ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол №2

Рабочая программа дисциплины (модуля)
«Архитектура компьютера и операционные системы»

Направление подготовки: *02.03.01 Математика и компьютерные науки*

Направленность (профиль) подготовки: *Дизайн и разработка информационных продуктов*

Квалификация (степень) выпускника: *бакалавр*

Форма обучения: *очная*

Срок освоения программы: *4 года*

Год набора: *2026*

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

Москва
2026

Содержание

1. **Краткая характеристика дисциплины (модуля)** 3
2. **Перечень планируемых результатов обучения** 5
3. **Тематический план** 5
4. **Содержание дисциплины (модуля)** 9
5. **Учебно-методическое обеспечение** 11
6. **Материально-техническое обеспечение** 11
7. **Методические и оценочные материалы** 13

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Архитектура компьютера и операционные системы» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 Математика и компьютерные науки, профиль Дизайн и разработка информационных продуктов, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Архитектура компьютера и операционные системы» позволяет сформировать фундаментальное понимание принципов работы вычислительной техники, навыки написания эффективного и безопасного низкоуровневого кода, способность анализировать производительность программ на уровне процессора и операционной системы.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 Математика и компьютерные науки, профиль Дизайн и разработка информационных продуктов и входит в обязательную часть Блока 1.

Дисциплина (модуль) изучается на 2 курсе в 3 и 4 семестре. Данная программа предназначена для 3-го семестра.

Цель изучения дисциплины (модуля): сформировать у обучающихся компетенции в области низкоуровневого программирования на языке Си, понимания архитектуры вычислительных систем, принципов работы операционных систем и навыков взаимодействия с аппаратным обеспечением через системные вызовы, управление памятью и анализ исполняемого кода.

Задачи изучения дисциплины (модуля):

- освоить синтаксис и семантику языка Си для ручного управления памятью, включая выделение и освобождение ресурсов, предотвращение утечек и неопределённого поведения;
- изучить этапы компиляции программы, организацию статических и динамических библиотек, а также принципы работы системы пакетов и заголовочных файлов в Linux;
- понять принципы организации процессора, архитектуры наборов команд (ISA) и научиться читать и писать фрагменты кода на ассемблере с соблюдением соглашений о вызовах;
- научиться взаимодействовать с ядром операционной системы через прямые системные вызовы и управлять динамической загрузкой библиотек на этапе выполнения;
- приобрести навыки отладки низкоуровневого кода, дизассемблирования программ, анализа численных ошибок и использования векторных инструкций для оптимизации.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- синтаксис и семантику языка Си в части управления памятью;
- этапы компиляции программы и роль каждого артефакта;
- принципы организации статических и динамических библиотек, заголовочных файлов и системы пакетов в Linux;
- базовые логические блоки процессора и побитовые операции, способы представления целых и вещественных чисел, текстовых и бинарных данных, кодировок;

— концепцию архитектур набора команд (ISA), отличия архитектур (RISC, CISC); основы кросс-компиляции;
— структуру и семантику инструкций ассемблера: работу с памятью, условные переходы, циклы, стек вызовов и соглашение о вызовах;
— оптимизации процессора и векторных инструкций (SIMD);
— механизм системных вызовов как интерфейс между прикладными программами и ядром ОС;
— принципы работы с динамическими библиотеками: загрузка, символы, связывание на этапе выполнения.

уметь:

— писать программы на языке Си с ручным управлением памятью: выделять и освобождать память, избегать утечек и неопределённого поведения;
— читать и интерпретировать вывод компилятора (предупреждения, ошибки);
— применять побитовые операции для кодирования, маскирования и упаковки данных в память;
— читать и писать простые фрагменты на ассемблере: реализовывать условия, циклы, вызовы функций с соблюдением соглашения о вызовах;
— производить системные вызовы напрямую;
— подключать и использовать динамические библиотеки;
— объяснять, почему вещественная арифметика неточна, и применять правила работы с IEEE 754 при отладке численных ошибок.

владеть:

— навыками разработки программ через системные вызовы;
— практикой отладки низкоуровневого кода;
— навыками разработки программ на языке Си;
— навыками дизассеблирования программ и анализа их поведения;
— практикой безопасной работы с памятью при разработке низкоуровневых программ.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области искусственного интеллекта, основные принципы критической оценки источников информации и их релевантности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
УК-3.	Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	УК-3.1.	Знает основные принципы социального взаимодействия и командной работы, роли и ответственности в команде
		УК-3.2.	Умеет эффективно взаимодействовать с членами команды, разрешать конфликты и достигать согласия в группе
		УК-3.3.	Имеет практический опыт участия в командной работе, реализации своей роли в команде для достижения общих целей

ОПК-4.	Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	ОПК-4.1.	Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
		ОПК-4.2.	Умеет использовать этот математический аппарат в профессиональной деятельности
		ОПК-4.3.	Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
ПК-2.	Способен решать типовые задачи профессиональной деятельности, опираясь на информационную и библиографическую культуру, используя информационно-коммуникационные технологии и учитывая основные требования информационной безопасности	ПК-2.1.	Знает основы информационной и библиографической культуры, а также принципы информационной безопасности и применения информационно-коммуникационных технологий в профессиональной деятельности
		ПК-2.2.	Умеет эффективно использовать информационно-коммуникационные технологии для решения стандартных задач профессиональной деятельности, учитывая требования информационной безопасности

		ПК-2.3.	Имеет опыт работы с информационными ресурсами и технологиями в области профессиональной деятельности, включая соблюдение норм информационной безопасности
ПК-3.	Способен применять методы математического и алгоритмического моделирования для решения как теоретических, так и практических задач в рамках профессиональной деятельности	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет применять методы математического и алгоритмического моделирования для анализа и решения различных задач в области математики и компьютерных наук
		ПК-3.3.	Имеет опыт использования методов математического и алгоритмического моделирования при решении теоретических и прикладных задач в профессиональной деятельности
ПК-4.	Способен разрабатывать программное обеспечение для решения прикладных задач в области профессиональной деятельности под руководством более опытного специалиста	ПК-4.1.	Знает основные принципы разработки программного обеспечения и методы решения прикладных задач в сфере профессиональной деятельности
		ПК-4.2.	Умеет разрабатывать программное обеспечение под руководством специалистов более высокой категории, используя современные технологии и инструменты
		ПК-4.3.	Имеет опыт участия в разработке программного обеспечения для решения прикладных задач в команде под руководством более

			опытных специалистов

3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Контактная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	Основы низкоуровненого программирования в Linux	8	12	4	16	Аудиторная работа Домашние задания Контрольная работа
2	Машинное представление данных	7	10	4	15	Аудиторная работа Домашние задания Контрольная работа
3	Архитектуры команд процессоров и ассемблер	8	12	4	16	Аудиторная работа Домашние задания Контрольная работа
4	Взаимодействие с операционной системой	7	10		15	Аудиторная работа Домашние задания
	Экзамен			4		
Итого:		***30***	***44***	***16***	***62***	
Объем дисциплины (модуля) (в ак. ч.)		***152***				
Объем дисциплины (модуля) (в зач. ед.)		***4***				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Основы низкоуровненого программирования в Linux	Введенние в язык Си и инструменты разработки. Процесс и артефакты компиляции. Препроцессор Си. Арифметика указателей, строки и преобразования строк. Структуры данных и работа с данными на куче. Библиотеки, заголовочные файлы и пакеты в Linux.
2	Машинное представление данных	Базовые логические блоки процессора и целочисленная арифметика. Текстовые и бинарные данные, кодировки и работа с файлами. Вещественная арифметика и квантизация.
3	Архитектуры команд процессоров и ассемблер	Трансляция программ с высокоуровневых языков на линейный код. Архитектуры команд процессора, кросс-компиляция и основы ассемблера Ассемблер: работа с памятью и реализация условий и циклов Ассемблер: стек вызовов и реализация функций Оптимизации процессора и векторные инструкции.

4	Взаимодействие с операционной системой	Системные вызовы. Динамические библиотеки и их использование.

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Таненбаум, Э. С. Компьютерные сети / Э. С. Таненбаум, Д. Уэзеролл. - 5-е изд. - Санкт-Петербург : Питер, 2021. - 960 с. - ISBN 978-5-4461-1248-7. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2141427.

2. Кузин, А. В. Программирование на языке Си : учебное пособие / А.В. Кузин, Е.В. Чумакова. — Москва : ФОРУМ : ИНФРА-М, 2024. — 143 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-556-1. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2137197.

3. Рабчевский, А. Н. Компьютерные сети и системы связи. Вводный курс : учебное пособие для вузов / А. Н. Рабчевский. — Москва : Издательство Юрайт, 2025. — 207 с. — (Высшее образование). — ISBN 978-5-534-21489-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/572633.

Дополнительная литература:

1. Зараменских, Е. П. Информационные системы: управление жизненным циклом : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 486 с. — (Высшее образование). — ISBN 978-5-534-21415-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/571328.

2. Чекмарев А. В. Управление цифровыми проектами и процессами : учебник для вузов / А. В. Чекмарев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 424 с. — (Высшее образование). — ISBN 978-5-534-18522-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/564520.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

— столами и стульями;

— компьютерной техникой;

— механическими калькуляторами;

— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование"	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		

КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Архитектура компьютера и операционные системы» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, аудиторная работа, домашние задания, контрольные работы, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Аудиторная работа – активная работа студента на семинаре, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их.

При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Контрольная работа – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы – получить специальные знания по одной или нескольким темам дисциплины и продемонстрировать навыки их практического применения.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Архитектура компьютера и операционные системы»

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***экзамена***, при этом проводится оценка компетенций, сформированных по дисциплине. Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с
9	Отлично	
8	Отлично	

Десятибалльная оценка	Пятибалльная оценка	Общая характеристика результата обучения по дисциплине (модулю)
		практическими задачами.
7	Хорошо	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	
5	Удовлетворительно	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	
3	Не сдан	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	
1	Не сдан	

Дисциплина (модуль) «Архитектура компьютера и операционные системы» оценивается следующим образом:

Активность	Вес	Количество	Описание
Аудиторная работа	10%	1	Активная работа студента на занятии
Домашние задания	20%	11	Набор задач по темам недели
Контрольные работы	30%	3	Письменная работа с набором задач, которые нужно решить за ограниченное время
Экзамен	40%	1	Письменная или устная работа над заданием, направленным на проверку полученных знаний и навыков по дисциплине (модулю)

Формула расчёта итоговой оценки по дисциплине (модулю) «Архитектура компьютера и операционные системы» «0,1 × аудиторная работа + 0,2 × среднее за домашние задания + 0,3 × среднее за контрольные работы + 0,4 × экзамен».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные вопросы для подготовки к занятию

Занятие 1. Основы низкоуровневого программирования в Linux

1. Опишите этапы компиляции программы на языке Си (препроцессинг, компиляция, ассемблирование, линковка).
2. В чем разница между статической и динамической линковкой библиотек?
3. Как работает арифметика указателей и почему важно знать тип данных указателя?
4. Какие функции стандартной библиотеки используются для управления памятью в куче (malloc, free)?
5. Какова роль заголовочных файлов и как препроцессор обрабатывает директиву #include?

Занятие 2. Машинное представление данных и Ассемблер

1. Как представляются отрицательные целые числа в двоичном коде (дополнительный код)?
2. В чем заключается проблема точности вещественной арифметики (IEEE 754)?
3. Что такое порядок байт (Endianness) и как он влияет на чтение бинарных данных?
4. Как реализуется условный переход (if/else) на языке ассемблера?
5. Что такое стек вызовов и какую информацию он хранит при вызове функции?

Занятие 3. Взаимодействие с операционной системой

1. Что такое системный вызов и чем он отличается от обычного вызова функции?
2. Как происходит загрузка динамической библиотеки во время выполнения программы (dlopen)?
3. Какие существуют режимы работы процессора (user mode, kernel mode)?
4. Что такое векторные инструкции (SIMD) и для чего они используются?
5. Как можно отследить системные вызовы программы в Linux (strace)?

Примерные домашние задания

Домашнее задание 1. Управление памятью и структуры данных в Си

Разработать программу на языке Си, реализующую связный список (Linked List) с динамическим выделением памяти. Программа должна поддерживать операции добавления, удаления и поиска элементов. Необходимо вручную управлять памятью (malloc/free), обеспечить отсутствие утечек памяти (проверить через Valgrind). Реализовать функцию сериализации структуры в бинарный файл и обратно. Предоставить исходный код, Makefile для сборки и отчет об отсутствии утечек памяти.

Домашнее задание 2. Анализ машинного кода и ассемблер

Написать функцию на языке Си для вычисления скалярного произведения векторов. Скомпилировать программу с флагом -S для получения ассемблерного кода. Проанализировать полученный листинг: найти циклы, условные переходы и работу со стеком. Внедрить блок кода на встроенном ассемблере (inline assembly) для оптимизации критического участка. Сравнить производительность чистой Си-версии и версии с ассемблером. Предоставить отчет с анализом регистров и инструкций.

Домашнее задание 3. Системные вызовы и динамические библиотеки

Разработать собственную динамическую библиотеку (.so), реализующую набор математических функций. Написать клиентское приложение, которое загружает эту библиотеку во время выполнения через dlopen и вызывает функции через dlsym. Реализовать обертку над системным вызовом write для вывода данных в консоль без использования стандартной библиотеки printf. Продемонстрировать работу программы через strace. Предоставить исходный код, скрипт сборки и скриншоты работы strace.

Примерные задания для контрольной работы

Контрольная работа 1. Программирование на Си и компиляция

1. Написать программу на Си, которая принимает аргументы командной строки и выводит их в обратном порядке.
2. Исправить ошибки управления памятью в предоставленном фрагменте кода (утечки, двойное освобождение).
3. Написать Makefile для сборки проекта из нескольких исходных файлов в один исполняемый бинарник.
4. Реализовать функцию копирования строки без использования библиотечных функций (работа с указателями).
5. Объяснить вывод препроцессора для заданного макроса с условиями #ifdef.

Контрольная работа 2. Архитектура данных и Ассемблер

1. Перевести десятичное число -45 в 8-битное двоичное представление в дополнительном коде.
2. Определить размер структуры в байтах с учетом выравнивания (struct padding) для заданной архитектуры.
3. Написать фрагмент кода на ассемблере x86_64, который суммирует значения в массиве из 5 элементов.
4. Проанализировать дамп памяти и определить значение переменной типа float согласно IEEE 754.
5. Восстановить логику функции на Си по предоставленному фрагменту ассемблерного кода (цикл и условие).

Контрольная работа 3. Операционная система и интеграция

1. Написать программу, которая создает файл и записывает в него данные, используя только системный вызов syscall (без libc wrapper).
2. Скомпилировать программу для другой архитектуры (кросс-компиляция) и объяснить необходимые флаги.
3. Реализовать проверку наличия символа в строке с использованием побитовых операций.
4. Настроить переменные окружения (LD_PRELOAD) для перехвата вызова стандартной функции (например, malloc).
5. Объяснить разницу в поведении программы при статической и динамической линковке при отсутствии библиотеки в системе.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Каков системный риск утечки памяти в фоновом процессе?	Исчерпание ресурсов и отказ службы	УК-1
2.	Как порядок байт влияет на сетевые протоколы?	Требует приведения к сетевому порядку	УК-1
3.	Зачем соблюдать стабильность ABI в библиотеках?	Совместимость для команд разработки	УК-3
4.	Какова ответственность разработчика при работе с root?	Минимизация привилегий и аудит	УК-3
5.	Какая утилита трассирует системные вызовы в Linux?	strace	ОПК-4
6.	На каком этапе обрабатываются макросы препроцессора?	До начала компиляции	ОПК-4
7.	Что содержит объектный файл (.o)?	Машинный код и таблицу символов	ОПК-4
8.	Какой инструмент показывает символы в бинарнике?	nm	ОПК-4
9.	Как кодируется -1 в 8-битном дополнительном коде?	11111111	ПК-2
10.	Почему нельзя сравнивать float через ==?	Ошибки округления IEEE 754	ПК-2
11.	Какая операция обнуляет младшие 4 бита?	AND с маской 0xF0	ПК-2
12.	Чему кратно выравнивание типа int в 32-бит?	4 байтам	ПК-2
13.	Какой регистр хранит результат функции в x86_64?	RAX	ПК-3
14.	Какая инструкция уменьшает указатель стека?	push	ПК-3
15.	Когда устанавливается флаг Zero Flag (ZF)?	При результате операции равном 0	ПК-3
16.	Что вызывает конвейерную опасность данных?	Зависимость по данным между инструкциями	ПК-3
17.	Какой системный вызов читает из файла?	read	ПК-4
18.	Какая функция загружает библиотеку runtime?	dlopen	ПК-4
19.	Какой флаг mprotect разрешает исполнение?	PROT_EXEC	ПК-4
20.	Как передаются аргументы в функцию main?	Через стек и регистры (argc, argv)	ПК-4