

Белорусский государственный университет

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

О.Г. Прохоренко

«05» июля 2023 г.

Регистрационный № УД- 12710/уч.

Системное программирование

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

**1-31 03 07 Прикладная информатика (по направлениям)
направления специальности**

**1-31 03 07-01 Прикладная информатика (программное обеспечение
компьютерных систем)**

2023 г.

Учебная программа составлена на основе ОСВО 1-31 03 07-2021, типового учебного плана № G 31- 1- 030/пр-тип., утверждённого 01.07.2021, учебных планов БГУ №G31-1-034/уч. от 23.07.2021, №G31-1-023/уч. ин. от 09.08.2021.

СОСТАВИТЕЛИ:

Давидовская М. И., старший преподаватель кафедры технологий программирования факультета прикладной математики и информатики Белорусского государственного университета.

РЕЦЕНЗЕНТЫ:

Н. В. Лапицкая, заведующая кафедрой программного обеспечения информационных технологий УО «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент.

Т. В. Соболева, доцент кафедры многопроцессорных систем и сетей факультета прикладной математики и информатики Белорусского государственного университета, кандидат физико-математических наук.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой технологий программирования
(протокол № 16 от 18 мая 2023 г.);

Научно-методическим советом БГУ
(протокол № 9 от 29 июня 2023 г.)

Заведующий кафедрой



А.Н. Курбацкий

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Цель учебной дисциплины — подготовка специалиста по проектированию и разработке системных программ на языке С для операционных систем на основе ядра Linux. Формирование и развитие системы знаний по теоретическим и практическим аспектам проектирования и разработки системных программ, получаемых будущими специалистами в процессе учёбы и необходимых им в дальнейшем для успешной профессиональной деятельности в области разработки операционных систем и системного программирования.

Задачи учебной дисциплины:

1. Изучить инструментарий операционной системы Linux для системного администрирования и обеспечения безопасности, программирования на языке сценариев bash и обработки данных с помощью awk, sed и других утилит;
2. Изучить системные вызовы Linux API и библиотечные функции.
3. Изучить подходы к проектированию библиотек.
4. Научить эффективно применять полученные навыки проектирования и разработки системного программного обеспечения в профессиональной деятельности;
5. Сформировать навыки проектирования системного программного обеспечения.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Дисциплина «Системное программирование» относится к модулю «Программирование» государственного компонента. Организационные, управленческие и производственные процессы в любой отрасли насыщены вычислительными устройствами. Важной составляющей процесса автоматизации производства является операционная система и системное программное обеспечение. Как следствие, повышаются требования к уровню компетентности выпускников вузов в области разработки системных программных средств. Без теоретических знаний и практических навыков системного администрирования и программирования в среде операционной системы Linux невозможно эффективно решать профессиональные задачи. Наиболее востребованными навыками являются управление пользователями, правами доступа и процессами в среде ОС Linux, автоматизация задач администрирования и обработки данных за счёт применения интерпретируемых сценариев, проектирование, разработка и отладка системных программ, проектирование и разработка системных библиотек.

Связи с другими учебными дисциплинами, включая учебные дисциплины компонента учреждения высшего образования, дисциплины специализации и др.

Содержание учебной программы соответствует уровню подготовки студентов к изучению дисциплины «Системное программирование» и основывается на следующих учебных дисциплинах: «Основы и методологии программирования», «Разработка кросс-платформенных приложений», «Операционные системы», «Технологии программирования для мобильных приложений».

Изучение дисциплины «Системное программирование» будет востребовано при изучении дисциплины «Программирование мобильных и встраиваемых систем», «Проектирование человеко-машинных интерфейсов» и других дисциплин модуля «Прикладное программирование», так же дисциплин специализации «Проектирование приложений под Linux», «Сетевые приложения в среде Linux», «Информационная безопасность мобильных приложений» и при выполнении курсовых и дипломных работ.

Требования к компетенциям

Освоение учебной дисциплины «Системное программирование» должно обеспечить формирование следующих универсальных, базовых профессиональных и специализированных компетенций:

***универсальные* компетенции:**

УК-1. Владеть основами исследовательской деятельности, осуществлять поиск, анализ и синтез информации.

УК-2. Решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий.

УК-4. Работать в команде, толерантно воспринимать социальные, этнические, конфессиональные, культурные и иные различия.

УК-5. Быть способным к саморазвитию и совершенствованию в профессиональной деятельности.

УК-6. Проявлять инициативу и адаптироваться к изменениям в профессиональной деятельности.

***базовые профессиональные* компетенции:**

БПК-2. Строить, анализировать и тестировать алгоритмы и программы решения типовых задач обработки информации с использованием структурного, объектно-ориентированного подходов и иных парадигм программирования.

БПК-4. Применять знания в области принципов функционирования, архитектур и программных реализация операционных систем, структурной организации компьютеров и компьютерных систем, методах обработки данных для выбора вычислительных средств решения практических задач.

***специализированные* компетенции:**

СК-5. Использовать программные средства и технологии для создания прикладного программного обеспечения.

В результате освоения учебной дисциплины студент должен:

знать:

- методы системного администрирования ОС Linux;

- методы обработки текстовых данных;
 - методы разработки сценариев на языке Bash;
 - методы проектирования и разработки статических, разделяемых и динамических библиотек;
 - методы разработки, отладки и тестирования системных программ;
- уметь:**
- анализировать требования, предъявляемые к сценариям и системным программам, и составлять техническое задание;
 - применять инструментарий для обработки текстовых данных;
 - создавать сценарии на языке Bash и тестировать их;
 - проектировать и разрабатывать системное программное обеспечение для ОС Linux;
- владеть:**
- навыками системного администрирования ОС Linux;
 - навыками разработки сценариев на языке Bash;
 - навыками разработки системного программного обеспечения в среде ОС Linux;
 - навыками анализа и проектирования системных программ.

Структура учебной дисциплины

Дисциплина изучается в 5 семестре. Всего на изучение учебной дисциплины «Системное программирование» отведено:

– для очной формы получения высшего образования – 108 часов, в том числе 68 аудиторных часов, из них: лекции – 34 часа, лабораторные занятия – 30 часов, управляемая самостоятельная работа – 4 часа.

Трудоёмкость учебной дисциплины составляет 3 зачётные единицы.

Форма промежуточной аттестации – зачёт.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Основы системного администрирования и программирования ОС Linux

Тема 1.1. Введение в системное программирование и ОС Linux

Операционная система. Интерфейс операционной системы. Задачи операционной системы. Архитектура Unix-подобных систем. Интерфейс системных вызовов.

Введение в системное программирование в среде Linux. Понятие системного вызова и библиотечной функции и их примеры. Разделение обязанностей библиотечных функций и системных вызовов.

Тема 1.2. Пользовательское окружение ОС Linux и управление пакетами

Операционная система Unix и история развития Unix-подобных операционных систем. Свободное программное обеспечение. Программное обеспечение с открытыми исходными кодами. Семейство BSD-семейство.

Операционная система GNU/Linux. Предпосылки возникновения. Дистрибутивы GNU/Linux и история развития. Ядро Linux, версии ядра и правила нумерации. Установка ОС Linux.

Пользовательский и программный интерфейс. Графический интерфейс пользователя и основные компоненты. Командный интерфейс. Терминал. Физическая и виртуальная консоль. Эмулятор терминала. Командный интерпретатор и его виды. Интерфейс командной строки, синтаксис команд и основные команды. Справочная документация.

Сжатие и архивация в ОС Linux. Сжатие и основные утилиты. Архивация и основные утилиты для архивации. Утилита tar. Утилиты gzip и gunzip.

Установка ПО в ОС Linux. Типы репозитория. Классификация ПО в системных репозиториях. Форматы пакетов в ОС Linux. Преимущества и недостатки классического формата пакетов. Универсальные форматы пакетов и их особенности на примере snap и flatpak. Формат пакета переносимых приложений AppImage. Установка ПО и пакетные менеджеры. Менеджер пакетов APT и управление пакетами приложений. Управление snap-пакетами, flatpak-пакетами и AppImage.

Тема 1.3. Основы управления файлами, пользователями, правами доступа и процессами

Конфигурационные файлы ОС Linux и файлы устройств. Подсистема TTY. Файлы конфигурации терминалов. Типы файлов и права доступа. Модель хранения данных в ОС Linux. Типы файлов. Жёсткие и символические ссылки. Форматы файлов.

Учётная запись пользователя. Типы пользователей. Ключевые параметры учётной записи. Системы управления учётными записями. Файл /etc/passwd. Файл /etc/shadow. Хеширование пароля в ОС Linux. Управление пользователями. Основные команды управления учётными записями. Библиотека PAM и управление пользователями. Модели управления правами доступа. Модель дискреционного контроля доступа. Классы пользователей и

разрешений. Права доступа на файлы и каталоги. Специальные разрешения. Основные команды управления правами доступа. Маска разрешений. Списки контроля доступа ACL и их применение. Модель обязательного контроля доступа на примере AppArmor, SELinux.

Идея многозадачности. Процесс, задание, поток. Процесс и его жизненный цикл. Классификация процессов. Группы процессов и сессии. Атрибуты и идентификаторы процесса. Мониторинг процессов. Управление заданиями. Потоки и их преимущества. Файловая система /proc. Сигналы и их применение для управления процессами и заданиями. Приоритет и специальный приоритет (коэффициент уступчивости) процесса и управление ими. Стандартные потоки ввода-вывода и их перенаправление.

Тема 1.4. Инструментальные средства обработки текстовых данных и программирования

Обработка текстовых данных в Linux. Метасимволы POSIX BRE. Версии команды grep. Фильтрация по строкам и поиск с помощью grep. Рекурсивный поиск с rgrep. Фильтрация по блокам с помощью cut.

Программируемые фильтры на примере awk. Язык сценариев AWK для управления данными. Интерпретаторы языка AWK. Синтаксис языка AWK. Переменные и константы в AWK. Поля в AWK. Встроенные функции и переменные AWK. Структура программы на языке AWK. Операторы сравнения и логические операции. Арифметические операторы. Ветвления и циклы. Рекомендации по разработке сценариев на языке AWK.

Потоковый редактор sed и его характеристика. Синтаксис sed. Обработка потока в sed и правила обработки. Сценарии sed и их применение для обработки текстовых файлов.

Характеристика утилит nl, sort, tsort, xargs, diff, split, paste, tr, uniq, wc и других и их применение для обработки текстовых данных.

Тема 1.5. Командный интерпретатор Bash и программирование сценариев

Командные интерпретаторы и их характеристика. Введение в командный интерпретатор Bash. Исполнительная среда интерпретатора. Типы запуска командного интерпретатора Bash. Инициализация пользовательского окружения при запуске Bash. Конфигурационные сценарии Bash. Сравнительный анализ командных интерпретаторов их конфигурационных сценариев.

Переменные командной оболочки и их типы. Правила и рекомендации по созданию переменных. Локальные и глобальные переменные. Операции над переменными. Массив и операции над массивами. Переменные командной оболочки и их настройка. Псевдонимы и команда alias.

Типы команд в интерпретаторе Bash. Системные и встроенные команды. Справочная документация по системным и встроенным командам. Определение типа команда. История команд. Обработка командной строки в Bash. Расширения аргументов командной строки. Синтаксис регулярных выражений.

Сценарий командной оболочки Bash. Правила и ограничения. Составная команда как пример простого сценария. Алгоритм создания сценария и его запуска. Команда source.

Условное и безусловное выполнение команд. Позиционные параметры, константы и переменные языка Bash. Ввод и вывод данных в Bash. Форматированный вывод. Арифметические выражения и калькулятор Bash (bc). Условный оператор и виды ветвления. Команда if и проверка условий. Утилита test в инструкции if-then-else. Условное выполнение команд. Условные выражения. Инструкция выбора case. Составные команды в виде циклов. Циклы for, while, until. Функции в языке Bash. Обработка сигналов в сценариях. Механизм программных прерываний и команда trap. Отладка сценария. Вызов одного сценария из другого.

Раздел 2. Системное программирование в среде ОС Linux

Тема 2.1. Общие принципы Linux API и инструментарий отладки и сборки пакетов

Стандартизация как основа для упрощения портирования приложений. Стандарты в основе Linux API. Стандарт языка C. Стандарты POSIX. Стандарты LSB.

Задачи Linux API. Ядро, его концепция и выполняемые задачи. Режим ядра и пользовательский режим. Модель памяти процесса, его создание и выполнение. Отличия пользовательских процессов и процессов ядра. Особенности выполнения системных вызовов Linux API. Системные типы данных Linux API и их вывод.

Инструментарий для разработки, отладки и сборки. Установка компилятора C и библиотек. Отладчик gdb. Утилита make и синтаксис Makefile. Сценарий configure и сборка программ на языке C. Набор утилит Autotools и его применение.

Тема 2.2. Файловые операции средствами системных вызовов и файловые системы

Общее представление о файловом вводе-выводе. Дескриптор файла и стандартные дескрипторы. Ключевые системные вызовы для выполнения файлового ввода-вывода. Универсальность ввода-вывода. Открытие файла open(). Чтение из файла read(). Запись в файл write(). Закрытие файла close(). Изменение файлового смещения lseek().

Блокировка доступа к файлу, описание и виды. Блокировка функцией fcntl(). Блокировка функцией lockf().

Введение в файловые системы и разделы в ОС Linux. Таблицы разделов. Файловые системы и управление файловыми системами. Специальные файловые системы. Сетевые файловые системы. Внеядерные файловые системы. Архитектура виртуальной файловой системы. Библиотека fuse и разработка файловой системы.

Тема 2.3. Статические и динамические библиотеки

Компиляция проекта на языке C и характеристика её этапов. Библиотека объектов. Введение в статические библиотеки. Создание, редактирование и использование.

Концепция разделяемой библиотеки. Создание и использование разделяемых библиотек. Команды `objdump` и `readelf`. Команда `nm`. Общепринятые методики создания разделяемой библиотеки.

Динамически загружаемые библиотеки. Открытие разделяемой библиотеки `dlopen()`. Системный вызов получения адреса функции или переменной `dlsym()`. Выгрузка динамической библиотеки `dlclose()`. Инициализация и деинициализация динамических библиотек.

Тема 2.4. Многозадачное программирование

Введение в управление процессами и основные понятия. Основные системные вызовы для реализации многозадачности. Идентификаторы процессов в Linux. Порождение процессов. Методы синхронизации процессов. Завершение процесса. Функции и программы в порождённых процессах. Управление приоритетами процессов. Завершение процесса. Сигналы. Процессы и использование ресурсов

Тема 2.5. Многопоточное программирование

Основы многопоточности и модели многопоточности. Введение в создание потоков и управление ими. Создание и завершение потоков. Особенности главного потока. Жизненный цикл потоков. Атрибуты потоков. Средства синхронизации потоков в Linux.

Тема 2.6. Межпроцессное взаимодействие

Основы межпроцессного взаимодействия. Механизмы синхронизации между процессами. Сигналы как средства управления процессами. Механизмы взаимодействия между процессами. Сравнение IPC-механизмов. Совместно использование информации процессами. Неименованные и именованные каналы.

Введение в System V IPC. Очереди сообщений, семафоры и разделяемая память System V IPC.

Основы POSIX IPC. Очереди сообщений, семафоры и разделяемая память POSIX IPC.

Тема 2.7. Сетевое взаимодействие процессов

Введение в сетевое взаимодействие процессов в Linux. Сокеты, дейтаграммы и потоки передачи данных.

Серверные функции сокета. Связывание сокета. Функции работы с DNS. Перевод связанного сокета в состояние прослушивания. Клиентские функции сокета. Приём клиентских запросов соединения. Отключение и закрытие сокетов. Отправка и получение данных. Сокеты домена Unix.

Тема 2.8. Системные службы в ОС Linux

Понятие «системная служба». Характеристика системных служб (демонов). Создание демона. Функция `daemon()`. Рекомендации по написанию демонов. Демоны и `systemd`. Модули (Units) `systemd`. Управление демонами (службами) в Linux.

Запись в журнал сообщений и ошибок с помощью системы syslog. Подсистема аудита и журналирование в Linux. Файл журнала /var/run/utmp (who). Файл журнала /var/log/wtmp (last). Файл журнала /var/log/lastlog (lastlog). Файл журнала /var/log/btmp (lastb). Журналирование событий аутентификации. Файлы журналов. Утилита journalctl и инструментарий для управления журналами.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением дистанционных образовательных технологий (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
1	Основы системного администрирования и программирования ОС Linux	16			12		2	
1.1	Введение в системное программирование и ОС Linux	1						Собеседование
1.2	Пользовательское окружение ОС Linux и управление пакетами	3			2			Собеседование Отчет по лабораторной работе
1.3	Основы управления файлами, пользователями, правами доступа и процессами	4			4		2	Собеседование Отчет по лабораторной работе
1.4	Инструментальные средства обработки текстовых данных и программирования	2			2		2	Собеседование Отчет по лабораторной работе
1.5	Командный интерпретатор Bash и программирование сценариев	6			4			Собеседование Отчет по лабораторной работе Контрольная работа

2	Системное программирование в среде ОС Linux	18			18			
2.1	Общие принципы Linux API и инструментарий отладки и сборки пакетов	2						Собеседование
2.2	Файловые операции средствами системных вызовов и файловые системы	2			2			Собеседование Отчет по лабораторной работе
2.3	Статические и динамические библиотеки	2			4			Собеседование Отчет по лабораторной работе Коллоквиум
2.4	Многозадачное программирование	2			2			Собеседование Отчет
2.5	Многопоточное программирование	2			2			Собеседование
2.6	Межпроцессное взаимодействие	4			4			Собеседование Отчет по лабораторной работе Контрольная работа
2.7	Сетевое взаимодействие процессов	2			4			Собеседование Отчет по лабораторной работе
2.8	Системные службы в ОС Linux	2						Собеседование

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Кенин А. М. Самоучитель системного администратора / А. Кенин, Д. Колисниченко. — 6-е изд., перераб. и доп. — Санкт-Петербург: БХВ-Петербург, 2021. — 606 с.: ил. — URL: <https://ibooks.ru/reading.php?short=1&productid=380054>
2. Кобылянский В. Г. Операционные системы, среды и оболочки: учебное пособие / В. Г. Кобылянский. — Санкт-Петербург; Москва; Краснодар: Лань, 2020. — 117 с.: ил. — URL: <https://e.lanbook.com/book/254651>
3. Староверова, Н.А. Операционные системы: учебник / Н. А. Староверова. — Санкт-Петербург; Москва; Краснодар: Лань, 2019. — 307 с. — URL: <https://e.lanbook.com/book/207089>.
4. Таненбаум Э. С. Современные операционные системы = Modern Operating Systems / Э. Таненбаум, Х. Бос; [пер. с англ.: А. Леонтьева, М. Малышева, Н. Вильчинский]. — 4-е изд. — Санкт-Петербург [и др.]: Питер, 2020. — 1119 с.: ил. — URL: <https://ibooks.ru/bookshelf/364626/reading>

Перечень дополнительной литературы

1. Гунько, А. В. Системное программирование в среде Linux : учебное пособие / А. В. Гунько. — Новосибирск : НГТУ, 2020. — 235 с. — URL: <https://e.lanbook.com/book/152228>
2. Кетов Д. В. Внутреннее устройство Linux. / Д. В. Кетов. — 2-ое изд., перераб. и доп. — СПб.: БХВ-Петербург, 2021. — 400 с.: ил.
3. Лав, Роберт. Linux. Системное программирование / Роберт Лав; [пер. с англ. О. Сивченко]. - 2-е изд. - Санкт-Петербург [и др.]: Питер, 2015. - 445 с.; 23х16 см. - (Бестселлеры O'Reilly). - На обл. также: Изучаем ядро Linux и библиотеки C. - Библиогр.: с. 443–445.
4. Операционные системы. Программное обеспечение : учебник / сост. Т. П. Куль. — Санкт-Петербург ; Москва ; Краснодар : Лань, 2020. — 245 с.: ил.
5. Тронкон П. Bash и кибербезопасность. Атака, защита и анализ из командной строки Linux = Cybersecurity Ops with Bash. Attack, Defend, and Analyze from the Command Line / Пол Тронкон, Карл Олбинг; [пер. с англ. А. Герасименко]. — СПб[и др.]: Питер, 2020. — 286 с.: ил.

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Для диагностики компетенций в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: собеседование.
2. Письменная форма: коллоквиум, контрольная работа.

3. Устно-письменная форма: отчёт по лабораторным работам с устной защитой и оцениванием на основе проектного метода.

Формой промежуточной аттестации по дисциплине «Системное программирование» учебным планом предусмотрен **зачёт**.

Текущий контроль самостоятельной работы студентов по усвоению знаний по учебной дисциплине «Системное программирование» рекомендуется осуществлять преподавателем на лабораторных занятиях в виде проверки выполнения индивидуальных заданий, а также в процессе собеседования со студентом в ходе такой проверки. Такое совмещение проверки, собеседования и индивидуальной консультации позволяет наиболее полно оценить уровень знаний и практических навыков студента по рассматриваемой теме. Однако оно возможно при проведении занятий в небольших группах, предъявляет требования к организации учебного процесса и высокому уровню мотивации и отдаче преподавателя.

При формировании итоговой отметки используется рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения.

Рейтинговая система предусматривает использование весовых коэффициентов в ходе проведения контрольных мероприятий текущей аттестации.

Примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации:

Формирование отметки за текущую аттестацию:

- коллоквиум – 15%;
- отчёт по лабораторной работе – 70%;
- контрольная работа – 15%

Итоговая отметка по дисциплине рассчитывается на основе отметки текущей аттестации (рейтинговой системы оценки знаний) и отметки на зачёте с учётом их весовых коэффициентов. Вес отметки по текущей аттестации составляет 60 %, отметки на зачёте – 40%.

Примерный перечень заданий для управляемой самостоятельной работы студентов

Тема № 1.3. Основы управления файлами, пользователями, правами доступа и процессами (2 часа)

Примерный перечень вопросов:

1. Типы файлов, их форматы и команды управления файлами.
2. Жёсткие и символические ссылки.
3. Управление пользователями.

Примерный перечень заданий:

1. Выполнить операция над файлами согласно индивидуальному заданию.
2. Выполнить операции над ссылками согласно индивидуальному заданию.

3. Выполнить операции над учётными записями пользователей согласно индивидуальному заданию.

Форма контроля: собеседование.

Тема № 1.4. Инструментальные средства обработки текстовых данных и программирования (2 часа)

Примерный перечень вопросов:

1. Обработка текстовых данных в Linux.
2. Фильтрация и поиск по строкам и блокам.
3. Регулярные выражения POSIX BRE.
4. Язык сценариев AWK и его синтаксис.

Примерный перечень заданий:

1. Выполнить обработку текстовых файлов с использованием grep согласно индивидуальному заданию.
2. Выполнить обработку текстовых файлов с использованием cut согласно индивидуальному заданию.
3. Разработать сценарии на языке AWK для обработки файлов согласно индивидуальному заданию.

Форма контроля: собеседование.

Рекомендуемая тематика коллоквиума

1. «Проектирование и реализация библиотек в ОС Linux».

Примерная тематика контрольных работ

Контрольная работа 1. Разработка сценариев в ОС Linux.

Контрольная работа 2. Реализация приложений с поддержкой Linux IPC.

Примерная тематика лабораторных занятий

Лабораторная работа 1. Пользовательское окружение ОС LINUX и основные команды.

Лабораторная работа 2. Основы управления учётными записями, правами доступа и процессами.

Лабораторная работа 3. Инструментальные средства обработки данных и программирования.

Лабораторная работа 4. Синтаксис языка Bash и разработка сценариев.

Лабораторная работа 5. Проектирование и реализация статической и динамической библиотеки для файловых операций и разработка файловой системы.

Лабораторная работа 6. Многозадачность и многопоточность в Linux.

Лабораторная работа 7. Организация обмена данными между процессами.

Лабораторная работа 8. Сетевое взаимодействие процессов.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса большинства занятий используется **практико-ориентированный подход**, который предполагает освоение содержания образования через решение практических задач; приобретение навыков эффективного выполнения разных видов профессиональной деятельности.

При организации образовательного процесса **используется метод проектного обучения**, который предполагает способ организации учебной деятельности студентов, развивающий актуальные для учебной и профессиональной деятельности навыки планирования, самоорганизации и сотрудничества, включает приобретение навыков для решения исследовательских и творческих задач.

При проведении занятий в компьютерном классе основной формой работы является работа студентов над заданиями под руководством и контролем преподавателя. При этом должно обеспечиваться достаточное количество вариантов каждого задания. В основном, предполагается, что вариант задания является индивидуальным, т. е., рассчитанным на выполнение одним студентом, в некоторых случаях, например, проектных работах, задание может выполняться небольшой группой студентов (2-3 студента). Преподаватель должен оперативно консультировать выполнение заданий и принимать выполненное задание (оценивать результаты его выполнения) посредством визуальной проверки полученных результатов и собеседованием со студентом (группой студентов).

В силу различного уровня готовности студентов к восприятию новых понятий, на занятиях по дисциплине рекомендуется при необходимости проводить дополнительные консультации в малых группах студентов для объяснения и закрепления сложного материала

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные технологии.

Разместить в сетевом доступе на образовательном портале БГУ комплекс учебных и учебно-методических материалов: учебно-программные комплексы, учебные издания для теоретического изучения дисциплины, презентации лекций, методические указания к лабораторным занятиям, электронные версии домашних заданий, материалы текущего контроля и промежуточной аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательного стандарта высшего образования и учебно-программной документации, в том числе

вопросы для подготовки к зачету, задания и вопросы для самоконтроля, список рекомендуемой литературы, информационных ресурсов и др.

Примерный перечень вопросов к зачету

1. Архитектура ОС Linux и интерфейс системных вызовов.
2. Характеристика дистрибутивов ОС Linux и их история развития.
3. Пользовательское окружение и графический интерфейс в ОС Linux.
4. Командная интерфейс и основные команды.
5. Управление пакетами в ОС Linux на примере пакетов классического формата.
6. Управление пакетами в ОС Linux на примере пакетов универсального формата.
7. Характеристика конфигурационных файлов в ОС Linux.
8. Управление файлами и типы файлов.
9. Управление пользователями.
10. Библиотека PAM и управление пользователями.
11. Управление правами доступа на примере дискреционного разграничения прав доступа.
12. Управление правами доступа на примере списков контроля доступа ACL.
13. Управление разделами диска и таблицы разделов.
14. Типы файловых систем и монтирование.
15. Процессы и их атрибуты.
16. Приоритеты процессов и их настройка.
17. Механизм сигналов для управления процессами.
18. Инструментальные средства обработки текста на примере `grep`, `cut`, `awk`, `sed`.
19. Синтаксис языка AWK и обработка структурированных данных.
20. Поточковый редактор `sed` и его возможности.
21. Командная оболочка `bash` и настройка окружения.
22. Сценарии на языке командного интерпретатора.
23. Локальные и глобальные переменные, позиционные параметры в `bash`.
24. Основные конструкции ветвления и циклы в `bash`.
25. Функции в `bash`.
26. Стандарты, лежащие в основе Linux API.
27. Задачи, выполняемые Linux API.
28. Системные типы данных Linux API.
29. Общее представление о файловом вводе-выводе и его универсальность.
30. Системные вызовы `open()`, `read()`, `write()`, `close()`, `lseek()` для операций над файлами.
31. Блокировка доступа к файлу на примере функций `fcntl()` и `lockf()`.
32. Создание и использование статических библиотек.
33. Создание и применение разделяемых библиотек.

- 34. Динамически загружаемые библиотеки и интерфейс dlopen().
- 35. Основы управления процессами в ОС Linux.
- 36. Системные вызовы для реализации многозадачности и идентификаторы процессов.
- 37. Системные вызовы для порождения, синхронизации и завершения процессов.
- 38. Функции и программы в порожденных процессах.
- 39. Управление приоритетами процессов и сигналы.
- 40. Концепция, модели и шаблоны потоков.
- 41. Создание потоков и управление ими.
- 42. Средства синхронизации потоков в ОС Linux.
- 43. Концепции межпроцессного взаимодействия в Linux и каналы передачи данных.
- 44. Очереди сообщений, семафоры, разделяемая память в System V IPC.
- 45. Очереди сообщений, семафоры, разделяемая память в POSIX IPC.
- 46. Сетевое взаимодействие процессов в Linux.
- 47. Серверные и клиентские функции сокета.
- 48. Системные службы (демоны) в Linux и их создание.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
1. Программирование мобильных и встраиваемых систем	Кафедра технологий программирования	Нет	Оставить содержание учебной дисциплины без изменения (протокол № 16 от 18.05.2023)

**ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ ПО
ИЗУЧАЕМОЙ УЧЕБНОЙ ДИСЦИПЛИНЕ**

на ____ / ____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № ____ от _____ 202_ г.)

Заведующий кафедрой

УТВЕРЖДАЮ
Декан факультета
