Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гриб Владислав Валерьевич
Должность: Ректор
Дата подписания: 23.09.2026 19:01:43
Уникальный программный ключ:
637517d24e103c3db032acf37e839d98ec1c5bb2f5eb89c29abfcd7f4398544

**Образовательное частное учреждение высшего образования**

**«МОСКОВСКИЙ УНИВЕРСИТЕТ ИМЕНИ А.С. ГРИБОЕДОВА»**

(ИМПЭ им. А.С. Грибоедова)

**МЕЖДУНАРОДНЫЙ ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И БИЗНЕС-ИНФОРМАТИКИ**

**УТВЕРЖДАЮ**
Директор Международного института информационных технологий и бизнес-информатики
_____ Т.В. Новикова
«17» декабря 2025г.

# ОПЕРАЦИОННЫЕ СИСТЕМЫ

**Направление подготовки**
**09.03.03 Прикладная информатика**
**(уровень бакалавриат)**

**Направленность (профиль):**
**«Анализ данных»**

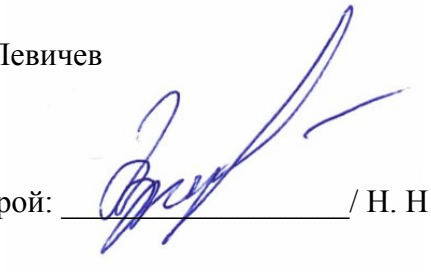
**Форма обучения**: **очная, заочная**

**Москва**

Фонд оценочных средств для дисциплины «Операционные системы». Направление подготовки/специальность 09.03.03 Прикладная информатика, направленность (профиль/специализация): «Анализ Данных» О.А. Левичев – М.: ИМПЭ им. А.С. Грибоедова.

Фонд оценочных средств является неотъемлемой частью рабочей программы дисциплины.

Разработчик: О.А. Левичев



Заведующий кафедрой: _____/ Н. Н. Загускин

## 1. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Настоящий Фонд оценочных средств (ФОС) по дисциплине «Операционные системы» является неотъемлемым приложением к рабочей программе дисциплины (РПД) «Операционные системы». На данный ФОС распространяются все реквизиты утверждения, представленные в РПД по данной дисциплине.

## 2. ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ

Для определения качества освоения обучающимися учебного материала по дисциплине используются следующие оценочные средства:

| № п/п | Оценочное средство | Краткая характеристика оценочного средства | Представление оценочного средства в ФОС |
|---|---|---|---|
| 1 | Тестирование | Вид контроля, позволяющий оценить изученный теоретический материал. | Вопросы для проведения тестирования |
| 2 | Практические задания | Вид контроля, позволяющий оценить умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач | Задания к практическому занятию |
| 3 | Контрольная работа | Вид контроля, позволяющий определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций | Задания контрольной работы |
| 4 | Самостоятельная работа | Вид контроля, позволяющий оценить проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборку нормативного и иного материала и выполнение других заданий | Задания самостоятельной работы |
| 5 | Курсовая работа | Вид контроля, позволяющий выявить степень владения базовыми знаниями, умениями и навыками, необходимыми для обучения, и определить уровень владения новым материалом | Индивидуальные задания (темы) для курсовой работы |
| 6 | Зачет/Зачет с оценкой/Экзамен | Вид контроля, позволяющий выявить степень овладения знаниями, умениями и навыками, необходимыми для дальнейшего освоения образовательной программы | Вопросы для подготовки к зачету/зачету с оценкой/экзамену |

| | | | |
|---|---|---|---|
| | | подготовки | |

## 3. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЕ

### 3.1. Сопроводительная информация.

| | |
|---|---|
| Разработчик | О.А. Левичев |
| Кафедра | Прикладной информатики и информационных технологий |
| Наименование дисциплины | Операционные системы |
| Факультет / институт | Институт международной экономики, лидерства и менеджмента |
| Направление подготовки / специальность | 09.03.03 Прикладная информатика |
| Количество вопросов в оценочных заданиях (диапазон) | 20-40 |
| Общее время тестирования (мин) | 90 мин |
| Общее количество вопросов/заданий в ФОС | 40 |
| Размещенность на сайте Университета примерного перечня вопросов, заданий ФОС – для подготовки обучающихся к прохождению оценки (да / нет) | Да |

### 3.1. Модели контролируемых компетенций:

| Код компетенции | Формулировка компетенции | Индикаторы достижения компетенции |
|---|---|---|
| ОПК-5 | Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | **ИОПК-5.1.** Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>**ИОПК-5.2.** Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем |

## 4. СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

### 4.1. ТИПОВЫЕ ТЕСТОВЫЕ ЗАДАНИЯ.

Тесты содержат набор вопросов, в полном объеме охватывающие изученный теоретический материал по указанной теме. Выполнение тестов позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций. Индивидуальный тестовый сеанс для каждого обучающегося формируется по специальному алгоритму, обеспечивающему заданную тематическую структуру и пропорциональное наличие вопросов разного типа и сложности.

***При формировании тестов необходимо использовать задания следующих типов***:

Тип задания 1. Задания закрытого типа на установление соответствия.

Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных.

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных.

Тип задания 5. Задания открытого типа с развернутым ответом.

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Тест по теме |
|---|---|---|---|
| Тема 1 Введение в ОС Linux. Понятие операционной системы. Лицензии на ПО | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 1.1**<br>Какие из следующих утверждений точно описывают понятие «операционная система» (ОС) в современной информатике?<br><br>1. ОС — это комплекс программ, который обеспечивает управление аппаратными ресурсами компьютера (процессором, памятью, устройствами ввода-вывода).<br>2. Основная функция ОС — предоставлять пользователю или прикладным программам удобный интерфейс для работы с вычислительной системой.<br>3. ОС всегда включает в себя графический пользовательский интерфейс (GUI), такой как рабочий стол с окнами.<br>4. ОС выполняет все прикладные задачи пользователя, такие как редактирование текстов или просмотр веб-страниц.<br><br>Ответ: 1, 2<br><br>Пояснение:<br>1 - Правильно. Это ядро определения ОС: управление ресурсами (resource management).<br>2 - Правильно. Это вторая ключевая функция ОС: предоставление интерфейса (abstraction layer) между «железом» и пользователем/программами.<br><br>**Задание 1.2**<br>Какие из следующих положений являются ключевыми принципами лицензии GNU GPL (General Public License), под которой распространяется ядро Linux и большая часть системного ПО в дистрибутивах?<br><br>1. Свобода запуска программы с любой целью.<br>2. Свобода изучения того, как программа работает, и адаптация её под свои нужды (доступ к исходному коду — обязательное условие).<br>3. Свобода распространять копии программы, но только в её исходной, неизменной форме.<br>4. Свобода улучшать программу и публиковать улучшения, но только для некоммерческого использования. |

| | | | |
|---|---|---|---|
| | | | Ответ: 1, 2<br><br>Пояснение:<br>1 - Правильно. Это первая свобода из четырёх, определяемых Фондом свободного ПО (FSF).<br>2 - Правильно. Это вторая свобода. Доступ к исходному коду — фундаментальное требование GPL для обеспечения возможности модификации.<br><br>**Задание 1.3**<br>Установите правильную историческую последовательность событий, связанных с созданием и развитием ОС Linux.<br><br>1. Ричард Столлман объявляет о проекте GNU и создании свободной UNIX-подобной операционной системы.<br>2. Линус Торвальдс объявляет в Usenet о создании небольшой любительской ОС (ядро Linux) и просит feedback.<br>3. Появление первых дистрибутивов Linux (таких как Slackware), объединяющих ядро Линуса и утилиты проекта GNU.<br>4. Эндрю Таненбаум создает учебную ОС Minix, которая вдохновляет Линуса Торвальдса.<br><br>Ответ: 4,1,2,3<br><br>Пояснение: Сначала появляется Minix (1987) как учебный пример. Затем стартует проект GNU (1983) с целью создания свободной ОС, но без ядра. Торвальдс, вдохновленный Minix и нуждаясь в свободном ядре для GNU, создает Linux (1991). И только после этого появляются полноценные дистрибутивы, сочетающие Linux (ядро) и GNU (утилиты).<br><br>**Задание 1.4**<br>Вопрос: Установите последовательность уровней абстракции в архитектуре вычислительной системы (снизу вверх, от аппаратуры к пользователю).<br><br>1. Прикладное программное обеспечение (браузер, текстовый редактор). |

| | | | |
|---|---|---|---|
| | | | 2. Аппаратное обеспечение (процессор, память, диски).<br>3. Пользователь.<br>4. Операционная система (ядро, системные библиотеки, драйверы).<br><br>Ответ: 2,4,1,3<br><br>Пояснение: Классическая слоистая модель: «железо» (2) — низший уровень. Над ним работает ОС (4), которая управляет ресурсами и предоставляет API. На основе этих API строятся приложения (1), с которыми напрямую взаимодействует конечный пользователь (3). |
| Тема 2 Атрибуты файлов. Работа с файлами | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 2.1**<br>Установите правильную последовательность проверки прав доступа к файлу для выполнения операции (например, чтения) в Linux, когда процесс пытается получить доступ к файлу.<br><br>1. Проверка прав доступа для «остальных» (others).<br>2. Проверка, является ли эффективный UID процесса владельцем файла.<br>3. Проверка, входит ли эффективный GID процесса или одна из дополнительных групп процесса в группу-владельца файла.<br>4. Проверка, является ли процесс суперпользователем (root, UID=0).<br><br>Ответ: 4,2,3,1<br><br>Пояснение: Ядро проверяет права по иерархии от наиболее привилегированных к наименее. Сначала проверяется суперпользователь (root обходит все проверки прав). Если процесс не root, проверяется, является ли он владельцем файла. Если нет, проверяется членство в группе-владельце. И только если все предыдущие проверки не прошли, применяются права для «остальных».<br><br>**Задание 2.2**<br>Установите последовательность действий для безопасного копирования файла с сохранением большинства его атрибутов (времени модификации, прав доступа), но без копирования специальных битов (SUID, SGID). |

| | | | |
|---|---|---|---|
| | | | Действия:<br>1. Скопировать содержимое файла: cat source > destination<br>2. Установить на целевом файле права доступа, скопированные с исходного: chmod --reference=source destination<br>3. Скопировать время модификации исходного файла на целевой: touch -r source destination<br>4. Удалить возможные SUID/SGID биты с целевого файла: chmod u-s,g-s destination<br><br>Ответ: 1,2,3,4<br>Пояснение: Сначала создается файл-назначение с содержимым (1). Затем устанавливаются базовые права (rwx) (2). Далее синхронизируется временная метка (3). Важный шаг безопасности — явное снятие SUID/SGID битов (4), чтобы случайно не создать уязвимость. Шаг 4 можно выполнить сразу после копирования содержимого, до установки прав.<br><br>**Задание 2.3**<br>Сопоставьте символьное представление права доступа в выводе команды ls -l с его числовым (восьмеричным) значением.<br><br>Список А (Символьное представление, например, для владельца):<br>A) r--<br>B) -w-<br>C) --x<br>D) rwx<br><br>Список Б (Числовое значение):<br>1. 4<br>2. 2<br>3. 7<br>4. 1<br><br>Ответ: A1, B2, C4, D3 |

| | | | |
|---|---|---|---|
| | | | **Задание 2.4**<br>Сопоставьте команду для работы с файлами и их атрибутами в Linux с ее основным назначением.<br><br>Список А (Команда):<br>A) chmod u+s file<br>B) chmod g+s dir<br>C) chmod o+t dir<br>D) chmod a=rx file<br><br>Список Б (Назначение):<br>1. Установить Sticky Bit на каталог dir.<br>2. Установить SetGID бит на каталог dir.<br>3. Установить SetUID бит на файл file.<br>4. Установить для всех категорий пользователей (владелец, группа, остальные) права на чтение и исполнение для файла file.<br><br>Ответ: A3, B2, C1, D4 |
| Тема 3 Архиваторы | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 3.1**<br>Сопоставьте расширение архивного/сжатого файла с утилитой (или комбинацией утилит), которая используется для его создания и распаковки.<br><br>Список А (Расширение файла):<br>A) .tar.gz или .tgz<br>B) .tar.bz2 или .tbz2<br>C) .tar.xz или .txz<br>D) .tar<br><br>Список Б (Основная утилита/ключ tar для распаковки):<br>1. tar -xf (только распаковка архива без сжатия)<br>2. tar -xzf (распаковка + декомпрессия gzip)<br>3. tar -xjf (распаковка + декомпрессия bzip2)<br>4. tar -xJf (распаковка + декомпрессия xz) |

| | | | |
|---|---|---|---|
| | | | Ответ: A2, B3, C4, D-1<br><br>**Задание 3.2**<br>Сопоставьте ключ команды tar с его описанием.<br><br>Список А (Ключ команды tar):<br>A) -c<br>B) -x<br>C) -t<br>D) -v<br><br>Список Б (Описание):<br>1. Распаковать (извлечь) содержимое архива.<br>2. Создать новый архив.<br>3. Вывести подробный (verbose) список обрабатываемых файлов.<br>4. Показать содержимое (оглавление) архива без его распаковки.<br><br>Ответ: A-2, B-1, C-4, D-3<br><br>**Задание 3.3**<br>Какая команда корректно создаст сжатый архив project.tar.bz2 из каталога project/, сохранив все атрибуты файлов и выведя в консоль список добавляемых файлов?<br><br>1. tar -cvjf project.tar.bz2 project/<br>2. tar -czvf project.tar.bz2 project/<br>3. tar -cJf project.tar.bz2 project/<br>4. tar -xvf project.tar.bz2 project/<br><br>Ответ: 1<br><br>Пояснение: 1 - Правильно. Ключи: -c (create), -v (verbose), -j (использовать bzip2), -f |

| | | | |
|---|---|---|---|
| | | | (указать имя файла). Порядок ключей после - не важен, но f должен идти перед именем файла.<br><br>**Задание 3.4**<br>Пользователь сжал текстовый файл data.txt с помощью gzip, получив data.txt.gz. Теперь ему нужно просмотреть первые 10 строк содержимого этого сжатого файла без его полной распаковки на диск. Какую команду следует использовать?<br><br>1. gunzip -c data.txt.gz \| tail -n 10<br>2. zcat data.txt.gz \| head -n 10<br>3. tar -tzf data.txt.gz \| head -n 10<br><br>Ответ: 2<br><br>Пояснение: 2 - Правильно. zcat (а также gzcat) — это утилита, входящая в пакет gzip, которая распаковывает файл и выводит его содержимое на стандартный вывод (stdout), после чего конвейером (\|) оно передается команде head -n 10. |
| Тема 4 Обработка текстовых файлов | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 4.1**<br>Какая из предложенных команд sed удалит все строки в файле config.txt, которые начинаются с символа # (комментарии) или являются пустыми (содержат только пробелы/табуляции)?<br><br>1. sed '/^#/d' config.txt<br>2. sed '/^#/d; /^$/d' config.txt<br>3. sed '/^#/d' config.txt \| sed '/^$/d'<br>4. sed '/^[[:space:]]*$/d; /^#/d' config.txt<br><br>Ответ: 4<br><br>Пояснение: 4 - Правильно. Выражение ^[[:space:]]*$ соответствует строкам, которые от начала (^) до конца ($) содержат ноль или более (*) пробельных символов ([[:space:]]), т.е. пустым или состоящим только из пробелов/табуляций. Вторая часть /^#/d удаляет строки, начинающиеся с #. Порядок может быть любым. |

| | | | |
|---|---|---|---|
| | | | **Задание 4.2**<br>Вам нужно в реальном времени отслеживать добавление новых записей в растущий лог-файл /var/log/app.log. Какая команда является наиболее подходящей?<br><br>1. tail /var/log/app.log<br>2. less /var/log/app.log<br>3. tail -f /var/log/app.log<br>4. head -f /var/log/app.log<br><br>Ответ: 3<br><br>Пояснение: 3 - Правильно. Ключ -f (follow) команды tail заставляет ее не завершаться после вывода последних строк, а продолжать работу и выводить новые данные по мере их добавления в файл. Это стандартный способ мониторинга логов.<br><br>**Задание 4.3**<br>Какой флаг (ключ) нужно добавить к команде sed для выполнения редактирования на месте (in-place), т.е. для сохранения изменений прямо в исходном файле, а не вывода их в консоль? (Ответ: короткий ключ из одной буквы или длинный из двух слов)<br><br>Ответ: -i (или длинный вариант --in-place)<br><br>Пояснение: Флаг -i (in-place) указывает sed записать результат обработки обратно в исходный файл. Для безопасности часто используют -i.bak, который создает резервную копию исходного файла с расширением .bak. Без этого флага sed выводит результат на стандартный вывод (обычно в терминал), оставляя исходный файл неизменным.<br><br>**Задание 4.4**<br>В языке awk какая специальная встроенная переменная содержит общее количество полей (столбцов) в текущей обрабатываемой строке? (Ответ: название переменной из двух символов) |

| | | | |
|---|---|---|---|
| | | | Ответ: NF<br><br>Пояснение: NF (Number of Fields) — одна из ключевых встроенных переменных awk. Она автоматически устанавливается для каждой обрабатываемой строки и содержит количество полей, на которые строка была разбита с использованием разделителя полей (FS). Например, awk '{print NF}' file.txt выведет количество столбцов в каждой строке. $NF ссылается на значение последнего поля в строке. |
| Тема 5 Текстовые редакторы. | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 5.1**<br>Какие из следующих утверждений о текстовом редакторе MCEDIT (в составе Midnight Commander) являются верными?<br><br>1. Это консольный (текстовый) редактор, работающий в терминале без графического интерфейса (TUI).<br>2. Он тесно интегрирован с файловым менеджером Midnight Commander (F1-F10, меню), что упрощает навигацию и пакетное редактирование.<br>3. Для работы ему обязательно требуется установленная графическая среда (X Window System или Wayland).<br>4. Он поддерживает подсветку синтаксиса для многих языков программирования и разметки.<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. MCEDIT — это классический TUI (Text-based User Interface) редактор, наследник Norton Editor, работающий в любой текстовой консоли или эмуляторе терминала.<br>2 - Правильно. Ключевая особенность. Функциональные клавиши (F3-F4 для копирования/перемещения и т.д.) и меню верхнего уровня делают его частью экосистемы MC, а не standalone-редактором.<br>4 - Правильно. Несмотря на простоту, MCEDIT имеет базовую, но полезную подсветку синтаксиса для C, Python, HTML, конфигов и др.<br><br>**Задание 5.2** |

| | | | |
|---|---|---|---|
| | | | Какие из следующих характеристик применимы к редактору GVIM (графическая версия Vim)?<br><br>1. Он сохраняет все основные режимы работы классического Vim (нормальный, вставки, визуальный, командной строки).<br>2. Он предоставляет стандартные элементы графического интерфейса: меню, панели инструментов, диалоги открытия/сохранения файлов.<br>3. Его интерфейс и поведение полностью идентичны редактору Kate, так как оба являются графическими.<br>4. Он может быть использован для удаленного редактирования файлов по SSH в чисто текстовом режиме с помощью команды vim (без префикса g).<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. GVIM — это Vim с графической оболочкой. Все фундаментальные концепции Vim (модальность, команды :, hjkl навигация) остаются.<br>2 - Правильно. В отличие от консольного Vim, GVIM добавляет привычные GUI-элементы, что облегчает начальное освоение для пользователей, привыкших к мыши и меню.<br>4 - Правильно. Это мощное преимущество Vim/GVIM. Одна и та же программа (vim) может работать и в графическом режиме (gvim), и в консольном. При SSH-сессии автоматически запускается консольная версия, что делает навыки работы с Vim универсальными.<br><br>**Задание 5.3**<br>Как называется характерный для редакторов семейства Vim режим работы, в котором нажатия клавиш интерпретируются не как ввод текста, а как команды для перемещения курсора, удаления, копирования и других действий? (Ответ: одно слово — название режима)<br><br>Ответ: Нормальный (Normal mode) или Командный (Command mode). |

| | | | |
|---|---|---|---|
| | | | Пояснение: Это фундаментальное понятие в философии Vim. «Нормальный режим» — это режим по умолчанию, в котором пользователь находится большую часть времени. В этом режиме клавиши h,j,k,l перемещают курсор, x удаляет символ, dd удаляет строку, i переходит в режим вставки и т.д. Это отличает Vim от модельных (modeless) редакторов вроде Kate или MCEDIT.<br><br>**Задание 5.4**<br>Какую функциональную клавишу в редакторе MCEDIT (и вообще в Midnight Commander) нужно нажать, чтобы вызвать меню с операциями над файлом (копирование, переименование, права доступа и т.д.)? (Ответ: обозначение клавиши в формате «F[цифра]»)<br><br>Ответ: F9<br><br>Пояснение: В Midnight Commander (и, соответственно, в MCEDIT, когда он запущен из MC) клавиша F9 активирует главное меню в верхней части экрана. Это основной способ доступа ко всем функциям файлового менеджера и редактора, кроме самых базовых. Знание основных функциональных клавиш (F3 — просмотр, F4 — редактировать, F5 — копировать, F6 — переименовать/переместить, F10 — выход) — ключевой навык для эффективной работы в MC. |
| Тема 6 Основы bash. | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия | **Задание 6.1**<br>Какие из следующих утверждений о стандартных потоках ввода-вывода в bash являются верными?<br><br>1. Стандартный поток вывода (stdout, дескриптор 1) по умолчано выводится в терминал.<br>2. Стандартный поток ошибок (stderr, дескриптор 2) не может быть перенаправлен отдельно от stdout.<br>3. Перенаправление 2>&1 означает, что stderr будет отправлен туда же, куда в данный момент направлен stdout.<br>4. Стандартный поток ввода (stdin, дескриптор 0) по умолчано читается с клавиатуры терминала. |

| | | | |
|---|---|---|---|
| | | систем на базовом уровне | Ответ: 1, 3, 4<br><br>Пояснение:<br>1 - Правильно. Это поведение по умолчанию для интерактивной оболочки.<br>3 - Правильно. Это ключевой синтаксис для объединения потоков. 2> перенаправляет stderr, &1 указывает на «то же место, где сейчас находится stdout».<br>4 - Правильно. В интерактивном режиме команда, ожидающая ввода, будет ждать его от пользователя через терминал.<br><br>**Задание 6.2**<br>Какие из следующих утверждений о специальных символах и операторах управления выполнением в bash верны?<br><br>1. Символ & в конце команды (command &) запускает ее в фоновом режиме, немедленно возвращая управление оболочке.<br>2. Оператор && (логическое И) выполняет команду справа только в том случае, если команда слева завершилась с нулевым кодом возврата (успешно).<br>3. Оператор \|\| (логическое ИЛИ) выполняет команду справа только в том случае, если команда слева завершилась с ненулевым кодом возврата (с ошибкой).<br>4. Символ ; используется для группировки команд в подобие функции.<br><br>Ответ: 1, 2, 3<br><br>Пояснение:<br>1 - Правильно. Это определение фонового запуска. Оболочка не ждет завершения такой команды.<br>2 - Правильно. Это основа цепочек выполнения. Например, make && make install выполнит установку только если сборка прошла успешно.<br>3 - Правильно. Обратная логика. Например, command \|\| echo "Ошибка" выведет сообщение только при неудаче команды. |

| | | | |
|---|---|---|---|
| | | | **Задание 6.1**<br>Установите правильную последовательность обработки следующей командной строки оболочкой bash: ls -l *.txt 2>/dev/null \| grep "report" > output.txt &<br><br>1. Подстановка шаблона (globbing): *.txt заменяется на список файлов.<br>2. Запуск команды ls -l со списком файлов, перенаправление ее stderr в /dev/null.<br>3. Создание канала (pipe) и запуск grep с аргументом "report", чтение его вывода из канала.<br>4. Перенаправление stdout команды grep в файл output.txt.<br>5. Помещение всего конвейера в фоновый режим.<br><br>Ответ: 1,2,3,4,5<br><br>Пояснение: Оболочка работает слева направо, выполняя подстановки, затем настройку перенаправлений и операторов, и только потом запускает команды. Сначала раскрывается шаблон (1). Затем настраивается перенаправление ошибок для ls (2). Далее создается конвейер между ls и grep (3). После этого настраивается перенаправление вывода grep (4). И, наконец, интерпретируется оператор &, который отправляет весь созданный конвейер (job) в фон (5).<br><br>**Задание 6.2**<br>Установите последовательность приоритетов (от самого высокого к низкому) для следующих действий оболочки bash при разборе командной строки.<br><br>1. Выполнение команд (запуск процессов).<br>2. Подстановка переменных ($VAR).<br>3. Подстановка команд (command substitution, $(...) или `...`).<br>4. Подстановка шаблонов (globbing, *, ?, []).<br><br>Ответ: 3,2,4,1<br><br>Пояснение: Сначала выполняется подстановка команд — их результат вычисляется и вставляется в строку. Затем раскрываются переменные. После этого происходит подстановка шаблонов на основе уже готовых путей. И только когда командная строка |

| | | | |
|---|---|---|---|
| | | | полностью обработана и сформирована, выполняется сама команда. |
| Тема 7 Оболочка bash. | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 7.1**<br>Установите правильную последовательность приоритетов (от самого высокого к низкому) для различных типов раскрытия (expansion), которые bash выполняет при обработке командной строки.<br><br>1. Раскрытие команд (Command Substitution) — $(...) или `...`.<br>2. Раскрытие переменных (Parameter Expansion) — $VAR или ${VAR}.<br>3. Арифметическое раскрытие (Arithmetic Expansion) — $((...)).<br>4. Разделение на слова (Word Splitting) и Подстановка путей (Pathname Expansion/Globbing) — *, ?.<br><br>Ответ: 1,3,2,4<br><br>Пояснение: Сначала выполняются самые «внутренние» подстановки: результат команды или арифметического выражения может содержать переменные, которые нужно раскрыть позже. Затем раскрываются переменные. И только после того, как все подстановки выполнены и получена итоговая строка, происходит разделение на слова (по разделителям из $IFS) и, наконец, подстановка путей (globbing).<br><br>**Задание 7.2**<br>Установите последовательность обработки переменной FILE в следующей команде: cp ${FILE:-default.txt} ${FILE%.txt}.backup 2>/dev/null<br><br>1. Проверка, определена ли переменная FILE и не является ли ее значение пустым. Если да, используется это значение. Если нет, используется строка "default.txt".<br>2. Удаление из значения переменной FILE (или строки "default.txt", если подстановка сработала) самого короткого совпадения суффикса .txt с конца строки, и добавление суффикса .backup.<br>3. Раскрытие параметра ${FILE:-default.txt} для первого аргумента cp.<br>4. Раскрытие параметра ${FILE%.txt}.backup для второго аргумента cp.<br><br>Ответ: 1,3,2,4 |

| | | | |
|---|---|---|---|
| | | | Пояснение: Оболочка обрабатывает каждый аргумент слева направо. Для первого аргумента (${FILE:-default.txt}) она выполняет подстановку с значением по умолчанию (шаг 1, результат шаг 3). Для второго аргумента (${FILE%.txt}.backup) она берет оригинальное значение $FILE (не из первой подстановки!) и применяет к нему операцию удаления суффикса (шаг 2, результат шаг 4). Важно: Вторая подстановка не зависит от результата первой, обе используют исходную переменную $FILE.<br><br>**Задание 7.3**<br>Сопоставьте конструкцию раскрытия параметра (parameter expansion) в bash с ее описанием.<br><br>Список А (Конструкция):<br>A) ${VAR:-default}<br>B) ${VAR:=default}<br>C) ${VAR:?error message}<br>D) ${VAR:+alternate}<br><br>Список Б (Описание):<br>1. Если VAR не задана или пуста, используется default, и это значение присваивается переменной VAR.<br>2. Если VAR не задана или пуста, выводится error message и выполнение скрипта прерывается.<br>3. Если VAR не задана или пуста, используется default, но сама VAR не изменяется.<br>4. Если VAR задана и не пуста, используется alternate, иначе используется пустая строка.<br><br>Ответ: A3, B1, C2, D4<br><br>**Задание 7.4**<br>Сопоставьте специальную переменную/параметр bash с его значением.<br><br>Список А (Переменная/параметр): |

| | | | |
|---|---|---|---|
| | | | A) $#<br>B) $@<br>C) $?<br>D) $$<br><br>Список Б (Значение):<br>1. Код возврата (exit status) последней выполненной команды.<br>2. Количество позиционных параметров (аргументов), переданных скрипту или функции.<br>3. PID (идентификатор процесса) текущей оболочки.<br>4. Список всех позиционных параметров, разделенных пробелами (или как отдельные слова в двойных кавычках).<br><br>Ответ: A2, B4, C1, D3 |
| Тема 8 Shell --- как язык программирования. | ОПК-5<br>С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 8.1**<br>Сопоставьте оператор цикла в bash с его описанием.<br><br>Список А (Оператор):<br>A) for var in list; do ... done<br>B) while condition; do ... done<br>C) until condition; do ... done<br>D) select var in list; do ... done<br><br>Список Б (Описание):<br>1. Цикл, который выполняется пока условие (код возврата команды) истинно (равно 0).<br>2. Цикл, который выполняется до тех пор, пока условие не станет истинным. Выполняется, пока условие ложно.<br>3. Цикл перебора элементов из заранее заданного списка.<br>4. Создает интерактивное меню на основе списка, выполняет тело цикла для выбранного пользователем пункта.<br><br>Ответ: A-3, B-1, C-2, D-4<br><br>**Задание 8.2** |

| | | | |
|---|---|---|---|
| | | | Сопоставьте конструкцию проверки условия для команды test/[ с ее смыслом.<br><br>Список А (Условие):<br>A) [ -z "$string" ]<br>B) [ -n "$string" ]<br>C) [ "$a" -eq "$b" ]<br>D) [ -r "$file" ]<br><br>Список Б (Смысл):<br>1. Проверяет, что строка $string не является пустой (ее длина больше нуля).<br>2. Проверяет, что строки $a и $b равны как целые числа.<br>3. Проверяет, что файл $file существует и доступен для чтения текущему пользователю.<br>4. Проверяет, что строка $string является пустой (ее длина равна нулю).<br><br>Ответ: A4, B1, C2, D3<br><br><br>**Задание 8.3**<br>Какая из следующих конструкций цикла for корректно переберет все файлы с расширением .txt в текущем каталоге и выведет их имена?<br><br>1. for file in "*.txt"; do echo $file; done<br>2. for file in $(ls *.txt); do echo "$file"; done<br>3. for file in *.txt; do echo "$file"; done<br>4. for file in ls *.txt; do echo $file; done<br><br>Ответ: 3<br><br>Пояснение:<br>3 - Правильно. Идиоматичный способ. Подстановка пути (globbing) *.txt выполняется оболочкой до запуска цикла, генерируя список файлов.<br><br>**Задание 8.4** |

| | | | |
|---|---|---|---|
| | | | Функция is_even() должна проверить, является ли переданный ей числовой аргумент четным, и вернуть 0 (истина), если да, и 1 (ложь) — если нет. Какая реализация является правильной и безопасной?<br><br>1.bash<br>is_even() {<br>  if [ $(($1 % 2)) -eq 0 ]; then<br>  return 0<br>  else<br>  return 1<br>  fi<br>}<br><br>2. bash<br>is_even() {<br>  if [ $1 % 2 == 0 ]; then<br>  return true<br>  else<br>  return false<br>  fi<br>}<br><br>3. bash<br>is_even() {<br>  if (( $1 % 2 == 0 )); then<br>  return 0<br>  else<br>  return 1<br>  fi<br>}<br><br>4. bash<br>is_even() { |

| | | | |
|---|---|---|---|
| | | | result=$(( $1 % 2 ))<br>return $result<br>}<br><br>Ответ: 3<br><br>Пояснение:<br>3 - Правильно (предпочтительно). Использует современную арифметику в двойных круглых скобках (( ... )), которая возвращает статус 0, если выражение истинно (не нуль). Чисто, понятно, без лишних команд test/[. |
| Тема 9 Элементы администрирования | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 9.1**<br>Какой файл является основным статическим конфигурационным файлом для автоматического монтирования файловых систем при загрузке в Linux?<br><br>1. /etc/mtab<br>2. /proc/mounts<br>3. /etc/fstab<br>4. /etc/filesystems<br><br>Ответ: 3<br><br>Пояснение: 3 - Правильно. /etc/fstab (FileSystem Table) — статический файл, который читается системными скриптами (например, systemd-fstab-generator) при загрузке для автоматического монтирования указанных разделов.<br><br>**Задание 9.2**<br>В процессе установки Ubuntu вы создаете разделы вручную. Какой тип раздела и файловую систему обязательно нужно создать для корректной загрузки в режиме UEFI?<br><br>1. Раздел с файловой системой ext4, помеченный как bootable (активный), смонтированный в /.<br>2. Раздел с файловой системой FAT32 (или vfat), смонтированный в /boot/efi, с флагами |

| | | | |
|---|---|---|---|
| | | | boot и esp.<br>3. Раздел с файловой системой ext4, смонтированный в /boot.<br>4. Раздел типа swap.<br><br>Ответ: 2<br><br>Пояснение: Для загрузки в режиме UEFI обязателен EFI System Partition (ESP). Он должен быть отформатирован в FAT32 (или vfat), так как это стандарт, читаемый прошивкой UEFI. Обычно он монтируется в /boot/efi. В таблице разделов GPT для него устанавливается тип EF00 (в gdisk) или флаги boot и esp. Раздел /boot (вариант 3) может быть отдельным, но не обязателен. Флаг bootable (вариант 1) относится к загрузке в режиме BIOS/MBR. Swap (вариант 4) не обязателен для загрузки.<br><br>**Задание 9.3**<br>Какой главный системный каталог в иерархии файловой системы Linux (FHS) традиционно содержит статичные файлы, специфичные для данного компьютера, такие как конфигурация загрузчика, таблица разделов (fstab) и информация о сети (hostname)? (Ответ: полный путь)<br><br>Ответ: /etc<br><br>Пояснение: Согласно Filesystem Hierarchy Standard (FHS), каталог /etc предназначен для хранения конфигурационных файлов всей системы, специфичных для данного хоста. Это статические файлы (не скрипты). Именно здесь находятся /etc/fstab, /etc/hostname, /etc/hosts, конфиги сетевых интерфейсов, passwd, shadow и большинства системных служб.<br><br>**Задание 9.4**<br>Какой ключ нужно добавить к команде apt install, чтобы автоматически подтвердить все запросы (такие как «Do you want to continue? [Y/n]») и установить пакет без интерактивного вмешательства? Этот ключ часто используется в скриптах автоматической установки. (Ответ: короткий ключ из двух букв или длинный из двух слов) |

| | | | |
|---|---|---|---|
| | | | Ответ: -y (или --yes)<br><br>Пояснение: Ключ -y автоматически отвечает «yes» на все запросы, которые в интерактивном режиме требуют подтверждения пользователя. Это критически важно для неинтерактивного выполнения в скриптах, например: apt update && apt install -y nginx. |
| Тема 1 Конфигурирование сетевых интерфейсов. | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 10.1**<br>Какие из следующих утверждений об IP-адресах и подсетях верны?<br><br>1. IP-адрес 192.168.1.0 с маской 255.255.255.0 (/24) является адресом сети и не может быть назначен интерфейсу хоста.<br>2. Широковещательный (broadcast) адрес для сети 10.0.0.0/8 — это 10.255.255.255.<br>3. Адрес 127.0.0.1 (loopback) всегда принадлежит устройству с физическим сетевым интерфейсом.<br>4. IPv6-адреса типа fe80::/10 (link-local) автоматически назначаются интерфейсу и используются для связи только в пределах одного канального сегмента (например, одной Ethernet-сети).<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. Адрес сети (первый адрес в подсети) зарезервирован для идентификации самой сети и не используется для узлов.<br>2 - Правильно. Для подсети /8 адрес сети 10.0.0.0, широковещательный адрес — 10.255.255.255 (все биты хостовой части равны 1).<br>4 - Правильно. Link-local адреса (fe80::/10) генерируются автоматически (например, из MAC-адреса) и не маршрутизируются за пределы сегмента L2.<br><br>**Задание 10.2**<br>Какие из следующих утверждений о конфигурировании сетевых интерфейсов и маршрутизации в Linux верны?<br><br>1. Команда ip addr add 192.168.1.10/24 dev eth0 добавит указанный IP-адрес и маску к |

| | | | |
|---|---|---|---|
| | | | интерфейсу eth0.<br>2. Для настройки статического маршрута по умолчанию (шлюза по умолчанию) используется команда вида ip route add default via 192.168.1.1.<br>3. Файл /etc/resolv.conf содержит конфигурацию сетевых интерфейсов (IP-адреса, маски).<br>4. После внесения изменений в конфигурационные файлы сетевых интерфейсов в /etc/network/interfaces (Debian/Ubuntu) для применения настроек необходимо перезагрузить компьютер.<br><br>Ответ: 1, 2<br><br>Пояснение:<br>1 - Правильно. Это современный синтаксис утилиты ip из пакета iproute2 для настройки адреса.<br>2 - Правильно. Команда для добавления маршрута по умолчанию через шлюз 192.168.1.1. Аналог старой команды route add default gw 192.168.1.1.<br><br>**Задание 10.3**<br>Какая утилита командной строки в современных дистрибутивах Linux (из пакета iproute2) является основным инструментом для отображения и изменения параметров сетевых интерфейсов, адресов, AR-кэша и маршрутов, заменяя собой устаревший набор ifconfig, route, arp? (Ответ: название команды из двух букв)<br><br>Ответ: ip<br><br>Пояснение: Утилита ip — это универсальный инструмент управления сетью в Linux. Её подкоманды: ip link (интерфейсы), ip addr (адреса), ip route (маршруты), ip neigh (ARP/NDP). Она предоставляет более полный доступ к функциям ядра, чем устаревшие (net-tools) аналоги, и является стандартом де-факто.<br><br>**Задание 10.4**<br>Какой тип DNS-записи (Resource Record) используется для прямого преобразования доменного имени (например, server.example.com) в IPv4-адрес? (Ответ: обозначени |

| | | | |
|---|---|---|---|
| | | | е типа записи одной заглавной буквой)<br><br>Ответ: A (Address record)<br><br>Пояснение: Запись типа A (Address) связывает имя хоста с IPv4-адресом. Для IPv6 используется запись AAAA (или «quad-A»). Это фундаментальные записи для разрешения имен. Другие важные типы: MX (для почтовых серверов), CNAME (алиас), NS (указывает на DNS-серверы зоны). |
| Тема 11 Сетевой экран. | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 11.1**<br>Какие из следующих утверждений о базовых концепциях сетевого экрана (брандмауэра) на основе фильтрации пакетов верны?<br><br>1. Фильтрация может осуществляться на основе различных атрибутов пакета: IP-адресов источника и назначения, номеров портов, интерфейса, состояния соединения.<br>2. Сетевой экран, работающий на уровне приложений (Application-level gateway), не способен анализировать содержимое пакетов (payload).<br>3. Состояние «ESTABLISHED» для TCP-соединения означает, что пакет является частью уже установленного и разрешенного ранее соединения, и такие пакеты часто разрешаются правилами stateful-фаервола.<br>4. Основная задача трансляции сетевых адресов (NAT) — это повышение скорости обработки пакетов.<br><br>Ответ: 1, 3<br><br>Пояснение:<br>1 - Правильно. Это основа фильтрации пакетов (packet filtering). Правила строятся на заголовках сетевого и транспортного уровней.<br>3 - Правильно. Stateful-инспекция (отслеживание состояния) — ключевая особенность современных фаерволов. Разрешая ответные пакеты установленных соединений (-m state --state ESTABLISHED,RELATED), мы значительно упрощаем правила и повышаем безопасность.<br><br>**Задание 11.2** |

| | | | |
|---|---|---|---|
| | | | Какие из следующих утверждений о таблицах и цепочках iptables являются верными?<br><br>1. Таблица filter используется для фильтрации пакетов и содержит стандартные цепочки INPUT, OUTPUT, FORWARD.<br>2. Таблица nat используется для трансляции адресов (NAT) и содержит цепочки PREROUTING (для DNAT) и POSTROUTING (для SNAT/Masquerade).<br>3. Цепочка INPUT обрабатывает пакеты, сгенерированные самой системой и отправляемые наружу.<br>4. Команда iptables -L по умолчанию показывает правила в таблице filter.<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. Это основное назначение таблицы filter.<br>2 - Правильно. Это основное назначение таблицы nat. PREROUTING изменяет адрес назначения до маршрутизации (DNAT), POSTROUTING — адрес источника после маршрутизации (SNAT).<br>4 - Правильно. iptables -L — это короткая запись для iptables -t filter -L, выводит правила фильтрации.<br><br>**Задание 11.3**<br>Установите правильную последовательность прохождения сетевого пакета через цепочки iptables на шлюзе (маршрутизаторе), который пересылает пакет из внешней сети во внутреннюю (Forwarding).<br><br>1. Цепочка FORWARD (таблица filter).<br>2. Цепочка POSTROUTING (таблица nat).<br>3. Цепочка PREROUTING (таблица nat).<br>4. Принятие решения о маршрутизации (Routing Decision).<br><br>Ответ: 3,4,1,2 |

| | | | |
|---|---|---|---|
| | | | Пояснение: Входящий пакет сначала попадает в PREROUTING (таблица nat, можно сделать DNAT) (3). Затем ядро определяет, куда направить пакет, и понимает, что он предназначен не локально, а для пересылки (4). Далее пакет проходит через FORWARD (таблица filter, здесь происходит основная фильтрация трафика между интерфейсами) (1). Перед отправкой в исходящий интерфейс пакет проходит POSTROUTING (таблица nat, можно сделать SNAT/Masquerade) (2).<br><br>**Задание 11.4**<br>Установите последовательность действий для настройки типичного правильного и безопасного базового набора правил iptables (политика DROP по умолчанию, разрешение loopback, установленных соединений и SSH).<br><br>1. Установить политику по умолчанию DROP для цепочек INPUT и FORWARD.<br>2. Разрешить весь трафик на интерфейсе loopback (-i lo -j ACCEPT).<br>3. Разрешить входящие пакеты, относящиеся к установленным и связанным соединениям (-m state --state ESTABLISHED,RELATED -j ACCEPT).<br>4. Разрешить входящие соединения на порт TCP/22 (SSH) (-p tcp --dport 22 -j ACCEPT).<br><br>Ответ: 2,3,4,1<br><br>Пояснение: При настройке iptables с консоли важно сначала добавить разрешающие правила, и только потом менять политику на DROP, чтобы не потерять управление. Типичный безопасный порядок правил в цепочке INPUT: разрешить loopback (2), затем ответы на установленные соединения (3), затем новые разрешенные службы (SSH, 4). В конце или в начале устанавливается политика DROP (1). Если установить DROP до добавления разрешений, SSH-сессия оборвется. |
| Тема 12 Сетевой экран (продолжение). | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, | **Задание 12.1**<br>Какая команда iptables добавит правило, разрешающее входящие SSH-соединения (TCP порт 22) только с IP-адреса 192.168.1.50 в цепочку INPUT таблицы filter?<br><br>1. iptables -A INPUT -s 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br>2. iptables -A INPUT -d 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br>3. iptables -t nat -A INPUT -s 192.168.1.50 -p tcp --dport 22 -j ACCEPT |

| | | | |
|---|---|---|---|
| | информационных и автоматизированных систем | современные стандарты информационного взаимодействия систем на базовом уровне | 4. iptables -A FORWARD -s 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br><br>Ответ: 1<br><br>Пояснение:<br>1 - Правильно. -A INPUT добавляет в конец цепочки INPUT таблицы filter (по умолчанию). -s 192.168.1.50 указывает адрес источника. -p tcp --dport 22 — протокол и порт назначения. -j ACCEPT — действие.<br><br>**Задание 12.2**<br>На маршрутизаторе с внешним интерфейсом eth0 (IP динамический) нужно настроить MASQUERADE для выхода внутренней сети 192.168.0.0/24 в интернет. В какую цепочку и таблицу нужно добавить правило?<br><br>1. iptables -t filter -A FORWARD -s 192.168.0.0/24 -j MASQUERADE<br>2. iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -o eth0 -j MASQUERADE<br>3. iptables -t nat -A PREROUTING -s 192.168.0.0/24 -j MASQUERADE<br>4. iptables -t nat -A INPUT -s 192.168.0.0/24 -j MASQUERADE<br><br>Ответ: 2<br><br>Пояснение:<br>2 - Правильно. Трансляция адреса источника (SNAT/MASQUERADE) выполняется после маршрутизации, в цепочке POSTROUTING таблицы nat. Ключ -o eth0 указывает исходящий интерфейс (опционально, но рекомендуется). MASQUERADE используется для динамических IP.<br><br>**Задание 12.3**<br>Какая команда iptables используется для очистки (удаления всех правил) во всех цепочках указанной таблицы? Например, для таблицы filter. (Ответ: синтаксис команды с ключом -F) |

| | | | |
|---|---|---|---|
| | | | Ответ: iptables -t filter -F (или iptables -F, так как filter таблица по умолчанию)<br><br>Пояснение: Ключ -F (flush) очищает все правила в выбранной цепочке. Если цепочка не указана, очищаются все цепочки указанной таблицы. Важно: Очистка не меняет политику по умолчанию (-P). После -F политики остаются такими, какими были (часто ACCEPT), что может открыть систему. Безопасная последовательность: iptables -F, затем iptables -X (удалить пользовательские цепочки), затем iptables -Z (обнулить счетчики), затем iptables -P INPUT DROP (установить безопасные политики).<br><br>**Задание 12.4**<br>Как называется специальный модуль iptables (указываемый ключом -m), который позволяет проверять состояние соединения и использовать такие критерии, как NEW, ESTABLISHED, RELATED? (Ответ: название модуля)<br><br>Ответ: state (в современных ядрах также conntrack, который является более новым именем того же модуля)<br><br>Пояснение: Модуль state (или conntrack) необходим для stateful-инспекции. Он позволяет писать правила вида:<br>iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT<br>Это правило разрешает все пакеты, которые являются частью уже установленных или связанных (например, ICMP-ошибки или FTP data-канал) соединений, что является краеугольным камнем безопасной и удобной настройки фаервола. |
| Тема 13 Скрипты с Web-интерфейсом. | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизиров | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационно | **Задание 13.1**<br>Установите правильную последовательность этапов обработки запроса веб-сервером Apache к bash-CGI скрипту, начиная с запроса пользователя.<br><br>1. Веб-сервер создает новое окружение для дочернего процесса, устанавливает CGI-переменные (REQUEST_METHOD, QUERY_STRING и др.).<br>2. Пользователь через браузер отправляет HTTP-запрос на адрес http://server/cgi-bin/status.cgi.<br>3. Веб-сервер отправляет полученный от скрипта HTTP-ответ обратно браузеру пользователя. |

| | | | |
|---|---|---|---|
| | анных систем | го взаимодействия систем на базовом уровне | 4. Bash-скрипт status.cgi выполняется, генерирует HTTP-заголовки и тело ответа, выводя их в стандартный поток вывода (STDOUT).<br>5. Apache, получив запрос, определяет по конфигурации, что status.cgi является CGI-скриптом.<br><br>Ответ: 2, 5, 1, 4, 3.<br><br>Пояснение: Последовательность демонстрирует принцип посредничества веб-сервера: он принимает запрос, организует выполнение внешней программы и возвращает её результат.<br><br>**Задание 13.2**<br>Установите правильную последовательность строк в теле bash-CGI скрипта для безопасной обработки поискового запроса из формы и вывода результатов.<br><br>1. echo "<h2>Результаты поиска для '$filtered_query':</h2>"<br>2. echo "Content-Type: text/html; charset=utf-8"<br>3. log_entry=$(grep -i "$filtered_query" /var/log/syslog \| head -20)<br>4. echo -e "\n\n<html><body>"<br>5. filtered_query=$(echo "$query" \| sed 's/[^a-zA-Z0-9 ]//g')<br>6. query="${QUERY_STRING#*=}" # Извлекаем значение после 'q='<br>7. echo "<pre>$log_entry</pre></body></html>"<br><br>Ответ: 2, 4, 6, 5, 1, 3, 7.<br><br>Пояснение: Никогда не доверяй пользовательскому вводу. Порядок гарантирует, что любая обработка (вывод или выполнение) происходит только с очищенными данными, разделяя этапы получения, валидации и использования.<br><br>**Задание 13.3**<br>Сопоставьте метод HTTP-запроса с особенностью его обработки в bash-CGI скрипте.<br><br>Список А (Метод запроса): |

| | | | |
|---|---|---|---|
| | | | 1. GET<br>2. POST<br><br>Список Б (Особенность обработки):<br>A. Параметры запроса передаются через переменную окружения QUERY_STRING.<br>B. Длина тела запроса передается через переменную окружения CONTENT_LENGTH.<br>C. Для чтения параметров запроса необходимо читать данные из стандартного потока ввода (STDIN).<br>D. Параметры запроса видны в адресной строке браузера.<br><br>Ответ: 1A, 1D, 2B, 2C.<br><br>**Задание 13.4**<br>Сопоставьте элемент технологии CGI с его описанием.<br><br>Список А (Элемент):<br>1. Шебанг (#!/bin/bash)<br>2. Заголовок Content-Type: text/html<br>3. Переменная окружения SCRIPT_NAME<br>4. Пустая строка после заголовков<br><br>Список Б (Описание):<br>A. Служит разделителем между HTTP-заголовками и телом ответа в выводе CGI-скрипта.<br>B. Указывает веб-браузеру, как интерпретировать полученные данные.<br>C. Содержит путь к выполняемому скрипту относительно корня CGI.<br>D. Директива для ядра ОС, указывающая, какой интерпретатор должен выполнять код в файле.<br><br>Ответ: 1D, 2B, 3C, 4A. |
| Тема 14 Web-сервер Apache. | ОПК-5 Способен инсталлировать программное | ИОПК-5.1. Знать основы системного администрирова | **Задание 14.1**<br>Сопоставьте директиву Apache с ее основной функцией в контексте настройки виртуальных хостов. |

| | | | |
|---|---|---|---|
| | и аппаратное обеспечение для информационных и автоматизированных систем | ния, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | Список А (Директива):<br>1. Listen<br>2. ServerAdmin<br>3. DirectoryIndex<br>4. ErrorDocument<br><br>Список Б (Функция):<br>A. Задает порядок поиска индексных файлов (например, index.html, index.php) при запросе к директории.<br>B. Определяет, на каких IP-адресах и портах Apache будет принимать входящие соединения.<br>C. Позволяет настроить кастомную страницу или действие для определенных HTTP-кодов ошибок (404, 500).<br>D. Указывает email-адрес администратора, который может отображаться на страницах ошибок.<br><br>Ответ: 1B, 2D, 3A, 4C.<br><br>**Задание 14.2**<br>Сопоставьте проблему с наиболее вероятной конфигурационной причиной в Apache.<br><br>Список А (Проблема):<br>1. При запросе site.com открывается содержимое другого сайта, размещенного на этом же сервере.<br>2. При запросе PHP-файла браузер предлагает его скачать, а не выполняет.<br>3. Apache выдает ошибку AH00558: apache2: Could not reliably determine the server's fully qualified domain name.<br>4. Страница загружается, но все стили (CSS) и скрипты (JS) не применяются (ошибка 403).<br><br>Список Б (Причина):<br>A. Не задана глобальная директива ServerName или не указан ServerName в основном виртуальном хосте.<br>B. В конфигурации виртуального хоста для site.com отсутствует или неверно указана |

| | | | |
|---|---|---|---|
| | | | директива ServerName, и запрос попадает в хост по умолчанию (первый в списке).<br>C. В конфигурации виртуального хоста отсутствует директива AddType или SetHandler для обработки PHP, либо не загружен модуль mod_php/mod_proxy_fcgi.<br>D. Для директорий со статическими файлами (CSS, JS) в конфигурации не хватает секции <Directory> с директивами Require all granted и Options Indexes или FollowSymLinks.<br><br>Ответ: 1B, 2C, 3A, 4D.<br><br>**Задание 14.3**<br>Какой из предложенных фрагментов конфигурации Apache наиболее корректно настраивает обработку статических файлов (изображений, CSS) с включенным кэшированием на стороне клиента?<br><br>1.apache<br><Directory "/var/www/static"><br>Options -Indexes<br>AllowOverride None<br>Require all granted<br></Directory><br><br>2. apache<br><Directory "/var/www/static"><br>Options -Indexes<br>AllowOverride None<br>Require all granted<br>ExpiresActive On<br>ExpiresDefault "access plus 1 month"<br></Directory><br><br>3. apache<br>Alias "/static/" "/var/www/static/"<br><Location "/static"><br>Order allow,deny |

| | | | |
|---|---|---|---|
| | | | Allow from all<br></Location><br><br>4. apache<br><Directory "/var/www/static"><br>Options +Indexes<br>AllowOverride All<br>Require all granted<br></Directory><br><br>Ответ: 2.<br><br>Пояснение: Только вариант 2 включает директивы модуля mod_expires (ExpiresActive On, ExpiresDefault), которые добавляют HTTP-заголовки Cache-Control и Expires, инструктируя браузер кэшировать статические ресурсы на месяц. Это стандартная практика для повышения производительности.<br><br><br>**Задание 14.4**<br>При настройке виртуального хоста администратор получает ошибку Invalid command 'Require', perhaps misspelled or defined by a module not included in the server configuration. В чем наиболее вероятная причина?<br><br>1. Директива Require использована внутри секции <Files>, где она недопустима.<br>2. Не загружен модуль mod_authz_core, который предоставляет эту директиву.<br>3. Использована устаревшая форма директивы (Require all granted) вместо корректной для этой версии Apache.<br>4. Допущена синтаксическая ошибка в написании слова Require.<br><br>Ответ: 2.<br><br>Пояснение: Директива Require является частью системы авторизации Apache 2.4 и предоставляется модулем mod_authz_core. Если этот модуль не загружен (например, в |

| | | | |
|---|---|---|---|
| | | | минимальной сборке), данная директива будет неизвестна серверу. Ошибка прямо указывает на это: «perhaps defined by a module not included». |
| Тема 15 Общие сведения об операционных системах | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 15.1**<br>Какой из перечисленных алгоритмов планирования процессов в операционной системе является непрерывным (non-preemptive)?<br><br>1. Round Robin (RR, циклическое планирование).<br>2. Shortest Job First (SJF, наим. время выполнения).<br>3. Priority Scheduling with Preemption (приоритетное с вытеснением).<br>4. Multilevel Feedback Queue (многоуровневая очередь с обратной связью).<br><br>Ответ: 2 (Shortest Job First в своей классической форме).<br><br>Пояснение: Невытесняющее (непрерывное) планирование означает, что процесс, получивший процессор, работает до своего завершения или до явной блокировки (например, на операцию ввода-вывода). Классический SJF (без вытеснения) именно так и работает: он выбирает процесс с наименьшим предсказанным временем выполнения и отдает ему процессор до конца. Все остальные алгоритмы из списка (1, 3, 4) являются вытесняющими (preemptive): процесс может быть принудительно снят с выполнения по истечению кванта времени (1), при появлении более приоритетного процесса (3) или при перемещении между очередями (4).<br><br>**Задание 15.2**<br>В чем заключается основное назначение диспетчера (dispatcher) в подсистеме управления процессами операционной системы?<br><br>1. Принимать решение о том, какой процесс из очереди готовых получить процессор следующим (стратегия планирования).<br>2. Выполнять переключение контекста между процессами, т.е. сохранять состояние текущего процесса и загружать состояние выбранного для выполнения.<br>3. Управлять созданием и завершением процессов (вызовы fork(), exit() в UNIX).<br>4. Обеспечивать взаимодействие и синхронизацию между процессами (семафоры, мьютексы). |

| | | | |
|---|---|---|---|
| | | | Ответ: 2.<br><br>Пояснение: Диспетчер — это низкоуровневый модуль ядра, который выполняет механическую работу по переключению контекста. Планировщик (scheduler) принимает стратегическое решение (1), а диспетчер реализует его. Созданием и завершением процессов управляет другой компонент ядра (3). Взаимодействие процессов (4) обеспечивается подсистемой межпроцессного взаимодействия (IPC). Путаница между функциями планировщика и диспетчера — частая ошибка, которую важно устранять на фундаментальном уровне.<br><br>**Задание 15.3**<br>Вопрос: Как называется ключевой принцип построения операционной системы, при котором ОС состоит из набора небольших, изолированных модулей (сервисов), взаимодействующих через строго определенные интерфейсы, что повышает надежность и упрощает разработку? Этот принцип противопоставляется монолитной архитектуре. Укажите термин.<br><br>Ответ: Микроядерная архитектура (или принцип микроядра).<br><br>Пояснение: Именно микроядерная архитектура (microkernel architecture) воплощает принцип минимализма в ядре. В привилегированном режиме (ядре) выполняется только самый необходимый код: управление процессами, базовое планирование, межпроцессное взаимодействие (IPC). Все остальные сервисы (драйверы, файловые системы, сетевой стек) работают как изолированные пользовательские процессы (серверы). Это повышает отказоустойчивость (сбой драйвера не крашит всё ядро) и упрощает разработку/отладку компонентов.<br><br>**Задание 15.4**<br>Как называется абстракция, предоставляемая операционной системой, которая позволяет представлять различные устройства хранения (жесткие диски, SSD, USB-накопители) в виде иерархической структуры каталогов и файлов для удобства доступа и управления? |

| | | | |
|---|---|---|---|
| | | | Укажите термин.<br><br>Ответ: Файловая система.<br><br>Пояснение: Файловая система (File System) — это именно та абстракция, которая скрывает физические особенности хранения данных на блочных устройствах, предоставляя пользователю и программам логическую модель в виде именованных файлов, организованных в директории (папки). Это одна из фундаментальных абстракций ОС, обеспечивающая единообразный доступ к данным. |
| Тема 16 Интерфейс пользователя. Операционное окружение | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 16.1**<br>Какие из следующих утверждений о программном интерфейсе (API) операционной системы являются корректным? Выберите верные.<br><br>1. API — это набор функций, структур данных и констант, предоставляемых ОС для разработки прикладных программ.<br>2. Основное назначение API — скрыть детали реализации аппаратных и системных вызовов, обеспечивая переносимость и упрощение разработки.<br>3. Программный интерфейс (API) и интерфейс системных вызовов (System Call Interface) — это одно и то же.<br>4. Примером API операционной системы является Win32 API для Windows или POSIX API для UNIX-подобных систем.<br>5. API относится исключительно к графическим библиотекам для создания оконных интерфейсов (например, Qt, GTK).<br><br>Ответ: 1, 2, 4.<br><br>Пояснение:<br><br>1 — Это классическое определение API.<br><br>2 — Это ключевое назначение абстракции, которую предоставляет API.<br><br>4 — Это конкретные, широко известные примеры API операционных систем. |

| | | | |
|---|---|---|---|
| | | | **Задание 16.2**<br>Какие из перечисленных компонентов обычно входят в состав операционного окружения (operating environment) или оболочки (shell) ОС, предоставляя пользователю и программам стандартные сервисные функции? Выберите верные.<br><br>1. Командный интерпретатор (bash, cmd.exe, PowerShell).<br>2. Библиотеки времени выполнения (C runtime library, .NET Framework, Java Runtime Environment).<br>3. Утилиты для работы с файловой системой (cp, mv, dir, find).<br>4. Ядро операционной системы и его драйверы устройств.<br>5. Текстовые редакторы, компиляторы и отладчики.<br><br>Ответ: 1, 3.<br><br>Пояснение:<br>1 — Командный интерпретатор — центральный элемент операционного окружения, обеспечивающий прямой диалог пользователя с ОС.<br>3 — Стандартный набор утилит для управления файлами и данными — неотъемлемая часть любого операционного окружения (как CLI, так и GUI).<br><br>**Задание 16.3**<br>Как называется основной сервисный механизм в составе операционного окружения UNIX-подобных систем, который позволяет перенаправлять вывод одной программы на ввод другой, создавая конвейеры (pipes) для обработки данных? Укажите термин, а также символ, которым он обозначается в командной строке.<br><br>Ответ: Конвейер (pipe). Символ: \| (вертикальная черта).<br><br>Пояснение: Конвейер — это фундаментальная абстракция и сервис, предоставляемый командной оболочкой (shell). Он позволяет соединять стандартный вывод (stdout) одной |

| | | | |
|---|---|---|---|
| | | | программы со стандартным вводом (stdin) другой, создавая мощные и гибкие цепочки обработки данных (например, ls -l \| grep ".txt" \| sort). Это яркий пример того, как операционное окружение предоставляет больше, чем просто сумму отдельных утилит.<br><br>**Задание 16.4**<br>Вопрос: Как называется специальный тип переменных, управляемый операционным окружением и доступный всем процессам пользовательской сессии, который хранит информацию о настройках системы и пользователя, такие как путь к домашнему каталогу (HOME), список директорий для поиска программ (PATH) или текущая локаль (LANG)? Укажите общепринятый англоязычный термин.<br><br>Ответ: Environment variables (Переменные окружения).<br><br>Пояснение: Переменные окружения — это ключевой механизм настройки и передачи информации о среде выполнения (операционном окружении) от родительских процессов дочерним. Они являются частью контекста каждого процесса и широко используются как системой, так и прикладными программами для адаптации своего поведения. Это неотъемлемая часть понятия «операционное окружение». |
| Тема 17 Обработка прерываний | ОПК-5<br>С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 17.1**<br>Какие из следующих утверждений о прерываниях (interrupts) являются корректными? Выберите верные.<br><br>1; Прерывание — это сигнал от аппаратного устройства или программы, требующий немедленного внимания процессора.<br>2. Обработка прерывания всегда приводит к принудительному переключению контекста на другой пользовательский процесс.<br>3. Основная цель системы прерываний — реализация мультипрограммирования и эффективное использование процессорного времени.<br>4. Программные прерывания (исключения, traps) инициируются самим процессором при особых условиях выполнения инструкций (деление на ноль, доступ к защищенной памяти).<br>5. Обработчик прерывания (ISR) выполняется в непривилегированном пользовательском режиме для безопасности. |

| | | | |
|---|---|---|---|
| | | | Ответ: 1, 3, 4.<br><br>Пояснение:<br>1 — Это базовое определение прерывания.<br>3 — Именно благодаря прерываниям от таймера ОС может переключать процессы, а благодаря прерываниям от устройств ввода-вывода — не ждать их завершения, занимая CPU.<br>4 — Это точное определение программных прерываний (исключений). Они синхронны и связаны с выполнением конкретной инструкции.<br><br>**Задание 17.2**<br>Вопрос: Какие из следующих элементов обязательно участвуют или используются в ходе аппаратной обработки прерывания на уровне процессора? Выберите верные.<br><br>1. Вектор прерывания (номер), уникальный для каждого источника прерывания.<br>2. Таблица векторов прерываний (Interrupt Vector Table - IVT), содержащая адреса обработчиков.<br>3. Стек ядра (kernel stack) для сохранения контекста прерванного потока.<br>4. Очередь готовых процессов планировщика.<br>5. Счетчик команд (Instruction Pointer - IP) и слово состояния процессора (FLAGS/PSW).<br><br>Ответ: 1, 2, 5.<br><br>Пояснение:<br>1 — Устройство или условие процессора генерирует уникальный номер (вектор), идентифицирующий тип прерывания.<br>2 — Процессор использует этот номер как индекс в таблице IVT, чтобы найти адрес программы-обработчика (ISR).<br>5 — Аппаратура процессора автоматически сохраняет критический контекст (как минимум IP и FLAGS) перед передачей управления обработчику, чтобы можно было later вернуться к прерванной инструкции. |

| | | | |
|---|---|---|---|
| | | | **Задание 17.3**<br>Установите правильную последовательность аппаратных и начальных программных действий процессора и ОС при возникновении маскируемого аппаратного прерывания (например, от сетевой карты).<br><br>1. Процессор приостанавливает выполнение текущего потока команд.<br>2. Обработчик прерывания (ISR) сохраняет на стеке ядра остальные регистры процессора (RAX, RBX и т.д.) для полного сохранения контекста.<br>3. Процессор определяет номер вектора прерывания от устройства.<br>4. Устройство выставляет сигнал на линии запроса прерывания (IRQ).<br>5. Процессор загружает из таблицы векторов прерываний (IVT) адрес обработчика (ISR) и передает ему управление.<br>6. Процессор автоматически сохраняет на стеке текущие значения счетчика команд (IP) и регистра флагов (FLAGS/PSW).<br><br>Ответ: 4, 1, 3, 6, 5, 2.<br><br>Пояснение: Эта последовательность разделяет аппаратные (4,1,3,6,5) и начальные программные (2) действия, что критически важно для понимания. Устройство инициирует процесс (4). Процессор реагирует (1), идентифицирует источник (3), аппаратно сохраняет минимальный, но достаточный для возврата контекст (6), и аппаратно передает управление (5). Только после этого начинает работу код ОС (обработчик), и его первая задача — сохранить все остальное (2). Нарушение этого порядка (например, сохранение регистров до передачи управления) противоречит архитектуре процессора.<br><br>**Задание 17.4**<br>Установите последовательность этапов в полном цикле обработки аппаратного прерывания, включая действия операционной системы после работы основного обработчика (ISR).<br><br>1. Обработчик прерывания (ISR) выполняет минимально необходимые действия (например, чтение данных из порта устройства, отправка подтверждения). |

| | | | |
|---|---|---|---|
| | | | 2. Процессор, выполнив инструкцию возврата из прерывания (IRET), восстанавливает контекст прерванного потока из стека и продолжает его выполнение.<br>3. Планировщик ОС, если это необходимо, принимает решение о переключении на другой процесс с более высоким приоритетом, который мог быть разблокирован данным прерыванием.<br>4. Управление возвращается из ISR в общий код ядра, который отвечает за обработку «нижней половины» (bottom half) — более тяжелой обработки, которую нельзя делать в контексте прерывания.<br>5. Обработчик прерывания (ISR) восстанавливает ранее сохраненные регистры общего назначения из стека ядра.<br><br>Ответ: 1, 5, 2, 4, 3.<br><br>Пояснение: Эта последовательность иллюстрирует ключевую концепцию разделения обработки на «верхнюю» (ISR, быстрая, в контексте прерывания) и «нижнюю» (отложенную, в контексте процесса/ядра). ISR делает срочное (1), затем готовит возврат (5). Аппаратный возврат (2) восстанавливает прерванный мир. Затем уже в «нормальном» режиме ядро доделывает тяжелую работу (4). И только на основе результатов этой работы планировщик может принять решение (3). Это показывает, как ОС обеспечивает и быстроту реакции, и стабильность системы. |
| Тема 18 Работа с файлами | ОПК-5 С пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на | **Задание 18.1**<br>Установите последовательность основных шагов, которые выполняет ОС при системном вызове open("/home/user/report.txt", O_RDWR) для существующего файла.<br><br>1. Поиск индексного дескриптора (inode) файла report.txt в каталоге user, используя структуры данных файловой системы.<br>2. Проверка прав доступа процесса к файлу (чтение и запись) на основе uid/gid процесса и прав, хранящихся в метаданных файла.<br>3. Разрешение абсолютного пути /home/user/report.txt, начиная с корневого каталога (/).<br>4. Создание новой записи в таблице открытых файлов ядра для процесса и связывание ее с найденным индексным дескриптором.<br>5. Возврат пользовательскому процессу файлового дескриптора — целого числа, являющегося индексом в локальной для процесса таблице дескрипторов. |

| | | | |
|---|---|---|---|
| | | базовом уровне | Ответ: 3, 1, 2, 4, 5.<br><br>Пояснение: Эта последовательность отражает процесс «логического открытия». Сначала ОС должна пройти по логическому пути (3), на каждом этапе проверяя права на поиск (execute) в каталогах. Найдя запись о файле, она получает ключ к его метаданным (например, номер inode) (1). Затем проверяет конкретные права на запрошенную операцию (2). Только после успешных проверок файл считается открытым — для этого в системных структурах создаются необходимые записи (4), и пользовательский код получает удобный handle — дескриптор (5). Этот порядок обеспечивает безопасность и корректность.<br><br>**Задание 18.2**<br>Установите последовательность действий на уровне физической организации файловой системы при первом чтении данных из только что открытого файла (системный вызов read()).<br><br>1. Драйвер блочного устройства выполняет низкоуровневую команду чтения секторов в память (в кэш).<br>2. Подсистема управления файловой системой в ядре определяет, что нужные блоки отсутствуют в кэше (cache miss).<br>3. Код файловой системы, используя структуры (например, прямые/косвенные блоки в inode), вычисляет номера логических блоков файловой системы, соответствующих запрошенному смещению.<br>4. Драйвер файловой системы преобразует логические номера блоков ФС в адреса физических секторов на носителе (LBA).<br>5. Данные из кэша копируются в буфер пользовательского процесса.<br><br>Ответ: 3, 2, 4, 1, 5.<br><br>Пояснение: Эта последовательность показывает «путь данных» от абстрактного смещения в файле до физического диска. Сначала на основе метаданных файла (inode) определяется, какие логические блоки ФС нужны (3). Затем проверяется кэш (2) — если данных там нет, инициируется физическое чтение. Для этого логические блоки ФС переводятся в адреса, |

| | | | |
|---|---|---|---|
| | | | понятные диску (4), и отправляется аппаратная команда (1). После выполнения данные помещаются в кэш и отдаются процессу (5). Этот порядок иллюстрирует многоуровневую абстракцию и важность кэширования.<br><br>**Задание 18.3**<br>Сопоставьте элемент файловой системы с уровнем организации, к которому он в первую очередь относится.<br><br>Список А (Элемент):<br>1. Имя файла и расширение<br>2. Индексный дескриптор (inode)<br>3. Права доступа в формате rwxr-xr--<br>4. Таблица размещения файлов (FAT)<br>5. Путь /usr/bin/bash<br><br>Список Б (Уровень организации):<br>А. Логическая организация (представление для пользователя)<br>В. Физическая организация (внутренняя реализация на диске)<br><br>Ответ: 1А, 2В, 3А, 4В, 5А.<br><br>**Задание 18.4**<br>Сопоставьте метод физического размещения файла на диске с его ключевой характеристикой.<br><br>Список А (Метод размещения):<br>1. Непрерывное (последовательное) размещение<br>2. Размещение связным списком (через FAT)<br>3. Индексное размещение (через inode)<br><br>Список Б (Характеристика):<br>А. Для доступа к произвольному блоку требуется последовательное чтение цепочки указателей, что замедляет случайный доступ. |

| | | | |
|---|---|---|---|
| | | | B. Позволяет эффективный прямой доступ к любому блоку, храня все указатели централизованно, но размер файла может быть ограничен размером индексного блока.<br>C. Обеспечивает максимальную скорость последовательного чтения, но страдает от внешней фрагментации и сложности расширения файла.<br><br>Ответ: 1C, 2A, 3B. |
| Тема 19 Планирование заданий. Распределение ресурсов | ОПК-5<br>С<br>пособен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | **Задание 19.1**<br>Сопоставьте алгоритм планирования с типом системы, для которого он наиболее характерен или эффективен.<br><br>Список А (Алгоритм планирования):<br>1. First-Come, First-Served (FIFO)<br>2. Round Robin (RR)<br>3.Shortest Job First (SJF)<br>4. Earliest Deadline First (EDF)<br><br>Список Б (Тип системы):<br>A. Интерактивные time-sharing системы (универсальный fairness).<br>B. Системы реального времени с динамическими приоритетами.<br>C. Системы пакетной обработки (простой и предсказуемый).<br>D. Системы пакетной обработки (оптимальный по среднему времени выполнения).<br><br>Ответ: 1C, 2A, 3D, 4B.<br><br>**Задание 19.2**<br>Сопоставьте метрику производительности планировщика с ее определением.<br><br>Список А (Метрика):<br>1. Время оборачивания (Turnaround Time)<br>2. Время ожидания (Waiting Time)<br>3. Время отклика (Response Time)<br>4. Пропускная способность (Throughput) |

| | | | |
|---|---|---|---|
| | | | Список Б (Определение):<br>A. Интервал от момента поступления задачи в систему до момента выдачи последнего результата (время от подачи до завершения).<br>B. Количество задач, завершенных системой в единицу времени.<br>C. Интервал от момента поступления задачи до момента, когда она впервые получает процессор для выполнения.<br>D. Общее время, которое процесс провел в состоянии готовности (готов, но не выполняется).<br><br>Ответ: 1A, 2D, 3C, 4B.<br><br>**Задание 19.3**<br>В чем заключается основная проблема алгоритма Shortest Job First (SJF), которая препятствует его прямому применению в реальных интерактивных системах, даже несмотря на его теоретическую оптимальность?<br><br>1. Он требует знания времени выполнения задач в момент их поступления, что часто невозможно предсказать.<br>2. Он является строго невытесняющим, что делает систему абсолютно неотзывчивой.<br>3. Он приводит к «голоданию» (starvation) длинных задач, если в систему постоянно поступают короткие задачи.<br>4. Он имеет слишком высокие накладные расходы на принятие решений по сравнению с Round Robin.<br><br>Ответ: 1.<br><br>Пояснение: Это фундаментальное ограничение SJF. Алгоритм требует априорного знания или точной оценки времени выполнения (burst time) каждого процесса. В реальных системах (особенно интерактивных, где задачи — это программы пользователя) это время неизвестно и непредсказуемо. Поэтому SJF используется как теоретический эталон или в адаптированном виде (например, в MLFQ с эвристической оценкой по прошлому поведению). Хотя варианты 3 и 2 также являются недостатками, они вторичны или не всегда верны (существует вытесняющая версия — SRTF). Вариант 4 не является основной |

| | | | |
|---|---|---|---|
| | | | проблемой.<br><br>**Задание 19.4**<br>Какой из перечисленных алгоритмов планирования наиболее эффективно решает проблему обеспечения хорошего времени отклика для интерактивных задач (I/O-bound) и одновременно приемлемой пропускной способности для фоновых вычислений (CPU-bound) в смешанной рабочей нагрузке?<br><br>1. First-Come, First-Served (FIFO) — гарантирует честность и простоту.<br>2. Round Robin (RR) с фиксированным квантом времени — обеспечивает равномерное распределение времени.<br>3. Multilevel Feedback Queue (MLFQ) — использует несколько очередей с разными приоритетами и квантами, перемещая процессы между ними на основе наблюдаемого поведения.<br>4. Priority Scheduling (по статическому приоритету) — позволяет отдавать приоритет интерактивным задачам вручную.<br><br>Ответ: 3 (Multilevel Feedback Queue).<br><br>Пояснение: MLFQ был разработан именно для решения этой дилеммы. Он адаптируется: короткие интерактивные задачи (I/O-bound) быстро завершаются в очередях с высоким приоритетом и малым квантом, обеспечивая хороший response time. Долгие вычислительные задачи (CPU-bound) постепенно «скатываются» в низкоприоритетные очереди с большим квантом, где они не мешают интерактивности, но все же получают свое процессорное время. RR (2) обеспечивает fairness, но не делает различий между типами задач. Статический приоритет (4) не адаптируется к изменению поведения задачи. FIFO (1) совершенно не подходит для интерактивных систем. |
| Тема 20 Защищенность и отказоустойчивость операционн | ОПК-5<br>С пособен инсталлировать программное и аппаратное | ИОПК-5.1. Знать основы системного администрирования, администрирова | **Задание 20.1**<br>Какая из перечисленных технологий хранения данных является наиболее отказоустойчивой к одновременному отказу двух дисков в массиве без потери данных и доступности?<br><br>1. RAID 0 (чередование, striping) — повышает производительность, но не |

| | | | |
|---|---|---|---|
| ых систем | обеспечение для информационных и автоматизированных систем | ния СУБД, современные стандарты информационного взаимодействия систем на базовом уровне | отказоустойчивость.<br>2. RAID 1 (зеркалирование, mirroring) — полная копия данных на двух дисках.<br>3. RAID 5 (чередование с контролем четности) — распределенная четность, позволяет пережить отказ одного диска.<br>4. RAID 6 (чередование с двойной четностью) — позволяет пережить одновременный отказ любых двух дисков в массиве.<br>5. JBOD (Just a Bunch Of Disks) — простое объединение дисков в один том.<br><br>Ответ: 4 (RAID 6).<br><br>Пояснение: RAID 6 использует две независимые схемы контроля четности (например, Reed-Solomon коды), что позволяет восстановить данные при отказе любых двух дисков в массиве одновременно. RAID 1 выдержит отказ одного из двух дисков, но при отказе обоих данные будут потеряны. RAID 5 выдерживает отказ только одного диска. RAID 0 и JBOD не обеспечивают отказоустойчивости вообще. Для требовательных сред, где важна доступность данных даже при множественных сбоях, RAID 6 является стандартным выбором.<br><br>**Задание 20.2**<br>В чем заключается ключевая разница между дискреционным (Discretionary Access Control - DAC) и мандатным (Mandatory Access Control - MAC) управлением доступом в операционных системах?<br><br>1. DAC основан на идентификаторе пользователя (UID), а MAC — на ролях (Roles).<br>2. В DAC владелец объекта (файла) сам определяет политику доступа к нему. В MAC политика доступа централизованно задается администратором безопасности и не может быть изменена пользователями.<br>3. DAC реализуется через списки контроля доступа (ACL), а MAC — через матрицу доступа.<br>4. DAC менее производителен, чем MAC, из-за необходимости проверять сложные ACL для каждого запроса.<br><br>Ответ: 2. |

| | | | |
|---|---|---|---|
| | | | Пояснение: Это фундаментальное концептуальное различие. В DAC (как в классических UNIX правах или Windows ACL) владелец файла имеет полный контроль над тем, кто и что может с ним делать. В MAC (SELinux, AppArmor) пользователи и процессы (субъекты) получают метки безопасности, объекты (файлы, сокеты) также имеют метки. Правила доступа между метками определяются глобальной политикой, которую устанавливает администратор. Пользователь, даже будучи владельцем файла, не может изменить эти правила — отсюда и название «мандатное» (обязательное). Варианты 1 и 3 описывают частные реализации, но не ключевую разницу. Вариант 4 неверен — производительность зависит от конкретной реализации.<br><br>**Задание 20.3**<br>Как называется основной механизм в современных файловых системах (ZFS, Btrfs), который обеспечивает высокую целостность данных и восстанавливаемость за счет того, что при любом обновлении данных всегда записывается новый блок, а старый блок остается неизменным до освобождения? Укажите общепринятый англоязычный термин.<br><br>Ответ: Copy-on-Write (CoW) / Копирование при записи.<br><br>Пояснение: Механизм Copy-on-Write является краеугольным камнем для обеспечения целостности. Он предотвращает ситуации, когда сбой питания может привести к частичной записи блока, повреждающей старые данные. Поскольку новые данные пишутся в новое место, предыдущее состояние файловой системы (или снапшот) остается неизменным и согласованным. Это позволяет легко откатить изменения или восстановить данные после сбоя. CoW также лежит в основе эффективных снапшотов.<br><br>**Задание 20.4**<br>Как называется компонент подсистемы безопасности ядра Linux, который отвечает за регистрацию (логирование) всех значимых событий безопасности, таких как успешные и неуспешные попытки входа, использование привилегий, изменения политик доступа? Укажите его точное название. |

| | | | |
|---|---|---|---|
| | | | Ответ: Audit daemon (auditd) и подсистема аудита ядра Linux (Linux Audit Subsystem).<br><br>Пояснение: Подсистема аудита Linux (во главе с демоном auditd) является ключевым механизмом аудита (accountability). Она собирает события от ядра, связанные с безопасностью, в защищенный от модификации журнал (/var/log/audit/audit.log). Эти логи критически важны для расследования инцидентов, соответствия стандартам (например, PCI DSS) и анализа действий в системе. Без эффективного аудита даже самые строгие механизмы аутентификации и авторизации не обеспечивают полной подотчетности. |

***Критерии оценивания тестового задания:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | от 90 до 100 % правильно выполненных заданий |
| **хорошо** | от 70 до 89 % правильно выполненных заданий |
| **удовлетворительно** | от 50 до 69 % правильно выполненных заданий |
| **неудовлетворительно** | менее 50 % правильно выполненных заданий |

**4.2 ТИПОВЫЕ ПРАКТИЧЕСКИЕ ЗАДАНИЯ**

Практические задания должны отражать умение обучающегося применять осваиваемую компетенцию в практических ситуациях и при решении производственных задач.

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| Тема 1. Введение в ОС Linux. Понятие операционной системы. Лицензии на ПО<br>Тема 2. Атрибуты файлов. Работа с файлами<br>Тема 3. Архиваторы<br>Тема 4. Обработка текстовых файлов | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание №1**. Анализ лог-файла и безопасное извлечение данных<br>Формулировка: В каталоге /var/log/ находится файл secure.log.<br>Ваша задача:<br>● Найти все строки в этом файле, содержащие слово "FAILED".<br>● Сохранить только первые 5 таких найденных строк в новый файл failed_attempts.txt в вашем домашнем каталоге.<br>● Установить права доступа на файл failed_attempts.txt так, чтобы владелец мог читать и писать, группа — только читать, остальные — не имели никаких прав.<br>● Создать сжатый архив logs_backup.tar.gz, содержащий исходный файл secure.log и созданный вами failed_attempts.txt.<br><br>Ответ:<br># 1. Поиск строк и сохранение первых 5 в новый файл<br>grep "FAILED" /var/log/secure.log \| head -5 > ~/failed_attempts.txt<br><br># 2. Установка прав доступа (rw-r----- или 640)<br>chmod 640 ~/failed_attempts.txt<br># или chmod u=rw,g=r,o= ~/failed_attempts.txt<br><br># 3. Создание сжатого архива<br>tar -czf ~/logs_backup.tar.gz /var/log/secure.log ~/failed_attempts.txt |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | 2>/dev/null<br># 2>/dev/null убирает предупреждение о ведущем '/' в именах<br><br>**Задание №2.** Структурирование данных из CSV<br>Формулировка: Дан файл employees.csv с содержимым:<br>*id;name;department;salary*<br>*101;Ivanov;IT;80000*<br>*102;Petrov;Sales;60000*<br>*103;Sidorov;IT;95000*<br>*104;Kuznetsova;HR;55000*<br>Напишите команду (или конвейер команд), которая:<br>● Отсортирует строки по убыванию зарплаты (четвертое поле).<br>● Выведет только имена и зарплаты сотрудников из отдела "IT".<br>● Разделитель полей — точка с запятой.<br><br>Ответ:<br># Решение с использованием sort, awk и head/tail для пропуска заголовка<br>awk -F';' '$3 == "IT" {print $2, $4}' employees.csv \| sort -t' ' -k2 -nr<br># или, если нужно корректно обработать заголовок:<br>awk -F';' 'NR>1 && $3=="IT" {print $2, $4}' employees.csv \| sort -t' ' -k2 -nr<br><br>Ожидаемый вывод: |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | *Sidorov 95000*<br>*Ivanov 80000* |
| Тема 5. Текстовые редакторы.<br>Тема 6. Основы bash.<br>Тема 7. Оболочка bash.<br>Тема 8. Shell как язык программирования. | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание №3.** Создание скрипта для мониторинга дискового пространства<br>Формулировка: Напишите bash-скрипт check_disk.sh, который:<br>● Принимает один обязательный аргумент командной строки — путь к каталогу для проверки.<br>● Если аргумент не передан, выводит сообщение об ошибке "Usage: $0 <directory_path>" и завершается с кодом 1.<br>● Проверяет, существует ли переданный каталог. Если нет — выводит ошибку и завершается с кодом 2.<br>● Вычисляет общий размер всех файлов в этом каталоге (включая подкаталоги) в мегабайтах (МБ).<br>● Если размер превышает 100 МБ, выводит предупреждение "WARNING: Directory size is <SIZE> MB. Consider cleanup.".<br>● В любом случае выводит строку "Directory <PATH> size: <SIZE> MB".<br><br>Ответ (check_disk.sh):<br>#!/bin/bash<br><br># Проверка количества аргументов<br>if [ $# -ne 1 ]; then<br>  echo "Usage: $0 <directory_path>"<br>  exit 1 |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | fi<br><br>target_dir="$1"<br><br># Проверка существования каталога<br>if [ ! -d "$target_dir" ]; then<br>  echo "Error: Directory '$target_dir' does not exist."<br>  exit 2<br>fi<br><br># Вычисление размера в МБ (суммарный размер файлов, округленный)<br># 'du' возвращает размер в килобайтах по умолчанию, 'awk' переводит в МБ<br>size_mb=$(du -sk "$target_dir" 2>/dev/null \| awk '{printf "%.1f", $1/1024}')<br><br># Проверка размера и вывод<br>if (( $(echo "$size_mb > 100" \| bc -l) )); then<br>  echo "WARNING: Directory size is $size_mb MB. Consider cleanup."<br>fi<br><br>echo "Directory $target_dir size: $size_mb MB"<br><br>**Задание №4.** Скрипт обработки списка пользователей |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | Формулировка: Напишите скрипт user_report.sh, который читает файл users.list (формат: username:group:shell). Скрипт должен:<br>● Подсчитать общее количество пользователей.<br>● Вывести количество пользователей, использующих shell /bin/bash.<br>● Для каждой уникальной группы вывести количество пользователей в ней.<br><br>Ответ (user_report.sh):<br>#!/bin/bash<br><br>file="users.list"<br><br>if [ ! -f "$file" ]; then<br>  echo "Error: File $file not found."<br>  exit 1<br>fi<br><br>echo "=== User Report ==="<br># 1. Общее количество пользователей<br>total_users=$(wc -l < "$file")<br>echo "Total users: $total_users"<br><br># 2. Количество пользователей с /bin/bash<br>bash_users=$(grep -c ":/bin/bash$" "$file")<br>echo "Users with /bin/bash: $bash_users" |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | echo ""<br>echo "=== Users per group ==="<br># 3. Количество пользователей в каждой группе (второе поле)<br># Извлекаем группу, сортируем, подсчитываем уникальные<br>cut -d: -f2 "$file" \| sort \| uniq -c \| while read count group; do<br>  echo "Group '$group': $count user(s)"<br>done |
| Тема 9. Элементы администрирования.<br>Тема 10. Конфигурирование сетевых интерфейсов.<br>Тема 11. Сетевой экран.<br>Тема 12. Сетевой экран (продолжение)<br>Тема 13. Скрипты с Web-интерфейсом.<br>Тема 14 Web-сервер Apache. | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание №5.** Настройка статической сети и базового фаервола<br>Формулировка: Вы администратор сервера с интерфейсом eth0. Необходимо:<br>● Назначить интерфейсу eth0 статический IP-адрес 192.168.1.100/24 и шлюз по умолчанию 192.168.1.1.<br>● Настроить iptables для обеспечения базовой безопасности:<br>○ Разрешить весь исходящий трафик.<br>○ Разрешить входящие SSH-подключения (порт 22) только с подсети 192.168.1.0/24.<br>○ Разрешить входящие HTTP/HTTPS (порты 80, 443) со всех адресов.<br>○ Запретить все остальные входящие подключения.<br>○ Разрешить "ответный" трафик (ESTABLISHED, RELATED).<br>● Сохранить правила iptables, чтобы они загружались при старте системы (команда для сохранения в Ubuntu/Debian).<br><br>Ответ: |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | # 1. Настройка статического IP (пример для netplan в Ubuntu 18.04+)<br># Файл: /etc/netplan/01-netcfg.yaml<br># После редактирования: sudo netplan apply<br>cat <<EOF \| sudo tee /etc/netplan/01-netcfg.yaml > /dev/null<br>network:<br>  version: 2<br>  ethernets:<br>  eth0:<br>  addresses: [192.168.1.100/24]<br>  gateway4: 192.168.1.1<br>  nameservers:<br>  addresses: [8.8.8.8, 1.1.1.1]<br>EOF<br>sudo netplan apply<br><br># 2. Настройка iptables<br># Очистка старых правил и установка политик по умолчанию DROP на вход<br>sudo iptables -F<br>sudo iptables -X<br>sudo iptables -P INPUT DROP<br>sudo iptables -P FORWARD DROP<br>sudo iptables -P OUTPUT ACCEPT<br><br># Разрешение локального интерфейса (loopback) |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | sudo iptables -A INPUT -i lo -j ACCEPT<br><br># Разрешение ответного и связанного трафика<br>sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT<br><br># Разрешение SSH только с локальной подсети<br>sudo iptables -A INPUT -p tcp --dport 22 -s 192.168.1.0/24 -j ACCEPT<br><br># Разрешение HTTP/HTTPS со всех адресов<br>sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT<br>sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT<br><br># 3. Сохранение правил (для Ubuntu/Debian с пакетом iptables-persistent)<br>sudo netfilter-persistent save<br># Или альтернатива: sudo iptables-save > /etc/iptables/rules.v4<br><br>**Задание №6**. Создание простого CGI-скрипта для мониторинга<br>Формулировка: Настройте веб-сервер Apache для выполнения CGI-скриптов. Создайте bash-CGI скрипт status.cgi в директории /usr/lib/cgi-bin/, который отображает в браузере:<br>● Текущую дату и время.<br>● Загрузку системы (вывод команды uptime).<br>● Использование дискового пространства (вывод df -h). |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | ● Список 5 самых "тяжелых" процессов по использованию памяти.<br><br>Ответ:<br>Настройка Apache (Debian/Ubuntu):<br>sudo a2enmod cgi<br>sudo systemctl restart apache2<br><br>Создание скрипта /usr/lib/cgi-bin/status.cgi:<br>#!/bin/bash<br><br>echo "Content-type: text/html"<br>echo ""<br><br>cat <<EOF<br><html><br><head><title>System Status</title></head><br><body><br><h2>System Status Report</h2><br><pre><br>$(date)<br>==========<br>Uptime:<br>$(uptime)<br>==========<br>Disk Usage: |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | $(df -h)<br>==========<br>Top 5 Processes by Memory:<br>$(ps aux --sort=-%mem \| head -6)<br></pre><br></body><br></html><br>EOF<br><br>Установка прав:<br>sudo chmod +x /usr/lib/cgi-bin/status.cgi<br>sudo chown www-data:www-data /usr/lib/cgi-bin/status.cgi |
| Тема 15 Общие сведения об операционных системах<br>Тема 16. Интерфейс пользователя. Операционное окружение<br>Тема 17. Обработка прерываний | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание №7**. Создание командного файла для автоматизации задач<br>Формулировка: Создайте командный файл (.bat) system_report.bat, который:<br>● Выводит в файл C:\report_%DATE%.txt заголовок "System Report".<br>● Добавляет в этот файл текущие дату и время.<br>● Добавляет список запущенных служб (службы Windows) в состоянии "Running".<br>● Добавляет информацию о свободном месте на диске C:.<br><br>Ответ (system_report.bat):<br>@echo off<br>set |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | report_file=C:\report_%DATE:~0,2%%DATE:~3,2%%DATE:~6,4%.txt<br><br>echo System Report > %report_file%<br>echo ============== >> %report_file%<br>echo Date/Time: %DATE% %TIME% >> %report_file%<br>echo. >> %report_file%<br><br>echo Running Services: >> %report_file%<br>sc query state= all \| findstr /C:"SERVICE_NAME" /C:"STATE.*: 4" >> %report_file%<br>echo. >> %report_file%<br><br>echo Disk C: Free Space: >> %report_file%<br>wmic logicaldisk where caption="C:" get size,freespace /format:value >> %report_file%<br><br>echo Report saved to: %report_file%<br>pause<br><br>Примечание: Для более читаемого формата даты можно использовать %DATE%, но для имени файла лучше убрать разделители. sc query выводит все службы, findstr фильтрует строки с именем и состоянием "4" (Running).<br><br>**Задание №8**. Анализ событий (Event Log) с помощью PowerShell |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | Формулировка: Напишите сценарий PowerShell check_errors.ps1, который находит в журнале "System" за последние 24 часа все события с уровнем "Error" (Уровень 2) и выводит их в консоль в формате: "Время \| Источник \| ID события \| Сообщение".<br><br>Ответ (check_errors.ps1):<br># Определяем время сутки назад<br>$StartTime = (Get-Date).AddHours(-24)<br><br># Получаем события уровня Error (Level = 2) из лога System за последние 24 часа<br>$ErrorEvents = Get-WinEvent -LogName "System" -MaxEvents 1000 \| Where-Object {<br>  $_.TimeCreated -ge $StartTime -and $_.Level -eq 2<br>}<br><br># Выводим отформатированные результаты<br>foreach ($Event in $ErrorEvents) {<br>  $Time = $Event.TimeCreated.ToString("yyyy-MM-dd HH:mm:ss")<br>  $Source = $Event.ProviderName<br>  $ID = $Event.Id<br>  $Message = $Event.Message -replace "`n", " " -replace "`r", " " # Убираем переносы строк для компактности<br><br>  Write-Host ("{0} \| {1} \| {2} \| {3}" -f $Time, $Source, $ID, $Message) |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | }<br><br># Если событий нет<br>if ($ErrorEvents.Count -eq 0) {<br>  Write-Host "No Error events found in System log for the last 24 hours."<br>}<br> |
| Тема 18. Работа с файлами<br>Тема 19. Планирование заданий. Распределение ресурсов<br>Тема 20. Защищенность и отказоустойчивость операционных систем | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание №9**. Моделирование планировщика процессов (Round Robin)<br>Формулировка: Напишите скрипт на Python или bash, который имитирует вытесняющий алгоритм планирования Round Robin (RR). На вход подается квант времени Q=2 и список процессов: P1(0, 5), P2(1, 3), P3(2, 8), где первое число — время появления, второе — общее время выполнения (burst time). Скрипт должен вывести порядок выполнения процессов и среднее время ожидания.<br><br>Ответ (rr_scheduler.py):<br>#!/usr/bin/env python3<br><br>class Process:<br>  def __init__(self, name, arrival, burst):<br>  self.name = name<br>  self.arrival = arrival<br>  self.burst = burst<br>  self.remaining = burst |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | self.waiting = 0<br>self.finished = False<br><br>def round_robin(processes, quantum):<br>time = 0<br>queue = []<br>order = []<br>total_waiting = 0<br>n = len(processes)<br><br># Инициализация: все процессы еще не готовы<br>remaining_procs = processes[:]<br><br>while remaining_procs or queue:<br># Добавляем процессы, которые появились к текущему времени<br>for p in remaining_procs[:]:<br>if p.arrival <= time:<br>queue.append(p)<br>remaining_procs.remove(p)<br><br>if not queue:<br>time += 1<br>continue<br><br>current = queue.pop(0) |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | order.append(current.name)<br><br># Время выполнения в этом кванте<br>exec_time = min(quantum, current.remaining)<br>current.remaining -= exec_time<br>time += exec_time<br><br># Увеличиваем время ожидания для процессов в очереди<br>for p in queue:<br>p.waiting += exec_time<br><br># Добавляем новые процессы, пока выполнялся текущий<br>for p in remaining_procs[:]:<br>if p.arrival <= time:<br>queue.append(p)<br>remaining_procs.remove(p)<br><br># Если процесс не завершен, ставим в конец очереди<br>if current.remaining > 0:<br>queue.append(current)<br>else:<br>current.finished = True<br>total_waiting += current.waiting<br><br># Вывод результатов<br>print("Execution order:", ' -> '.join(order)) |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | avg_wait = total_waiting / n<br>print(f"Average waiting time: {avg_wait:.2f}")<br><br>if __name__ == "__main__":<br># Процессы: (Имя, Время появления, Burst time)<br>procs = [<br>Process("P1", 0, 5),<br>Process("P2", 1, 3),<br>Process("P3", 2, 8)<br>]<br>round_robin(procs, quantum=2)<br><br>**Задание №10.** Создание отказоустойчивой конфигурации (RAID 1) и проверка целостности файлов<br>Формулировка:<br>● Создайте два файла-диска (/tmp/disk1.img, /tmp/disk2.img) размером по 100 МБ.<br>● Объедините их в программный RAID 1 (зеркало) с помощью mdadm.<br>● Создайте на массиве файловую систему ext4 и смонтируйте её в /mnt/raid_test.<br>● Создайте тестовый файл с контрольной суммой.<br>● Имитируйте отказ одного из "дисков" (отсоедините его от массива).<br>● Убедитесь, что данные доступны, и проверьте их |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | целостность.<br><br>Ответ (последовательность команд для bash):<br># Внимание: Выполнять с root-правами (sudo). Подходит для учебного тестирования.<br><br># 1. Создание файлов-дисков<br>dd if=/dev/zero of=/tmp/disk1.img bs=1M count=100<br>dd if=/dev/zero of=/tmp/disk2.img bs=1M count=100<br><br># 2. Создание RAID 1 массива<br>sudo losetup -fP /tmp/disk1.img<br>sudo losetup -fP /tmp/disk2.img<br># Узнаем имена loop-устройств (обычно loop0, loop1)<br>LOOP1=$(losetup -l \| grep disk1.img \| awk '{print $1}')<br>LOOP2=$(losetup -l \| grep disk2.img \| awk '{print $1}')<br><br>sudo mdadm --create /dev/md0 --level=mirror --raid-devices=2 $LOOP1 $LOOP2<br><br># 3. Создание ФС и монтирование<br>sudo mkfs.ext4 /dev/md0<br>sudo mkdir -p /mnt/raid_test<br>sudo mount /dev/md0 /mnt/raid_test<br><br># 4. Создание тестового файла и его контрольной суммы |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | echo "Very important data" \| sudo tee /mnt/raid_test/testfile.txt > /dev/null<br>ORIGINAL_SUM=$(sudo md5sum /mnt/raid_test/testfile.txt \| awk '{print $1}')<br>echo "Original MD5: $ORIGINAL_SUM"<br><br># 5. Имитация отказа первого диска (помечаем как faulty и удаляем)<br>sudo mdadm /dev/md0 --fail $LOOP1<br>sudo mdadm /dev/md0 --remove $LOOP1<br><br># 6. Проверка доступности данных и целостности<br>echo "RAID status after failure:"<br>sudo mdadm --detail /dev/md0 \| grep -A5 "State :"<br>CURRENT_SUM=$(sudo md5sum /mnt/raid_test/testfile.txt \| awk '{print $1}')<br>echo "Current MD5: $CURRENT_SUM"<br><br>if [ "$ORIGINAL_SUM" = "$CURRENT_SUM" ]; then<br>  echo "SUCCESS: Data integrity is maintained after disk failure."<br>else<br>  echo "FAIL: Data corruption detected!"<br>fi<br><br># 7. Очистка (демонтирование и остановка)<br>sudo umount /mnt/raid_test |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Задания к практическому занятию |
|---|---|---|---|
| | | | sudo mdadm --stop /dev/md0<br>sudo losetup -d $LOOP1 $LOOP2<br>rm /tmp/disk1.img /tmp/disk2.img |

***Критерии оценивания практических занятий:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт правильные ответы на вопросы преподавателя |
| **хорошо** | Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя |
| **удовлетворительно** | Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя |
| **неудовлетворительно** | Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя |

## 4.3. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ КОНТРОЛЬНЫХ РАБОТ

Контрольные работы содержат несколько практических заданий по индивидуальным вариантам, в полном объеме охватывающих изученный материал по указанной теме. Выполнение контрольных работ позволяет определить результат освоения компетенций по дисциплине в рамках рассматриваемой темы, оцениваемый с помощью соответствующих индикаторов достижения компетенций.

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| Тема 5 Текстовые редакторы. | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание 5.1**<br>Какие из следующих утверждений о текстовом редакторе MCEDIT (в составе Midnight Commander) являются верными?<br><br>1. Это консольный (текстовый) редактор, работающий в терминале без графического интерфейса (TUI).<br>2. Он тесно интегрирован с файловым менеджером Midnight Commander (F1-F10, меню), что упрощает навигацию и пакетное редактирование.<br>3. Для работы ему обязательно требуется установленная графическая среда (X Window System или Wayland).<br>4. Он поддерживает подсветку синтаксиса для многих языков программирования и разметки.<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. MCEDIT — это классический TUI (Text-based User Interface) редактор, наследник Norton Editor, работающий в любой текстовой консоли или эмуляторе терминала.<br>2 - Правильно. Ключевая особенность. Функциональные клавиши (F3-F4 для копирования/перемещения и т.д.) и меню верхнего уровня делают его частью экосистемы MC, а не standalone-редактором.<br>4 - Правильно. Несмотря на простоту, MCEDIT имеет базовую, но полезную подсветку синтаксиса для C, Python, HTML, конфигов и др.<br><br>**Задание 5.2**<br>Какие из следующих характеристик применимы к редактору GVIM (графическая версия Vim)?<br><br>1. Он сохраняет все основные режимы работы классического Vim |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | (нормальный, вставки, визуальный, командной строки).<br>2. Он предоставляет стандартные элементы графического интерфейса: меню, панели инструментов, диалоги открытия/сохранения файлов.<br>3. Его интерфейс и поведение полностью идентичны редактору Kate, так как оба являются графическими.<br>4. Он может быть использован для удаленного редактирования файлов по SSH в чисто текстовом режиме с помощью команды vim (без префикса g).<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. GVIM — это Vim с графической оболочкой. Все фундаментальные концепции Vim (модальность, команды :, hjkl навигация) остаются.<br>2 - Правильно. В отличие от консольного Vim, GVIM добавляет привычные GUI-элементы, что облегчает начальное освоение для пользователей, привыкших к мыши и меню.<br>4 - Правильно. Это мощное преимущество Vim/GVIM. Одна и та же программа (vim) может работать и в графическом режиме (gvim), и в консольном. При SSH-сессии автоматически запускается консольная версия, что делает навыки работы с Vim универсальными.<br><br>**Задание 5.3**<br>Как называется характерный для редакторов семейства Vim режим работы, в котором нажатия клавиш интерпретируются не как ввод текста, а как команды для перемещения курсора, удаления, копирования и других действий? (Ответ: одно слово — название режима)<br><br>Ответ: Нормальный (Normal mode) или Командный (Command mode). |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | Пояснение: Это фундаментальное понятие в философии Vim. «Нормальный режим» — это режим по умолчанию, в котором пользователь находится большую часть времени. В этом режиме клавиши h,j,k,l перемещают курсор, x удаляет символ, dd удаляет строку, i переходит в режим вставки и т.д. Это отличает Vim от модельных (modeless) редакторов вроде Kate или MCEDIT.<br><br>**Задание 5.4**<br>Какую функциональную клавишу в редакторе MCEDIT (и вообще в Midnight Commander) нужно нажать, чтобы вызвать меню с операциями над файлом (копирование, переименование, права доступа и т.д.)? (Ответ: обозначение клавиши в формате «F[цифра]»)<br><br>Ответ: F9<br><br>Пояснение: В Midnight Commander (и, соответственно, в MCEDIT, когда он запущен из MC) клавиша F9 активирует главное меню в верхней части экрана. Это основной способ доступа ко всем функциям файлового менеджера и редактора, кроме самых базовых. Знание основных функциональных клавиш (F3 — просмотр, F4 — редактировать, F5 — копировать, F6 — переименовать/переместить, F10 — выход) — ключевой навык для эффективной работы в MC. |
| Тема 10 Конфигурирование сетевых интерфейсов. | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированны | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем | **Задание 10.1**<br>Какие из следующих утверждений об IP-адресах и подсетях верны?<br><br>1. IP-адрес 192.168.1.0 с маской 255.255.255.0 (/24) является адресом сети и не может быть назначен интерфейсу хоста.<br>2. Широковещательный (broadcast) адрес для сети 10.0.0.0/8 — это 10.255.255.255. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | х систем | на базовом уровне, реализовывать техническое сопровождение информационных систем | 3. Адрес 127.0.0.1 (loopback) всегда принадлежит устройству с физическим сетевым интерфейсом.<br>4. IPv6-адреса типа fe80::/10 (link-local) автоматически назначаются интерфейсу и используются для связи только в пределах одного канального сегмента (например, одной Ethernet-сети).<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. Адрес сети (первый адрес в подсети) зарезервирован для идентификации самой сети и не используется для узлов.<br>2 - Правильно. Для подсети /8 адрес сети 10.0.0.0, широковещательный адрес — 10.255.255.255 (все биты хостовой части равны 1).<br>4 - Правильно. Link-local адреса (fe80::/10) генерируются автоматически (например, из MAC-адреса) и не маршрутизируются за пределы сегмента L2.<br><br>**Задание 10.2**<br>Какие из следующих утверждений о конфигурировании сетевых интерфейсов и маршрутизации в Linux верны?<br><br>1. Команда ip addr add 192.168.1.10/24 dev eth0 добавит указанный IP-адрес и маску к интерфейсу eth0.<br>2. Для настройки статического маршрута по умолчанию (шлюза по умолчанию) используется команда вида ip route add default via 192.168.1.1.<br>3. Файл /etc/resolv.conf содержит конфигурацию сетевых интерфейсов (IP-адреса, маски).<br>4. После внесения изменений в конфигурационные файлы сетевых интерфейсов в /etc/network/interfaces (Debian/Ubuntu) для применения |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | настроек необходимо перезагрузить компьютер.<br><br>Ответ: 1, 2<br><br>Пояснение:<br>1 - Правильно. Это современный синтаксис утилиты ip из пакета iproute2 для настройки адреса.<br>2 - Правильно. Команда для добавления маршрута по умолчанию через шлюз 192.168.1.1. Аналог старой команды route add default gw 192.168.1.1.<br><br>**Задание 10.3**<br>Какая утилита командной строки в современных дистрибутивах Linux (из пакета iproute2) является основным инструментом для отображения и изменения параметров сетевых интерфейсов, адресов, AR-кэша и маршрутов, заменяя собой устаревший набор ifconfig, route, arp? (Ответ: название команды из двух букв)<br><br>Ответ: ip<br><br>Пояснение: Утилита ip — это универсальный инструмент управления сетью в Linux. Её подкоманды: ip link (интерфейсы), ip addr (адреса), ip route (маршруты), ip neigh (ARP/NDP). Она предоставляет более полный доступ к функциям ядра, чем устаревшие (net-tools) аналоги, и является стандартом де-факто.<br><br>**Задание 10.4**<br>Какой тип DNS-записи (Resource Record) используется для прямого преобразования доменного имени (например, server.example.com) в IPv4-адрес? (Ответ: обозначени |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | е типа записи одной заглавной буквой)<br><br>Ответ: A (Address record)<br><br>Пояснение: Запись типа A (Address) связывает имя хоста с IPv4-адресом. Для IPv6 используется запись AAAA (или «quad-A»). Это фундаментальные записи для разрешения имен. Другие важные типы: MX (для почтовых серверов), CNAME (алиас), NS (указывает на DNS-серверы зоны). |
| Тема 16 Интерфейс пользователя. Операционное окружение | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированны х систем | ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | **Задание 16.1**<br>Какие из следующих утверждений о программном интерфейсе (API) операционной системы являются корректным? Выберите верные.<br><br>1. API — это набор функций, структур данных и констант, предоставляемых ОС для разработки прикладных программ.<br>2. Основное назначение API — скрыть детали реализации аппаратных и системных вызовов, обеспечивая переносимость и упрощение разработки.<br>3. Программный интерфейс (API) и интерфейс системных вызовов (System Call Interface) — это одно и то же.<br>4. Примером API операционной системы является Win32 API для Windows или POSIX API для UNIX-подобных систем.<br>5. API относится исключительно к графическим библиотекам для создания оконных интерфейсов (например, Qt, GTK).<br><br>Ответ: 1, 2, 4.<br><br>Пояснение:<br><br>1 — Это классическое определение API.<br><br>2 — Это ключевое назначение абстракции, которую предоставляет API. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | 4 — Это конкретные, широко известные примеры API операционных систем.<br><br>**Задание 16.2**<br>Какие из перечисленных компонентов обычно входят в состав операционного окружения (operating environment) или оболочки (shell) ОС, предоставляя пользователю и программам стандартные сервисные функции? Выберите верные.<br><br>1. Командный интерпретатор (bash, cmd.exe, PowerShell).<br>2. Библиотеки времени выполнения (C runtime library, .NET Framework, Java Runtime Environment).<br>3. Утилиты для работы с файловой системой (cp, mv, dir, find).<br>4. Ядро операционной системы и его драйверы устройств.<br>5. Текстовые редакторы, компиляторы и отладчики.<br><br>Ответ: 1, 3.<br><br>Пояснение:<br>1 — Командный интерпретатор — центральный элемент операционного окружения, обеспечивающий прямой диалог пользователя с ОС.<br>3 — Стандартный набор утилит для управления файлами и данными — неотъемлемая часть любого операционного окружения (как CLI, так и GUI).<br><br>**Задание 16.3**<br>Как называется основной сервисный механизм в составе операционного окружения UNIX-подобных систем, который позволяет перенаправлять |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | вывод одной программы на ввод другой, создавая конвейеры (pipes) для обработки данных? Укажите термин, а также символ, которым он обозначается в командной строке.<br><br>Ответ: Конвейер (pipe). Символ: \| (вертикальная черта).<br><br>Пояснение: Конвейер — это фундаментальная абстракция и сервис, предоставляемый командной оболочкой (shell). Он позволяет соединять стандартный вывод (stdout) одной программы со стандартным вводом (stdin) другой, создавая мощные и гибкие цепочки обработки данных (например, ls -l \| grep ".txt" \| sort). Это яркий пример того, как операционное окружение предоставляет больше, чем просто сумму отдельных утилит.<br><br>**Задание 16.4**<br>Вопрос: Как называется специальный тип переменных, управляемый операционным окружением и доступный всем процессам пользовательской сессии, который хранит информацию о настройках системы и пользователя, такие как путь к домашнему каталогу (HOME), список директорий для поиска программ (PATH) или текущая локаль (LANG)? Укажите общепринятый англоязычный термин.<br><br>Ответ: Environment variables (Переменные окружения).<br><br>Пояснение: Переменные окружения — это ключевой механизм настройки и передачи информации о среде выполнения (операционном окружении) от родительских процессов дочерним. Они являются частью контекста каждого процесса и широко используются как системой, так и прикладными программами для адаптации своего поведения. Это неотъемлемая часть понятия «операционное окружение». |
| Тема 20 | ОПК-5Способен | ИОПК-5.2. Уметь | **Задание 20.1** |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| Защищенность и отказоустойчивость операционных систем | инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | выполнять параметрическую настройку информационных и автоматизированных систем<br>на базовом уровне, реализовывать техническое сопровождение информационных систем | Какая из перечисленных технологий хранения данных является наиболее отказоустойчивой к одновременному отказу двух дисков в массиве без потери данных и доступности?<br><br>1. RAID 0 (чередование, striping) — повышает производительность, но не отказоустойчивость.<br>2. RAID 1 (зеркалирование, mirroring) — полная копия данных на двух дисках.<br>3. RAID 5 (чередование с контролем четности) — распределенная четность, позволяет пережить отказ одного диска.<br>4. RAID 6 (чередование с двойной четностью) — позволяет пережить одновременный отказ любых двух дисков в массиве.<br>5. JBOD (Just a Bunch Of Disks) — простое объединение дисков в один том.<br><br>Ответ: 4 (RAID 6).<br><br>Пояснение: RAID 6 использует две независимые схемы контроля четности (например, Reed-Solomon коды), что позволяет восстановить данные при отказе любых двух дисков в массиве одновременно. RAID 1 выдержит отказ одного из двух дисков, но при отказе обоих данные будут потеряны. RAID 5 выдерживает отказ только одного диска. RAID 0 и JBOD не обеспечивают отказоустойчивости вообще. Для требовательных сред, где важна доступность данных даже при множественных сбоях, RAID 6 является стандартным выбором.<br><br>**Задание 20.2**<br>В чем заключается ключевая разница между дискреционным (Discretionary Access Control - DAC) и мандатным (Mandatory Access Control - MAC) управлением доступом в операционных системах? |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | 1. DAC основан на идентификаторе пользователя (UID), а MAC — на ролях (Roles).<br>2. В DAC владелец объекта (файла) сам определяет политику доступа к нему. В MAC политика доступа централизованно задается администратором безопасности и не может быть изменена пользователями.<br>3. DAC реализуется через списки контроля доступа (ACL), а MAC — через матрицу доступа.<br>4. DAC менее производителен, чем MAC, из-за необходимости проверять сложные ACL для каждого запроса.<br><br>Ответ: 2.<br><br>Пояснение: Это фундаментальное концептуальное различие. В DAC (как в классических UNIX правах или Windows ACL) владелец файла имеет полный контроль над тем, кто и что может с ним делать. В MAC (SELinux, AppArmor) пользователи и процессы (субъекты) получают метки безопасности, объекты (файлы, сокеты) также имеют метки. Правила доступа между метками определяются глобальной политикой, которую устанавливает администратор. Пользователь, даже будучи владельцем файла, не может изменить эти правила — отсюда и название «мандатное» (обязательное). Варианты 1 и 3 описывают частные реализации, но не ключевую разницу. Вариант 4 неверен — производительность зависит от конкретной реализации.<br><br>**Задание 20.3**<br>Как называется основной механизм в современных файловых системах (ZFS, Btrfs), который обеспечивает высокую целостность данных и восстанавливаемость за счет того, что при любом обновлении данных всегда записывается новый блок, а старый блок остается неизменным до |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовой вариант контрольной работы |
|---|---|---|---|
| | | | освобождения? Укажите общепринятый англоязычный термин.<br><br>Ответ: Copy-on-Write (CoW) / Копирование при записи.<br><br>Пояснение: Механизм Copy-on-Write является краеугольным камнем для обеспечения целостности. Он предотвращает ситуации, когда сбой питания может привести к частичной записи блока, повреждающей старые данные. Поскольку новые данные пишутся в новое место, предыдущее состояние файловой системы (или снапшот) остается неизменным и согласованным. Это позволяет легко откатить изменения или восстановить данные после сбоя. CoW также лежит в основе эффективных снапшотов.<br><br>**Задание 20.4**<br>Как называется компонент подсистемы безопасности ядра Linux, который отвечает за регистрацию (логирование) всех значимых событий безопасности, таких как успешные и неуспешные попытки входа, использование привилегий, изменения политик доступа? Укажите его точное название.<br><br>Ответ: Audit daemon (auditd) и подсистема аудита ядра Linux (Linux Audit Subsystem).<br><br>Пояснение: Подсистема аудита Linux (во главе с демоном auditd) является ключевым механизмом аудита (accountability). Она собирает события от ядра, связанные с безопасностью, в защищенный от модификации журнал (/var/log/audit/audit.log). Эти логи критически важны для расследования инцидентов, соответствия стандартам (например, PCI DSS) и анализа действий в системе. Без эффективного аудита даже самые строгие механизмы аутентификации и авторизации не обеспечивают полной подотчетности. |

***Критерии оценивания контрольной работы:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи, формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, обосновывает свои суждения и даёт<br>правильные ответы на вопросы преподавателя |
| **хорошо** | Выставляется, если обучающийся умеет увязывать теорию с практикой (решает задачи и формулирует выводы, умеет пояснить полученные результаты), владеет понятийным аппаратом, полно и глубоко овладел материалом по заданной теме, но содержание ответов имеют некоторые неточности и требуют уточнения и комментария со стороны преподавателя. |
| **удовлетворительно** | Выставляется, если обучающийся знает и понимает материал по заданной теме, но изложение неполное, непоследовательное, допускаются неточности в определении понятий, студент не может обосновать свои ответы на уточняющие вопросы преподавателя |
| **неудовлетворительно** | Выставляется, если обучающийся допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал. Делает ошибки в ответах на уточняющие вопросы преподавателя |

## 4.4. ТИПОВЫЕ ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Самостоятельная работа включает в себя проработку теоретического материала, изучение рекомендуемой литературы, выполнение практико-ориентированных заданий (заполнение таблиц, проведение сравнительного анализа, составление схем и др.), решение практических задач, создание презентаций, написание рефератов, подборка нормативного и иного материала и выполнение других заданий.

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| Тема 3. Архиваторы | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | Подготовить доклад на одну из 3 тем<br><br>**Тема 1** “Опции архиваторов tar”<br>**Тема 2** “Опции архиваторов zip,<br>**Тема 3** “Опции архиваторов bzip2”<br><br>tar, gzip и bzip2 — утилит для работы с архивами и сжатия файлов. Ниже приведены примеры опций для каждой из них.<br><br>**Тема 1 “Опции архиваторов tar”**<br><br>Некоторые опции утилиты tar:<br>-c, --create — создаёт новый архив и записывает в него файлы.<br>-t, --list — выводит содержимое архива.<br>-x, --extract, --get — извлекает файлы из архива.<br>-v, --verbose — выводит подробный список обработанных файлов.<br>-f, --file=ARCHIVE — в качестве имени файла архива использует ARCHIVE.<br>-C, --directory — позволяет задать каталог, в который нужно поместить извлекаемые из архива файлы.<br>Важно: tar только создаёт архив, но не сжимает файлы — для сжатия используются внешние утилиты.<br><br>**Тема 2 “Опции архиваторов zip,**<br>Некоторые опции утилиты gzip:<br>-c, --stdout — печатать в стандартный вывод, сохранять исходные файлы без изменений.<br>-d, --decompress — распаковать архив.<br>-k, --keep — сохранить исходный файл. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| | | | -l, --list — отобразить содержимое архива.<br>-t, --test — проверить целостность архива.<br>-1 — -9 — установить степень сжатия (чем больше число, тем выше степень сжатия).<br>-r, --recursive — рекурсивная работа с каталогом.<br>-h, --help — вывести справку по использованию утилиты.<br><br>**Тема 3 “Опции архиваторов bzip2”**<br>Некоторые опции утилиты bzip2:<br>-d, --decompress — принудительная распаковка.<br>-z, --compress — принудительная компрессия, если требуется архивация уже сжатого файла.<br>-t, --test — проверить целостность указанных файлов, но не распаковывать их.<br>-f, --force — принудительная перезапись выходных файлов.<br>-k, --keep — сохранить входные файлы при сжатии или распаковке.<br>-s, --small — уменьшить использование памяти (для систем с ограниченными ресурсами).<br>-q, --quiet — подавить несущественные сообщения.<br>-v, --verbose — подробный режим — показывает степень сжатия для каждого обрабатываемого файла.<br><br>Важно: утилита bzip2 может работать только с одиночными файлами — для работы с несколькими файлами или папками можно использовать эту утилиту вместе с tar |
| Тема 9. Элементы | ОПК-5Способен инсталлировать | ИОПК-5.1. Знать основы системного администрирования, | Подготовьте реферат на тему<br><br>**Вариант 1 “Истоия Ubuntu”** |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| администрирования. | программное и аппаратное обеспечение для информационных и автоматизированных систем | администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | Что может быть в реферате<br><br>История<br>Описание Ubuntu как бесплатного дистрибутива ОС с открытым исходным кодом. Система разработана компанией Canonical на основе Linux и Unix-подобной операционной системы Debian. otus.ru<br>Первая версия — Ubuntu 4.10 «Warty Warthog» — вышла в открытом доступе в 2004 году. Автор проекта — Марк Шаттлворт, который хотел создать доступную для конечного пользователя настольную операционную систему на базе ядра Linux.<br>Влияние Ubuntu на экосистему свободного ПО: с её выходом операционную систему на Linux стало проще установить и использовать на домашних и рабочих компьютерах.<br><br>**Вариант 2 “Debian. Структура, репозиторий, область применения”**<br><br>Структура<br>Многоуровневая структура. Компоненты системы организованы по иерархическим уровням: аппаратное обеспечение, ядро, системные утилиты и приложения.<br>Основная система управления пакетами — APT (Advanced Package Tool). Она позволяет устанавливать, обновлять и удалять программное обеспечение, обеспечивает |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| | | | автоматическую установку зависимостей и контроль версий.<br>Формат упаковки — пакеты Debian используют формат .deb, который содержит исполняемые файлы, файлы конфигурации, документацию и метаданные. Каждый пакет содержит информацию о зависимостях, номера версий и сценарии установки.<br><br>Репозитории<br>В основе системы лежит концепция репозиториев — специальных хранилищ, из которых пользователи могут устанавливать и обновлять программы.<br>Некоторые типы официальных репозиториев в Debian:<br>Stable — здесь хранятся рекомендованные версии программ, которые попадают сюда из ветки Testing после проверки и доработки.<br>Testing — в этой ветке находятся разработанные программы, которые не прошли достаточно тестов, чтобы попасть в ветку Stable.<br>Unstable (sid) — здесь находятся самые свежие версии ПО, для которых не выпускаются обновления.<br>Experimental — этот репозиторий содержит программы, которые находятся на начальной стадии разработки.<br>Backports — компромиссный вариант между ветками Stable и Testing, включает в себя большое количество пакетов из Testing и некоторые из Unstable.<br><br>Применение<br>Для серверов — Debian можно встретить на веб-серверах, почтовых серверах, файлохранилищах и других критически важных системах. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| | | | Для разработчиков — Debian поддерживает множество языков программирования, библиотек и инструментов, в этой системе удобно собирать и тестировать проекты.<br>Для обучения — из-за своей открытости и гибкости Debian часто используют в образовательных учреждениях.<br>Для восстановления и диагностики — Debian подойдёт как инструмент для восстановления данных, тестирования оборудования или устранения неполадок на других системах.<br>Для персонального использования — Debian можно установить как обычную настольную систему, он поддерживает различные графические окружения, такие как GNOME, KDE, Xfce и другие. |
| Тема 14 Web-сервер Apache. | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | Изучите самостоятельно тему “Настройка виртуальных хостов со статическим и динамическим содержанием в Web-сервере Apache2” и подготовьте краткий доклад по изученному материалу.<br><br>Что может быть отражено:<br><br>Виртуальные хосты — это механизм, который связывает доменное имя сайта с директорией, в которой находятся нужные файлы. Веб-сервер сравнивает домен, переданный в заголовке «Host», с доступными виртуальными доменами и, если находит совпадение, возвращает содержимое настроенной папки или содержимое по умолчанию, или ошибку 404.<br><br>Статическое содержание<br>Создание конфигурационных файлов для виртуальных хостов. По умолчанию такой файл — 000-default.conf, но для каждого |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| | | | виртуального хоста необходимо скопировать его, задать произвольное имя (рекомендуется использовать имя домена) и произвести определённые настройки. В файле можно указать, например:<br>ServerName — домен; losst.proreg.cloud<br>ServerAlias — псевдоним домена; losst.proreg.cloud<br>DocumentRoot — каталог домена, созданный ранее; nic.rureg.cloud<br>ErrorLog и CustomLog — файлы логов, в названии которых тоже должен содержаться домен. nic.rureg.cloud<br>Активацию файлов виртуального хоста — чтобы Apache использовал новые файлы, их нужно активировать, а файл, использующийся по умолчанию, — отключить. Изменения вступят в силу только после перезапуска веб-сервера. jino.ru<br>Динамическое содержание<br>Динамическую настройку виртуальных хостов — механизм, который автоматически вставляет информацию о виртуальном хосте в путь к файлу, используемому для удовлетворения запроса. Это делается с помощью модулей mod_vhost_alias или mod_rewrite. По умолчанию эти модули отключены, их нужно включить при настройке и построении Apache.<br>Настройка единого виртуального хоста, который может динамически обрабатывать все запросы на основе имени хоста, используемого для доступа к нему. Например, можно настроить виртуальный хост, который включает имя хоста в имена файлов, используемых для удовлетворения запросов.<br>Контроль scope различных конфигураций виртуальных хостов — в сложных настройках можно использовать директивы Apache, чтобы контролировать область конфигураций. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| Тема 18. Работа с файлами | ОПК-5Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | Самостоятельно изучите тему<br>«Примеры файловых систем» и подготовьте краткий доклад.<br><br>Что может быть отражено:<br><br>Определение файловой системы.<br>Это метод организации, хранения и управления данными на носителе информации (жёсткий диск, SSD, флеш-накопитель и т. д.).Основные функции файловой системы. К ним относятся управление файлами и каталогами, обеспечение структуры хранения, контроль доступа к файлам, защита данных от потери и оптимизация использования памяти.<br>Типы файловых систем. Некоторые из них и их особенности:<br>FAT32. Поддерживает старые устройства, имеет ограничение размера файлов (до 4 ГБ). Используется в Windows и Linux.<br>NTFS. Поддерживает права доступа, шифрование, журналирование. Используется в Windows. ext4.<br>Журналируемая, имеет высокую скорость работы, поддерживает большие файлы. Применяется в Linux.<br>HFS+. Оптимизирована для работы с macOS. APFS.<br>Обеспечивает высокую скорость, шифрование, моментальные снимки файловой системы. Используется в macOS.<br>Другие файловые системы. К ним относятся, например:<br>JFS. Стабильная и надёжная файловая система, которая потребляет минимум мощностей.<br>XFS. Быстро работает с крупными файлами и легко расширяет разделы, но более чувствительна к потере данных.<br>ReiserFS. Существует только для Linux, в ней удобно работать с данными, но можно потерять информацию при аварийном отключении. |

| Тема | Код и формулировка компетенции | Индикаторы достижения компетенции | Типовое задание для самостоятельной работы |
|---|---|---|---|
| | | | Btrfs. Удобная в администрировании система для OpenSUSE и SUSE Linux.<br>Кластерные файловые системы. К ним относятся, например: ZFS, Apple Xsan, VMFS, GFS и JFS1. timeweb.com<br>Также можно рассмотреть применение файловых систем в различных операционных системах, что позволит понять их реальные возможности и ограничения |

***Критерии оценивания самостоятельной работы:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | выставляется если работа носит научно-исследовательский характер, проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры |
| **хорошо** | выставляется если проанализирован и сделан сравнительный анализ нескольких литературных источников, приведены примеры |
| **удовлетворительно** | выставляется если проведен сравнительный анализ научно-методической литературы, приведены примеры |
| **неудовлетворительно** | выставляется если работа прошла проверку на антиплагиат и соответствует требованиям оформления |

## 4.5. ИНДИВИДУАЛЬНЫЕ ЗАДАНИЯ ДЛЯ КУРСОВЫХ РАБОТ

**Курсовые работы не предусмотрены**

Курсовая работа позволяет выявить степень владения базовыми знаниями, умениями и навыками, необходимыми для обучения, и определить уровень владения новым материалом.

Курсовая работа не предусмотрена учебным планом

***Критерии оценивания курсовой работы:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. Все аналитические расчеты выполнены верно, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Все выводы и предложения убедительно аргументированы. При защите курсовой работы обучающийся правильно и уверенно отвечает на вопросы преподавателя, демонстрирует глубокое знание теоретического материала, способен аргументировать собственные утверждения и выводы |
| **хорошо** | Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Представлены результаты структурированного и логически последовательного обзора литературных и иных источников по теме исследования. Структура курсовой работы логически и методически выдержана. Верно определены исходные данные для расчетов. В расчетах допускаются незначительные (не искажающие общего итога оценки) погрешности/ошибки. Большинство выводов и предложений аргументировано, корректно применены методы экономического анализа, не нарушена методика анализа предмета исследования. Оформление курсовой работы и полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две несущественные ошибки в использовании терминов, в построенных диаграммах и схемах, в |

| | |
|---|---|
| | оформлении таблиц. Наличествует незначительное количество грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся правильно и уверенно отвечает на большинство вопросов преподавателя, демонстрирует хорошее знание теоретического материала, но не всегда способен аргументировать собственные утверждения и выводы. При наводящих вопросах преподавателя исправляет ошибки в ответе |
| **удовлетворительно** | Содержание курсовой работы полностью соответствует заданию, содержащемуся в методических указаниях, и плану. Результаты обзора литературных и иных источников представлены недостаточно полно, недостаточно логично и последовательно. Верно определены исходные данные для расчетов, но имеются грубые ошибки в расчетах. Аргументация выводов и предложений слабая или отсутствует. Экономические выводы носят констатирующий (описательный) характер. Имеются одно-два существенных отклонений от требований в оформлении курсовой работы. Полученные результаты в целом отвечают требованиям, изложенным в методических указаниях. Имеются одна-две существенных ошибки в использовании терминов, в построенных диаграммах и схемах. Много грамматических и/или стилистических ошибок. При защите курсовой работы обучающийся допускает грубые ошибки при ответах на вопросы преподавателя, демонстрирует слабое знание теоретического материала, в большинстве случаев не способен уверенно аргументировать собственные утверждения и выводы |
| **неудовлетворительно** | Содержание курсовой работы не соответствует заданию, содержащемуся в методических указаниях, и плану. Неверно определены исходные данные для расчетов, неверно и не корректно применены методы экономического анализа. Экономические выводы содержат неверную экономическую оценку. Имеются более двух существенных отклонений от требований в оформлении курсовой работы. Большое количество существенных ошибок по сути работы, много грамматических и стилистических ошибок и др. Полученные результаты не отвечают требованиям, изложенным в методических указаниях. При защите курсовой работы обучающийся демонстрирует слабое понимание программного материала, студент не может защитить свои решения, допускает грубые фактические ошибки при ответах на поставленные вопросы или вовсе не отвечает на них. Курсовая работа не представлена преподавателю. Обучающийся не явился на защиту курсовой работы |

## 5. МАТЕРИАЛЫ ДЛЯ ОЦЕНКИ РЕЗУЛЬТАТОВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Промежуточная аттестация проводится в форме зачета/зачета с оценкой/экзамена.

**5.1. Вопросы к зачету с оценкой:**

1. Сколько первичных разделов можно создать на диске со схемой разбиения MBR (Master Boot Record)?

2. Как в Linux обозначается первый раздел на втором SATA или SCSI диске?
3. Какая команда используется для создания файловой системы ext4 на разделе /dev/sda1?

4. Что такое монтирование (mounting) файловой системы?
5. Какая команда монтирует все файловые системы, описанные в файле /etc/fstab?
6. Какая журналируемая файловая система является стандартом де-факто для современных дистрибутивов Linux?
7. Почему перед физическим извлечением флешки необходимо выполнить ее размонтирование?
8. Какой параметр в файле /etc/fstab указывает, что файловую систему нужно проверять при загрузке с помощью fsck?
9. Какой каталог в иерархии FHS (Filesystem Hierarchy Standard) предназначен для хранения исполняемых программ, доступных всем пользователям?
10. Какой символ в первом столбце вывода ls -l обозначает обычный файл?
11. На какие три основные категории делятся пользователи при проверке прав доступа к файлу в Linux?
12. Что означает право на выполнение (execute, x) для каталога?
13. Какой специальный бит прав доступа, установленный на исполняемый файл, заставляет процесс запускаться с правами владельца этого файла, а не того пользователя, который его запустил?
14. Каково числовое (абсолютное) представление прав rwxr-xr--?
15. Приведите пример команды chmod, которая с помощью текстового режима добавит право на запись для группы к файлу report.txt.
16. Какой командой сменить владельца файла data.log на пользователя www-data и одновременно его группу на www-data?
17. Какая команда копирует файл source.txt в каталог /backup/ с сохранением всех атрибутов (владельца, прав, временных меток)?
18. Чем отличается абсолютный (полный) путь от относительного?
19. Какая команда создаст сжатый архив (с использованием gzip) с именем backup.tar.gz, содержащий каталог /home/user/docs/?
20. Какой командой можно вывести последние 10 строк файла server.log?
21. Какой символ в регулярном выражении обозначает "любой одиночный символ"?
22. Какой командой с помощью sed заменить все вхождения слова "error" на "warning" в файле messages.txt?
23. Как выглядит регулярное выражение, которое соответствует пустой строке?
24. Какая команда awk выведет первый столбец из каждой строки файла data.csv, где разделителем является запятая?
25. Как присвоить переменной MYVAR строковое значение "Hello World"?
26. Какая специальная переменная bash содержит количество аргументов, переданных сценарию?
27. Какой оператор перенаправит стандартный вывод (stdout) и стандартный вывод ошибок (stderr) команды в один файл log.txt?
28. Какой escape-код используется в ANSI-последовательностях для сброса всех атрибутов (цвета, яркости) текста в терминале?
29. В какую встроенную переменную bash по умолчанию помещается ввод пользователя при использовании команды read без указания переменной?

30. Как вычислить результат выражения 5 + 3 и сохранить его в переменную RES с использованием конструкции $((...))?
31. Какой оператор в условии test (или в квадратных скобках [ ]) проверяет, что файл /etc/passwd существует и является обычным файлом?
32. Какой ключевое слово завершает условную конструкцию if в bash?
33. Цикл for i in 1 2 3; do ... выполнит тело цикла сколько раз?
34. Как объявить функцию с именем greet в bash-скрипте?
35. Какая встроенная утилита bash предназначена для корректного разбора опций командной строки (например, -a -f file)?
36. Какой оператор (символ) используется для выполнения второй команды только в случае успешного завершения (с кодом 0) первой команды?
37. Каковы числовые дескрипторы стандартных потоков: ввода (stdin), вывода (stdout) и ошибок (stderr)?
38. Что обозначает маска подсети /24 в нотации CIDR?
39. Какой командой ifconfig отключить (деактивировать) сетевой интерфейс eth0?
40. Какой демон или команда используется в большинстве современных дистрибутивов Linux для автоматического получения IP-адреса по DHCP?
41. Какой командой добавить статический маршрут к сети 192.168.2.0/24 через шлюз 192.168.1.1?
42. Каковы три основные функции (действия) сетевого экрана (брандмауэра) по отношению к проходящему через него трафику?
43. Какая стандартная цепочка таблицы filter в iptables обрабатывает пакеты, предназначенные локальному хосту?
44. Какое состояние (state) в iptables соответствует пакету, который является частью уже установленного соединения?
45. Какой критерий (ключ) в команде iptables указывает на протокол (например, TCP или UDP)?
46. Какая команда iptables покажет все правила в цепочке INPUT таблицы filter с номерами строк?
47. Какой командой удалить правило номер 3 из цепочки INPUT?

***Критерии оценивания зачета с оценкой/экзамена:***

| Оценка | Критерии оценивания |
|---|---|
| **зачтено** | Обучающийся должен: уметь строить ответ в соответствии со структурой излагаемого вопроса; продемонстрировать прочное, достаточно полное усвоение знаний программного материала; продемонстрировать знание основных теоретических понятий; правильно формулировать определения; последовательно, грамотно и логически стройно изложить теоретический материал; продемонстрировать умения самостоятельной работы с литературой; уметь сделать достаточно обоснованные выводы по излагаемому материалу. |
| **не зачтено** | Обучающийся демонстрирует: незнание значительной части программного материала; не владение понятийным аппаратом дисциплины; существенные ошибки при изложении учебного материала; неумение строить ответ в |

| | |
|---|---|
| | соответствии со структурой излагаемого вопроса; неумение делать выводы по излагаемому материалу. |

***Критерии оценивания зачета с оценкой/экзамена:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | Выставляется, если обучающимся правильно и полностью раскрыто содержание материала в пределах программы, чётко и правильно даны определения и раскрыто содержание понятий, точно использованы научные и технические термины, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения |
| **хорошо** | Выставляется, если обучающимся раскрыто основное содержание материала в пределах программы, даны определения и раскрыто содержание понятий, в ответе использованы ранее приобретённые теоретические знания, сделаны необходимые выводы и обобщения, но присутствуют незначительные нарушения в последовательности изложения, имеются одна-две неточности в содержании ответа |
| **удовлетворительно** | Выставляется, если обучающимся содержание учебного материала изложено фрагментарно, не всегда последовательно, не даны определения, не раскрыто содержание понятий, или они изложены с ошибками, допускаются ошибки и неточности в использовании научной терминологии, отсутствуют выводы и обобщения из предыдущего материала, или возможны ошибки в их изложении |
| **неудовлетворительно** | Выставляется, если обучающимся основное содержание учебного материала не раскрыто, не даются ответы на основные вопросы, допускаются грубые ошибки в определении понятий, в использовании терминологии, отсутствуют выводы и обобщения |

## 0. МАТЕРИАЛЫ ДЛЯ ДИАГНОСТИЧЕСКОЙ РАБОТЫ

Задания для диагностической работы должны обеспечивать оценку полностью или частично сформированных компетенций. Каждое задание должно быть привязано к тому или иному индикатору сформированности компетенций.

***При формировании заданий для диагностической работы необходимо использовать тестовые задания следующих типов***:

Тип задания 1. Задания закрытого типа на установление соответствия.

Тип задания 2. Задания закрытого типа на установление последовательности.

Тип задания 3. Задания комбинированного типа, предполагающие выбор одного правильного ответа из предложенных.

Тип задания 4. Задания комбинированного типа, предполагающие выбор нескольких ответов из предложенных.

Тип задания 5. Задания открытого типа с развернутым ответом.

**Все типы заданий должны быть представлены не менее одного раза.**

| № п/п | Тема занятия | Код компетенции | Индикатор | Тип задания | Задание | | Уровень освоения компетенции |
|---|---|---|---|---|---|---|---|
| | | | | | Вариант 1 | Вариант 2 | |
| 1 | Тема 1 Введение в ОС Linux. Понятие операционной системы. Лицензии на ПО | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение | 4,4 | **1.**<br>Какие из следующих утверждений точно описывают понятие «операционная система» (ОС) в современной информатике?<br><br>1. ОС — это комплекс программ, который обеспечивает управление аппаратными ресурсами компьютера (процессором, памятью, устройствами ввода-вывода).<br>2. Основная функция ОС — предоставлять пользователю или прикладным программам удобный интерфейс для работы с вычислительной системой.<br>3. ОС всегда включает в себя графический пользовательский интерфейс (GUI), такой как рабочий стол с окнами.<br>4. ОС выполняет все прикладные задачи пользователя, такие как редактирование текстов или просмотр веб-страниц.<br><br>Ответ: 1, 2 | **1.**<br>Какие из следующих положений являются ключевыми принципами лицензии GNU GPL (General Public License), под которой распространяется ядро Linux и большая часть системного ПО в дистрибутивах?<br><br>1. Свобода запуска программы с любой целью.<br>2. Свобода изучения того, как программа работает, и адаптация её под свои нужды (доступ к исходному коду — обязательное условие).<br>3. Свобода распространять копии программы, но только в её исходной, неизменной форме.<br>4. Свобода улучшать программу и публиковать улучшения, но только для некоммерческого использования.<br><br>Ответ: 1, 2<br><br>Пояснение:<br>1 - Правильно. Это первая | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | информационных систем | | Пояснение:<br>1 - Правильно. Это ядро определения ОС: управление ресурсами (resource management).<br>2 - Правильно. Это вторая ключевая функция ОС: предоставление интерфейса (abstraction layer) между «железом» и пользователем/программами. | свобода из четырёх, определяемых Фондом свободного ПО (FSF).<br>2 - Правильно. Это вторая свобода. Доступ к исходному коду — фундаментальное требование GPL для обеспечения возможности модификации | |
| 2 | Тема 2 Атрибуты файлов. Работа с файлами | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и | 2,2 | **2.**<br>Установите правильную последовательность проверки прав доступа к файлу для выполнения операции (например, чтения) в Linux, когда процесс пытается получить доступ к файлу.<br><br>1. Проверка прав доступа для «остальных» (others).<br>2. Проверка, является ли эффективный UID процесса владельцем файла.<br>3. Проверка, входит ли эффективный GID процесса или одна из дополнительных групп процесса в группу-владельца файла.<br>4. Проверка, является ли процесс суперпользователем (root, | **2.**<br>Установите последовательность действий для безопасного копирования файла с сохранением большинства его атрибутов (времени модификации, прав доступа), но без копирования специальных битов (SUID, SGID).<br><br>Действия:<br>1. Скопировать содержимое файла: cat source > destination<br>2. Установить на целевом файле права доступа, скопированные с исходного: chmod --reference=source destination<br>3. Скопировать время модификации исходного файла на целевой: touch -r source destination | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | UID=0).<br><br>Ответ: 4231<br><br>Пояснение: Ядро проверяет права по иерархии от наиболее привилегированных к наименее. Сначала проверяется суперпользователь (root обходит все проверки прав). Если процесс не root, проверяется, является ли он владельцем файла. Если нет, проверяется членство в группе-владельце. И только если все предыдущие проверки не прошли, применяются права для «остальных». | 4. Удалить возможные SUID/SGID биты с целевого файла: chmod u-s,g-s destination<br><br>Ответ: 1234<br>Пояснение: Сначала создается файл-назначение с содержимым (1). Затем устанавливаются базовые права (rwx) (2). Далее синхронизируется временная метка (3). Важный шаг безопасности — явное снятие SUID/SGID битов (4), чтобы случайно не создать уязвимость. Шаг 4 можно выполнить сразу после копирования содержимого, до установки прав. | |
| 3 | Тема 3 Архиваторы | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь | 1,1 | **3.**<br>Сопоставьте расширение архивного/сжатого файла с утилитой (или комбинацией утилит), которая используется для его создания и распаковки.<br><br>Список А (Расширение файла):<br>A) .tar.gz или .tgz<br>B) .tar.bz2 или .tbz2<br>C) .tar.xz или .txz<br>D) .tar<br><br>Список Б (Основная утилита/ключ tar для распаковки): | **3.**<br>Сопоставьте ключ команды tar с его описанием.<br><br>Список А (Ключ команды tar):<br>A) -c<br>B) -x<br>C) -t<br>D) -v<br><br>Список Б (Описание):<br>1. Распаковать (извлечь) содержимое архива.<br>2. Создать новый архив.<br>3. Вывести подробный (verbose) список обрабатываемых файлов. | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | 1. tar -xf (только распаковка архива без сжатия)<br>2. tar -xzf (распаковка + декомпрессия gzip)<br>3. tar -xjf (распаковка + декомпрессия bzip2)<br>4. tar -xJf (распаковка + декомпрессия xz)<br><br>Ответ: A2, B3, C4, D-1 | 4. Показать содержимое (оглавление) архива без его распаковки.<br><br>Ответ: A-2, B-1, C-4, D-3 | |
| 4 | Тема 4 Обработка текстовых файлов | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку | 3,3 | **4.**<br>Какая из предложенных команд sed удалит все строки в файле config.txt, которые начинаются с символа # (комментарии) или являются пустыми (содержат только пробелы/табуляции)?<br><br>1. sed '/^#/d' config.txt<br>2. sed '/^#/d; /^$/d' config.txt<br>3. sed '/^#/d' config.txt \| sed '/^$/d'<br>4. sed '/^[[:space:]]*$/d; /^#/d' config.txt<br><br>Ответ: 4<br><br>Пояснение: 4 - Правильно. Выражение ^[[:space:]]*$ соответствует | **4.**<br>Вам нужно в реальном времени отслеживать добавление новых записей в растущий лог-файл /var/log/app.log. Какая команда является наиболее подходящей?<br><br>1. tail /var/log/app.log<br>2. less /var/log/app.log<br>3. tail -f /var/log/app.log<br>4. head -f /var/log/app.log<br><br>Ответ: 3<br><br>Пояснение: 3 - Правильно. Ключ -f (follow) команды tail заставляет ее не завершаться после вывода последних строк, а продолжать работу и выводить | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | строкам, которые от начала (^) до конца ($) содержат ноль или более (*) пробельных символов ([[:space:]]), т.е. пустым или состоящим только из пробелов/табуляций. Вторая часть /^#/d удаляет строки, начинающиеся с #. Порядок может быть любым. | новые данные по мере их добавления в файл. Это стандартный способ мониторинга логов. | |
| 5 | Тема 5 Текстовые редакторы. | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизирован | 5,5 | **5.** Как называется характерный для редакторов семейства Vim режим работы, в котором нажатия клавиш интерпретируются не как ввод текста, а как команды для перемещения курсора, удаления, копирования и других действий? (Ответ: одно слово — название режима)<br><br>Ответ: Нормальный (Normal mode) или Командный (Command mode).<br><br>Пояснение: Это фундаментальное понятие в философии Vim. «Нормальный режим» — это режим по умолчанию, в котором пользователь находится большую часть времени. В этом | **5.** Какую функциональную клавишу в редакторе MCEDIT (и вообще в Midnight Commander) нужно нажать, чтобы вызвать меню с операциями над файлом (копирование, переименование, права доступа и т.д.)? (Ответ: обозначение клавиши в формате «F[цифра]»)<br><br>Ответ: F9<br><br>Пояснение: В Midnight Commander (и, соответственно, в MCEDIT, когда он запущен из MC) клавиша F9 активирует главное меню в верхней части экрана. Это основной способ доступа ко всем функциям файлового менеджера и редактора, кроме самых базовых. | **Высокий (5-10 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | ных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | режиме клавиши h,j,k,l перемещают курсор, x удаляет символ, dd удаляет строку, i переходит в режим вставки и т.д. Это отличает Vim от модельных (modeless) редакторов вроде Kate или MCEDIT. | Знание основных функциональных клавиш (F3 — просмотр, F4 — редактировать, F5 — копировать, F6 — переименовать/переместить, F10 — выход) — ключевой навык для эффективной работы в MC. | |
| 6 | Тема 6 Основы bash. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, | 2,2 | **6.**<br>Какие из следующих утверждений о стандартных потоках ввода-вывода в bash являются верными?<br><br>1. Стандартный поток вывода (stdout, дескриптор 1) по умолчано выводится в терминал.<br>2. Стандартный поток ошибок (stderr, дескриптор 2) не может быть перенаправлен отдельно от stdout.<br>3. Перенаправление 2>&1 означает, что stderr будет отправлен туда же, куда в данный момент направлен stdout.<br>4. Стандартный поток ввода (stdin, дескриптор 0) по умолчано читается с клавиатуры терминала.<br><br>Ответ: 1, 3, 4<br><br>Пояснение: | **6.**<br>Какие из следующих утверждений о специальных символах и операторах управления выполнением в bash верны?<br><br>1. Символ & в конце команды (command &) запускает ее в фоновом режиме, немедленно возвращая управление оболочке.<br>2. Оператор && (логическое И) выполняет команду справа только в том случае, если команда слева завершилась с нулевым кодом возврата (успешно).<br>3. Оператор \|\| (логическое ИЛИ) выполняет команду справа только в том случае, если команда слева завершилась с ненулевым кодом возврата (с ошибкой).<br>4. Символ ; используется для группировки команд в подобие | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | реализовывать техническое сопровождение информационных систем | | 1 - Правильно. Это поведение по умолчанию для интерактивной оболочки.<br>3 - Правильно. Это ключевой синтаксис для объединения потоков. 2> перенаправляет stderr, &1 указывает на «то же место, где сейчас находится stdout».<br>4 - Правильно. В интерактивном режиме команда, ожидающая ввода, будет ждать его от пользователя через терминал. | функции.<br><br>Ответ: 1, 2, 3<br><br>Пояснение:<br>1 - Правильно. Это определение фонового запуска. Оболочка не ждет завершения такой команды.<br>2 - Правильно. Это основа цепочек выполнения. Например, make && make install выполнит установку только если сборка прошла успешно.<br>3 - Правильно. Обратная логика. Например, command \|\| echo "Ошибка" выведет сообщение только при неудаче команды. | |
| 7 | Тема 7 Оболочка bash. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь | 1,1 | **7.**<br>Сопоставьте конструкцию раскрытия параметра (parameter expansion) в bash с ее описанием.<br><br>Список А (Конструкция):<br>A) ${VAR:-default}<br>B) ${VAR:=default}<br>C) ${VAR:?error message}<br>D) ${VAR:+alternate}<br><br>Список Б (Описание):<br>1. Если VAR не задана или пуста, используется default, и это значение присваивается переменной VAR. | **7.**<br>Сопоставьте специальную переменную/параметр bash с его значением.<br><br>Список А (Переменная/параметр):<br>A) $#<br>B) $@<br>C) $?<br>D) $$<br><br>Список Б (Значение):<br>1. Код возврата (exit status) последней выполненной команды. | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | 2. Если VAR не задана или пуста, выводится error message и выполнение скрипта прерывается.<br>3. Если VAR не задана или пуста, используется default, но сама VAR не изменяется.<br>4. Если VAR задана и не пуста, используется alternate, иначе используется пустая строка.<br><br>Ответ: A3, B1, C2, D4 | 2. Количество позиционных параметров (аргументов), переданных скрипту или функции.<br>3. PID (идентификатор процесса) текущей оболочки.<br>4. Список всех позиционных параметров, разделенных пробелами (или как отдельные слова в двойных кавычках).<br><br>Ответ: A2, B4, C1, D3 | |
| 8 | Тема 8 Shell --- как язык программирования. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку | 3,3 | **8.**<br>Какая из следующих конструкций цикла for корректно переберет все файлы с расширением .txt в текущем каталоге и выведет их имена?<br><br>1. for file in "*.txt"; do echo $file; done<br>2. for file in $(ls *.txt); do echo "$file"; done<br>3. for file in *.txt; do echo "$file"; done<br>4. for file in ls *.txt; do echo $file; done<br><br>Ответ: 3<br><br>Пояснение: | **8.**<br>Функция is_even() должна проверить, является ли переданный ей числовой аргумент четным, и вернуть 0 (истина), если да, и 1 (ложь) — если нет. Какая реализация является правильной и безопасной?<br><br>1.bash<br>is_even() {<br>  if [ $(($1 % 2)) -eq 0 ]; then<br>  return 0<br>  else<br>  return 1<br>  fi<br>} | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | 3 - Правильно. Идиоматичный способ. Подстановка пути (globbing) *.txt выполняется оболочкой до запуска цикла, генерируя список файлов. | 2. bash<br>is_even() {<br>  if [ $1 % 2 == 0 ]; then<br>  return true<br>  else<br>  return false<br>  fi<br>}<br><br>3. bash<br>is_even() {<br>  if (( $1 % 2 == 0 )); then<br>  return 0<br>  else<br>  return 1<br>  fi<br>}<br><br>4. bash<br>is_even() {<br>  result=$(( $1 % 2 ))<br>  return $result<br>}<br><br>Ответ: 3<br><br>Пояснение:<br>3 - Правильно (предпочтительно). Использует современную арифметику в двойных круглых скобках (( ... )), которая возвращает статус 0, если выражение истинно (не | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | нуль). Чисто, понятно, без лишних команд test/[. | |
| 9 | Тема 9 Элементы администрирования. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | 3,3 | **9.**<br>Какой файл является основным статическим конфигурационным файлом для автоматического монтирования файловых систем при загрузке в Linux?<br><br>1. /etc/mtab<br>2. /proc/mounts<br>3. /etc/fstab<br>4. /etc/filesystems<br><br>Ответ: 3<br><br>Пояснение: 3 - Правильно. /etc/fstab (FileSystem Table) — статический файл, который читается системными скриптами (например, systemd-fstab-generator) при загрузке для автоматического монтирования указанных разделов.<br><br>не обязателен для загрузки. | **9.**<br>В процессе установки Ubuntu вы создаете разделы вручную. Какой тип раздела и файловую систему обязательно нужно создать для корректной загрузки в режиме UEFI?<br><br>1. Раздел с файловой системой ext4, помеченный как bootable (активный), смонтированный в /.<br>2. Раздел с файловой системой FAT32 (или vfat), смонтированный в /boot/efi, с флагами boot и esp.<br>3. Раздел с файловой системой ext4, смонтированный в /boot.<br>4. Раздел типа swap.<br><br>Ответ: 2<br><br>Пояснение: Для загрузки в режиме UEFI обязателен EFI System Partition (ESP). Он должен быть отформатирован в FAT32 (или vfat), так как это стандарт, читаемый прошивкой UEFI. Обычно он монтируется в /boot/efi. В таблице разделов GPT для него устанавливается тип EF00 (в gdisk) или флаги boot и | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | esp. Раздел /boot (вариант 3) может быть отдельным, но не обязателен. Флаг bootable (вариант 1) относится к загрузке в режиме BIOS/MBR. Swap (вариант 4) | |
| 10 | Тема 10 Конфигурирование сетевых интерфейсов. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое | 5,5 | **10.**<br>Какая утилита командной строки в современных дистрибутивах Linux (из пакета iproute2) является основным инструментом для отображения и изменения параметров сетевых интерфейсов, адресов, AR-кэша и маршрутов, заменяя собой устаревший набор ifconfig, route, arp? (Ответ: название команды из двух букв)<br><br>Ответ: ip<br><br>Пояснение: Утилита ip — это универсальный инструмент управления сетью в Linux. Её подкоманды: ip link (интерфейсы), ip addr (адреса), ip route (маршруты), ip neigh (ARP/NDP). Она предоставляет более полный доступ к функциям ядра, чем устаревшие (net-tools) аналоги, и является стандартом де-факто. | **10.**<br>Какой тип DNS-записи (Resource Record) используется для прямого преобразования доменного имени (например, server.example.com) в IPv4-адрес? (Ответ: обозначение типа записи одной заглавной буквой)<br><br>Ответ: A (Address record)<br><br>Пояснение: Запись типа A (Address) связывает имя хоста с IPv4-адресом. Для IPv6 используется запись AAAA (или «quad-A»). Это фундаментальные записи для разрешения имен. Другие важные типы: MX (для почтовых серверов), CNAME (алиас), NS (указывает на DNS-серверы зоны). | **Высокий (5-10 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | сопровождение информационных систем | | | | |
| 11 | Тема 11 Сетевой экран. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | 4,4 | **11.**<br>Какие из следующих утверждений о базовых концепциях сетевого экрана (брандмауэра) на основе фильтрации пакетов верны?<br><br>1. Фильтрация может осуществляться на основе различных атрибутов пакета: IP-адресов источника и назначения, номеров портов, интерфейса, состояния соединения.<br>2. Сетевой экран, работающий на уровне приложений (Application-level gateway), не способен анализировать содержимое пакетов (payload).<br>3. Состояние «ESTABLISHED» для TCP-соединения означает, что пакет является частью уже установленного и разрешенного ранее соединения, и такие пакеты часто разрешаются правилами stateful-фаервола.<br>4. Основная задача трансляции сетевых адресов (NAT) — это повышение скорости обработки пакетов. | **11.**<br>Какие из следующих утверждений о таблицах и цепочках iptables являются верными?<br><br>1. Таблица filter используется для фильтрации пакетов и содержит стандартные цепочки INPUT, OUTPUT, FORWARD.<br>2. Таблица nat используется для трансляции адресов (NAT) и содержит цепочки PREROUTING (для DNAT) и POSTROUTING (для SNAT/Masquerade).<br>3. Цепочка INPUT обрабатывает пакеты, сгенерированные самой системой и отправляемые наружу.<br>4. Команда iptables -L по умолчанию показывает правила в таблице filter.<br><br>Ответ: 1, 2, 4<br><br>Пояснение:<br>1 - Правильно. Это основное назначение таблицы filter.<br>2 - Правильно. Это основное назначение таблицы nat. | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | Ответ: 1, 3<br><br>Пояснение:<br>1 - Правильно. Это основа фильтрации пакетов (packet filtering). Правила строятся на заголовках сетевого и транспортного уровней.<br>3 - Правильно. Stateful-инспекция (отслеживание состояния) — ключевая особенность современных фаерволов. Разрешая ответные пакеты установленных соединений (-m state --state ESTABLISHED,RELATED), мы значительно упрощаем правила и повышаем безопасность. | PREROUTING изменяет адрес назначения до маршрутизации (DNAT), POSTROUTING — адрес источника после маршрутизации (SNAT).<br>4 - Правильно. iptables -L — это короткая запись для iptables -t filter -L, выводит правила фильтрации. | |
| 12 | Тема 12 Сетевой экран (продолжение). | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. | 3,3 | **12.**<br>Какая команда iptables добавит правило, разрешающее входящие SSH-соединения (TCP порт 22) только с IP-адреса 192.168.1.50 в цепочку INPUT таблицы filter?<br><br>1. iptables -A INPUT -s 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br>2. iptables -A INPUT -d 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br>3. iptables -t nat -A INPUT -s 192.168.1.50 -p tcp --dport 22 -j | **12.**<br>На маршрутизаторе с внешним интерфейсом eth0 (IP динамический) нужно настроить MASQUERADE для выхода внутренней сети 192.168.0.0/24 в интернет. В какую цепочку и таблицу нужно добавить правило?<br><br>1. iptables -t filter -A FORWARD -s 192.168.0.0/24 -j MASQUERADE<br>2. iptables -t nat -A POSTROUTING -s 192.168.0.0/24 | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | систем | Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | ACCEPT<br>4. iptables -A FORWARD -s 192.168.1.50 -p tcp --dport 22 -j ACCEPT<br><br>Ответ: 1<br><br>Пояснение:<br>1 - Правильно. -A INPUT добавляет в конец цепочки INPUT таблицы filter (по умолчанию). -s 192.168.1.50 указывает адрес источника. -p tcp --dport 22 — протокол и порт назначения. -j ACCEPT — действие. | -o eth0 -j MASQUERADE<br>3. iptables -t nat -A PREROUTING -s 192.168.0.0/24 -j MASQUERADE<br>4. iptables -t nat -A INPUT -s 192.168.0.0/24 -j MASQUERADE<br><br>Ответ: 2<br><br>Пояснение:<br>2 - Правильно. Трансляция адреса источника (SNAT/MASQUERADE) выполняется после маршрутизации, в цепочке POSTROUTING таблицы nat. Ключ -o eth0 указывает исходящий интерфейс (опционально, но рекомендуется). MASQUERADE используется для динамических IP. | |
| 13 | Тема 13 Скрипты с Web-интерфейсом. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информацион | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного | 2,2 | **13.**<br>Установите правильную последовательность этапов обработки запроса веб-сервером Apache к bash-CGI скрипту, начиная с запроса пользователя.<br><br>1. Веб-сервер создает новое окружение для дочернего процесса, устанавливает CGI-переменные | **13.**<br>Установите правильную последовательность строк в теле bash-CGI скрипта для безопасной обработки поискового запроса из формы и вывода результатов.<br><br>1. echo "<h2>Результаты поиска для '$filtered_query':</h2>"<br>2. echo "Content-Type: text/html; charset=utf-8" | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ных и автоматизированных систем | взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | (REQUEST_METHOD, QUERY_STRING и др.).<br>2. Пользователь через браузер отправляет HTTP-запрос на адрес http://server/cgi-bin/status.cgi.<br>3. Веб-сервер отправляет полученный от скрипта HTTP-ответ обратно браузеру пользователя.<br>4. Bash-скрипт status.cgi выполняется, генерирует HTTP-заголовки и тело ответа, выводя их в стандартный поток вывода (STDOUT).<br>5. Apache, получив запрос, определяет по конфигурации, что status.cgi является CGI-скриптом.<br><br>Ответ: 2, 5, 1, 4, 3.<br><br>Пояснение: Последовательность демонстрирует принцип посредничества веб-сервера: он принимает запрос, организует выполнение внешней программы и возвращает её результат. | 3. log_entry=$(grep -i "$filtered_query" /var/log/syslog \| head -20)<br>4. echo -e "\n\n<html><body>"<br>5. filtered_query=$(echo "$query" \| sed 's/[^a-zA-Z0-9 ]//g')<br>6. query="${QUERY_STRING#*=}" # Извлекаем значение после 'q='<br>7. echo "<pre>$log_entry</pre></body></html>"<br><br>Ответ: 2, 4, 6, 5, 1, 3, 7.<br><br>Пояснение: Никогда не доверяй пользовательскому вводу. Порядок гарантирует, что любая обработка (вывод или выполнение) происходит только с очищенными данными, разделяя этапы получения, валидации и использования. | |
| 14 | Тема 14 Web-сервер Apache. | ОПК-5<br><br>Способен инсталлировать программное и аппаратное | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, | 1,1 | **14.**<br>Сопоставьте директиву Apache с ее основной функцией в контексте настройки виртуальных хостов.<br><br>Список А (Директива): | **14.**<br>Сопоставьте проблему с наиболее вероятной конфигурационной причиной в Apache.<br><br>Список А (Проблема): | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | обеспечение для информационных и автоматизированных систем | современные стандарты информационного взаимодействия систем на базовом уровне<br>ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | 1. Listen<br>2. ServerAdmin<br>3. DirectoryIndex<br>4. ErrorDocument<br><br>Список Б (Функция):<br>A. Задает порядок поиска индексных файлов (например, index.html, index.php) при запросе к директории.<br>B. Определяет, на каких IP-адресах и портах Apache будет принимать входящие соединения.<br>C. Позволяет настроить кастомную страницу или действие для определенных HTTP-кодов ошибок (404, 500).<br>D. Указывает email-адрес администратора, который может отображаться на страницах ошибок.<br><br>Ответ: 1B, 2D, 3A, 4C. | 1. При запросе site.com открывается содержимое другого сайта, размещенного на этом же сервере.<br>2. При запросе PHP-файла браузер предлагает его скачать, а не выполняет.<br>3. Apache выдает ошибку AH00558: apache2: Could not reliably determine the server's fully qualified domain name.<br>4. Страница загружается, но все стили (CSS) и скрипты (JS) не применяются (ошибка 403).<br><br>Список Б (Причина):<br>A. Не задана глобальная директива ServerName или не указан ServerName в основном виртуальном хосте.<br>B. В конфигурации виртуального хоста для site.com отсутствует или неверно указана директива ServerName, и запрос попадает в хост по умолчанию (первый в списке).<br>C. В конфигурации виртуального хоста отсутствует директива AddType или SetHandler для обработки PHP, либо не загружен модуль mod_php/mod_proxy_fcgi.<br>D. Для директорий со статическими файлами (CSS, JS) | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | в конфигурации не хватает секции <Directory> с директивами Require all granted и Options Indexes или FollowSymLinks.<br><br>Ответ: 1B, 2C, 3A, 4D. | |
| 15 | Тема 15 Общие сведения об операционных системах | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать | 3,3 | **15.**<br>Какой из перечисленных алгоритмов планирования процессов в операционной системе является непрерывным (non-preemptive)?<br><br>1. Round Robin (RR, циклическое планирование).<br>2. Shortest Job First (SJF, наим. время выполнения).<br>3. Priority Scheduling with Preemption (приоритетное с вытеснением).<br>4. Multilevel Feedback Queue (многоуровневая очередь с обратной связью).<br><br>Ответ: 2 (Shortest Job First в своей классической форме).<br><br>Пояснение: Невытесняющее (непрерывное) планирование означает, что процесс, получивший процессор, работает до своего завершения или до | **15.**<br>В чем заключается основное назначение диспетчера (dispatcher) в подсистеме управления процессами операционной системы?<br><br>1. Принимать решение о том, какой процесс из очереди готовых получить процессор следующим (стратегия планирования).<br>2. Выполнять переключение контекста между процессами, т.е. сохранять состояние текущего процесса и загружать состояние выбранного для выполнения.<br>3. Управлять созданием и завершением процессов (вызовы fork(), exit() в UNIX).<br>4. Обеспечивать взаимодействие и синхронизацию между процессами (семафоры, мьютексы).<br><br>Ответ: 2. | **Повышенный (3-5 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | техническое сопровождение информационных систем | | явной блокировки (например, на операцию ввода-вывода). Классический SJF (без вытеснения) именно так и работает: он выбирает процесс с наименьшим предсказанным временем выполнения и отдает ему процессор до конца. Все остальные алгоритмы из списка (1, 3, 4) являются вытесняющими (preemptive): процесс может быть принудительно снят с выполнения по истечению кванта времени (1), при появлении более приоритетного процесса (3) или при перемещении между очередями (4). | Пояснение: Диспетчер — это низкоуровневый модуль ядра, который выполняет механическую работу по переключению контекста. Планировщик (scheduler) принимает стратегическое решение (1), а диспетчер реализует его. Созданием и завершением процессов управляет другой компонент ядра (3). Взаимодействие процессов (4) обеспечивается подсистемой межпроцессного взаимодействия (IPC). Путаница между функциями планировщика и диспетчера — частая ошибка, которую важно устранять на фундаментальном уровне. | |
| 16 | Тема 16 Интерфейс пользователя. Операционное окружение | ОПК-5<br><br>Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизиро | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на | 4,4 | **16.**<br>Какие из следующих утверждений о программном интерфейсе (API) операционной системы являются корректным? Выберите верные.<br><br>1. API — это набор функций, структур данных и констант, предоставляемых ОС для разработки прикладных программ.<br>2. Основное назначение API — | **16.**<br>Какие из перечисленных компонентов обычно входят в состав операционного окружения (operating environment) или оболочки (shell) ОС, предоставляя пользователю и программам стандартные сервисные функции? Выберите верные.<br><br>1. Командный интерпретатор (bash, cmd.exe, PowerShell). | **Высокий (5-10 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | ванных систем | базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | скрыть детали реализации аппаратных и системных вызовов, обеспечивая переносимость и упрощение разработки.<br>3. Программный интерфейс (API) и интерфейс системных вызовов (System Call Interface) — это одно и то же.<br>4. Примером API операционной системы является Win32 API для Windows или POSIX API для UNIX-подобных систем.<br>5. API относится исключительно к графическим библиотекам для создания оконных интерфейсов (например, Qt, GTK).<br><br>Ответ: 1, 2, 4.<br><br>Пояснение:<br><br>1 — Это классическое определение API.<br><br>2 — Это ключевое назначение абстракции, которую предоставляет API.<br><br>4 — Это конкретные, широко известные примеры API операционных систем. | 2. Библиотеки времени выполнения (C runtime library, .NET Framework, Java Runtime Environment).<br>3. Утилиты для работы с файловой системой (cp, mv, dir, find).<br>4. Ядро операционной системы и его драйверы устройств.<br>5. Текстовые редакторы, компиляторы и отладчики.<br><br>Ответ: 1, 3.<br><br>Пояснение:<br>1 — Командный интерпретатор — центральный элемент операционного окружения, обеспечивающий прямой диалог пользователя с ОС.<br>3 — Стандартный набор утилит для управления файлами и данными — неотъемлемая часть любого операционного окружения (как CLI, так и GUI). | |
| 17 | Тема 17 | ОПК-5 | ИОПК-5.1. Знать | 2,2 | **17.** | 17. | **Базовый (1-3** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | Обработка прерываний | Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем | основы системного администрирования, администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | Установите правильную последовательность аппаратных и начальных программных действий процессора и ОС при возникновении маскируемого аппаратного прерывания (например, от сетевой карты).<br><br>1. Процессор приостанавливает выполнение текущего потока команд.<br>2. Обработчик прерывания (ISR) сохраняет на стеке ядра остальные регистры процессора (RAX, RBX и т.д.) для полного сохранения контекста.<br>3. Процессор определяет номер вектора прерывания от устройства.<br>4. Устройство выставляет сигнал на линии запроса прерывания (IRQ).<br>5. Процессор загружает из таблицы векторов прерываний (IVT) адрес обработчика (ISR) и передает ему управление.<br>6. Процессор автоматически сохраняет на стеке текущие значения счетчика команд (IP) и регистра флагов (FLAGS/PSW).<br><br>Ответ: 4, 1, 3, 6, 5, 2. | Установите последовательность этапов в полном цикле обработки аппаратного прерывания, включая действия операционной системы после работы основного обработчика (ISR).<br><br>1. Обработчик прерывания (ISR) выполняет минимально необходимые действия (например, чтение данных из порта устройства, отправка подтверждения).<br>2. Процессор, выполнив инструкцию возврата из прерывания (IRET), восстанавливает контекст прерванного потока из стека и продолжает его выполнение.<br>3. Планировщик ОС, если это необходимо, принимает решение о переключении на другой процесс с более высоким приоритетом, который мог быть разблокирован данным прерыванием.<br>4. Управление возвращается из ISR в общий код ядра, который отвечает за обработку «нижней половины» (bottom half) — более тяжелой обработки, которую нельзя делать в контексте прерывания. | **минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | Пояснение: Эта последовательность разделяет аппаратные (4,1,3,6,5) и начальные программные (2) действия, что критически важно для понимания. Устройство инициирует процесс (4). Процессор реагирует (1), идентифицирует источник (3), аппаратно сохраняет минимальный, но достаточный для возврата контекст (6), и аппаратно передает управление (5). Только после этого начинает работу код ОС (обработчик), и его первая задача — сохранить все остальное (2). Нарушение этого порядка (например, сохранение регистров до передачи управления) противоречит архитектуре процессора. | 5. Обработчик прерывания (ISR) восстанавливает ранее сохраненные регистры общего назначения из стека ядра.<br><br>Ответ: 1, 5, 2, 4, 3.<br><br>Пояснение: Эта последовательность иллюстрирует ключевую концепцию разделения обработки на «верхнюю» (ISR, быстрая, в контексте прерывания) и «нижнюю» (отложенную, в контексте процесса/ядра). ISR делает срочное (1), затем готовит возврат (5). Аппаратный возврат (2) восстанавливает прерванный мир. Затем уже в «нормальном» режиме ядро доделывает тяжелую работу (4). И только на основе результатов этой работы планировщик может принять решение (3). Это показывает, как ОС обеспечивает и быстроту реакции, и стабильность системы. | |
| 18 | Тема 18 Работа с файлами | ОПК-5<br><br>Способен инсталлировать | ИОПК-5.1. Знать основы системного администрирования, | 1,1 | **18.**<br>Сопоставьте элемент файловой системы с уровнем организации, к которому он в первую очередь относится. | **18.**<br>Сопоставьте метод физического размещения файла на диске с его ключевой характеристикой. | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | программное и аппаратное обеспечение для информационных и автоматизированных систем | администрирования СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | Список А (Элемент):<br>1. Имя файла и расширение<br>2. Индексный дескриптор (inode)<br>3. Права доступа в формате rwxr-xr--<br>4. Таблица размещения файлов (FAT)<br>5. Путь /usr/bin/bash<br><br>Список Б (Уровень организации):<br>A. Логическая организация (представление для пользователя)<br>B. Физическая организация (внутренняя реализация на диске)<br><br>Ответ: 1A, 2B, 3A, 4B, 5A. | Список А (Метод размещения):<br>1. Непрерывное (последовательное) размещение<br>2. Размещение связным списком (через FAT)<br>3. Индексное размещение (через inode)<br><br>Список Б (Характеристика):<br>A. Для доступа к произвольному блоку требуется последовательное чтение цепочки указателей, что замедляет случайный доступ.<br>B. Позволяет эффективный прямой доступ к любому блоку, храня все указатели централизованно, но размер файла может быть ограничен размером индексного блока.<br>C. Обеспечивает максимальную скорость последовательного чтения, но страдает от внешней фрагментации и сложности расширения файла.<br><br>Ответ: 1C, 2A, 3B. | |
| 19 | Тема 19. Планирование заданий. Распределение | ОПК-5 Способен инсталлировать программное и аппаратное | ИОПК-5.1. Знать основы системного администрирования, администрирова | 1,1 | **19.**<br>Сопоставьте алгоритм планирования с типом системы, для которого он наиболее характерен или эффективен.<br>Список А (Алгоритм | **19.**<br>Сопоставьте метрику производительности планировщика с ее определением.<br>Список А (Метрика): | **Базовый (1-3 минуты)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | ресурсов | обеспечение для информационных и автоматизированных систем | ния СУБД, современные стандарты информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | планирования):<br>1. First-Come, First-Served (FIFO)<br>2. Round Robin (RR)<br>3.Shortest Job First (SJF)<br>4. Earliest Deadline First (EDF)<br>Список Б (Тип системы):<br>A. Интерактивные time-sharing системы (универсальный fairness).<br>B. Системы реального времени с динамическими приоритетами.<br>C. Системы пакетной обработки (простой и предсказуемый).<br>D. Системы пакетной обработки (оптимальный по среднему времени выполнения).<br>Ответ: 1C, 2A, 3D, 4B.<br><br>. | 1. Время оборачивания (Turnaround Time)<br>2. Время ожидания (Waiting Time)<br>3. Время отклика (Response Time)<br>4. Пропускная способность (Throughput)<br>Список Б (Определение):<br>A. Интервал от момента поступления задачи в систему до момента выдачи последнего результата (время от подачи до завершения).<br>B. Количество задач, завершенных системой в единицу времени.<br>C. Интервал от момента поступления задачи до момента, когда она впервые получает процессор для выполнения.<br>D. Общее время, которое процесс провел в состоянии готовности (готов, но не выполняется).<br>Ответ: 1A, 2D, 3C, 4B | |
| 20 | Тема 20. Защищенность и отказоустойчивость операционных систем | ОПК-5 Способен инсталлировать программное и аппаратное обеспечение для | ИОПК-5.1. Знать основы системного администрирования, администрирования СУБД, современные стандарты | 5,5 | **20.**<br>Как называется основной механизм в современных файловых системах (ZFS, Btrfs), который обеспечивает высокую целостность данных и восстанавливаемость за счет того, что при любом обновлении данных всегда записывается | **20.**<br>Как называется компонент подсистемы безопасности ядра Linux, который отвечает за регистрацию (логирование) всех значимых событий безопасности, таких как успешные и неуспешные попытки входа, использование привилегий, | **Высокий (5-10 минут)** |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | информационных и автоматизированных систем | информационного взаимодействия систем на базовом уровне ИОПК-5.2. Уметь выполнять параметрическую настройку информационных и автоматизированных систем на базовом уровне, реализовывать техническое сопровождение информационных систем | | новый блок, а старый блок остается неизменным до освобождения? Укажите общепринятый англоязычный термин.<br>Ответ: Copy-on-Write (CoW) / Копирование при записи.<br>Пояснение: Механизм Copy-on-Write является краеугольным камнем для обеспечения целостности. Он предотвращает ситуации, когда сбой питания может привести к частичной записи блока, повреждающей старые данные. Поскольку новые данные пишутся в новое место, предыдущее состояние файловой системы (или снапшот) остается неизменным и согласованным. Это позволяет легко откатить изменения или восстановить данные после сбоя. CoW также лежит в основе эффективных снапшотов. | изменения политик доступа? Укажите его точное название.<br>Ответ: Audit daemon (auditd) и подсистема аудита ядра Linux (Linux Audit Subsystem).<br>Пояснение: Подсистема аудита Linux (во главе с демоном auditd) является ключевым механизмом аудита (accountability). Она собирает события от ядра, связанные с безопасностью, в защищенный от модификации журнал (/var/log/audit/audit.log). Эти логи критически важны для расследования инцидентов, соответствия стандартам (например, PCI DSS) и анализа действий в системе. Без эффективного аудита даже самые строгие механизмы аутентификации и авторизации не обеспечивают полной подотчетности. | |

***Критерии оценивания диагностической работы:***

| Оценка | Критерии оценивания |
|---|---|
| **отлично** | от 90 до 100 % правильно выполненных заданий |
| **хорошо** | от 70 до 89 % правильно выполненных заданий |
| **удовлетворительно** | от 50 до 69 % правильно выполненных заданий |
| **неудовлетворительно** | менее 50 % правильно выполненных заданий |