

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 27.08.2025

Рабочая программа дисциплины
Архитектуры и операционные системы

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
02.03.02 - Фундаментальная информатика и информационные технологии

Направленность образовательной программы
Системное программирование

Форма обучения
очная

г. Нижний Новгород

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.ДВ.07.02 Архитектуры и операционные системы относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-Опер_1: Осуществляет управление архитектурой изолированной (неинтегрированной) программной системы	<p>ПК-Опер_1.1: Выявляет и согласовывает требования к программной системе с точки зрения архитектуры</p> <p>ПК-Опер_1.2: Осуществляет выбор и моделирование архитектурного решения для реализации программной системы</p> <p>ПК-Опер_1.3: Разрабатывает разделы по архитектуре проектных и эксплуатационных документов программной системы</p> <p>ПК-Опер_1.4: Контролирует реализацию и испытания программной системы с точки зрения архитектуры</p> <p>ПК-Опер_1.5: Осуществляет сопровождение эксплуатации программной системы с точки зрения архитектуры</p>	<p>ПК-Опер_1.1:</p> <p>ПК-Опер_1.1. У-1. Способен выявлять несоответствия требований заказчика к программной системе с точки зрения архитектуры</p> <p>ПК-Опер_1.1. У-2. Способен описывать требования к программной системе с точки зрения архитектуры</p> <p>ПК-Опер_1.1. У-3. Умеет проверять требования на соответствие архитектуре программной системы</p> <p>ПК-Опер_1.1. У-4. Умеет выявлять требования к архитектуре программной системы путем проведения интервью с заинтересованными сторонами</p> <p>ПК-Опер_1.1. У-5. Умеет формулировать архитектурные требования к программной системе</p> <p>ПК-Опер_1.1. З-1. Знает методы управления требованиями.</p> <p>ПК-Опер_1.1. З-2. Знает методы моделирования архитектуры программной системы</p> <p>ПК-Опер_1.1. З-3. Знает методы проектирования архитектуры программной системы</p>	Контрольная работа	Зачёт: Контрольные вопросы

		<p>ПК-Опер_1.2: ПК-Опер_1.2: ПК-ОПЕР_1.2. У-1. Способен выбрать оптимальное архитектурное решение с учетом особенностей программной системы и принципов её организации. ПК-ОПЕР_1.2. У-2. Способен определить архитектуру системы, ее, бизнес-процессов, структуру данных и отдельных компонентов программной системы и методы их интеграции. ПК-ОПЕР_1.2. У-3. Способен определить перечень элементов архитектуры, которые должны быть защищены от угроз безопасности информации, связанных с нарушением конфиденциальности, целостности и доступности. ПК-ОПЕР_1.2. У-4. Способен моделировать архитектурное решение для изолированной программной системы. ПК-ОПЕР_1.2. У-5. Умеет проектировать бизнес-архитектуру программных систем с применением лучших практик, шаблонов и стилей архитектурного проектирования. ПК-ОПЕР_1.2. У-6. Умеет проектировать архитектуру интегрированной программной системы с учетом устойчивости к воздействиям внутреннего и внешнего нарушителя (хакер, неосторожный пользователь, программист, поставщик компонентов) на любую из подсистем и с использованием методов и шаблонов конструктивной (встроенной) безопасности. ПК-ОПЕР_1.2. З-1. Знает методы моделирования архитектуры программных систем и критерии сравнения</p>		
--	--	---	--	--

		<p>архитектурных решений. ПК-ОПЕР_1.2. 3-2. Знает протоколы взаимодействия программных систем. ПК-ОПЕР_1.2. 3-3. Знает нормативные правовые акты, организационно-распорядительные документы и методические рекомендации, определяющие требования к безопасности программного обеспечения. ПК-ОПЕР_1.2. 3-4. Знает методики определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных ПК-ОПЕР_1.2. 3-5. Знает лучшие практики и шаблоны создания конструктивно-безопасных интегрированных информационных систем. ПК-ОПЕР_1.2. 3-1. Знает методы моделирования архитектуры программных систем и критерии сравнения архитектурных решений. ПК-ОПЕР_1.2. 3-2. Знает протоколы взаимодействия программных систем. ПК-ОПЕР_1.2. 3-3. Знает нормативные правовые акты, организационно-распорядительные документы и методические рекомендации, определяющие требования к безопасности программного обеспечения. ПК-ОПЕР_1.2. 3-4. Знает методики определения актуальных угроз безопасности персональных данных при их обработке в информационных системах персональных данных.</p> <p>ПК-Опер_1.3: ПК-Опер_1.3: ПК-ОПЕР_1.3. У-1. Способен описывать технические и организационные меры, обеспечивающие сохранение и</p>		
--	--	---	--	--

		<p>восстановление программного обеспечения.</p> <p>ПК-ОПЕР_1.3. У-2. Умеет проектировать и моделировать архитектурные элементы программных систем и их взаимосвязи.</p> <p>ПК-ОПЕР_1.3. У-3. Умеет формировать технические и организационные меры для защиты программной системы от несанкционированного доступа к элементам конфигурации.</p> <p>ПК-ОПЕР_1.3. З-1. Знает методы моделирования и технического описания архитектуры программных систем</p> <p>ПК-ОПЕР_1.3. З-2. Знает нормативные правовые акты, организационно-распорядительные документы и методические рекомендации, определяющие требования к безопасности программного обеспечения.</p> <p>ПК-Опер_1.4:</p> <p>ПК-Опер_1.4:</p> <p>ПК-ОПЕР_1.4. У-1. Способен проверять соответствие реализации программной системы выбранному архитектурному решению.</p> <p>ПК-ОПЕР_1.4. У-2. Способен проверять результаты испытаний программной системы на соответствие архитектуре и архитектурным решениям.</p> <p>ПК-ОПЕР_1.4. У-3. Умеет проверять характеристики реализованной программной системы на соответствие архитектурным требованиям.</p> <p>ПК-ОПЕР_1.4. З-1. Знает способы определения характеристик работающей программной системы.</p> <p>ПК-ОПЕР_1.4. З-2. Знает методы параметризации архитектуры программных</p>		
--	--	--	--	--

		<p>систем. ПК-ОПЕР_1.4. 3-3. Знает основы процесса управления изменениями программных систем.</p> <p>ПК-Опер_1.5: ПК-Опер_1.5: ПК-ОПЕР_1.5. У-1. Способен проверять запросы на изменения программной системы на реализуемость с точки зрения архитектуры программной системы. ПК-ОПЕР_1.5. У-2. Способен согласовывать запросы на изменения программной системы с точки зрения архитектуры. ПК-ОПЕР_1.5. У-3. Умеет взаимодействовать с авторами запросов на изменения программной системы для уточнения содержания запросов. ПК-ОПЕР_1.5. У-4. Умеет выявлять несоответствия и изменять запросы на изменения программной системы для обеспечения их соответствия выбранной архитектуре ПК-ОПЕР_1.5. 3-1. Знает основы процесса управления изменениями программных систем. ПК-ОПЕР_1.5. 3-2. Знает методы обеспечения устойчивости функционирования программной системы.</p>		
<p>ПК-ТОП_11: Способен разрабатывать, оптимизировать и отлаживать системное программное обеспечение</p>	<p>ПК-ТОП_11.1: Разрабатывает низкоуровневый код для встраиваемых систем и драйверов ПК-ТОП_11.2: Оптимизирует код под ограниченные ресурсы ПК-ТОП_11.3: Работает с ОС, загрузчиками и аппаратурой</p>	<p>ПК-ТОП_11.1: ПК-ТОП_11.1. 3-1. Знает архитектуру современных процессоров, особенности работы с оборудованием процессора ПК-ТОП_11.1. 3-2. Знает принципы взаимодействия ПО с аппаратурой ПК-ТОП_11.1. У-1. Умеет разрабатывать</p>	<p>Контрольная работа</p>	<p>Зачёт: Контрольные вопросы</p>

	<p>ПК-ТОП_11.4: Отлаживает системное ПО без ОС</p>	<p>низкоуровневый код для встроенного программного обеспечения и драйверов</p> <p>ПК-ТОП_11.1. У-2. Умеет разрабатывать драйверы для аппаратных устройств</p> <p>ПК-ТОП_11.2:</p> <p>ПК-ТОП_11.2. 3-1. Знает методы оптимизации (кэш, память, тактовая частота процессора)</p> <p>ПК-ТОП_11.2. 3-2. Знает инструменты профилирования</p> <p>ПК-ТОП_11.5. 3-2. Знает архитектуру компиляторов, оптимизации для выбранных архитектур процессоров</p> <p>ПК-ТОП_11.2. У-1. Умеет анализировать бенчмарки, выявлять узкие места</p> <p>ПК-ТОП_11.2. У-2. Умеет разрабатывать код, оптимизированный для выбранной аппаратной архитектуры</p> <p>ПК-ТОП_11.3:</p> <p>ПК-ТОП_11.3. 3-1. Знает архитектуру ядра Linux.</p> <p>ПК-ТОП_11.3. 3-2. Знает принципы работы встраиваемых операционных систем</p> <p>ПК-ТОП_11.3. У-1. Умеет портировать код между различными вариантами загрузчиков и ОС</p> <p>ПК-ТОП_11.3. У-2. Умеет выполнять разработку на стыке программного обеспечения и оборудования</p> <p>ПК-ТОП_11.4:</p> <p>ПК-ТОП_11.4. 3-1. Знает методы анализа bare-metal сбоев</p> <p>ПК-ТОП_11.4. 3-2. Знает форматы бинарных файлов, работу с памятью</p> <p>ПК-ТОП_11.4. У-1. Умеет отлаживать код через</p>		
--	--	---	--	--

		специализированные отладчики системного программного обеспечения ПК-ТОП_11.4. У-2. Умеет анализировать дампы памяти, дизассемблированный код		
--	--	--	--	--

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очная
Общая трудоемкость, з.е.	2
Часов по учебному плану	72
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	16
- занятия семинарского типа (практические занятия / лабораторные работы)	16
- КСР	1
самостоятельная работа	39
Промежуточная аттестация	0 Зачёт

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабораторные работы), часы	Всего	
0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	0 Ф 0	
Суперскалярная архитектура CPU в современных процессорах	22	6	4	10	12
Архитектура VLIW	3	1	0	1	2
Популярные расширения архитектуры системы команд	10	2	2	4	6
Механизм прерываний и их обработка в процессорах с разной микроархитектурой	4	1	0	1	3
Оптимизация реализации кеша	10	4	2	6	4
Использование эмуляторов архитектур и микроархитектур	22	2	8	10	12
Аттестация	0				
КСР	1			1	

Итого	72	16	16	33	39
-------	----	----	----	----	----

Содержание разделов и тем дисциплины

Цели и задачи изучения дисциплины.

Целями освоения дисциплины являются:

- формирование у обучающихся системных знаний о современных архитектурах вычислительных систем и микроархитектурных решениях процессоров;
- формирование знаний о современных расширениях архитектур системы команд, включая векторные, матричные и специализированные расширения;
- приобретение практических навыков анализа, моделирования и исследования поведения процессоров и подсистем памяти с использованием эмуляторов и симуляторов архитектур.

Задачами освоения дисциплины являются:

- освоение алгоритмов работы стадий выборки, декодирования, выдачи, исполнения и завершения инструкций в современных CPU;
- изучение механизмов динамического планирования инструкций, переименования регистров и управления зависимостями;
- изучение современных расширений архитектур системы команд, включая SIMD-, векторные, матричные и другие расширения;
- изучение методов оптимизации работы с памятью при разработке приложений;
- приобретение практических навыков работы с эмуляторами и симуляторами архитектур и микроархитектур процессоров.

Содержание разделов и тем дисциплины.

1. Суперскалярная архитектура CPU в современных процессорах

Общая структура современного суперскалярного процессора. Стадия выборки инструкций (Fetch), предвыборка инструкций, очереди инструкций. Стадия декодирования (Decode), микрооперации, ограничения пропускной способности декодеров. Стадия выдачи инструкций (Dispatch/Issue), динамическое планирование, переименование регистров. Стадия исполнения (Execute), конвейеризация на стадии, параллельное исполнение. Стадия записи результатов (Writeback/Commit), буфер переупорядочивания (ROB), подтверждение инструкций, поддержание точного состояния процессора, исключения и откат выполнения. Производительность суперскалярных CPU.

2. Архитектура VLIW (Very Long Instruction Word)

Принцип длинного машинного слова. Статическое планирование инструкций. Пакетирование операций, роль компилятора. Динамическое планирование инструкций. Преимущества и недостатки VLIW.

3. Популярные расширения архитектуры системы команд

Причины появления расширений ISA. Векторные расширения. Матричные расширения.

Специализированные расширения: аппаратное ускорение шифрования, инструкции упаковки данных, расширения для виртуализации. Проблемы совместимости.

4. Механизм прерываний и их обработка

Классификация прерываний. Обработка прерываний в in-order CPU. Обработка прерываний в out-of-order CPU, точное состояние процессора. Прерывания в суперскалярных процессорах. Контроллеры прерываний. Влияние прерываний на производительность.

5. Оптимизация реализации кеша

Иерархия памяти, принцип локальности, организация кеша. Конвейеризованная запись в кеш. Буфер записи, объединение операций записи, конфликты и переполнение буфера. Аппаратная предвыборка (Hardware Prefetching), предсказание шаблонов доступа, адаптивные предвыборщики. Многопортовость кеша. Банковая организация кеша, конфликты банков.

6. Использование эмуляторов архитектур и микроархитектур

Назначение эмуляторов. Типы эмуляторов (функциональные / тактово-точные / полносистемные).
Моделирование микроархитектуры. Анализ производительности с использованием эмуляторов.
Популярные инструменты.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

Для обеспечения самостоятельной работы обучающихся используются:
Электронные курсы, созданные в системе электронного обучения ННГУ:

Архитектура и организация ЭВМ, <https://e-learning.unn.ru/course/view.php?id=2435>.

Иные учебно-методические материалы:

А.В. Кудин, А.В. Линёв. Архитектура и операционные системы параллельных вычислительных систем. Нижний Новгород, 2007. <http://www.unn.ru/books/resources.html>

С.Бастраков, В.Гергель, А.Горшков, Е.Козинов, А.Линев, И.Мееров, А.Сиднев, А.Сысоев. Введение в принципы функционирования и применения современных мультиядерных архитектур (на примере Intel Xeon Phi). <http://www.intuit.ru/studies/courses/10611/1095/info>

В.Гуров, В.Чуканов. Архитектура и организация ЭВМ.
<http://www.intuit.ru/studies/courses/60/60/info>

В.Гуров. Архитектура микропроцессоров. <http://www.intuit.ru/studies/courses/604/460/info>

Д.Северов. Архитектура вычислительных систем и язык ассемблера.
<http://www.intuit.ru/studies/courses/535/391/info>

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Контрольная работа) для оценки сформированности компетенции ПК-Опер_1:

Что такое суперскалярная архитектура процессора?

Какие стадии включает типичный конвейер современного CPU?

Как организована стадия выборки инструкций (Fetch)?

Какие задачи выполняет стадия декодирования (Decode)?

Что такое диспетчеризация и выдача инструкций (Dispatch/Issue)?

Что такое окно инструкций (instruction window)?

Как работает переименование регистров (register renaming)?

Для чего используются reservation stations?

Что такое ROB (Reorder Buffer) и зачем он нужен?

Чем VLIW отличается от суперскалярной архитектуры?

Кто отвечает за планирование инструкций в VLIW?

5.1.2 Типовые задания (оценочное средство - Контрольная работа) для оценки сформированности компетенции ПК-ТОП_11:

Для чего применяются SIMD-расширения?

Для каких задач используются tensor/matrix instructions?

Что такое packed instructions?

Какие инструкции применяются для криптографии?

Как процессор обрабатывает прерывание в in-order CPU?

Какие сложности возникают при обработке прерываний в out-of-order CPU?

Что такое precise interrupt? Как ROB помогает реализовать precise exceptions?

Как суперскалярность влияет на обработку исключений?

Для чего нужен store queue?

Что такое конвейеризованная запись в кеш?

Как работает аппаратная предвыборка (hardware prefetch)?

Как многопортовость влияет на производительность кеша?

Что такое banking cache memory?

Чем отличается эмуляция от симуляции?

Какие задачи решают микроархитектурные симуляторы?

Критерии оценивания (оценочное средство - Контрольная работа)

Оценка	Критерии оценивания
превосходно	Студент получил верный ответ во всех заданиях. При этом студент продемонстрировал знание дополнительного материала.
отлично	Студент получил верный ответ во всех заданиях. При этом студент

Оценка	Критерии оценивания
	продемонстрировал знание дополнительного материала.
очень хорошо	Студент получил верный ответ в большинстве заданий.
хорошо	Студент решил большую часть задач с незначительными недочетами.
удовлетворительно	Студент решил большую часть задач с существенными недочетами.
неудовлетворительно	Студент допускает грубые ошибки в решении стандартных задач.
плохо	Отсутствие знаний материала, отсутствует способность решения стандартных задач.

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено			зачтено			
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами.	Продемонстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнен	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов

				недочетами		ы все задания в полном объеме	
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторым и недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторым и недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой
	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-Опер_1

Что такое суперскалярная архитектура процессора?

Какие стадии включает типичный конвейер современного CPU?

Как организована стадия выборки инструкций (Fetch)?

Какие задачи выполняет стадия декодирования (Decode)?

Что такое диспетчеризация и выдача инструкций (Dispatch/Issue)?

Что такое окно инструкций (instruction window)?

Как работает переименование регистров (register renaming)?

Для чего используются reservation stations?

Что такое ROB (Reorder Buffer) и зачем он нужен?

Чем VLIW отличается от суперскалярной архитектуры?

Кто отвечает за планирование инструкций в VLIW?

5.3.2 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-ТОП_11

Для чего применяются SIMD-расширения?

Для каких задач используются tensor/matrix instructions?

Что такое packed instructions?

Какие инструкции применяются для криптографии?

Как процессор обрабатывает прерывание в in-order CPU?

Какие сложности возникают при обработке прерываний в out-of-order CPU?

Что такое precise interrupt? Как ROB помогает реализовать precise exceptions?

Как суперскалярность влияет на обработку исключений?

Для чего нужен store queue?

Что такое конвейеризованная запись в кэш?

Как работает аппаратная предвыборка (hardware prefetch)?

Как многопортовость влияет на производительность кеша?

Что такое banking cache memory?

Чем отличается эмуляция от симуляции?

Какие задачи решают микроархитектурные симуляторы?

Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	Выполнены все или большая часть этапов решения задачи или задача решена с незначительными недочетами. Результаты работы представлены преподавателю в срок.
не зачтено	Выполнены не все практические задания или выполнены не в полном объеме (представлено не полное описание этапов выполнения заданий, получен неверный ответ, результаты работы не представлены преподавателю).

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Максимов Николай Вениаминович. Архитектура ЭВМ и вычислительных систем : учеб. для студентов учреждений сред. проф. образования, обучающихся по группе специальностей "Информатика и вычисл. техника". - 3-е изд., перераб. и доп. - М. : Форум, 2010. - 512 с. : ил. - (Профессиональное образование). - На обл. и корешке кн. загл.: Архитектура ЭВМ и вычислительные системы. - ISBN 978-5-91134-374-3 : 169.95. - Текст : непосредственный., 14 экз.

Дополнительная литература:

1. Паттерсон Дэвид. Архитектура компьютера и проектирование компьютерных систем = Computer Organization and Design. - 4-е изд. - СПб. : Питер, 2012. - 784 с. : ил. - (Классика Computer Science). - ISBN 978-5-459-00291-1 : 500.00. - Текст : непосредственный., 1 экз.
2. Гуров Валерий Валентинович. Архитектура микропроцессоров : учеб. пособие. - М. : Интернет-Университет Информационных Технологий : Бином. Лаборатория знаний, 2010. - 272 с. : ил., табл. - (Основы информационных технологий). - ISBN 978-5-9963-0267-3 : 246.40. - Текст : непосредственный., 1 экз.
3. Кудин А. В. Архитектура и операционные системы параллельных вычислительных систем : учебно-методические материалы по программе повышения квалификации «информационные технологии и компьютерное моделирование в прикладной математике» / Кудин А. В., Линёв А. В. - Нижний Новгород : ННГУ им. Н. И. Лобачевского, 2007. - 73 с. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции ННГУ им. Н. И. Лобачевского - Информатика. - Текст : электронный., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=730170&idb=0>.
4. Введение в принципы функционирования и применения современных мультиядерных архитектур (на примере Intel Xeon Phi) / Гергель В.П., Мееров И.Б., Бахраков С.И., Горшков А.В., Козин Е.А., Линев А.В., Сиднев А.А., Сысоев А.В. - Москва : ИНТУИТ, 2016. - Текст : электронный., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=662900&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

1. Операционные системы семейства Microsoft Windows, лицензия по подписке Microsoft Imagine.
2. Браузер Google Chrome, предоставляется бесплатно на условиях лицензионных соглашений на программное обеспечение с открытым исходным кодом.
3. Среда разработки семейства Microsoft Visual Studio <https://visualstudio.microsoft.com/ru/>

4. ПО виртуализации Oracle VM VirtualBox (распространяется бесплатно в соответствии с условиями лицензии GPL версии 3).
5. ОС семейства Linux с установленным компилятором языка C (есть несколько бесплатных дистрибутивов).
6. Симулятор микроархитектур gem5 (распространяется бесплатно).

Интернет-ресурсы:

- <http://e-learning.unn.ru/>
- <https://e-lib.unn.ru>
- <https://intuit.ru/>

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 02.03.02 - Фундаментальная информатика и информационные технологии.

Автор(ы): Линев Алексей Владимирович.

Заведующий кафедрой: Мееров Иосиф Борисович, кандидат технических наук.

Программа одобрена на заседании методической комиссии от 25.06.2025, протокол № Протокол №11.