ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ИМЕНИ. В.Ф. УТКИНА

Кафедра «Вычислительная и прикладная математика»

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ**

**«Разработка системных утилит»**

Направление подготовки

09.03.04 «Программная инженерия»

Направленность (профиль) подготовки

Программная инженерия

Квалификация выпускника - бакалавр

Форма обучения - очная

Рязань

**1. ОБЩИЕ ПОЛОЖЕНИЯ**

*Оценочные материалы* - это совокупность учебно-методических материалов и процедур, предназначенных для оценки качестваосвоения обучающимися данной дисциплины как части основной образовательной программы.

*Цель* - оценить соответствие знаний, умений и уровня приобретенных компетенций, обучающихся целям и требованиям основной образовательной программы в ходе проведения текущего контроля и промежуточной аттестации.

*Основная задача* - обеспечить оценку уровня сформированности компетенций, приобретаемых обучающимся в соответствии с этими требованиями.

Контроль знаний обучающихся проводится в форме экзамена в 7-м семестре.

**2. ОПИСАНИЕ ПОКАЗАТЕЛЕЙ И КРИТЕРИЕВ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ**

Сформированность каждой компетенции в рамках освоения данной дисциплины оценивается по трехуровневой шкале:

1) пороговый уровень является обязательным для всех обучающихся по завершении освоения дисциплины;

2) продвинутый уровень характеризуется превышением минимальных характеристик сформированности компетенций по завершении освоения дисциплины;

3) эталонный уровень характеризуется максимально возможной выраженностью компетенций и является важным качественным ориентиром для самосовершенствования.

**Уровень освоения компетенций, формируемых дисциплиной**

*а) описание критериев и шкалы оценивания тестирования:*

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| 3 балла  (эталонный уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 85 до 100% |
| 2 балла  (продвинутый уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 75 до 84% |
| 1 балл  (пороговый уровень) | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 60 до 74% |
| 0 баллов | уровень усвоения материала, предусмотренного программой: процент верных ответов на тестовые вопросы от 0 до 59% |

*б) описание критериев и шкалы оценивания теоретического вопроса:*

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| 3 балла  (эталонный уровень) | выставляется студенту, который дал полный ответ на вопрос, показал глубокие систематизированные знания, смог привести примеры, ответил на дополнительные вопросы преподавателя. |
| 2 балла  (продвинутый уровень) | выставляется студенту, который дал полный ответ на вопрос, но на некоторые дополнительные вопросы преподавателя ответил только с помощью наводящих вопросов. |
| 1 балл  (пороговый уровень) | выставляется студенту, который дал неполный ответ на вопрос в билете и смог ответить на дополнительные вопросы только с помощью преподавателя. |
| 0 баллов | выставляется студенту, который не смог ответить на вопрос. |

*в) описание критериев и шкалы оценивания практического задания:*

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| 3 балла  (эталонный уровень) | Задание решено верно. |
| 2 балла  (продвинутый уровень) | Задание решено верно, но имеются технические неточности в выполнении. |
| 1 балл  (пороговый уровень) | Задание решено верно, с дополнительными наводящими вопросами преподавателя. |
| 0 баллов | Задание не решено. |

На экзамен выносится: тестовое задание, 1 практическое задание и 1 теоретический вопрос. Студент может набрать максимум 9 баллов. Итоговый суммарный балл студента, полученный при прохождении промежуточной аттестации, переводится в традиционную форму по системе «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Обязательным условием получения оценок «отлично», «хорошо» и «удовлетворительно» является выполнениевсех предусмотренных в течение семестра практических заданий и лабораторных работ. Студент не выполнивший всех предусмотренных в течение семестра текущих заданий получает оценку «неудовлетворительно»

|  |  |
| --- | --- |
| **Шкала оценивания** | **Критерий** |
| отлично  (эталонный уровень) | 8 - 9 баллов (все задания и лабораторные работы выполнены) |
| хорошо  (продвинутый уровень) | 6 - 7 баллов (все задания и лабораторные работы выполнены) |
| удовлетворительно  (пороговый уровень) | 4 - 5 баллов (все задания и лабораторные работы выполнены) |
| неудовлетворительно | 0 - 3 баллов (студент не выполнил все задания и лабораторные работы) |

**3. ПАСПОРТ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ДИСЦИПЛИНЕ**

|  |  |  |
| --- | --- | --- |
| **Контролируемые разделы (темы) дисциплины** | **Код контролируемой компетенции (или её части)** | **Наименование оценочного средства** |
| Тема 1. Языки программирования. Классификация языков программирования. Парадигмы языков программирования | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 2. Архитектура и аппаратно-программные компоненты компьютера | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 3. Операционные системы | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 4. Системные утилиты. Создание системных утилит | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 5. Драйверы | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 6. Распределение памяти. Управление памятью | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 7. Трансляторы. Компиляторы и интерпретаторы | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 8. Ассемблер. Компоновщик. Загрузчик | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |
| Тема 9. Оптимизация кода. | ПК-1.2 ПК-3.1 ПК-3.2 | Экзамен |

**4. ТИПОВЫЕ КОНТРОЛЬНЫЕ ЗАДАНИЯ ИЛИ ИНЫЕ МАТЕРИАЛЫ**

**4.1 Промежуточная аттестация**

*Типовые тестовые вопросы закрытого типа:*

|  |
| --- |
| **ПК-1 Способен выполнять постановку задач, обосновывать технические условия и задания на проектирование автоматизированных систем с учетом требований к автоматизированным системам специального назначения.** |
| **ПК-1.2 Разрабатывает технические спецификации на программные компоненты** |

Вопрос 1. Какая область программирования занимается разработкой искусственного интеллекта?

Веб-разработка

Фронтенд-разработка

**Машинное обучение**

Бэкенд-разработка

Вопрос 2. Какая область программирования занимается разработкой операционных систем?

Бэкенд-разработка

**Системное программирование**

Фронтенд-разработка

Веб-разработка

Вопрос 3. Кто разрабатывает технические спецификации на программные компоненты?

**Бизнес-аналитики**

Системные администраторы

Программисты

Тестировщики

Вопрос 4. Какая область программирования занимается разработкой клиентской части веб-сайтов?

Системная разработка

Мобильная разработка

**Фронтенд-разработка**

Бэкенд-разработка

Вопрос 5. Какая область программирования занимается разработкой баз данных?

Фронтенд-разработка

Системная разработка

**Бэкенд-разработка**

Мобильная разработка

Вопрос 6. Какие методы можно использовать для запуска службы операционной системы?

Запуск из графического интерфейса

Запуск через браузер

Ручной запуск из командной строки

**Автоматический запуск при загрузке системы**

Вопрос 7. Что такое служба операционной системы?

Модуль операционной системы

**Программа, выполняющаяся в фоновом режиме**

Устройство для хранения данных

Виртуальная машина

Вопрос 8. Какой язык программирования можно использовать для разработки системных утилит?

Java

**C++**

HTML

Python

Вопрос 9. Какие протоколы могут использоваться для взаимодействия с службами операционной системы?

FTP

**RPC**

TCP/IP

HTTP

|  |
| --- |
| **ПК-3 Способен выполнять работы и управление работами по созданию и сопровождению информационных систем.** |
| **ПК-3.1 Разрабатывает, анализирует и утверждает требования к информационной системе.** |

Вопрос 1. Нефункциональные требования определяют:

функционалные возможности интерфейса системы

**какие функции должна выполнять система**

ограничения и качественные характеристики системы

функциональный способ хранения данных в системе

Вопрос 2. то утверждает требования к информационной системе?

Программист

**Представитель заказчика**

Бизнес-аналитик

QA-инженер

Вопрос 3. Кто отвечает за разработку требований к информационной системе?

Проектный менеджер

Тестировщик

**Бизнес-аналитик**

Архитектор

Вопрос 4. Функциональные требования определяют:

ограничения и качественные характеристики системы

способ хранения данных в системе

**какие функции должна выполнять система**

функционалные возможности интерфейса системы

Вопрос 5. Кто отвечает за обновление требований к информационной системе в процессе разработки?

Заказчик

Технический директор

Руководитель проекта

**Бизнес-аналитик**

Вопрос 6. Кто разрабатывает документацию по требованиям к информационной системе?

Бизнес-аналитик

Программист

**Технический писатель**

QA-инженер

Вопрос 7. Какие функции выполняет драйвер в операционной системе?

Управление сетью

**Управление устройствами**

Управление памятью

Управление процессами

Вопрос 8. Что такое драйвер?

Программа для создания пользовательского интерфейса

Программа для установки новых устройств

**Программа для управления аппаратными ресурсами**

Программа для отладки кода

Вопрос 9. Какие файлы используются для конфигурации служб операционной системы?

.txt

.exe

**.conf**

.dll

Вопрос 10. Какие операции нельзя выполнять над службами операционной системы?

**Гибернация**

Перезапуск

Остановка

Запуск

|  |
| --- |
| **ПК-3 Способен выполнять работы и управление работами по созданию и сопровождению информационных систем.** |
| **ПК-3.2 Выполняет проектирование и реализацию информационной системы.** |

Вопрос 1. Какой протокол используется для передачи веб-страниц между клиентом и сервером?

TCP

**HTTP**

IP

FTP

Вопрос 2. Какой методологии разработки программного обеспечения уделяет особое внимание гибкости и быстрому реагированию на изменения требований?

Каскадная модель

**Agile**

Прототипирование

Водопадная модель

Вопрос 3. Кто отвечает за проектирование и реализацию информационной системы?

Бизнес-аналитик

**Архитектор**

Программист

Системный администратор

Вопрос 4. Какой тип тестирования проверяет, соответствует ли программное обеспечение заранее определенным требованиям?

Модульное тестирование

Интеграционное тестирование

Системное тестирование

**Приемочное тестирование**

Вопрос 5. Какой процесс включает в себя проектирование информационной системы?

Проектирование архитектуры

Сбор требований

**Все перечисленное**

Анализ бизнес-процессов

Вопрос 6. Какая операционная система является наиболее распространенной для серверов?

macOS

**Linux**

Windows

Android

Вопрос 7. Какой инструмент используется для отладки и анализа кода?

CMS (Content Management System)

SDK (Software Development Kit)

API (Application Programming Interface)

**IDE (Integrated Development Environment)**

Вопрос 8. Какой тип архитектуры программного обеспечения предполагает разделение приложения на логические компоненты?

ORM (Object-Relational Mapping)

**MVC (Model-View-Controller)**

MVVM (Model-View-ViewModel)

MVP (Model-View-Presenter)

Вопрос 9. Какая операционная система является открытой и бесплатной?

**Linux**

macOS

Windows

iOS

Вопрос 10. Какая операционная система используется в большинстве серверов?

Android

**Linux**

Windows

macOS

Вопрос 11. Какая операционная система наиболее распространена в мобильных устройствах?

macOS

Windows

Linux

**Android**

Вопрос 12. Какие типы драйверов существуют?

**Графические драйверы**

Файловые драйверы

Процессорные драйверы

HDD драйверы

*Типовые тестовые вопросы открытого типа:*

|  |
| --- |
| **ПК-1 Способен выполнять постановку задач, обосновывать технические условия и задания на проектирование автоматизированных систем с учетом требований к автоматизированным системам специального назначения.** |
| **ПК-1.2 Разрабатывает технические спецификации на программные компоненты** |

Вопрос 1. Какие тестовые требования должны быть указаны в технической спецификации?

Ответ: Тестовые требования определяют, какие тесты должны быть проведены для проверки работоспособности и соответствия программного компонента требованиям.

Вопрос 2. Какие архитектурные решения должны быть описаны в технической спецификации?

Ответ: В технической спецификации должны быть описаны архитектурные решения, такие как выбор языка программирования, базы данных, платформы, а также структура компонента и его модули.

Вопрос 3. Какие методы и инструменты могут использоваться при разработке технической спецификации?

Ответ: При разработке технической спецификации могут использоваться методы и инструменты, такие как UML-диаграммы, CASE-средства, текстовые редакторы, системы управления требованиями и другие.

Вопрос 4. Какие информационные требования должны быть указаны в технической спецификации?

Ответ: Информационные требования включают в себя описание входных и выходных данных, форматы данных, базы данных, интерфейсы и протоколы обмена информацией.

Вопрос 5. Какова роль разработчика программного компонента в создании технической спецификации?

Ответ: Разработчик программного компонента играет ключевую роль в создании технической спецификации, так как он отвечает за определение требований, архитектуры и функциональности компонента.

Вопрос 6. Что такое функциональные требования в технической спецификации?

Ответ: Функциональные требования определяют, какие функции и возможности должны быть реализованы в программном компоненте, включая взаимодействие с пользователем и другими системами.

Вопрос 7. Какие основные разделы должны присутствовать в технической спецификации?

Ответ: Основные разделы технической спецификации включают в себя введение, общее описание, требования, архитектуру, функциональность, тестирование и дополнительные материалы.

Вопрос 8. Что такое техническая спецификация в разработке программных компонентов?

Ответ: Техническая спецификация - это документ, который описывает требования, архитектуру, функциональность и другие технические детали программного компонента.

Вопрос 9. Какие компоненты входят в операционную систему?

Ответ: Операционная система состоит из нескольких компонентов, включая ядро операционной системы, файловую систему, драйверы устройств, системные утилиты, пользовательский интерфейс и множество других компонентов, которые обеспечивают функциональность и управление ресурсами компьютера.

Вопрос 10. Какие языки программирования часто используются при разработке операционных систем?

Ответ: При разработке операционных систем часто используются языки программирования, такие как C и C++. Эти языки предоставляют низкоуровневый доступ к аппаратным ресурсам компьютера и обеспечивают эффективность и производительность операционной системы.

Вопрос 11. Какие задачи выполняют системные утилиты?

Ответ: Системные утилиты выполняют различные задачи, такие как управление памятью, дисковым пространством, процессами, сетевыми соединениями и безопасностью. Они также могут предоставлять информацию о системе, обнаруживать и исправлять ошибки, а также оптимизировать работу операционной системы.

Вопрос 12. Что такое драйвер?

Ответ: Драйвер - это программное обеспечение, которое позволяет операционной системе взаимодействовать с аппаратными устройствами, такими как принтеры, сканеры, видеокарты и т. д. Драйверы обеспечивают правильную работу устройств и переводят команды операционной системы в понятный для устройства формат.

Вопрос 13. Что такое системная утилита?

Ответ: Системная утилита - это программное обеспечение, разработанное для управления и оптимизации работы операционной системы. Она может выполнять различные задачи, такие как дефрагментация диска, очистка реестра, мониторинг ресурсов и т. д.

|  |
| --- |
| **ПК-3 Способен выполнять работы и управление работами по созданию и сопровождению информационных систем.** |
| **ПК-3.1 Разрабатывает, анализирует и утверждает требования к информационной системе.** |

Вопрос 1. Какие инструменты могут быть использованы для документирования требований к информационной системе?

Ответ: Инструменты, такие как диаграммы вариантов использования, диаграммы классов, диаграммы последовательностей, таблицы требований, прототипы и спецификации требований, могут быть использованы для документирования требований.

Вопрос 2. Кто является основным ответственным за разработку требований к информационной системе?

Ответ: Бизнес-аналитик. Бизнес-аналитик проводит анализ бизнес-потребностей и взаимодействует с заинтересованными сторонами для определения требований к информационной системе.

Вопрос 3. Какие методы используются для разработки требований к информационной системе?

Ответ: Методы, такие как сбор требований через интервью и обсуждения, анализ документации и бизнес-процессов, проведение опросов и анкетирования, а также применение моделирования и прототипирования.

Вопрос 4. Каким образом требования к информационной системе могут быть проверены на соответствие?

Ответ: Требования могут быть проверены через проведение тестирования, проведение ревизий и аудитов требований, а также сравнение требований с бизнес-потребностями и ожиданиями заказчика.

Вопрос 5. Какие факторы могут повлиять на изменение требований к информационной системе?

Ответ: Изменение бизнес-потребностей, новые технологии, изменение регулятивных требований, обратная связь от пользователей и заинтересованных сторон, а также ошибки и недоразумения в исходных требованиях могут привести к изменению требований.

Вопрос 6. Какой процесс включает в себя анализ требований к информационной системе?

Ответ: Анализ требований включает в себя изучение и понимание бизнес-процессов, определение функциональных и нефункциональных требований, а также выявление возможных рисков и ограничений.

Вопрос 7. Какие типы системных утилит существуют?

Ответ: Существует множество типов системных утилит, включая утилиты для управления памятью, дисковым пространством, процессами, сетевыми соединениями, безопасностью, резервным копированием данных и многие другие. Каждая утилита выполняет определенную задачу для оптимизации работы операционной системы.

Вопрос 8. Что такое многозадачность в операционной системе?

Ответ: Многозадачность в операционной системе - это возможность выполнять несколько задач одновременно. Операционная система распределяет ресурсы компьютера между различными задачами, позволяя им работать параллельно и обеспечивая эффективное использование процессора и памяти.

Вопрос 9. Что такое файловая система?

Ответ: Файловая система - это метод организации и хранения файлов на компьютере. Она определяет структуру и формат файлов, а также способы доступа к ним. Файловая система также обеспечивает управление пространством на диске и контроль доступа к файлам.

Вопрос 10. Что такое API операционной системы?

Ответ: API (Application Programming Interface) операционной системы - это набор функций и процедур, предоставляемых операционной системой для взаимодействия с приложениями. Они определяют способы вызова функций операционной системы и передачи параметров для выполнения различных операций.

Вопрос 11. Что такое операционная система?

Ответ: Операционная система - это программное обеспечение, которое управляет ресурсами компьютера и обеспечивает взаимодействие между пользователем и аппаратным обеспечением. Она предоставляет интерфейс для запуска и управления приложениями, а также обеспечивает безопасность и управление файлами.

|  |
| --- |
| **ПК-3 Способен выполнять работы и управление работами по созданию и сопровождению информационных систем.** |
| **ПК-3.2 Выполняет проектирование и реализацию информационной системы.** |

Вопрос 1. Что включает в себя процесс проектирования информационной системы?

Ответ: Процесс проектирования информационной системы включает в себя определение требований, анализ бизнес-процессов, разработку архитектуры, проектирование базы данных и интерфейса пользователя.

Вопрос 2. Какие базы данных могут быть использованы при реализации информационной системы?

Ответ: Некоторые из баз данных, которые могут быть использованы при реализации информационной системы, включают MySQL, PostgreSQL, Oracle и MongoDB.

Вопрос 3. Какие инструменты разработки программного обеспечения могут быть использованы при реализации информационной системы?

Ответ: Некоторые из инструментов разработки программного обеспечения, которые могут быть использованы при реализации информационной системы, включают IDE (Integrated Development Environment), системы контроля версий (например, Git), инструменты для автоматизации сборки и развертывания (например, Jenkins), и инструменты для тестирования (например, JUnit).

Вопрос 4. Какие архитектурные паттерны могут быть использованы при проектировании информационной системы?

Ответ: Некоторые из архитектурных паттернов, которые могут быть использованы при проектировании информационной системы, включают MVC (Model-View-Controller), MVVM (Model-View-ViewModel), и SOA (Service-Oriented Architecture).

Вопрос 5. Какие методологии разработки программного обеспечения могут быть применены при проектировании информационной системы?

Ответ: Некоторые из методологий разработки программного обеспечения, которые могут быть применены при проектировании информационной системы, включают в себя водопадную модель, Agile, Scrum и DevOps.

Вопрос 6. Какие задачи выполняет системный монитор?

Ответ: Системный монитор - это системная утилита, которая отслеживает и анализирует работу компьютера и его компонентов. Он может отображать информацию о загрузке процессора, использовании памяти, активности дисков, сетевом трафике и других системных параметрах.

Вопрос 7. Что такое динамическая загрузка драйверов?

Ответ: Динамическая загрузка драйверов - это механизм, который позволяет операционной системе загружать драйверы устройств по мере необходимости. Это позволяет экономить память и ускоряет процесс загрузки операционной системы, так как только необходимые драйверы загружаются в память при запуске.

Вопрос 8. Что такое межпроцессное взаимодействие?

Ответ: Межпроцессное взаимодействие - это механизм, который позволяет процессам в операционной системе обмениваться данными и синхронизировать свою работу. Он может быть реализован с помощью различных методов, таких как сигналы, сокеты, каналы и разделяемая память.

Вопрос 9. Что такое виртуальная память?

Ответ: Виртуальная память - это механизм, который позволяет операционной системе использовать дополнительное пространство на жестком диске в качестве расширенной памяти. Она позволяет запускать приложения, которые требуют больше памяти, чем доступно физически.

Вопрос 10. Какие задачи выполняет операционная система?

Ответ: Операционная система выполняет ряд задач, включая управление ресурсами компьютера (процессором, памятью, дисками), планирование и управление процессами, обеспечение безопасности и защиты данных, управление файлами и директориями, а также обеспечение взаимодействия с пользователем.

Вопрос 11. Какие задачи выполняет операционная система при загрузке компьютера?

Ответ: При загрузке компьютера операционная система выполняет ряд задач, включая инициализацию аппаратного обеспечения, загрузку ядра операционной системы, запуск системных служб и драйверов устройств, а также инициализацию пользовательского интерфейса.

**4.2 Типовые контрольные вопросы и задания**

1. Какие методы тестирования используются для драйверов?

2. Операционные системы:

3. Какие алгоритмы управления памятью используются в операционных системах?

4. Какие системные утилиты используются для управления файлами и дисками?

5. Какие типы пользовательского интерфейса существуют?

6. Распределение памяти. Управление памятью:

7. Какие языки программирования относятся к мультипарадигмальной парадигме?

8. Системные утилиты. Создание системных утилит:

9. Что такое пользовательский интерфейс операционной системы?

10. Какие языки программирования относятся к функциональной парадигме?

11. Какие устройства требуют драйверов?

12. Какие инструменты разработки используются для создания системных утилит?

13. Какие устройства ввода-вывода используются в компьютере?

14. Какие языки программирования используются для разработки драйверов?

15. Что такое центральный процессор?

16. Какие инструменты разработки используются для создания драйверов?

17. Какие проблемы могут возникнуть при распределении памяти?

18. Какие функции выполняет операционная система?

19. Какие функции выполняет оперативная память?

20. Какие языки программирования относятся к объектно-ориентированной парадигме?

21. Что такое виртуальная память?

22. Какие принципы следует учитывать при разработке драйверов?

23. Какие инструменты разработки используются для анализа использования памяти?

24. Что такое шина данных?

25. Какие системные утилиты используются для настройки сети?

26. Какие языки программирования предоставляют возможности для управления памятью?

27. Какие системные утилиты используются для обеспечения безопасности?

28. Какие проблемы могут возникнуть при работе с драйверами?

29. Что такое операционная система?

30. Что такое жесткий диск?

31. Какие методы распределения памяти существуют?

32. Что такое драйвер?

33. Какие операционные системы относятся к семейству UNIX?

34. Что такое многозадачность?

35. Какие языки программирования относятся к структурной парадигме?

36. Какие языки программирования относятся к логической парадигме?

37. Какие принципы следует учитывать при управлении памятью?

38. Какие языки программирования используются для создания системных утилит?

39. Какие языки программирования относятся к скриптовым языкам?

40. Какие типы многозадачности существуют?

41. Какие операционные системы вы знаете?

42. Какие компоненты входят в аппаратную часть компьютера?

43. Что такое файловая система?

44. Какие проблемы могут возникнуть при управлении памятью?

45. Какие языки программирования вы знаете?

46. Какие типы драйверов существуют?

47. Какие функции выполняют системные утилиты?

48. Какие функции выполняют драйверы?

49. Какие системные утилиты используются для управления процессами?

50. Какие языки программирования относятся к компилируемым языкам?

51. Какие функции выполняет материнская плата?

52. Какие операционные системы используют драйверы?

53. Какие типы шин данных существуют?

54. Какие методы оптимизации использования памяти существуют?

55. Какие системные утилиты используются для диагностики компьютера?

56. Что такое периферийные устройства?

57. Какие языки программирования относятся к императивной парадигме?

58. Какие принципы следует учитывать при создании системных утилит?

59. Что такое виртуальная машина?

60. Что такое архитектура компьютера?

61. Какие языки программирования относятся к интерпретируемым языкам?

62. Что такое системные утилиты?

63. Что такое распределение памяти?

**4.3 Типовые задачи по дисциплине**

1. Создайте утилиту для мониторинга сетевых портов и отображения информации о текущих соединениях.

2. Напишите утилиту для генерации отчетов о системных журналах, включая информацию о событиях, ошибках и предупреждениях.

3. Напишите утилиту для автоматического переименования файлов в заданной директории на основе заданных правил.

4. Создайте утилиту для шифрования и дешифрования файлов с использованием алгоритма AES.

5. Напишите утилиту для генерации случайных паролей заданной длины. Учтите возможность настройки сложности пароля.

6. Разработайте утилиту для автоматического создания скриншотов экрана с заданной периодичностью.

7. Напишите утилиту для удаления временных файлов и очистки системного кэша для повышения производительности.

8. Создайте утилиту для архивации файлов. Реализуйте функционал сжатия и распаковки файлов в формате ZIP.

9. Напишите утилиту для удаления дубликатов файлов в заданной директории на основе их содержимого.

10. Создайте утилиту для анализа и оптимизации использования оперативной памяти системы.

11. Разработайте утилиту для поиска и замены текста в заданных файлах с использованием регулярных выражений.

12. Разработайте утилиту для мониторинга системных ресурсов, таких как использование процессора, памяти и дискового пространства.

13. Напишите утилиту для автоматического создания и отправки отчетов о состоянии системы на заданный адрес электронной почты.

14. Создайте утилиту для мониторинга и анализа сетевого трафика, включая отображение активных соединений и использование протоколов.

15. Создайте утилиту для автоматического резервного копирования файлов и папок на заданное сетевое хранилище.

16. Разработайте утилиту для контроля доступа к файлам и папкам на основе заданных правил и ролей пользователей.

17. Создайте утилиту для генерации и проверки хеш-сумм файлов для обеспечения целостности данных.