

Министерство науки и высшего образования Российской Федерации
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный университет промышленных технологий и дизайна»
(СПбГУПТД)

УТВЕРЖДАЮ
Первый проректор, проректор
по УР

_____ А.Е. Рудин

Рабочая программа дисциплины

Б1.О.19

Системное программирование

Учебный план: 2026-2027 09.03.04 ИИТА Программ инженер ОО №1-1-195.plx

Кафедра: **33** Цифровых и аддитивных технологий

Направление подготовки:
(специальность) 09.03.04 Программная инженерия

Профиль подготовки:
(специализация) Программная инженерия

Уровень образования: бакалавриат

Форма обучения: очная

План учебного процесса

Семестр (курс для ЗАО)		Контактная работа обучающихся		Сам. работа	Контроль, час.	Трудоём- кость, ЗЕТ	Форма промежуточной аттестации
		Лекции	Практ. занятия				
5	УП	16	32	69	27	4	Экзамен
	РПД	16	32	69	27	4	
Итого	УП	16	32	69	27	4	
	РПД	16	32	69	27	4	

Рабочая программа дисциплины составлена в соответствии с федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.04 Программная инженерия, утверждённым приказом Минобрнауки России от 19.09.2017 г. № 920

Составитель (и):

к.т.н., заведующий кафедрой

Сошников Антон
Владимирович

старший преподаватель

Зверев Владислав
Витальевич

От кафедры составителя:

Заведующий кафедрой цифровых и аддитивных технологий

Сошников Антон
Владимирович

От выпускающей кафедры:

Заведующий кафедрой

Сошников Антон
Владимирович

Методический отдел:

1 ВВЕДЕНИЕ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

1.1 Цель дисциплины: Формирование у обучающихся теоретических знаний и практических навыков в области системного программирования, необходимых для разработки, отладки и оптимизации программного обеспечения, взаимодействующего с операционной системой и аппаратными ресурсами, с учётом современных стандартов и отечественных технологий.

1.2 Задачи дисциплины:

Сформировать представление об архитектуре и функционировании современных операционных систем.

Ознакомить с принципами работы системного программного обеспечения и его взаимодействием с аппаратной платформой.

Развить навыки использования системных вызовов, POSIX API и стандартных библиотек.

Обеспечить освоение основных инструментов разработки, отладки и профилирования системных приложений.

Способствовать формированию компетенций, необходимых для реализации прикладных и системных компонентов ПО в соответствии с требованиями профессиональной деятельности.

1.3 Требования к предварительной подготовке обучающегося:

Предварительная подготовка предполагает создание основы для формирования компетенций, указанных в п. 2, при изучении дисциплин:

Алгоритмизация и программирование

Архитектура ЭВМ и систем

Операционные системы, сети и телекоммуникации

2 КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	
Знать: Принципы архитектуры и функционирования современных операционных систем; особенности системного программного обеспечения и его взаимодействия с аппаратными ресурсами.	
Уметь: Выбирать и обосновывать использование соответствующих информационных технологий и программных средств для решения задач, связанных с разработкой, отладкой и оптимизацией системного программного обеспечения.	
Владеть: Навыками практического применения современных инструментов и сред разработки, включая отечественные, для создания, тестирования и отладки системного программного обеспечения.	
ОПК-6: Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	
Знать: Современные программные среды и инструменты разработки системного программного обеспечения.	
Уметь: Разрабатывать и настраивать системные компоненты (драйверы, утилиты, модули ОС) с использованием соответствующих языков программирования и средств разработки.	
Владеть: Навыками отладки, профилирования и тестирования системных приложений и прототипов программно-технических решений на уровне взаимодействия с операционной средой и аппаратной платформой.	

3 РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Наименование и содержание разделов, тем и учебных занятий	Семестр (курс для ЗАО)	Контактная работа		СР (часы)	Инновац. формы занятий	Форма текущего контроля
		Лек. (часы)	Пр. (часы)			
Раздел 1. Основы взаимодействия программ с операционной системой.	5					О
Тема 1. Архитектура ОС: ядро, системные вызовы, пользовательское и привилегированное пространство. Практические занятия: Создание и управление процессами с помощью fork(), exec(), wait().		2	4	8		
Тема 2. Управление процессами и потоками. Механизмы синхронизации. Практические занятия: Программирование простых сценариев синхронизации: мьютексы, семафоры.		2	4	8		
Тема 3. Работа с файловой системой на уровне системных вызовов. Практические занятия: Низкоуровневая работа с файлами: открытие, чтение, запись, позиционирование.		2	4	8		

Тема 4. Средства отладки и анализа. Практические занятия: Отладка программ и анализ системных вызовов с использованием GDB и strace.		2	4	8		
Раздел 2. Разработка и тестирование системных приложений.						
Тема 5. Принципы разработки системных утилит: надёжность, переносимость, стандарты. Практические занятия: Разработка системной утилиты командной строки.		2	4	8		
Тема 6. Особенности системного программирования на отечественных платформах. Практические занятия: Сборка и адаптация программы под отечественную ОС (с использованием виртуальной машины или контейнера).		2	4	8		О
Тема 7. Основы безопасности системного ПО: типовые уязвимости и методы защиты. Практические занятия: Поиск и устранение ошибок памяти (утечки, обращение к освобождённой памяти) с помощью valgrind.		2	4	8		
Тема 8. Тестирование системных приложений: unit-тестирование, статический и динамический анализ. Практические занятия: Написание unit-тестов для модуля системной утилиты.		2	4	13		
Итого в семестре (на курсе для ЗАО)		16	32	69		
Консультации и промежуточная аттестация (Экзамен)		2,5		24,5		
Всего контактная работа и СР по дисциплине		50,5		93,5		

4 КУРСОВОЕ ПРОЕКТИРОВАНИЕ

Курсовое проектирование учебным планом не предусмотрено

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

5.1 Описание показателей, критериев и системы оценивания результатов обучения

5.1.1 Показатели оценивания

Код компетенции	Показатели оценивания результатов обучения	Наименование оценочного средства
ОПК-2	Формулирует основные принципы работы ОС и взаимодействия программного обеспечения с оборудованием. Использует современные и отечественные инструменты для решения задач системного программирования. Демонстрирует результаты выбора технологий в зависимости от целей разработки системного программного продукта.	
ОПК-6	Раскрывает сущность системных утилит и отдельных компонентов. Применяет отладку, профилирование и тестирование на уровне ОС. Демонстрирует программный продукт пригодный к эксплуатации, соответствующие стандартам надёжности и безопасности.	

5.1.2 Система и критерии оценивания

Шкала оценивания	Критерии оценивания сформированности компетенций	
	Устное собеседование	Письменная работа
5 (отлично)	Полный, исчерпывающий ответ, явно демонстрирующий глубокое понимание предмета и широкую эрудицию в оцениваемой области. Критический, оригинальный подход к материалу. Учитываются баллы, накопленные в течение семестра.	

4 (хорошо)	Ответ полный, основанный на проработке всех обязательных источников информации. Подход к материалу ответственный, но стандартный. Учитываются баллы, накопленные в течение семестра.	
3 (удовлетворительно)	Ответ воспроизводит в основном только лекционные материалы, без самостоятельной работы с рекомендованной литературой. Демонстрирует понимание предмета в целом, без углубления в детали. Присутствуют существенные ошибки или пробелы в знаниях по некоторым темам. Учитываются баллы, накопленные в течение семестра.	
2 (неудовлетворительно)	Неспособность ответить на вопрос без помощи экзаменатора. Незнание значительной части принципиально важных элементов дисциплины. Многочисленные грубые ошибки. Не учитываются баллы, накопленные в течение семестра.	

5.2 Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности

5.2.1 Перечень контрольных вопросов

№ п/п	Формулировки вопросов
Семестр 5	
1	Что такое системный вызов и как он реализуется в Linux?
2	В чём разница между пространством пользователя и пространством ядра?
3	Как создаются новые процессы в Unix-подобных системах?
4	Какие функции используются для замены образа процесса (ехес-семейство)?
5	Что такое поток выполнения? Чем он отличается от процесса?
6	Какие примитивы синхронизации вы знаете? Как работает мьютекс?
7	Что такое файловый дескриптор и как он используется?
8	Какие основные системные вызовы для работы с файлами в POSIX?
9	Для чего используется функция lseek()?
10	Как работает утилита strace и для чего она применяется?
11	Как отлаживать программу с помощью GDB? Как поставить точку останова?
12	Что позволяет выявить valgrind при запуске программы?
13	Какие ошибки типичны при неправильной работе с памятью?
14	Что такое race condition и как его избежать?
15	Какие стандарты регулируют совместимость ПО в Unix-средах?
16	Какие принципы лежат в основе надёжного системного ПО?
17	Что такое POSIX и зачем соблюдать его при разработке?
18	Какие особенности разработки ПО под ОС «Астра Линукс»?
19	Почему важно учитывать переносимость при написании системных утилит?
20	Какие уязвимости характерны для программ на C (buffer overflow, use-after-free)?
21	Что такое secure coding и какие его правила применимы к системному ПО?
22	Как проверить, соответствует ли программа стандарту LSB?
23	Какие виды тестирования применяются к системным приложениям?
24	Что такое unit-тест и как он пишется для функции, использующей системные вызовы?
25	Как работают sanitizers (ASan, UBSan) и зачем они нужны?
26	Как протестировать корректность обработки ошибок системных вызовов?
27	Какие права требуются для запуска системной утилиты с повышенными привилегиями?
28	Как обеспечить корректную обработку сигналов в системной программе?
29	Какие инструменты статического анализа кода вы знаете (например, cppcheck)?

30	Какие метрики качества применимы к системному ПО (надёжность, производительность, безопасность)?
----	--

5.2.2 Типовые тестовые задания

Не предусмотрено.

5.2.3 Типовые практико-ориентированные задания (задачи, кейсы)

1. Утилита копирования файлов с использованием только системных вызовов (open, read, write).
2. Проведение аудита памяти для заданной программы с использованием valgrind и устранение.

найденных ошибок.

5.3 Методические материалы, определяющие процедуры оценивания знаний, умений, владений (навыков и (или) практического опыта деятельности)

5.3.1 Условия допуска обучающегося к промежуточной аттестации и порядок ликвидации академической задолженности

Проведение промежуточной аттестации регламентировано локальным нормативным актом СПбГУПТД «Положение о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся»

5.3.2 Форма проведения промежуточной аттестации по дисциплине

Устная ☐ + Письменная ☐ Компьютерное тестирование ☐ Иная ☐

5.3.3 Особенности проведения промежуточной аттестации по дисциплине

Время на подготовку к устному собеседованию составляет 30 минут;
Выполнение тестовых заданий осуществляется за 30 минут.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Учебная литература

Автор	Заглавие	Издательство	Год издания	Ссылка
6.1.1 Основная учебная литература				
Бизянов, Е. Е.	Системное программирование	Москва, Вологда: Инфра-Инженерия	2024	https://www.iprbooks.hop.ru/143252.html
Гунько, А. В.	Системное программирование в среде Linux	Новосибирск: Новосибирский государственный технический университет	2020	http://www.iprbookshop.ru/98735.html
Страуструп, Б.	Язык программирования C++ для профессионалов	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2025	https://www.iprbooks.hop.ru/146412.html
Городняя, Л. В.	Основы функционального программирования	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2025	https://www.iprbooks.hop.ru/146377.html
6.1.2 Дополнительная учебная литература				
Назаров, С. В., Широков, А. И.	Современные операционные системы	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2024	https://www.iprbooks.hop.ru/133980.html
Курячий, Г. В.	Операционная система UNIX	Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа	2024	https://www.iprbooks.hop.ru/142280.html

6.2 Перечень профессиональных баз данных и информационно-справочных систем

Информационная система «Единое окно доступа к образовательным ресурсам. Раздел. Информатика и информационные технологии» [Электронный ресурс]. URL: http://window.edu.ru/catalog/?p_rubr=2.2.75.6
Электронно-библиотечная система IPRbooks [Электронный ресурс]. URL: <http://www.iprbookshop.ru/>

6.3 Перечень лицензионного и свободно распространяемого программного обеспечения

MicrosoftOfficeProfessional
Microsoft Windows
Microsoft Visual C++ 2010 Express

6.4 Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Аудитория	Оснащение
Компьютерный класс	Мультимедийное оборудование, компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду
Лекционная аудитория	Мультимедийное оборудование, специализированная мебель, доска