

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ ВЛАДИМИРА ДАЛЯ»

Краснодонский факультет инженерии и менеджмента (филиал)
Кафедра информационных технологий и транспорта



УТВЕРЖДАЮ:

Директор

Панайотов К.К.

(подпись)

«14» марта 2025 года

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по учебной дисциплине
Системное программное обеспечение

(наименование учебной дисциплины, практики)

09.03.01 Информатика и вычислительная техника

(код и наименование направления подготовки (специальности))

«Компьютерные системы и сети»

(наименование профиля подготовки (специальности, магистерской программы); при отсутствии ставится прочерк)

Разработчик(разработчики):

ст. преподаватель

(подпись)

Крупница О.В.

ФОС рассмотрен и одобрен на заседании кафедры информационных технологий и транспорта от «26» февраля 2025 г., протокол № 7

Заведующий кафедрой

информационных

технологий и транспорта

(подпись)

Верительник Е.А.

Краснодон 2025

**Комплект оценочных материалов по дисциплине
«Системное программное обеспечение»**

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

Выберите один правильный ответ.

1. Какой компонент ОС отвечает за выделение памяти процессам?

- А) Драйвер устройства.
- Б) Диспетчер памяти.
- В) Планировщик задач.
- Г) Файловая система.

Правильный ответ: Б

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. На каком этапе компиляции создается промежуточный объектный код?

- А) Лексический анализ.
- Б) Синтаксический анализ.
- В) Генерация кода.
- Г) Оптимизация.

Правильный ответ: В

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Какой тип гипервизора работает непосредственно на аппаратном обеспечении?

- А) Хост-ориентированный.
- Б) "Голый металл" (bare-metal).
- В) Контейнерный.
- Г) Паравиртуализация.

Правильный ответ: Б

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Какая библиотека предоставляет API для работы с файлами в Unix-системах?

- А) OpenGL.
- Б) POSIX.
- В) Boost.
- Г) STL.

Правильный ответ: Б

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Задания закрытого типа на установление соответствия

Установите правильное соответствие.

Каждому элементу левого столбца соответствует только один элемент правого столбца.

1. Установите соответствие между компонентами ОС и их функциями:

Компонентами ОС		Функции ОС	
1)	Планировщик задач	А)	Управление физической и виртуальной памятью.
2)	Драйвер устройства	Б)	Организация хранения данных на носителях.
3)	Файловая система	В)	Обеспечение взаимодействия с аппаратными устройствами.
4)	Менеджер памяти	Г)	Распределение процессорного времени между процессами.

Правильный ответ: 1Г, 2В, 3Б, 4А

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. Установите соответствие между видами трансляторов и их характеристиками:

Вид транслятора		Характеристики	
1)	Компилятор	А)	Построчное выполнение исходного кода.
2)	Интерпретатор	Б)	Преобразование программы в машинный код перед выполнением.
3)	Ассемблер	В)	Динамическая компиляция "на лету".
4)	ЛТ-компилятор	Г)	Трансляция с языка ассемблера.

Правильный ответ: 1Б, 2А, 3Г, 4В

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Установите соответствие между технологиями виртуализации и их описанием:

Технологии виртуализации		Описание	
1)	Гипервизор типа 1	А)	Работает поверх хостовой ОС.
2)	Гипервизор типа 2	Б)	Изоляция процессов без виртуализации оборудования.
3)	Контейнеризация	В)	Модификация гостевой ОС для повышения производительности.
4)	Паравиртуализация	Г)	Работает непосредственно на аппаратном обеспечении.

Правильный ответ: 1Г, 2А, 3Б, 4В

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Установите соответствие между системными библиотеками и их назначением:

Системные библиотеки		Назначение	
1)	POSIX	А)	Стандартный интерфейс Unix-систем.
2)	GLIBC	Б)	Основная библиотека языка С в Windows.
3)	WinAPI	В)	Реализация стандартной библиотеки С в Linux.
4)	CRT	Г)	API для разработки под Windows.

Правильный ответ: 1А, 2В, 3Г, 4Б

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Задания закрытого типа на установление правильной последовательности

Установите правильную последовательность.

Запишите правильную последовательность букв слева направо.

1. Установите правильную последовательность этапов загрузки операционной системы:

- А) Инициализация ядра.
- Б) Загрузка bootloader.
- В) Проверка оборудования (POST).
- Г) Запуск системных служб.

Правильный ответ: В, Б, А, Г

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. Установите последовательность этапов компиляции исходного кода:

- А) Генерация объектного кода.
- Б) Оптимизация кода.
- В) Лексический анализ.
- Г) Синтаксический анализ.

Правильный ответ: В, Г, Б, А

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Установите правильную последовательность действий при создании процесса:

- А) Выделение ресурсов.
- Б) Инициализация PCB.
- В) Загрузка программы в память.

Г) Постановка в очередь готовых процессов.

Правильный ответ: В, А, Б, Г

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Установите последовательность действий при обработке аппаратного прерывания:

А) Сохранение контекста.

Б) Возврат из прерывания.

В) Выполнение обработчика.

Г) Идентификация источника.

Правильный ответ: А, Г, В, Б

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Задания открытого типа

Задания открытого типа на дополнение

Напишите пропущенное слово (словосочетание).

1. Программный компонент ОС, отвечающий за распределение процессорного времени между задачами, называется _____.

Правильный ответ: Планировщик задач (или диспетчер задач).

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. Этап компиляции, на котором выполняется проверка синтаксической правильности исходного кода, называется _____.

Правильный ответ: Синтаксический анализ (или parsing).

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Технология изоляции процессов, не требующая полной виртуализации оборудования, называется _____.

Правильный ответ: Контейнеризация.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Интерфейс между приложением и ядром ОС для доступа к системным ресурсам называется _____.

Правильный ответ: Системный вызов (system call).

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Задания открытого типа с кратким свободным ответом

Дайте ответ на вопрос.

1. Какие компоненты являются обязательными для современной ОС?

Правильные ответы:

Драйверы устройств.

Менеджер памяти.

Планировщик задач.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. Какие виды трансляторов преобразуют исходный код в исполняемый?

Правильные ответы:

Компиляторы.

Ассемблеры.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Какие технологии используются для виртуализации?

Правильные ответы:

Контейнеры Docker.

Гипервизор VMware ESXi.

Паравиртуализация Xen.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Какие библиотеки относятся к системным?

Правильные ответы:

POSIX.

GLIBC.

WinAPI.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Задания открытого типа с развернутым ответом

Дайте развернутый ответ на вопрос.

1. Опишите эволюцию архитектуры ОС от монолитных ядер до микроядерных и гибридных систем. Какие преимущества и недостатки имеет каждый подход? Приведите конкретные примеры ОС каждого типа (не менее 2 для каждого) и обоснуйте выбор архитектуры в современных системах (Windows 11, Linux 6.x, macOS Sonoma).

Время выполнения – 20 мин.

Ожидаемый результат:

Типы архитектур ядер:

1. Монолитные ядра (Linux, FreeBSD):

- Все компоненты работают в одном адресном пространстве
- Преимущества: высокая производительность, простой дизайн
- Недостатки: сложность отладки, низкая отказоустойчивость

2. Микроядерные системы (QNX, MINIX 3):

- Минимальное ядро + сервисы в пользовательском пространстве

- Преимущества: стабильность, безопасность
 - Недостатки: накладные расходы на IPC
3. Гибридные ядра (Windows NT, macOS XNU):
- Компромисс между производительностью и модульностью
 - Пример: Windows 11 использует гибридное ядро для совместимости с драйверами

Современные тенденции:

- Linux сохраняет монолитную модель с модулями
- macOS сочетает микроядро Mach и BSD-компоненты
- Windows развивает концепцию "ядро как библиотека"

Критерии оценивания: Ответ должен содержательно соответствовать ожидаемому результату.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

2. Проанализируйте современные методы оптимизации кода в компиляторах (LLVM, GCC). Какие трансформации выполняются на разных уровнях оптимизации (-O1, -O2, -O3)? Как технология LTO (Link Time Optimization) меняет традиционный процесс компиляции? Приведите примеры конкретных оптимизаций для x86-64 и ARM архитектур.

Время выполнения – 20 мин.

Ожидаемый результат:

Уровни оптимизации в GCC:

1. -O1: базовые оптимизации (удаление мертвого кода, inline простых функций)
2. -O2: векторные операции, разворот циклов
3. -O3: агрессивные преобразования (автовекторизация)

LTO-технологии:

- Анализ всей программы перед генерацией кода
- Пример: объединение одинаковых функций из разных модулей

Архитектурно-зависимые оптимизации:

- x86-64: использование AVX-512 инструкций
- ARM: оптимизация ветвлений для процессоров Cortex

Примеры:

- Замена деления на умножение (x86)
- Оптимизация доступа к памяти (ARM big.LITTLE)

Критерии оценивания: Ответ должен содержательно соответствовать ожидаемому результату.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

3. Сравните технологии аппаратной виртуализации (Intel VT-x, AMD-V) и контейнеризации (Docker, LXC). Какие сценарии использования предпочтительны для каждого подхода? Опишите проблемы безопасности в контейнерах и методы их решения (не менее 3 реальных примеров уязвимостей и способов защиты).

Время выполнения – 20 мин.

Ожидаемый результат:

Сравнение технологий:

1. Аппаратная виртуализация:

- Полная изоляция ВМ
- Пример: облачные сервисы AWS EC2
- Недостаток: overhead ~15-20%

2. Контейнеризация:

- Легковесные изолированные процессы
- Пример: микросервисные архитектуры
- Недостаток: общее ядро ОС

Проблемы безопасности:

1. Уязвимость CVE-2019-5736 (escape из контейнера)

- Решение: использование rootless-контейнеров

2. Утечка данных через shared volumes

- Решение: шифрование томов

3. Атаки на ядро через syscalls

- Решение: seccomp-фильтры

Критерии оценивания: Ответ должен содержательно соответствовать ожидаемому результату.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

4. Проанализируйте эволюцию системных API от POSIX до современных кросс-платформенных интерфейсов. Как развивались стандарты системных вызовов в Linux (syscalls), Windows (WinAPI) и macOS (Darwin)? Какие проблемы решают новые технологии типа eBPF в Linux и DriverKit в macOS?

Время выполнения – 20 мин.

Ожидаемый результат:

Эволюция API:

1. POSIX (1988):

- Стандартизация Unix-интерфейсов
- Пример: fork(), exec()

2. Современные тенденции:

- Linux: 400+ системных вызовов (включая io_uring)
- Windows: Win32 API + UWP
- macOS: XNU + Darwin

Новые технологии:

1. eBPF в Linux:

- Безопасное выполнение кода в ядре
- Применение: мониторинг сети, безопасность

2. DriverKit в macOS:

- Перенос драйверов в пользовательское пространство
- Преимущество: стабильность системы

Проблемы совместимости:

- WSL2: реализация Linux syscalls в Windows
- Darling: POSIX-слой для macOS

Критерии оценивания: Ответ должен содержательно соответствовать ожидаемому результату.

Компетенции (индикаторы): ПК-2. (ПК-2.2)

Экспертное заключение

Представленный фонд оценочных средств (далее – ФОС) по дисциплине «Системное программное обеспечение» соответствует требованиям ФГОС ВО.

Предлагаемые формы и средства текущего и промежуточного контроля адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 09.03.01 Информатика и вычислительная техника.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины представлены в полном объеме.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

Разработанный и представленный для экспертизы фонд оценочных средств рекомендуется к использованию в процессе подготовки обучающихся по указанному направлению 09.03.01 Информатика и вычислительная техника.

Председатель учебно-методической
комиссии Краснодонского факультета
инженерии и менеджмента (филиала)

 Родионова О.Ю.

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)