к оперативной памяти.
43. Способы отображения основной памяти на кэш. Проблема согласования данных.
44. Основные понятия информационной безопасности ОС. Классификация возможных угроз.
45. Свойства безопасных систем. Классы информационной безопасности.
46. Политика информационной безопасности. Базовые принципы разработки ОС с точки зрения обеспечения безопасности.
47. Криптография. Методы шифрования информации в ОС.
48. Система защиты в ОС Windows NT/2000.

49. Устройства внутренней памяти ПК. Недостатки и пути их устранения. Основные параметры подсистемы памяти.
50. Устройства внешней памяти ПК. Недостатки и пути их устранения. Основные параметры подсистемы памяти.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ДИСЦИПЛИНЕ ДЛЯ ПРЕПОДАВАТЕЛЕЙ

Лекционный курс должен строиться таким образом, чтобы, приступая к изучению каждой новой темы, студенты знали, какие вопросы ранее изученного материала будут использованы при изучении нового. Каждая лекция должна носить изучение практической стороны программного комплекса. Студенты должны привлекаться к постановке новой задачи и к поиску путей ее решения, обосновывают практическое применение форм. Используемые методы должны ориентировать будущего специалиста на их усвоение и применение в будущей профессиональной деятельности.

В начале каждой лекции необходимо уяснить цель, которую лектор ставит перед собой и перед студентами. Необходимо ориентировать студентов на сравнение того, что он слышит на лекции с тем, что им было изучено ранее, укладывать новую информацию в собственную, уже имеющуюся у него систему знаний. По ходу лекции целесообразно подчеркивать новые понятия, выяснять их смысл.

Наряду с традиционными формами преподнесения материала разделы дисциплины целесообразно сопровождать демонстрацией на экране через мультимедийный проектор, а студентам предоставить возможность на своих рабочих местах выполнять соответствующие задания. На лабораторных работах группа должна быть разделена таким образом, чтобы каждому предоставлялась возможность индивидуальной работы. Выполняемый в течении одного занятия информационный ресурс должен быть надежно сохранен и продолжение его формирования осуществляется на последующих лабораторных занятиях. Итоговый контроль может осуществляться как с экрана (информационный ресурс при этом целесообразно сохранить), так и в распечатанном виде на твердом носителе.

Важная роль должна быть отведена на лекции дискуссии. С этой целью в процессе подготовки к лекции целесообразно продумать систему вопросов, на которые должны ответить студенты, с полным обоснованием своих утверждений.

В конце лекции вместе со студентами целесообразно подвести ее итоги и убедиться, что поставленная цель достигнута.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ДИСЦИПЛИНЕ ДЛЯ СТУДЕНТОВ

Осваивая курс студенту необходимо научиться работать на лекциях, лабораторных занятиях организовывать самостоятельную внеаудиторную деятельность.

В начале лекции необходимо уяснить цель, которую лектор ставит перед собой и студентами. Важно внимательно слушать лектора, отмечать наиболее существенную информацию и кратко записывать ее в тетрадь. Сравнить то, что услышано на лекции с прочитанным и усвоенным ранее, укладывать новую информацию в собственную, уже имеющуюся, систему знаний. По ходу лекции важно подчеркивать новые термины, устанавливать их взаимосвязь с изученными ранее понятиями.

Необходимо очень тщательно вслед за лектором делать записи, таблицы. Если лектор приглашает к дискуссии, необходимо принять в ней участие.

Если на лекции студент не получил ответа на возникшие у него вопросы, необходимо в конце лекции задать их лектору.

В ходе самостоятельной подготовки к семинарскому занятию необходимо прочитать записанную лекцию, подчеркнуть наиболее важные моменты, составить словарь новых терминов, составить план ответа на каждый из предлагаемых для изучения вопросов. Для более глубокого усвоения темы необходимо прочесть рекомендованный преподавателем материал из учебной литературы.

В процессе подготовки к занятиям необходимо воспользоваться материалами учебно-методического комплекса дисциплины.

Эффективность самостоятельной работы зависит от таких факторов как:

- уровень мотивации студентов к овладению конкретными знаниями и умениями;

- - наличие навыка самостоятельной работы, сформированного на предыдущих этапах обучения;
- - наличие четких ориентиров самостоятельной работы.
- Приступая к самостоятельной работе, необходимо получить следующую информацию:
 - - цель изучения конкретного учебного материала;
 - место изучаемого материала в системе знаний, необходимых для формирования специалиста;
 - перечень знаний и умений, которыми должен овладеть студент;
 - порядок изучения учебного материала;
 - источники информации;
 - наличие контрольных заданий;
 - - форма и способ фиксации результатов выполнения учебных заданий;
 - сроки выполнения самостоятельной работы.
- Эта информация представлена в учебно-методическом комплексе дисциплины.

При выполнении самостоятельной работы рекомендуется:

- - записывать ключевые слова и основные термины,
- - составлять словарь основных понятий,

Результатом самостоятельной работы должна быть систематизация и структурирование учебного материала по изучаемой теме, включение его в уже имеющуюся у студента систему знаний.

После изучения учебного материала необходимо проверить усвоение учебного материала с помощью предлагаемых тестов текущего контроля и при необходимости повторить учебный материал.

В процессе подготовки к экзамену необходимо систематизировать, запомнить учебный материал, научиться применять его при решении конкретных задач по обеспечению безопасности бизнеса.

Важнейшим условием для успешного формирования прочных знаний является их упорядочивание, приведение их в единую систему. Это осуществляется в ходе выполнения учащимся следующих видов работ по самостоятельному структурированию учебного материала: запись ключевых слов, составление словаря терминов, составление классификаций, выявление причинно-следственных связей, составление опорных схем и конспектов,

Основная литература

1. В.Г. Олифер, Н.А. Олифер. Сетевые операционные системы. Учебник. – СПб.: Питер, 2002. (5 экз. - ч/з, 19 экз. - уч/а)
2. Д.В. Иртегов. Введение в операционные системы. – СПб.: БХВ-Петербург, 2002г.
3. А.Г. Андреев и др. Microsoft Windows 2000 Professional. - СПб.: БХВ-Петербург, 2001г.
4. Б. Леонтьев. Как установить и настроить Microsoft Windows 98. – М., Познавательная книга +, 1998. (1 экз. в ч/з).

5. Питер Нортон. Персональный компьютер фирмы IBM и ОС MS-DOS. – М.: Радиосвязь, 1991. (11 экз. – н/а)
6. А. Робачевский. ОС Unix. – БХВ, 1999г.
7. Х. Кастер. Основы Windows NT и NTFS. – М.: “Русская редакция” ТОО “Channel Trading Ltd”, 1996.

3.1.2. Дополнительная литература

8. М. Дансмур, Г. Дейвис. ОС UNIX и программирование на языке Си. – М., 1989. (7 экз. в ч/з)
9. У. Дэвис. Операционные системы. – М.: Мир, 1980. (2 экз. в ч/з)
10. М. Гук. Аппаратные средства IBM PC. Энциклопедия. - СПб.: Питер, 2001г.
11. М. Гук. Дисковая подсистема ПК. - СПб.: Питер, 2001г.
12. К. Ахметов. Знакомство с Microsoft Windows XP. – Microsoft Press, 2001г.

ГЛОССАРИЙ

Internet 🌐 — всемирная глобальная Сеть (Сеть сетей). Была создана в 1995 г., на первых этапах контролировалась National Science Foundation (NSF). Представляет собой совокупность взаимосвязанных коммуникационных центров, к которым подключаются региональные поставщики сетевых услуг и через которые осуществляется их взаимодействие.

Internet Explorer 🌐 — программа фирмы Microsoft, входящая в состав Windows, служащая для просмотра информации, размещенной в Интернете.

Intranet 🌐 — локальная (корпоративная) информационная сеть, построенная по принципам глобальной сети Internet.

IP (Internet Protocol) 🌐 — межсетевой рабочий протокол, являющийся основой для Интернета. Благодаря наличию IP возможна маршрутизация пакетов информации между сетями и последующая сборка пакетов после того, как они достигнут места назначения. IP не отвечает за надежность доставки информации, за ее целостность, за сохранение порядка потока пакетов.

IPX/SPX 🌐 — транспортные протоколы, применяемые в сетях Novell NetWare.

ISDN (Integrated Services Digital Network) 🌐 — цифровая сеть с интеграцией услуг, обеспечивающая цифровое соединение между оконечными устройствами сети для предоставления широкого набора услуг, к которым пользователи получают доступ через ограниченное число стандартных многофункциональных интерфейсов; относится к классу сетей, изначально предназначенных для передачи как данных, так и голоса.

OSI (Open Systems Interconnection reference model) 🌐 — эталонная модель взаимодействия открытых систем.

SDH (Synchronous Digital Hierarchy) 🌐 — сети с синхронной цифровой иерархией, реализующие технологию синхронных волоконно-оптических сетей. SDH — высокоскоростные сети цифровой связи, отличающиеся высоким уровнем стандартизации, высокой надежностью, наличием полного программного контроля, возможностью оперативного предоставления услуг по требованию, сравнительно простой схемой развития сети.

TCP (Transmission Control Protocol) 🌐 — высокоуровневый протокол (Transmission Control Protocol — протокол управления передачей) с установлением логического соединения между отправителем и получателем. TCP обеспечивает сеансовую связь между двумя узлами с

гарантированной доставкой информации, осуществляет контроль целостности передаваемой информации, сохраняет порядок потока пакетов.

ТСР/IP — протокол для взаимодействия сетей в Интернете; представляет собой семейство программно реализованных протоколов старшего уровня, не работающих с аппаратными прерываниями; состоит из двух частей — ТСР и IP.

Асимметричная операционная система (ОС) — система, которая целиком выполняется только на одном из процессоров системы, распределяя прикладные задачи по остальным процессорам.

Безопасность ОС — означает, что операционная система должна обладать средствами защиты ресурсов одних пользователей от других.

Браузер — прикладная программа, позволяющая получать из Интернета различные документы, просматривать и редактировать их содержимое.

Гипертекст (гипертекстовая связь) — средство соединения информации, содержащейся в одном документе, с информацией из того же или любого другого документа, в том числе объектами нетекстовой природы (звук, изображение, видео), а также система, позволяющая читать такой текст, отслеживать ссылки, отображать картинки и проигрывать звуковые и видеовставки.

Глобальная вычислительная сеть (ГВС) — объединяет абонентские системы, рассредоточенные на большой территории, охватывающей различные страны и континенты; решает проблему объединения информационных ресурсов всего человечества и организации доступа к ним.

Динамические свойства информации — свойства, которые характеризуют изменение информации во времени.

Децентрализованная (одноранговая) сеть — локальная вычислительная сеть, функции управления в которой поочередно передаются от одной рабочей станции к другой и которая не имеет выделенных серверов.

Знание — осознание, понимание и толкование определенной информации с учетом путей наилучшего ее использования для достижения конкретных целей.

Информатика — отрасль знаний, изучающая общие свойства и структуру информации, а также закономерности и принципы ее создания, преобразования, накопления, передачи и использования в различных областях человеческой деятельности на базе современных средств вычислительной и телекоммуникационной техники.

Информатизация — организационный, социально-экономический и научно-технический процесс создания оптимальных условий для удовлетворения потребностей и реализации прав граждан, органов государственной власти, органов местного самоуправления, организаций, общественных объединений на основе формирования и использования информационных ресурсов.

Информационные процессы — процессы сбора, обработки, накопления, хранения, поиска и распространения информации.

Информационные ресурсы — отдельные документы и отдельные массивы документов, документы и массивы документов в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах).

Источник информационного ресурса — определяет происхождение информации и в определенном смысле выступает как часть параметра охвата, ограничивающая содержание информации.

Каталоги Internet — средства хранения тематически систематизированных коллекций ссылок на различные сетевые ресурсы, в первую очередь на документы WWW.

Качество информационного ресурса — определяет (задает) совокупность свойств, отражающих степень пригодности конкретной информации об объектах и их взаимосвязях для достижения целей, стоящих перед пользователем, при реализации тех или иных видов деятельности.

Компьютерный вирус — программа (некоторая совокупность выполняемого кода и/или инструкций), которая способна создавать свои копии (не обязательно полностью совпадающие с оригиналом) и внедрять их в различные объекты и/или ресурсы компьютерных систем, сетей и т.д.

Мировые информационные ресурсы — информационные ресурсы, которые рассматриваются как совокупность информационных ресурсов различных государств.

Многозадачные ОС — системы, которые управляют разделением совместно используемых ресурсов, таких как процессор, оперативная память, файлы и внешние устройства.

Многонитевая ОС — система, которая разделяет процессорное время не между задачами, а между их отдельными ветвями (нитьями).

Монолитное ядро ОС — компонуется как одна программа, работающая в привилегированном режиме и использующая быстрые переходы с одной процедуры на другую, не требующие переключения между привилегированным и пользовательским режимами.

Надежность и отказоустойчивость ОС — означает, что система должна быть защищена как от внутренних, так и от внешних ошибок, сбоев и отказов. Ее действия должны быть всегда предсказуемыми, а приложения не должны быть способны наносить вред ОС.

Наличие нескольких прикладных сред — дает возможность в рамках одной ОС одновременно выполнять приложения, разработанные для нескольких ОС. Многие современные операционные системы поддерживают одновременно прикладные среды MS-DOS, Windows, UNIX (POSIX), OS/2 или хотя бы некоторого подмножества из этого популярного набора. Концепция множественных прикладных сред наиболее просто реализуется в ОС на базе микроядра, над которым работают различные серверы, часть которых реализует прикладную среду той или иной операционной системы.

Новый информационный ресурс — информационный ресурс, который порожден впервые и не представляет собой повторения тождественного или аналогичного.

Объем охвата информационного ресурса — общее количество информации по проблеме, доступной пользователю.

Однозадачные ОС — выполняют функцию предоставления пользователю виртуальной машины, делая более простым и удобным процесс взаимодействия пользователя с компьютером. Однозадачные ОС включают средства управления периферийными устройствами, средства управления файлами, средства общения с пользователем.

Одноранговая сеть — сеть, в которой два или несколько компьютеров могут взаимодействовать друг с другом, не прибегая к каким-либо промежуточным устройствам; в одноранговой сети компьютеры могут быть одновременно и клиентами, и серверами.

Операционная система (ОС) — это программный комплекс, одной из важнейших задач которого является предоставление пользователю возможности использовать ресурсы компьютера по своему усмотрению в максимально доступном объеме, не отвлекаясь на проблемы управления аппаратными ресурсами, находящиеся за гранью его возможностей.

ОС на базе микроядра — работает также в привилегированном режиме и выполняет только минимум функций по управлению аппаратурой, в то время как функции ОС более высокого уровня выполняют специализированные компоненты — ОС-серверы, работающие в пользовательском режиме.

Охват информационного ресурса — определяет, ограничивает и описывает содержание, уточняет или ограничивает его. В конкретном смысле охват можно рассматривать как часть параметра «содержание». Он как бы суживает и задает определенные рамки содержания.

Переносимость ОС — имеет место при условии, что код легко переносится с процессора одного типа на процессор другого типа и с аппаратной платформы (которая включает наряду с типом процессора и способ организации всей аппаратуры компьютера) одного типа на аппаратную платформу другого типа.

Подсеть — часть сети TCP/IP, в которой все устройства имеют одинаковый префикс.

Поисковый сервер (search engine) — специальное программное обеспечение, которое, автоматически просматривая все ресурсы сети Internet, может найти запрашиваемые ресурсы и проиндексировать их содержание.

Полезность информационного ресурса — характеризует пригодность для определенной цели, способность функционировать в чьих-либо интересах, в соответствии с чьими-нибудь выгодами.

Полнота охвата информационного ресурса — соотношение между имеющейся информацией по проблеме и информацией, доступной пользователю (т.е. той ее частью, которую он может получить).

Построение ОС на базе объектно ориентированного подхода — дает возможность использовать все достоинства подхода, хорошо зарекомендовавшие себя на уровне приложений, внутри операционной системы, а именно: аккумуляцию удачных решений в форме стандартных объектов, возможность создания новых объектов на базе имеющихся с помощью механизма наследования, хорошую защиту данных за счет их инкапсуляции во внутренние структуры объекта, что делает данные недоступными для несанкционированного использования извне, структурированность системы, состоящей из набора хорошо определенных объектов.

Почтовые списки — списки рассылки (Mailing Lists) — представляют собой один из видов сервиса глобальной сети, когда в Сети выделяется адрес электронной почты, который является общим для многих пользователей — подписчиков определенного списка рассылки; пользователи-подписчики посылают свои сообщения по общему адресу, и эти сообщения рассылаются всем, кто подписался на данный список рассылки.

Прагматические свойства информации — свойства, которые характеризуют степень полезности информации для пользователя, потребителя и практики.

Прикладной процесс — различные процедуры ввода, хранения, обработки и выдачи информации, выполняемые в интересах пользователей и описываемые прикладными программами.

Программные антивирусные средства 🌸, или **антивирусы** 🌸 — специальные программы, которые находят и уничтожают вирусы на компьютере.

Производительность ОС 🌸 — означает, что система должна обладать настолько хорошим быстродействием и временем реакции, насколько это позволяет аппаратная платформа.

Рабочая станция 🌸 — обычный персональный компьютер, на котором пользователи Сети реализуют прикладные задачи.

Распределенная организация операционной системы 🌸 — позволяет упростить работу пользователей и программистов в сетевых средах. В распределенной ОС реализованы механизмы, которые дают возможность пользователю представлять и воспринимать сеть в виде традиционного однопроцессорного компьютера. Характерными признаками распределенной организации ОС являются: наличие единой справочной службы разделяемых ресурсов, единой службы времени, использование механизма вызова удаленных процедур (RPC) для прозрачного распределения программных процедур по машинам, многократной обработки, позволяющей распараллеливать вычисления в рамках одной задачи и выполнять эту задачу сразу на нескольких компьютерах сети, а также наличие других распределенных служб.

Расширяемость ОС 🌸 — имеет место при условии, что код написан таким образом, чтобы можно было легко внести дополнения и изменения, если это потребуется, и не нарушить целостность системы.

Своевременность информационного ресурса 🌸 — поступление информации в пределах того времени, когда она полезна для принятия решения и когда она еще может повлиять на результат принятия решения (деятельность).

Сервер 🌸 — компьютер в Сети, обслуживающий другие компьютеры; его ресурсы доступны всем рабочим станциям Сети.

Сетевые серверы 🌸 — аппаратно-программные системы, выполняющие функции управления распределением сетевых ресурсов общего доступа, имеющие возможности работать в качестве обычной абонентской системы.

Сеть 🌸 — два или несколько компьютеров, соединенных друг с другом, благодаря чему они могут совместно использовать какие-либо ресурсы.

Симметричная ОС 🌸 — полностью децентрализованная ОС, которая использует весь пул процессоров, разделяя их между системными и прикладными задачами.

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

Системы разделения времени 🌸 — призваны исправить основной недостаток систем пакетной обработки — изоляцию пользователя-программиста от процесса выполнения его задач. Каждому пользователю системы разделения времени предоставляется терминал, с которого он может вести диалог со своей программой.

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

Служба каталогов 🌸 — предоставляет средства для хранения данных о каталогах и делает эти данные доступными для пользователей и администраторов сети.

Совместимость ОС 🍁 — означает, что ОС должна иметь средства для выполнения прикладных программ, написанных для других операционных систем. Кроме того, пользовательский интерфейс должен быть совместим с существующими системами и стандартами.

Содержание информационного ресурса 🍁 — определяет проблемную область, охватываемую информационными ресурсами (тему, идею, теорию, методику). Границы проблемной области зависят от задач, решаемых пользователем. Причем различные группы пользователей, решающие аналогичные задачи и реализующие одинаковые цели, различным образом определяют границы проблемной области (свои потребности в информационных ресурсах), что ведет, как правило, к различию в результатах их деятельности.

Ценность информационного ресурса 🍁 — его важность, необходимость для принятия решений.