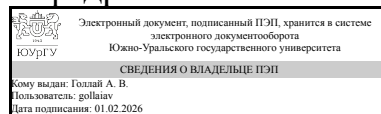


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Заведующий выпускающей
кафедрой



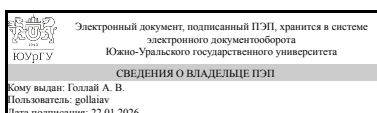
А. В. Голлай

РАБОЧАЯ ПРОГРАММА

дисциплины 1.Ф.ПО.11 Построение высоконагруженных и отказоустойчивых систем
для направления 09.03.02 Информационные системы и технологии
уровень Бакалавриат
профиль подготовки ИТ-инженерия
форма обучения очная
кафедра-разработчик Центр подготовки топ-специалистов в сфере ИТ "Цифровой Урал"

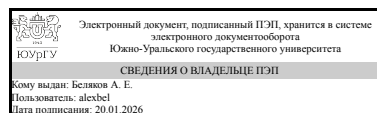
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, утверждённым приказом Минобрнауки от 19.09.2017 № 926

Зав.кафедрой разработчика,
Д.техн.н., доц.



А. В. Голлай

Разработчик программы,
старший преподаватель



А. Е. Беляков

1. Цели и задачи дисциплины

Цель дисциплины: Формирование у обучающихся системных знаний и практических навыков в области разработки, оптимизации и отладки системного программного обеспечения для создания высоконагруженных и отказоустойчивых вычислительных систем, работающих на уровне операционных систем, загрузчиков и аппаратного обеспечения. Задачи дисциплины: 1. Изучить архитектуру ядра Linux (подсистемы управления памятью, процессами, вводом-выводом, сетью); принципы работы и особенности встраиваемых и реального времени (ОСРВ) операционных систем. 2. Научиться анализировать и портировать низкоуровневый код между различными загрузчиками (U-Boot, GRUB, Barebox) и операционными системами; разрабатывать и отлаживать драйверы и системные компоненты на стыке программного обеспечения и аппаратного обеспечения (ПО и АО). 3. Получить практическими навыками (иметь практический опыт) работы с инструментами кросс-компиляции, отладки ядра и загрузчиков, профилирования производительности и диагностики системных сбоев в условиях высокой нагрузки.

Краткое содержание дисциплины

Дисциплина изучает полный стек технологий, необходимых для создания надежных и производительных системных решений. Курс охватывает внутреннее устройство современного ядра Linux, методы обеспечения отказоустойчивости (репликация, кластеризация, мониторинг), принципы работы встраиваемых систем и ОСРВ. Особое внимание уделяется практической работе с загрузчиками, портированию кода, разработке под специфичное аппаратное обеспечение и оптимизации системного ПО под высокие нагрузки. Курс носит ярко выраженный проектно-практический характер.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ПК-11 Способен разрабатывать, оптимизировать и отлаживать системное программное обеспечение	Знает: архитектуру ядра Linux; принципы работы встраиваемых операционных систем Умеет: портировать код между различными вариантами загрузчиков и ОС; выполнять разработку на стыке программного обеспечения и оборудования Имеет практический опыт: работы с ОС, загрузчиками и аппаратурой

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
Основы операционных систем и администрирование Linux, Языки программирования низкого уровня, Архитектура ЭВМ	Не предусмотрены

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
Архитектура ЭВМ	Знает: методику настройки и наладки программно-аппаратных комплексов, возможности существующей программно-технической архитектуры, возможности современных и перспективных средств разработки программных продуктов, технических средств, методологии разработки компьютерного программного обеспечения и технологии программирования, методы оптимизации (кэш, память, тактовая частота процессора), инструменты профилирования, архитектуру компиляторов, оптимизации для выбранных архитектур процессоров, методику настройки и наладки программно-аппаратных комплексов Умеет: производить коллективную настройку и наладку программно-аппаратных комплексов, проводить сбор и систематизацию требований к компьютерному программному обеспечению, выявлять взаимосвязи и документировать требования к компьютерному программному обеспечению, вырабатывать варианты реализации требований к компьютерному программному обеспечению, анализировать бенчмарки, выявлять узкие места, разрабатывать код, оптимизированный для выбранной аппаратной архитектуры Имеет практический опыт: коллективной настройки и наладки программно-аппаратных комплексов
Основы операционных систем и администрирование Linux	Знает: основные концепции и принципы построения современных операционных систем; архитектуру, функции и назначение подсистем ОС (управление процессами, памятью, вводом-выводом, файловыми системами); особенности организации многопользовательских и многозадачных систем, основные концепции современных операционных систем; особенности архитектуры и организации ОС семейства Linux, интерфейсы взаимодействия с внешней средой, архитектуру ядра Linux; принципы работы встраиваемых операционных систем, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий, основные широко распространенные операционные системы, принципы их работы Умеет: применять стандартные инструменты современных операционных систем (в том числе Linux и Windows) при решении задач профессиональной деятельности; использовать интерфейсы командной строки и API для

	взаимодействия с ОС, использовать стандартные инструменты современных ОС при решении задач профессиональной деятельности; устанавливать, настраивать и администрировать операционные системы на базе Linux, портировать код между различными вариантами загрузчиков и ОС; выполнять разработку на стыке программного обеспечения и оборудования, применять программные среды разработки информационных систем и технологий для решения прикладных задач различных классов, устанавливать и настраивать операционную систему, создавать прикладные программы в терминах API ОС Имеет практический опыт: работы с основными интерфейсами ОС (командный интерфейс и API); средствами диагностики и анализа состояния системы; приёмами управления вычислительными ресурсами ОС, навыками практической работы в среде Linux, владения технологиями, использования основных видов интерфейсов операционной системы Windows
Языки программирования низкого уровня	Знает: методы и средства оптимизации производительности компьютерного программного обеспечения; современные инструменты оптимизации производительности программного обеспечения, архитектуру современных процессоров, особенности работы с оборудованием процессора; принципы взаимодействия ПО с аппаратурой Умеет: оптимизировать программный код с использованием специализированных программных средств; вырабатывать варианты оптимизации производительности компьютерного программного обеспечения; проводить оценку и обоснование рекомендуемых решений по оптимизации производительности, разрабатывать низкоуровневый код для встроенного программного обеспечения и драйверов Имеет практический опыт:

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч., 72,5 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам
		в часах
		Номер семестра
		7
Общая трудоёмкость дисциплины	108	108
Аудиторные занятия:	64	64
Лекции (Л)	32	32

Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	32	32
Лабораторные работы (ЛР)	0	0
Самостоятельная работа (СРС)	35,5	35,5
Выполнение практических заданий	30	30
Подготовка к экзамену	5,5	5,5
Консультации и промежуточная аттестация	8,5	8,5
Вид контроля (зачет, диф.зачет, экзамен)	-	экзамен

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Архитектура ядра Linux и методы оптимизации	18	10	8	0
2	Отказоустойчивость на системном уровне	16	8	8	0
3	Встраиваемые системы и ОСРВ. Загрузчики и низкоуровневая разработка	18	10	8	0
4	Практикум по портированию и низкоуровневой отладке	12	4	8	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол-во часов
1	1	Введение. Особенности высоконагруженных и отказоустойчивых систем. Обзор стека технологий	2
2	1	Архитектура ядра Linux: системные вызовы, управление процессами и памятью	2
3	1	Подсистемы ввода-вывода и сетевой стек ядра. Модули ядра	2
4	1	Инструменты отладки и профилирования ядра (ftrace, perf, kprobes)	2
5	1	Методы оптимизации производительности ядра: тонкая настройка параметров, изоляция ресурсов (cgroups, namespaces)	2
6	2	Принципы отказоустойчивости: репликация состояния, кворумы, split-brain	2
7	2	Кластерные технологии на базе Linux (Pacemaker/Corosync, DRBD)	2
8	2	Мониторинг и алертинг для системного ПО (Prometheus, Grafana, кастомные экспортеры)	2
9	2	Механизмы обеспечения высокой доступности в сетевых стеках (bonding, teaming, keepalived)	2
10	3	Особенности встраиваемых систем и ОСРВ (FreeRTOS, Zephyr, Linux-based RT)	2
11	3	Архитектура и этапы загрузки системы: от ROM до пользовательского пространства	2
12	3	Современные загрузчики: U-Boot, GRUB2, Barebox. Device Tree (DTS)	2
13	3	Кросс-компиляция и отладка для целевых платформ (gdb, gdbserver, OpenOCD)	2
14	3	Разработка на стыке ПО и АО: основы работы с периферией, прерываниями, памятью	2
15	4	Методология портирования кода: абстракция аппаратного обеспечения, условная компиляция	1
16	4	Работа с энергонезависимой памятью (EEPROM, Flash)	1

17	4	Принципы отказоустойчивости на аппаратном уровне (watchdog timer, ECC память)	1
18	4	Паттерны для высоконагруженного системного ПО: lock-free алгоритмы, кэширование	1

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	1	Работа с системными вызовами strace, ltrace. Анализ работы процесса	4
2	1	Сборка кастомного ядра Linux. Настройка параметров ядра для специфичной нагрузки	4
3	2	Настройка отказоустойчивого сетевого интерфейса (bonding)	4
4	2	Развертывание простого кластера высокой доступности (HA) с виртуальным IP-адресом	4
5	3	Работа с Device Tree: модификация дерева устройств для виртуальной платформы (QEMU)	4
6	3	Сборка и загрузка минимального образа ОС (Buildroot/Yocto) через U-Boot в эмуляторе	4
7	4	Написание простого watchdog-драйвера для пользовательского пространства	4
8	4	Портируемость: адаптация кода драйвера символического устройства под разные версии ядра	4

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Выполнение практических заданий	Слайды лекций, методические указания к заданиям	7	30
Подготовка к экзамену	Слайды лекций, методические указания к заданиям	7	5,5

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-мestr	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учитывается в ПА
------	----------	--------------	-----------------------------------	-----	------------	---------------------------	------------------

1	7	Текущий контроль	ЛР №1	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	экзамен
2	7	Текущий контроль	ЛР №2	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	экзамен
3	7	Текущий контроль	ЛР №3	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	экзамен
4	7	Текущий контроль	ЛР №4	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена	экзамен

						в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	
5	7	Текущий контроль	ЛР №5	20	20	Баллы начисляются по результатам демонстрации программы и ответов на вопросы по исходному коду. Шкала начисления дискретная, оформлена в виде набора критериев "рубрика" системы MOODLE. Баллы за каждый критерий либо засчитываются в итоговую сумму баллов полностью, либо обнуляются. Критерии: 1. Наличие компилируемого без ошибок программного кода задания - 5 балла. 2. Понимание и комментирование функциональных блоков кода студентом - 5 баллов. 3. Ответы на вопросы по заданию - 10 балла.	экзамен
6	7	Промежуточная аттестация	Экзамен	-	100	По результатам ответов на вопросы устного зачета. Первый теоретический вопрос - максимум 30 баллов, второй теоретический вопрос - максимум 30 баллов, третий теоретический вопрос - максимум 40 баллов. За успешное выполнение ставится максимум баллов по заданию, если задание выполнено частично или с ошибками, то ставится от 0 до максимума по заданию пропорционально выполненному объему. Длительность зачета - 2 академических часа.	экзамен

6.2. Процедура проведения, критерии оценивания

Вид промежуточной аттестации	Процедура проведения	Критерии оценивания
экзамен	При оценивании результатов учебной деятельности обучающегося по дисциплине используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся (Положение о БРС утверждено приказом ректора от 24.05.2019 г. № 179, в редакции приказа ректора от 10.03.2022 г. № 25-13/09). Оценка за дисциплину	В соответствии с пп. 2.5, 2.6 Положения

	формируется на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля. Отлично: Величина рейтинга обучающегося по дисциплине 85...100 %. Хорошо: Величина рейтинга обучающегося по дисциплине 75...84 %. Удовлетворительно: Величина рейтинга обучающегося по дисциплине 60...74 %. Неудовлетворительно: Величина рейтинга обучающегося по дисциплине 0...59 %. Если студент согласен с оценкой, полученной по результатам текущего контроля, то он может в день, предшествующий промежуточной аттестации дать свое согласие в личном кабинете. В случае явки студента на промежуточную аттестацию, давшего свое согласие на автомат в личном кабинете, студент имеет право пройти мероприятия текущего контроля по дисциплине на промежуточной аттестации для улучшения своего рейтинга в день ее проведения. Снижение оценки в этом случае запрещено. Если студент не дал согласия в личном кабинете, то он может согласиться с оценкой лично на промежуточной аттестации в день ее проведения. Если студент не согласен с оценкой, то он имеет право пройти контрольно-рейтинговые мероприятия на промежуточной аттестации для улучшения своего рейтинга в день ее проведения. Фиксация результатов учебной деятельности по дисциплине проводится в день промежуточной аттестации на основе согласия студента, данного им в личном кабинете. При отсутствии согласия в журнале дисциплины фиксация результатов происходит при личном присутствии студента. Если студент не дал согласие в личном кабинете и не явился на промежуточную аттестацию – ему выставляется «неявка».	
--	--	--

6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ					
		1	2	3	4	5	6
ПК-11	Знает: архитектуру ядра Linux; принципы работы встраиваемых операционных систем	+	+	+	+	+	+
ПК-11	Умеет: портировать код между различными вариантами загрузчиков и ОС; выполнять разработку на стыке программного обеспечения и оборудования	+	+	+	+	+	+
ПК-11	Имеет практический опыт: работы с ОС, загрузчиками и аппаратурой	+	+	+	+	+	+

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) *основная литература:*

Не предусмотрена

б) *дополнительная литература:*

Не предусмотрена

в) *отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:*

Не предусмотрены

г) методические указания для студентов по освоению дисциплины:

1. Методические рекомендации по выполнению практических занятий

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. Методические рекомендации по выполнению практических занятий

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	ЭБС издательства Лань	Тормозов, В. С. Основы работы в операционной системе Linux : учебное пособие / В. С. Тормозов, А. Л. Золкин. — Самара : , 2023. — 66 с. — ISBN 978-5-907359-20-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/388838 . — Режим доступа: для авториз. пользователей.
2	Основная литература	ЭБС издательства Лань	Малахов, С. В. Принципы работы операционной системы Linux. Bash-скрипты : учебное пособие / С. В. Малахов, Д. О. Якупов. — Самара : ПГУТИ, 2024. — 134 с. — ISBN 978-5-907336-50-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/463574 . — Режим доступа: для авториз. пользователей.
3	Основная литература	ЭБС издательства Лань	Гулько, А. В. Системное программирование в среде Linux : учебное пособие / А. В. Гулько. — Новосибирск : НГТУ, 2020. — 235 с. — ISBN 978-5-7782-4160-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/152228 . — Режим доступа: для авториз. пользователей.

Перечень используемого программного обеспечения:

1. Rocky Enterprise Software Foundation (RESF)-Rocky Linux(бессрочно)
2. Docker-Docker(бессрочно)

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Вид занятий	№ ауд.	Основное оборудование, стенды, макеты, компьютерная техника, предустановленное программное обеспечение, используемое для различных видов занятий
Лекции	240 (36)	Компьютеры, ОС Windows, ОС Linux, VS Code, Docker, доска, проектор
Практические занятия и семинары	809 (36)	Компьютеры, ОС Windows, ОС Linux, VS Code, Docker

