

**Федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»**

УТВЕРЖДАЮ
Директор ИКНК
_____ Д.П. Зегжда
«17» июня 2024 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Системное программное обеспечение GNU/Linux»

| | |
|---|--|
| Разработчик | Высшая школа программной инженерии |
| Направление (специальность) подготовки | 09.03.04 Программная инженерия |
| Наименование ООП | 09.03.04_01 Технология разработки и сопровождения качественного программного продукта |
| Квалификация (степень) выпускника | бакалавр |
| Образовательный стандарт | СУОС |
| Форма обучения | Очная |

СОГЛАСОВАНО

Руководитель ОП

_____ А.В. Петров

«21» мая 2024 г.

Соответствует СУОС

Утверждена протоколом заседания
высшей школы "ВШПИ"

от «21» мая 2024 г. № № 1

РПД разработали:

Старший преподаватель С.А. Фёдоров

Старший преподаватель А.В. Петров

1. Цели и планируемые результаты изучения дисциплины

Цели освоения дисциплины

1. получение обучающимися необходимых знаний, умений и навыков для работы в пользовательском интерфейсе операционной среды GNU/Linux
2. получение обучающимися необходимых знаний, умений и навыков для программирования на уровне ядра Linux

Результаты обучения выпускника

| Код | Результат обучения (компетенция) выпускника ООП |
|---------------|---|
| ОПК-3 | Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности |
| ИД-3 ОПК-3 | Использует системные утилиты для решения профессиональных задач |
| ОПК-5 | Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем |
| ИД-1 ОПК-5 | Устанавливает и настраивает системное и прикладное программное обеспечение |

Планируемые результаты изучения дисциплины

знания:

- Знает возможности системных сервисов и языки для их применения
- Знает принципы установки ПО на различных операционных системах

умения:

- Умеет проводить инсталляцию программных пакетов для различных процессорных архитектур
- Умеет применять существующие и разрабатывать собственные программы с использованием системных сервисов

навыки:

- Владеет основными подходами к разработке системных инструментов

2. Место дисциплины в структуре ООП

В учебном плане дисциплина «Системное программное обеспечение GNU/Linux» относится к модулю «Модуль цифровых компетенций» / «Системное программное обеспечение».

Изучение дисциплины требует знания школьной программы, успешной сдачи вступительных или единых государственных экзаменов.

3. Распределение трудоёмкости освоения дисциплины по видам учебной работы и формы текущего контроля и промежуточной аттестации

3.1. Виды учебной работы

| Виды учебной работы | Трудоёмкость по семестрам |
|--|---------------------------|
| | Очная форма |
| Лекционные занятия | 30 |
| Лабораторные занятия | 30 |
| Самостоятельная работа | 53 |
| Часы на контроль | 16 |
| Промежуточная аттестация (экзамен) | 11 |
| Промежуточная аттестация (зачет) | 4 |
| Общая трудоёмкость освоения дисциплины | 144, ач |
| | 4, зет |

3.2. Формы текущего контроля и промежуточной аттестации

| Формы текущего контроля и промежуточной аттестации | Количество по семестрам |
|--|-------------------------|
| | Очная форма |
| Текущий контроль | |
| Расчетно-графические работы, шт. | 1 |
| Промежуточная аттестация | |
| Зачеты, шт. | 1 |
| Экзамены, шт. | 1 |

4. Содержание и результаты обучения

4.1 Разделы дисциплины и виды учебной работы

| № раздела | Разделы дисциплины, мероприятия текущего контроля | Очная форма | | |
|--------------|---|-------------|------------|-----------|
| | | Лек, ач | Лаб, ач | СР, ач |

| | | | | |
|--|--|---------|----|----|
| 1. | Компиляторы. Интерактивные системы | 2 | 5 | 5 |
| 2. | Программирование ядра Linux | | | |
| 2.1. | Управление задачами. Управление памятью. Управление вводом-выводом | 2 | 5 | 6 |
| 2.2. | Управление файлами | 2 | 5 | 10 |
| 2.3. | Средства трассировки и отладки программ | 2 | 5 | 1 |
| 3. | Дополнительные главы | | | |
| 3.1. | Встраиваемые и системы реального времени | 8 | 5 | 4 |
| 3.2. | Задачи администрирования, безопасность и защита | 4 | 5 | 4 |
| 3.3. | Основы построения сетей и сетевые службы | 4 | 0 | 4 |
| 3.4. | Отказоустойчивость | 0 | 0 | 8 |
| 4. | Оценка производительности системы | 6 | 0 | 6 |
| Итого по видам учебной работы: | | 30 | 30 | 53 |
| Зачеты, ач | | | | 9 |
| Экзамены, ач | | | | 12 |
| Часы на контроль, ач | | | | 16 |
| Промежуточная аттестация (экзамен) | | 11 | | |
| Промежуточная аттестация (зачет) | | 4 | | |
| Общая трудоёмкость освоения: ач / зет | | 144 / 4 | | |

4.2. Содержание разделов и результаты изучения дисциплины

| Раздел дисциплины | Содержание |
|--|---|
| 1. Компиляторы. Интерактивные системы | <p>Знания на уровне понятий, определений, формулировок Компиляторы и средства разработки. GNU Compiler Collection (GCC). Компиляторы корпорации Intel и Sun.</p> <p>Оптимизация приложений в машинных кодах. Оптимизации под типы приложений. Процессорные оптимизации. Дополнительные оптимизации (работа с плавающей точкой, математические библиотеки).</p> <p>Системные методы оптимизации приложений. Оптимизации уровня циклов. Межпроцедурные оптимизации. Ведомые профилем оптимизации (компиляция с обратной связью). Итегрированные среды разработки. Eclipse. CDT. Основы программирования.</p> |
| 2. Программирование ядра Linux | |
| 2.1. Управление задачами. Управление памятью. Управление вводом-выводом | <p>Знания на уровне понятий, определений, формулировок Ядро. Контроль и управление ядром и подгружаемыми модулями ядра. Настройка, сборка и установка ядра и подгружаемых модулей ядра из исходников.</p> <p>Оптимизация ядра Linux. Оптимизация ядра, включая оптимизацию под конкретный процессор и эффективную компиляцию на многоядерном процессоре.</p> <p>Программирование на уровне ядра и модули ядра. Модули ядра и прикладные программы. Функции, которые доступны из модулей. Пространство пользователя и пространство ядра. Пространство имен. Адресное пространство.</p> <p>Управление устройствами. Характеристики последовательных и параллельных устройств. Абстрагирование от различий между устройствами. Стратегии буферизации. Прямой доступ к памяти. Восстановление после сбоев. Модель представления устройств в Linux.</p> <p>Драйверы устройств. Драйвер как модуль ядра. Старший и младший номер устройства. Драйвер на примере клавиатуры и сетевой карты.</p> |

| | |
|--|--|
| 2.2. Управление файлами | <p>Знания на уровне понятий, определений, формулировок Создание разделов и файловых систем. Файловые системы: деление на разделы, поддержка целостности файловой системы, монтирование и размонтирование, виртуальные файловые системы. Стандартные методы реализации. Файлы, проецируемые в память. Специализированные файловые системы. Файлы: данные, метаданные, операции, организация, буферизация, файлы с последовательным и произвольным доступом. Директории: содержимое и структура. Управление дисковыми квотами. Использование прав доступа для управления доступом к файлам. Управление владельцами файлов. Создание и изменение жестких и символических ссылок. Поиск системных файлов и размещение файлов в правильное местоположение.</p> |
| 2.3. Средства трассировки и отладки программ | <p>Знания на уровне понятий, определений, формулировок Сложности отладки ядерного кода. Отладка через печать. Получение отладочной информации по запросу. Отладка с использованием strace – трассировка системных вызовов. Отладка системных ошибок (System Fault). Использование klogd. Использование ksymoops. Зависание системы. Отладчики и дополнительные отладочные инструменты. Использование gdb. Отладчик ядра kdb. Патч IKD (Integrated Kernel Debugger). Патч kgdb.</p> |
| 3. Дополнительные главы | |
| 3.1. Встраиваемые и системы реального времени | <p>Знания на уровне понятий, определений, формулировок Встраиваемая Linux. ucLinux. Отличия от базового ядра. Системы без блока управления памятью MMU (без виртуальной памяти). Исходный код встраиваемой Linux. Компиляция и оптимизация кода. Настройка и выбор исходных файлов для конкретной сборки. Встраивание Linux в микроконтроллеры, процессоры, программируемые системы на кристалле и другие высокотехнологичные устройства (фирм Altera, Analog Devices или Motorola).</p> <p>Встраиваемые системы реального времени. Планирование процессов и задач. Требования к управлению памятью и дисковыми устройствами в среде реального времени. Сбои, риски и восстановление. Специфические проблемы систем реального времени. Работа в режиме реального времени. Поточковая обработка данных.</p> |

| | |
|---|--|
| 3.2. Задачи администрирования, безопасность и защита | <p>Знания на уровне понятий, определений, формулировок Обзор безопасности системы. Механизмы и политики разграничения прав доступа. Управление учётными записями пользователей и групп и соответствующими системными файлами. Методы и устройства обеспечения безопасности. Защита, доступ и аутентификация. Настройка переменных окружения пользователя и системы. Конфигурирование и использование файлов журналирования системы в целях администрирования и безопасности. Расширенные регулярные выражения. Автоматизация задач системного администрирования с помощью составления расписания заданий для запуска в будущем. Следование стратегии эффективного резервирования данных. Управление восстановлением. Поддержание системного времени. Модели защиты. Выполнение административных задач по безопасности. Установка безопасности хоста. Установка уровня защиты пользователя. Защита памяти. Шифрование данных.</p> |
| 3.3. Основы построения сетей и сетевые службы | <p>Знания на уровне понятий, определений, формулировок Основы построения сетей. Основы TCP/IP. Конфигурация и поиск неисправностей TCP/IP. Конфигурация Linux как PPP-клиента. Сетевые службы. Конфигурация и управление xinetd, inetd и смежными службами. Работы и выполнение основной конфигурации Mail Transfer Agent (MTA) и Apache. Грамотное управление демонами NFS и SAMBA. Установка и конфигурация основных служб DNS. Установка безопасного командного процессора OpenSSH.</p> |
| 3.4. Отказоустойчивость | <p>Знания на уровне понятий, определений, формулировок Фундаментальные понятия: надежность и доступность системы. Пространственная и временная избыточность. Методы реализации отказоустойчивости. Примеры надежных систем.</p> |
| 4. Оценка производительности системы | <p>Почему требуется оценивать производительность системы. Какие параметры системы необходимо оценивать. Стратегии кэширования, подкачки, планирования, управления памятью, безопасности и другие. Модели оценки: детерминированная, аналитическая, имитационная и зависящая от реализации. Методы сбора данных для оценки (механизмы профилирования и трассировки).</p> |

5. Образовательные технологии

1. традиционные технологии (лекции в сочетании с практическими занятиями, семинарами и с лабораторными работами, самостоятельное изучение определённых разделов)

2. опережающая самостоятельная работа (самостоятельное изучение определённых разделов, выполнение лабораторных работ)

6. Лабораторный практикум

| № раздела | Наименование лабораторных работ | Трудоемкость, ач |
|--------------|---|---------------------|
| | | Очная форма |
| 1. | Компилятор GCC. Оптимизация приложений | 5 |
| 2. | Конфигурирование, сборка и установка ядра | 5 |
| 3. | Модули ядра | 5 |
| 4. | Драйвер простого устройства | 5 |
| 5. | Взаимодействие с драйверами устройств | 5 |
| 6. | Оценка производительности системы | 5 |
| Итого часов | | 30 |

7. Практические занятия

Не предусмотрено

8. Организация и учебно-методическое обеспечение самостоятельной работы

Примерное распределение времени самостоятельной работы студентов

| Вид самостоятельной работы | Примерная трудоемкость, ач |
|--|----------------------------------|
| | Очная форма |
| Текущая СР | |
| работа с лекционным материалом, с учебной литературой | 4 |
| опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях) | 4 |
| самостоятельное изучение разделов дисциплины | 4 |
| выполнение домашних заданий, домашних контрольных работ | 4 |
| подготовка к лабораторным работам, к практическим и семинарским занятиям | 4 |
| подготовка к контрольным работам, коллоквиумам | 8 |
| Итого текущей СР: | 28 |
| Творческая проблемно-ориентированная СР | |
| выполнение расчётно-графических работ | 4 |
| выполнение курсового проекта или курсовой работы | 4 |
| поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме | 1 |
| работа над междисциплинарным проектом | 5 |
| исследовательская работа, участие в конференциях, семинарах, олимпиадах | 2 |
| анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных | 4 |
| Итого творческой СР: | 20 |
| Общая трудоемкость СР: | 53 |

9. Учебно-методическое обеспечение дисциплины

9.1. Адрес сайта курса

<https://dl.spbstu.ru/course/view.php?id=4446>

9.2. Рекомендуемая литература

Основная литература

| № | Автор, название, место издания, издательство, год (годы) издания | Год изд. | Источник |
|---|--|----------|-----------|
| 1 | Померанц О. Ядро Linux. Программирование модулей: Москва: КУДИЦ-ОБРАЗ, 2000. | 2000 | ИБК СПбПУ |
| 2 | Бовет Д., Чезати М. Ядро Linux: СПб.: БХВ-Петербург, 2007. | 2007 | ИБК СПбПУ |

Ресурсы Интернета

1. Linux kernel: <http://www.kernel.org>

9.3. Технические средства обеспечения дисциплины

Свободно распространяемые программные инструменты для разработки программного обеспечения. Возможны демонстрационные анимационные flash-ролики для обучения различным разделам дисциплины.

10. Материально-техническое обеспечение дисциплины

Аудитория с мультимедийным проектором. Компьютерный класс, связанный в локальную сеть, с установленной операционной системой GNU/Linux.

11. Критерии оценивания и оценочные средства

11.1. Критерии оценивания

Для дисциплины «Системное программное обеспечение GNU/Linux» предусмотрены следующие формы аттестации: зачёт, экзамен. Дисциплина реализуется с применением системы индивидуальных достижений.

Текущий контроль успеваемости

Максимальное значение персонального суммарного результата обучения (ПСРО) по приведенной шкале - 100 баллов

Максимальное количество баллов приведенной шкалы по результатам прохождения двух точек контроля - 80 баллов.

Подробное описание правил проведения текущего контроля с указанием баллов по каждому контрольному мероприятию и критериев выставления оценки размещается в СДО в навигационном курсе дисциплины.

Промежуточная аттестация по дисциплине

Максимальное количество баллов по результатам проведения аттестационного испытания в период промежуточной аттестации – 20 баллов приведенной шкалы.

Промежуточная аттестация по дисциплине проводится в соответствии с расписанием.

В качестве критерия оценивания знаний и умений по дисциплине производится оценивание качества выполненных заданий по каждой лабораторной работе и оценивание качества самостоятельных работ.

Результаты промежуточной аттестации, определяются на основе баллов, набранных в рамках применения, СИД

| Баллы по приведенной шкале в рамках применения СИД (ПСРО+ ПА) | Оценка по результатам промежуточной аттестации |
|---|--|
| | Экзамен/диф.зачет/зачет |
| 0 - 60 баллов | Неудовлетворительно/не зачтено |
| 61 - 75 баллов | Удовлетворительно/зачтено |
| 76 - 89 баллов | Хорошо/зачтено |

| Баллы по приведенной шкале в рамках применения СИД (ПСРО+ ПА) | Оценка по результатам промежуточной аттестации |
|---|--|
| | Экзамен/диф.зачет/зачет |
| 90 и более | Отлично/зачтено |

11.2. Оценочные средства

Оценочные средства по дисциплине представлены в фонде оценочных средств, который является неотъемлемой частью основной образовательной программы и размещается в электронной информационно-образовательной среде СПбПУ на портале etk.spbstu.ru

12. Методические рекомендации по организации изучения дисциплины

Методические рекомендации по каждому разделу приводятся в лекциях в качестве дополнительных слайдов, со ссылкой на рекомендуемую литературу для самостоятельного изучения, и предварительного списка вопросов, заданий для самостоятельной работы.

Успешное изучение курса требует посещения лекций, активной работы на лабораторных работах, выполнения всех учебных заданий преподавателя, ознакомления с основной и дополнительной литературой.

Во время лекции студент должен вести краткий конспект. Если какие-то материалы конспекта вызывают затруднения, необходимо постараться найти ответы на затруднительные вопросы, используя рекомендуемую литературу. Если студенту самостоятельно не удалось разобраться в материале, необходимо сформулировать вопросы и обратиться за помощью к преподавателю на консультации или ближайшей лекции. Обучающемуся необходимо регулярно отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам.

Лабораторные работы составляют важную часть профессиональной подготовки студентов. Они направлены на подтверждение теоретических положений и формирование учебных и профессиональных практических умений.

При подготовке к экзамену в дополнение к изучению конспектов лекций и учебных пособий, необходимо пользоваться учебной литературой, рекомендованной в настоящей программе. При подготовке к экзамену нужно изучить теорию: определения всех понятий и подходы к оцениванию до состояния понимания материала и самостоятельно решить по несколько типовых задач из каждой темы. При решении задач всегда необходимо уметь качественно интерпретировать итог решения.

13. Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.