

Оборотная сторона титульного листа РПД

I. Рабочая программа пересмотрена на заседании департамента:

Протокол от « ____ » _____ 20__ г. № ____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

II. Рабочая программа пересмотрена на заседании департамента:

Протокол от « ____ » _____ 20__ г. № ____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

III. Рабочая программа пересмотрена на заседании департамента:

Протокол от « ____ » _____ 20__ г. № ____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

IV. Рабочая программа пересмотрена на заседании департамента:

Протокол от « ____ » _____ 20__ г. № ____

Заведующий кафедрой _____
(подпись) (И.О. Фамилия)

Цели и задачи освоения дисциплины:

Цель: раскрыть смысл ключевых понятий из области параллельных вычислений, сформировать представление о современных параллельных вычислительных архитектурах, моделях, методах и технологиях их программирования, привить навыки работы с параллельными вычислительными платформами.

Задачи:

- приобретение студентами базового набора знаний в областях параллельной алгоритмизации и параллельных вычислений;
- первичных навыков работы с современными параллельными вычислительными системами и инструментальными средствами разработки параллельного программного обеспечения.

Для успешного изучения дисциплины «Параллельное программирование» у обучающихся должны быть сформированы следующие предварительные компетенции:

- определяет роль и значение информации, информатизации общества, информационных технологий, использует теоретические основы информационных процессов преобразования информации (УК-1.1);
- применяет методики поиска, сбора и обработки информации с помощью современных компьютерных технологий, системный подход при работе с информацией в глобальных компьютерных сетях и корпоративных информационных системах, основы технологии создания баз данных для решения поставленных задач (УК-1.3);
- определяет круг задач в рамках поставленной цели, определяет связи между ними (УК-2.1);
- планирует реализацию задач в зоне своей ответственности с учетом имеющихся ресурсов и ограничений, действующих правовых норм (УК-2.2);
- представляет результаты проекта, предлагает возможности их использования и/или совершенствования (УК-2.3);
- использует фундаментальные основы математики, физики, вычислительной техники и программирования (ОПК-1.1);
- решает стандартные профессиональные задачи с применением фундаментальных основ математики, физики, вычислительной техники и программирования (ОПК-1.2);
- использует математические основы при разработке и эксплуатации программных продуктов и программных комплексов (ОПК-2.2);
- применяет математические основы при решении конкретных задач (ОПК-2.3).

В результате изучения данной дисциплины у обучающихся формируются следующие профессиональные компетенции:

Тип задач	Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
научно-исследовательский	ПК-1 Способность демонстрировать базовые знания математических и естественных наук, программирования и информационных технологий	ПК-1.1 Демонстрирует знания, полученные в области математических и (или) естественных наук, программирования и информационных технологий
		ПК-1.2 Формулирует стандартные задачи в собственной научно-исследовательской деятельности в области программирования и информационных технологий.
		ПК-1.3 Решает стандартные задачи в собственной научно-исследовательской деятельности в области программирования и информационных технологий
производственно-технологический	ПК-7 Способность использовать знания направлений развития компьютеров с традиционной (нетрадиционной) архитектурой; современных системных программных средств; операционных систем, операционных и сетевых оболочек, сервисных программ; тенденции развития функций и архитектур проблемно-ориентированных программных систем и комплексов в профессиональной деятельности	ПК-7.1 Демонстрирует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ; тенденции развития их функций
		ПК-7.2 Использует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ при создании информационных систем
		ПК-7.3 Выбирает подходящие архитектуру компьютеров и программное обеспечение при создании информационных систем
производственно-технологический	ПК-9 Способность использования современных методов разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных	ПК-9.1 Демонстрирует знание современных методов разработки и реализации алгоритмов математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ
		ПК-9.2 Использует современные методы разработки алгоритмов, математических моделей на базе языков

	программ моделирования	и современных инструментальных систем, в том числе пакетов прикладных программ
		ПК-9.3 Разрабатывает и реализует алгоритмы на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
ПК-1.1 Демонстрирует знания, полученные в области математических и (или) естественных наук, программирования и информационных технологий	Знает методы обоснования параллельных алгоритмов и программ
	Умеет представлять и формально обосновывать параллельные алгоритмы и программы применительно к основным вычислительным задачам
	Имеет навыки алгоритмической и программной реализации параллельных программ, демонстрируя навыки распознавания паттернов параллельного программирования применительно к задачам на ранних этапах поиска их алгоритмического и программного решения.
ПК-1.2 Формулирует стандартные задачи в собственной научно-исследовательской деятельности в области программирования и информационных технологий.	Знает методы формализации описания задач параллельного программирования, а также описания метода их реализации.
	Умеет формулировать обоснование методов параллельного программирования применительно к задачам в собственной научно-исследовательской деятельности
	Имеет навыки формального обоснования алгоритмов, выбора средств реализации, выбранного метода реализации методов параллельного программирования применительно к задачам научно-исследовательской деятельности в области программирования и информационных технологий
ПК-1.3 Решает стандартные задачи в собственной научно-исследовательской деятельности в области программирования и информационных технологий	Знает методы реализации задач параллельного программирования и методы оценки их реализуемости.
	Умеет выполнять реализацию стандартных задач параллельного программирования в собственной предметной области.
	Имеет навыки реализации и оценки реализуемости математических моделей задач методами параллельного программирования по их формальному описанию
ПК-7.1 Демонстрирует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ; тенденции развития их функций	Знает современные средства математического моделирования, описания и практической реализации параллельных вычислений с помощью программ, а также функциональные ограничения, налагаемые на реализацию применением этих средств.
	Умеет обосновывать применение методов реализации и программно-аппаратного обеспечения параллельных

	<p>вычислений для решения основных задач известных классов.</p> <p>Владеет навыками реализации задач параллельного программирования с использованием современного программного и аппаратного обеспечения параллельных вычислений с учетом тенденций его развития.</p>
ПК-7.2 Использует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ при создании информационных систем	<p>Знает направления развития параллельных вычислений, их современного программного и аппаратного обеспечения, математических моделей с точки зрения реализуемости параллельных вычислений и программирования.</p>
	<p>Умеет реализовывать задачи параллельного программирования с помощью современных языков программирования, а также современного программного и аппаратного обеспечения параллельного программирования и вычислений.</p>
	<p>Владеет методами применения современных методов и программно-аппаратного обеспечения параллельных вычислений, фундаментальными ограничениями применимости и эффективности параллельных вычислений, направления развития технических средств параллельных вычислений и параллельного программирования.</p>
ПК-7.3 Выбирает подходящие архитектуру компьютеров и программное обеспечение при создании информационных систем	<p>Знает архитектуру и функции современного аппаратного и программного обеспечения параллельных вычислений.</p>
	<p>Умеет обосновать и применить выбор архитектуры современного программно-аппаратного обеспечения параллельных вычислений с точки зрения особенностей применения и функциональной эффективности.</p>
	<p>Владеет навыками синтеза формальной модели параллельных алгоритмов и программ для заданной архитектуры программно-аппаратного обеспечения, поверхностной и детальной оценки эффективности обеспечения для задач параллельного программирования.</p>
ПК-9.1 Демонстрирует знание современных методов разработки и реализации алгоритмов математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	<p>Знает основные методы математического моделирования параллельных алгоритмов и программ, языки программирования для реализации параллельных вычислений, логические примитивы обеспечения параллельных вычислений и параллельного программирования.</p>
	<p>Умеет формально обосновать языковые средства и логические примитивы, реализуемые используемым обеспечением параллельных вычислений, для реализации задач параллельного программирования.</p>
	<p>Владеет навыками обоснования и формального описания используемых для реализации параллельного программирования языковых средства и логических примитивов обеспечения параллельных вычислений.</p>
ПК-9.2 Использует современные	Знает особенности применения методов

методы разработки алгоритмов, математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	математического моделирования параллельных алгоритмов и программ, языков программирования для реализации параллельных вычислений, логических примитивов обеспечения параллельных вычислений и параллельного программирования.
	Умеет использовать языковые средства и логические примитивы, реализуемые используемым обеспечением параллельных вычислений, для реализации задач параллельного программирования.
	Владеет навыками выбора и применения языкового и логического обеспечения параллельного программирования, реализации параллельных вычислений с помощью используемых языковых средств, выбранного программного и аппаратного обеспечения параллельных вычислений.
ПК-9.3 Разрабатывает и реализует алгоритмы на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	Знает современные инструментальные средства реализации параллельных вычислений, основные языки параллельного программирования, модели использования логических примитивов для реализации задач параллельного программирования.
	Умеет разрабатывать и реализовывать параллельные алгоритмы и программы с использованием языковых средств, логического, программного и аппаратного обеспечения параллельного программирования.
	Имеет навыки применения языковых средств, логического, программного и аппаратного обеспечения параллельного программирования для решения вычислительных задач, оценки, в том числе на ранних этапах проектирования программной реализации, эффективности использования инструментальных средств для задач параллельного программирования.

Трудоемкость дисциплины 2 зачетные единицы (72 часа), 1 зачетная единица соответствует 36 академическим часам. Дисциплина реализуется на 4 курсе, в 7 семестре.

Для формирования вышеуказанных компетенций в рамках дисциплины «Параллельное программирование» применяются неимитационные методы активного/интерактивного обучения: выполнение проектов с использованием компьютерных технологий и специализированного программного обеспечения.

Видами учебных занятий и работы обучающегося по дисциплине являются:

Обозначение	Виды учебных занятий и работы обучающегося
Лек	Лекции
Лаб	Лабораторные работы
СР	Самостоятельная работа обучающегося в период теоретического обучения
Контроль	Самостоятельная работа обучающегося и контактная работа обучающегося с преподавателем в период промежуточной аттестации

Структура дисциплины:

Форма обучения – очная.

№	Наименование раздела дисциплины	Семестр	Количество часов по видам учебных занятий и работы обучающегося						Формы промежуточной аттестации
			Лек	Лаб	Пр	ОК	СР	Контроль	
1	Параллельное программирование	7	16	16	0	0	13	27	Экзамен
	Итого:		16	16	0	0	13	27	

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА (16 ЧАСОВ)

Тема 1. Обоснование и теоретические основы параллельных вычислений, алгоритмов и программ (2 часа)

1. Актуальность параллельных вычислений. Закон Мура и гипотеза Минского.

2. Теоретическое обоснование параллельных вычислений. Параллелизуемость и масштабируемость параллельных алгоритмов. Закон Амдала. Параллельные формы алгоритмов.

3. Таксономия Флинна.

Тема 2. Архитектуры и интерфейсы платформ для параллельных вычислений (2 часа)

1. Параллельные системы с общей памятью. Симметричные мультипроцессоры. Кэш-память и когерентность кэша.

2. Векторизация вычислений для SIMD-расширений процессоров.

3. Системы с разделенной памятью. Архитектуры вычислительных систем с разделенной памятью. Особенности дизайна параллельных вычислений с использованием мультикомпьютеров.

4. Использование специализированных процессоров для решения задач, примеры реализации для цифровой обработки одномерных сигналов. Графические процессоры для вычислений общего назначения. Реализация GPGPU графическими процессорами AMD (ATI) и Nvidia. Архитектура и язык Nvidia CUDA. Пример переноса C-программы, реализующей сложение векторов, на графические процессоры Nvidia CUDA, реализация с помощью OpenCL для графических процессоров AMD (ATI).

Тема 3. Языковые средства реализации задач параллельного программирования (4 часа)

1. Языковое расширение и библиотека OpenMP для языков C и C++.

2. Средства реализации совместных и параллельных вычислений Windows и Posix.

3. Интерфейс реализации совместных и параллельных вычислений с помощью стандартной библиотеки C11.

4. Потоки C++11 и параллельные алгоритмы C++17.

5. Примеры реализации параллельных алгоритмов поиска минимума и аккумуляирования элементов вектора.

Тема 4. Средства управления параллельными вычислениями (6 часов)

1. Интерфейс и реализация взаимодействия единиц параллельного выполнения в системах с общей и разделенной памятью. Управление вычислениями с помощью передачи сообщений в системах с общей и разделенной памятью.

2. Проблемы управления и контроля доступа к разделяемым ресурсам. Состязательность и гонки. Ложное разделение памяти. Синхронизация доступа к разделяемым ресурсам.

3. Примитивы синхронизации параллельного выполнения в системах с общей памятью. Мьютексы. Мониторы: переменные условия, события и семафоры. Барьерная синхронизация.

4. Проектирование и реализация комплексных протоколов синхронизации доступа к разделяемым ресурсам. Реализация событий и семафоров на основе мьютексов и переменных условий. Реализация интерфейса передачи сообщений на основе многопоточной очереди сообщений.

5. Примитивы синхронизации, предоставляемые библиотеками языков C17 и C++14, платформами Windows и Linux (Pthreads). Масштабируемость алгоритмов, использующих синхронизацию. Облегченные мьютексы и спин-счетчики.

6. Атомарный неблокирующий доступ к элементарным ресурсам разделяемой памяти. Методы задания политик переупорядочивания языковых операций и инструкций в C++11. Видимость изменений в памяти. Примеры реализаций алгоритмов вставки/извлечения из очереди без использования мьютексной синхронизации.

7. Параллелизм, основанный на задачах. Интерфейс, балансировка нагрузки, пулы потоков, планирование выполнения. Динамический и статический параллелизм выполнения. Примеры реализации параллелизма, основанного на задачах, (расчет чисел последовательности Фибоначчи) с помощью конструкций OpenMP и C++17. Реализуемость для систем с общей памятью.

8. Особенности организации параллельных вычислений в системах с распределенной памятью посредством передачи сообщений.

Тема 5. Логическая модель и программная архитектура графических процессоров Nvidia CUDA (2 часа)

1. Логическая модель вычислителей CUDA. Поточковые мультипроцессоры CUDA. Блоки и потоки CUDA.

2. Арифметико-логические устройства и блоки трансцендентных операций CUDA. Параллелизм данных. Типы данных и параллелизм потокового мультипроцессора CUDA. Ветвления, условные переходы, рекурсия и прерывания на устройствах CUDA.

3. Память CUDA: общая память устройства, разделяемая память и кэш, текстурная память.

4. Компиляция CUDA кода. Взаимодействие компилятора nvcc, LLVM и распространенных компиляторов C++ Windows и Linux.

5. Обзор основных возможностей устройств CUDA различных Compute Capability.

6. Понятие группы потоков CUDA Warp. Оценка занятости мультипроцессоров CUDA при выполнении вычислений и доступе к памяти. Перекрывающийся доступ к разделяемой и общей памяти.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА

Лабораторные работы (16 час.)

Лабораторная работа № 1. Реализация параллельного численного интегрирования потоками выполнения центральных процессоров с помощью OpenMP, потоков C++11 и средств операционной системы (4 часа)

Реализовать последовательный и параллельный алгоритм численного интегрирования функции с записью локальных результатов интегрирования разделяемый массив и суммированием его элементов в основном потоке выполнения. Измерить показатели времени выполнения интегрирования от количества используемых рабочих потоков.

Лабораторная работа № 2. Доступ к разделяемой памяти: кеширование, синхронизация посредством взаимного исключения, состязательность и гонки (3 часа)

Модифицировать алгоритм численного интегрирования функции так, чтобы элементы массива результатов, получаемых рабочими потоками, были

выравнены по восьмибайтовой границе. Модифицировать алгоритм так, чтобы частные результаты интегрирования накапливались в разделяемой переменной, защищаемой мьютексом. Оценить масштабируемость обеих реализаций.

Лабораторная работа № 3. Параллелизм выполнения, реализуемый задачами (6 часов)

С помощью мьютекса и монитора реализовать блокирующую очередь задач, агрегирующий ее пул рабочих потоков, предоставляющий интерфейс для добавления задач и ожидания их выполнения, а также класс задач, рекурсивно реализующих вычисление заданного элемента последовательности Фибоначчи. Оценить и объяснить масштабируемость реализации.

Лабораторная работа № 4. Параллельные алгоритмы инициализации массива и вычисления среднего с помощью редукции на CUDA (3 часа)

В рамках одного ядра (kernel) CUDA реализовать алгоритм параллельной инициализации массива случайными числами с помощью линейного конгруэнтного генератора и выбранного источника энтропии, и затем, выполнив барьерную синхронизацию с помощью `__syncthreads`, осуществить вычисление среднего арифметического сгенерированных элементов через их редукцию.

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	2 неделя обучения	Составление и анализ параллельного алгоритма подсчета элементов с заданным свойством в векторе.	1	Проект
2.	4 неделя обучения	Составление и реализация параллельных алгоритмов длинной арифметики.	1	Проект
3.	4 неделя обучения	Быстрая сортировка с использованием параллельных	1	Проект

		вычислений.		
4.	4 неделя обучения	Инструментальные средства параллельных вычислений в C11 и C++17.	2	Собеседование
5.	6 неделя обучения	Инструменты автоматизированного анализа параллельного кода программ.	2	Собеседование
6.	8 неделя обучения	Оптимизация доступа к разделяемой памяти блока CUDA на примере задачи умножения матриц.	2	Проект
7.	8 неделя обучения	Методы ограничения переупорядочивания инструкций процессора и языковых конструкций при компиляции.	2	Собеседование
8.	9 неделя обучения	Распределение вычислений и инструменты MPI.	1	Проект
9.	9 неделя обучения	Реализация параллельного сложения векторов с помощью OpenCL.	1	Проект
	17-18 недели	Подготовка к экзамену	27	

III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Параллельное программирование» включает в себя:

- план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
- характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;
- требования к представлению и оформлению результатов самостоятельной работы;
- критерии оценки выполнения самостоятельной работы.

Самостоятельные работы проводятся на рабочих местах с доступом к ресурсам Internet и в домашних условиях. Порядок выполнения самостоятельной работы соответствует программе курса и контролируется в ходе аудиторных занятий. Самостоятельная работа подкрепляется учебно-методическим и информационным обеспечением, включающим рекомендованные учебники и учебно-методические пособия.

Самостоятельная работа считается выполненной, в отчете по проделанной работе представлено письменные пояснения к полученным выводам и, если требуется, код программной реализации, компилируемый и выполняющий задачу корректно.

Проводится проверка правильности выполнения заданий по самостоятельной работе. Задание зачтено, если нет ошибок. По текущим ошибкам даются пояснения.

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые модули/разделы / темы дисциплины	Код индикатора достижения компетенции	Результаты обучения	Оценочные средства – наименование	
				текущий контроль	промежуточная аттестация
1	Обоснование параллельных вычислений	ПК-1.1 Демонстрирует знания, полученные в области математических и (или) естественных наук, программирования и информационных технологий	Знает методы обоснования параллельных алгоритмов и программ	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 1, 5, 11, 15, 21.

		<p>Умеет представлять и формально обосновывать параллельные алгоритмы и программы применительно к основным вычислительным задачам.</p>	<p>Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).</p>	<p>Экзаменационные вопросы 1, 5, 11, 15, 21.</p>
		<p>Владеет точно и формально применять навыки алгоритмической и программной реализации параллельных программ, демонстрируя навыки распознавания паттернов параллельного программирования применительно к задачам на ранних этапах поиска их алгоритмического и программного решения.</p>	<p>Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).</p>	<p>Экзаменационные вопросы 1, 5, 11, 15, 21.</p>
	<p>ПК-7.1 Демонстрирует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ; тенденции развития их функций</p>	<p>Знает современные средства математического моделирования, описания и практической реализации параллельных вычислений с помощью программ, а также функциональные ограничения, налагаемые на реализацию применением этих средств.</p>	<p>Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).</p>	<p>Экзаменационные вопросы 1, 5, 11, 15, 21.</p>
		<p>Умеет обосновывать</p>	<p>Устный опрос (УО-</p>	<p>Экзаменацион</p>

			<p>применение методов реализаций и программно-аппаратного обеспечения параллельных вычислений для решения основных задач известных классов.</p>	<p>1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).</p>	<p>ные вопросы 1, 5, 11, 15, 21.</p>
			<p>Владеет навыками реализации задач параллельного программирования с использованием современного программного и аппаратного обеспечения параллельных вычислений с учетом тенденций его развития.</p>	<p>Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).</p>	<p>Экзаменационные вопросы 1, 5, 11, 15, 21.</p>
		ПК-9.1	Знает основные методы математического моделирования параллельных алгоритмов и программ, языки программирования для реализации параллельных вычислений, логические примитивы обеспечения параллельных вычислений и параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 1, 5, 11, 15, 21.
		Демонстрирует знание современных методов разработки и реализации алгоритмов математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	Умеет формально обосновать языковые средства и логические примитивы, реализуемые используемым обеспечением	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи	Экзаменационные вопросы 1, 5, 11, 15, 21.

			параллельных вычислений, для реализации задач параллельного программирования.	реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	
			Владеет навыками обоснования и формального описания используемых для реализации параллельного программирования языковых средства и логических примитивов обеспечения параллельных вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 1, 5, 11, 15, 21.
2	Архитектуры и интерфейсы платформ для параллельных вычислений	ПК-7.1 Демонстрирует знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ; тенденции развития их функций	Знает современные средства математического моделирования, описания и практической реализации параллельных вычислений с помощью программ, а также функциональные ограничения, налагаемые на реализацию применением этих средств.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Умеет обосновывать применение методов реализации и программно-аппаратного обеспечения параллельных вычислений для решения основных задач известных классов.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Владеет навыками реализации задач	Устный опрос (УО-	Экзаменационные вопросы 3,

			параллельного программирования с использованием современного программного и аппаратного обеспечения параллельных вычислений с учетом тенденций его развития.	1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	12, 16, 17, 19, 25, 27, 29.
		ПК-9.1 Демонстрирует знание современных методов разработки и реализации алгоритмов математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	Знает основные методы математического моделирования параллельных алгоритмов и программ, языки программирования для реализации параллельных вычислений, логические примитивы обеспечения параллельных вычислений и параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Умеет формально обосновать языковые средства и логические примитивы, реализуемые используемым обеспечением параллельных вычислений, для реализации задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Владеет навыками обоснования и формального описания используемых для реализации параллельного программирования	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.

			языковых средства и логических примитивов обеспечения параллельных вычислений.	реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	
		ПК-7.3 Выбирает подходящие архитектуру компьютеров и программное обеспечение при создании информационных систем	Знает архитектуру и функции современного аппаратного и программного обеспечения параллельных вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Умеет обосновать и применить выбор архитектуры современного программно-аппаратного обеспечения параллельных вычислений с точки зрения особенностей применения и функциональной эффективности.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
			Владеет навыками синтеза формальной модели параллельных алгоритмов и программ для заданной архитектуры программно-аппаратного обеспечения, поверхностной и детальной оценки эффективности обеспечения для задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 3, 12, 16, 17, 19, 25, 27, 29.
3	Языковые средства	ПК-7.2 Использует	Знает направления развития	Устный опрос (УО-	Экзаменационные вопросы 5,

реализации задач параллельного программирования	знание направлений развития компьютеров; современных системных программных средств; операционных систем и оболочек, сервисных программ при создании информационных систем	параллельных вычислений, их современного программного и аппаратного обеспечения, математических моделей с точки зрения реализуемости параллельных вычислений и программирования.	1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	7, 9, 10, 11, 13, 21, 23
		Умеет реализовывать задачи параллельного программирования с помощью современных языков программирования, а также современного программного и аппаратного обеспечения параллельного программирования и вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
		Владеет методами применения современных методов и программно-аппаратного обеспечения параллельных вычислений, фундаментальными ограничениями применимости и эффективности параллельных вычислений, направления развития технических средств параллельных вычислений и параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
	ПК-9.2 Использует	Знает особенности применения методов	Устный опрос (УО-	Экзаменационные вопросы 5,

		современные методы разработки алгоритмов, математических моделей на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	математического моделирования параллельных алгоритмов и программ, языков программирования для реализации параллельных вычислений, логических примитивов обеспечения параллельных вычислений и параллельного программирования.	1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	7, 9, 10, 11, 13, 21, 23
			Умеет использовать языковые средства и логические примитивы, реализуемые используемым обеспечением параллельных вычислений, для реализации задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
			Владеет навыками выбора и применения языкового и логического обеспечения параллельного программирования, реализации параллельных вычислений с помощью используемых языковых средств, выбранного программного и аппаратного обеспечения параллельных вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
		ПК-9.3 Разрабатывает	Знает современные инструментальные	Устный опрос (УО-1);	Экзаменационные вопросы 5,

		и реализует алгоритмы на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	средства реализации параллельных вычислений, основные языки параллельного программирования, модели использования логических примитивов для реализации задач параллельного программирования.	дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	7, 9, 10, 11, 13, 21, 23
			Умеет разрабатывать и реализовывать параллельные алгоритмы и программы с использованием языковых средств, логического, программного и аппаратного обеспечения параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
			Владеет навыками применения языковых средств, логического, программного и аппаратного обеспечения параллельного программирования для решения вычислительных задач, оценки, в том числе на ранних этапах проектирования программной реализации, эффективности использования инструментальных средств для задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 5, 7, 9, 10, 11, 13, 21, 23
4	Средства управления	ПК-1.2 Формулирует	Знает методы формализации	Устный опрос (УО-1);	Экзаменационные вопросы 2,

параллельными вычислениями	стандартные задачи в собственной научно-исследовательской деятельности в области программирования и информационных технологий.	описания задач параллельного программирования, а также описания метода их реализации.	дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32
		Умеет формулировать обоснование методов параллельного программирования применительно к задачам в собственной научно-исследовательской деятельности	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32
		Владеет навыками формального обоснования алгоритмов, выбора средств реализации, выбранного метода реализации методов параллельного программирования применительно к задачам научно-исследовательской деятельности в области программирования и информационных технологий	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32
	ПК-7.2 Использует знание направлений развития компьютеров; современных системных программных средств;	Знает направления развития параллельных вычислений, их современного программного и аппаратного обеспечения, математических моделей с точки	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32

	операционных систем и оболочек, сервисных программ при создании информационных систем	зрения реализуемости параллельных вычислений и программирования.	уровня (ПР-11); кейс-задачи (ПР-14).	
Умеет реализовывать задачи параллельного программирования с помощью современных языков программирования, а также современного программного и аппаратного обеспечения параллельного программирования и вычислений.		Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	
Владеет методами применения современных методов и программно-аппаратного обеспечения параллельных вычислений, фундаментальными ограничениями применимости и эффективности параллельных вычислений, направления развития технических средств параллельных вычислений и параллельного программирования.		Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	
	ПК-9.3 Разрабатывает и реализует алгоритмы на базе языков и современных инструментальных систем, в том числе пакетов	Знает современные инструментальные средства реализации параллельных вычислений, основные языки параллельного программирования, модели использования	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32

		прикладных программ	логических примитивов для реализации задач параллельного программирования.	уровня (ПР-11); кейс-задачи (ПР-14).	
			Умеет разрабатывать и реализовывать параллельные алгоритмы и программы с использованием языковых средств, логического, программного и аппаратного обеспечения параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32
			Владеет навыками применения языковых средств, логического, программного и аппаратного обеспечения параллельного программирования для решения вычислительных задач, оценки, в том числе на ранних этапах проектирования программной реализации, эффективности использования инструментальных средств для задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32
5	Логическая модель и программная архитектура графических процессоров Nvidia CUDA	ПК-7.2 Использует знание направлений развития компьютеров; современных системных программных средств;	Знает направления развития параллельных вычислений, их современного программного и аппаратного обеспечения, математических моделей с точки	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31

		операционных систем и оболочек, сервисных программ при создании информационных систем	зрения реализуемости параллельных вычислений и программирования.	уровня (ПР-11); кейс-задачи (ПР-14).	
			Умеет реализовывать задачи параллельного программирования с помощью современных языков программирования, а также современного программного и аппаратного обеспечения параллельного программирования и вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31
			Владеет методами применения современных методов и программно-аппаратного обеспечения параллельных вычислений, фундаментальными ограничениями применимости и эффективности параллельных вычислений, направления развития технических средств параллельных вычислений и параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31
		ПК-7.3 Выбирает подходящую архитектуру компьютеров и программное обеспечение при создании информационных систем	Знает архитектуру и функции современного аппаратного и программного обеспечения параллельных вычислений.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31

				уровня (ПР-11); кейс-задачи (ПР-14).	
			Умеет обосновать и применить выбор архитектуры современного программно-аппаратного обеспечения параллельных вычислений с точки зрения особенностей применения и функциональной эффективности.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31
			Владеет навыками синтеза формальной модели параллельных алгоритмов и программ для заданной архитектуры программно-аппаратного обеспечения, поверхностной и детальной оценки эффективности обеспечения для задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31
		ПК-9.3 Разрабатывает и реализует алгоритмы на базе языков и современных инструментальных систем, в том числе пакетов прикладных программ	Знает современные инструментальные средства реализации параллельных вычислений, основные языки параллельного программирования, модели использования логических примитивов для реализации задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31
			Умеет разрабатывать и реализовывать параллельные	Устный опрос (УО-1);	Экзаменационные вопросы 4, 14, 18, 20, 26,

			алгоритмы и программы с использованием языковых средств, логического, программного и аппаратного обеспечения параллельного программирования.	дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	29, 30, 31
			Владеет навыками применения языковых средств, логического, программного и аппаратного обеспечения параллельного программирования для решения вычислительных задач, оценки, в том числе на ранних этапах проектирования программной реализации, эффективности использования инструментальных средств для задач параллельного программирования.	Устный опрос (УО-1); дискуссия (УО-4); тесты (ПР-1); конспект (ПР-7); задачи реконструктивного уровня (ПР-11); кейс-задачи (ПР-14).	Экзаменационные вопросы 4, 14, 18, 20, 26, 29, 30, 31

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в разделе VIII.

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

(электронные и печатные издания)

1. Параллельные вычислительные системы: учебное пособие / Н.Ю. Сиротина [и др.]. — Красноярск: Сибирский федеральный университет, 2019. — 178 с. — ISBN 978-5-7638-4180-0. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/100081.html>
2. Биллиг В.А. Параллельные вычисления и многопоточное программирование: учебник / Биллиг В.А. — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 310 с. — ISBN 978-5-4497-0936-3. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/102044.html>
3. Гергель В.П. Теория и практика параллельных вычислений: учебное пособие / Гергель В.П. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 500 с. — ISBN 978-5-4497-0389-7. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/89478.html>
4. Федотов И.Е. Параллельное программирование. Модели и приемы / Федотов И.Е. — Москва: СОЛОН-ПРЕСС, 2018. — 390 с. — ISBN 978-5-91359-222-4. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/90420.html>
5. Арыков С.Б. Параллельное программирование над общей памятью. POSIX Threads: учебное пособие / Арыков С.Б., Городничев М.А., Щукин Г.А.. — Новосибирск: Новосибирский государственный технический университет, 2018. — 87 с. — ISBN 978-5-7782-3642-4. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/91650.html>
6. Левин М.П. Параллельное программирование с использованием OpenMP: учебное пособие / Левин М.П.. — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 133 с. — ISBN 978-5-4497-0685-0. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL: <http://www.iprbookshop.ru/97572.html> .

Дополнительная литература

1. Николаев Е.И. Параллельные вычисления: учебное пособие / Николаев Е.И. — Ставрополь: Северо-Кавказский федеральный университет, 2016. — 185 с. — ISBN 2227-8397. — Текст: электронный // Электронно-

библиотечная система IPR BOOKS: [сайт]. — URL:
<http://www.iprbookshop.ru/66086.html>

2. Параллельные вычисления на GPU. Архитектура и программная модель CUDA : учебное пособие / А.В. Боресков [и др.]. — Москва: Московский государственный университет имени М.В. Ломоносова, 2015. — 336 с. — ISBN 978-5-19-011058-6. — Текст: электронный // Электронно-библиотечная система IPR BOOKS: [сайт]. — URL:
<http://www.iprbookshop.ru/54647.html>

Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

1. Programming Guide:: CUDA Toolkit Documentation.
<https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html>
2. OpenMP 4.0 API C/C++ Syntax Quick Reference Card.
<https://www.openmp.org/wp-content/uploads/OpenMP-4.0-C.pdf>
3. Cppreference.com. <https://en.cppreference.com/w/>

Перечень информационных технологий и программного обеспечения

1. Среды разработки ПО Microsoft Visual Studio Community 2019.
2. Компиляторы gcc, g++ версии не ниже 9.0, а также отладчик gdb.
3. Пакет Nvidia CUDA Toolkit версии не ниже 10.0.

Профессиональные базы данных и информационные справочные системы

1. База данных Scopus <http://www.scopus.com/home.url>
2. База данных Web of Science <http://apps.webofknowledge.com/>
3. База данных полнотекстовых академических журналов Китая
<http://oversea.cnki.net/>
4. Федеральный портал «Российское Образование». Федеральный центр информационно-образовательных ресурсов. География.
[http://fcior.edu.ru/
catalog/osnovnoe_obshee?discipline_oo=16&class=&learning_character=&accessibility_restriction=](http://fcior.edu.ru/catalog/osnovnoe_obshee?discipline_oo=16&class=&learning_character=&accessibility_restriction=)
5. Электронные базы данных EBSCO <http://search.ebscohost.com/>

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Для достижения поставленных целей преподавания дисциплины реализуются следующие средства, способы и организационные мероприятия:

изучение теоретического материала дисциплины на лекциях с использованием компьютерных технологий;

самостоятельное изучение теоретического материала дисциплины с использованием Internet-ресурсов, методических разработок, специальной учебной и научной литературы;

закрепление теоретического материала при проведении лабораторных работ, выполнение проблемно-ориентированных, творческих заданий.

подготовка материалов для выступления на семинарах по темам курса, участие в дискуссиях.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Перечень материально-технического и программного обеспечения дисциплины приведен в таблице.

Материально-техническое и программное обеспечение дисциплины

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
690922, Приморский край, г. Владивосток, остров Русский, полуостров Саперный, поселок Аякс, 10, корпус D, ауд. D 733,733а. Учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Помещение укомплектовано специализированной учебной мебелью (посадочных мест – 13) Оборудование: ЖК-панель 47", Full HD, LG M4716 ССВА – 1 шт. Доска аудиторная, Моноблок Lenovo C360G- i34164G500UDK с лицензионными программами Microsoft Office 2013 (13 шт.) и аудиовизуальными средствами проектор Panasonic	1С Предприятие (8.2), 7-Zip, ABBYY Lingvo 12, Alice 3, Anaconda 3, Autodesk, CodeBlocks, CorelDRAW X7, Dia, Directum 4.8, DosBox-0.74, Farmanager, Firebird 2.5, FlameRobin, Foxit Reader, Free Pascal, Geany, Ghostscript, Git, Greenfoot, gsview, Inscapе 0.91, Java, Java development Kit, Kaspersky, Lazarus, LibreOffice 4.4, MatLab R2017b, Maxima 5.37.2, Microsoft Expression, Microsoft Office 2013, Microsoft Silverlight Microsoft System Center, Microsoft Visual Studio 2017, MikTeX 2.9, MySQL, NetBeans, Notepad++, Oracle VM VirtualBox, PascalABC.NET, PostgreSQL 9.4, PTC Mathcad, Putty, PyQt GPL v5.4.1 for Python 3.4, Python2.7(3.4, 3.6), QGIS Brighton, RStudio, SAM CoDeC Pack, SharePoint, Strawberry Perl, Tecnomatix, TeXnicCenter, TortoiseSVN, Unity 2017.3.1f1, Veusz, Vim 8.1, Visual Paradigm CE, Windows Kits, Windows Phone SDK 8.1, Xilinx Design Tools, Acrobat Reader DC, Adobe Bridge CS3, Adobe Device Central CS3, Adobe Extend Script Toolkit 2, Adobe Photoshop CS3, DVD-студия Windows, Google Chrome, Internet Explorer, ITMOproctor, Mozilla Firefox, Windows Media Center, WinSCP

Лекции проводятся с использованием проектора и внутренней системы портала ДВФУ. Лабораторные занятия проходят в аудиториях, оборудованных компьютерами типа Lenovo C360G-i34164G500UDK с лицензионными программами Microsoft Visual Studio 2017 и аудиовизуальными средствами проектор Panasonic DLPProjectorPT-D2110XE, плазма LG FLATRON M4716CCBAM4716CJ. Для выполнения самостоятельной работы студенты в жилых корпусах ДВФУ обеспечены Wi-Fi.

В целях обеспечения специальных условий обучения инвалидов и лиц с ограниченными возможностями здоровья в ДВФУ все здания оборудованы пандусами, лифтами, подъемниками, специализированными местами, оснащенными туалетными комнатами, табличками информационно-навигационной поддержки.

VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

Для дисциплины «Параллельное программирование» используются следующие оценочные средства.

Устный опрос:

- 1) устный опрос (УО-1);
- 2) дискуссия (УО-4);

Письменные работы:

- 1) тесты (ПР-1);
- 2) лабораторная работа (ПР-6);
- 3) конспект (ПР-7);
- 4) разноуровневые задачи и задания (ПР-11);
- 5) кейс-задачи (ПР-14).

Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

Оценочные средства для текущей аттестации

Проводится проверка отчетов по лабораторным работам и собеседования, в рамках которых студенты устно объясняют результаты измерений. Аналогично проводится проверка правильности выполнения

заданий по самостоятельной работе. Задание зачтено, если нет ошибок. По текущим ошибкам даются пояснения.

Оценочные средства для промежуточной аттестации

Промежуточная аттестация студентов по дисциплине «Параллельное программирование» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной. Форма отчётности по дисциплине – экзамен (7-й, осенний семестр). Экзамен по дисциплине включает ответы на 3 вопроса, как минимум один из которых направлен на оценку общих теоретических знаний по предмету, и как минимум один – на решение конкретной задачи по реализации параллельной обработки данных или по анализу представленной реализации.

Методические указания по сдаче экзамена

Экзамен принимается ведущим преподавателем. При большом количестве групп у одного преподавателя или при большой численности потока по распоряжению заведующего кафедрой (заместителя директора по учебной и воспитательной работе) допускается привлечение в помощь ведущему преподавателю других преподавателей. В первую очередь привлекаются преподаватели, которые проводили лабораторные занятия по дисциплине в группах.

В исключительных случаях, по согласованию с заместителем директора Школы по учебной и воспитательной работе, заведующий кафедрой имеет право принять экзамен в отсутствие ведущего преподавателя.

Форма проведения экзамена (устная, письменная и др.) утверждается на заседании кафедры по согласованию с руководителем в соответствии с рабочей программой дисциплины.

Во время проведения экзамена студенты могут пользоваться рабочей программой дисциплины, а также с разрешения преподавателя, проводящего экзамен, справочной литературой и другими пособиями (учебниками, учебными пособиями, рекомендованной литературой и т.п.).

Время, предоставляемое студенту на подготовку к ответу на экзамене, должно составлять не более 90 минут. По истечении данного времени студент должен быть готов к ответу.

Присутствие на экзамене посторонних лиц (кроме лиц, осуществляющих проверку) без разрешения соответствующих лиц (ректора либо проректора по учебной и воспитательной работе, директора Школы,

руководителя ОПОП или заведующего кафедрой), не допускается. Инвалиды и лица с ограниченными возможностями здоровья, не имеющие возможности самостоятельного передвижения, допускаются на экзамен с сопровождающими.

При промежуточной аттестации обучающимся устанавливается оценка «неудовлетворительно», «удовлетворительно» «хорошо» или «отлично».

Запись «неудовлетворительно» вносится только в экзаменационную ведомость. При неявке студента на экзамен в ведомости делается запись «не явился».

Критерии оценки устного ответа

✓ 100-86 баллов - если ответ показывает глубокое и систематическое знание всего программного материала и структуры конкретного вопроса, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой. Студент демонстрирует отчетливое и свободное владение концептуально-понятийным аппаратом, научным языком и терминологией соответствующей научной области. Знание основной литературы и знакомство с дополнительно рекомендованной литературой. Логически корректное и убедительное изложение ответа.

✓ 85-76 - баллов - знание узловых проблем программы и основного содержания лекционного курса; умение пользоваться концептуально-понятийным аппаратом в процессе анализа основных проблем в рамках данной темы; знание важнейших работ из списка рекомендованной литературы. В целом логически корректное, но не всегда точное и аргументированное изложение ответа.

✓ 75-61 - балл – фрагментарные, поверхностные знания важнейших разделов программы и содержания лекционного курса; затруднения с использованием научно-понятийного аппарата и терминологии учебной дисциплины; неполное знакомство с рекомендованной литературой; частичные затруднения с выполнением предусмотренных программой заданий; стремление логически определенно и последовательно изложить ответ.

✓ 60-50 баллов – незнание, либо отрывочное представление о данной проблеме в рамках учебно-программного материала; неумение использовать понятийный аппарат; отсутствие логической связи в ответе.

Критерии выставления оценки студенту на экзамене

Баллы (рейтинговой оценки)	Оценка экзамена (стандартная)	Требования к сформированным компетенциям
86-100	«отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач.
76-85	«хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения.
61-75	«удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ.
0-60	«неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.

Оценочные средства для промежуточной аттестации

Вопросы к экзамену

1. Классификация архитектур вычислительных систем. (Классификация Флинна).

2. Реализовать алгоритм

$$w_{ave}(x_1, x_2) = \frac{1}{x_2 - x_1} \int_{x_1}^{x_2} f(x)w(x)dx$$

расчета средневзвешенного значения функции $f(x)$ на интервале $x_1 \leq x \leq x_2$ параллельно, если задана весовая функция $w(x)$, также определенная на интервале $x_1 \leq x \leq x_2$. Вход и выход обеих функций – числа с плавающей точкой двойной точности (double).

Оценить эффективность параллельного выполнения такого вычисления T потоками выполнения.

Пусть функция $f(x)$ имеет побочные эффекты, которые могут приводить к гонкам. Какие изменения будет необходимо внести в реализацию, и как это повлияет на эффективность решения задачи?

3. Основные архитектуры параллельных компьютеров.

4. Пусть задана квадратная матрица A значений с N столбцами, N строками и с рангом N . Реализовать ядро CUDA (и вспомогательные функции устройства), вычисляющее обратную матрицу A^{-1} . При условии, что все элементы матрицы A могут быть помещены в собственной разделяемой памяти одного блока CUDA, а число $N^2 \leq T$, где T – максимальное число потоков CUDA на блок, реализовать алгоритм с использованием разделяемой памяти CUDA.

Интерфейс:

```
void Inverse(/*in*/ const double* A, size_t N, /*out*/ double* AInv);
```

5. Модели параллельного программирования. Основные свойства параллельных алгоритмов.

6. Пусть имеется оцифрованный звуковой сигнал, заданный набором уровней исходного сигнала на дискретном наборе временных отчетов. Такое цифровое представление звукового сигнала можно рассматривать как вектор уровней сигнала.

Пусть уровень звука момент времени определяется 32-х битовым числом с плавающей точкой. С помощью CUDA реализовать фильтр, осуществляющий вейвлет-преобразование Хаара $h(V, N)$, так, чтобы для вектора $V = \{v_i\}_{0 \leq i < N}$ из $N = 2^k$ элементов

$$h(V, N) = \begin{cases} \{v_0\}, N \equiv 1 \\ h\left(\bigcup_{i=0}^{\frac{N}{2}-1} \left\{\frac{v_{2i} + v_{2i+1}}{2}\right\}, \frac{N}{2}\right) \cup \bigcup_{i=0}^{\frac{N}{2}-1} \left\{\frac{v_{2i} - v_{2i+1}}{2}\right\}, N > 1 \end{cases}$$

Интерфейс:

```
void h(const float* input, size_t N, float* output);
```

7. Модель задача-канал. Основные свойства модели "задача/канал".

8. Пусть имеется некоторый канал, через который многопоточному серверу передаются запросы $M = \{m_i\}$ на последовательные вычисления, объем которых независим от запроса и равен t . Опишите возможную логическую организацию вычислений на сервере с T логическими процессорами. Опишите узкие места в предложенной организации и способ их устранения.

Если возможно атомарное получение количества потоков выполнения, единовременно выполняющих ожидание прихода через канал сообщения, то

как повлияет на реализацию и оперативность сервера введение дополнительного параллельного и независимого канала?

9. Модель передачи сообщений.

10. С помощью переменных условий реализовать (с помощью Posix или C11 или C++11 threads) событие, под которым понимается механизм синхронизации со следующим интерфейсом:

Создание экземпляра: без параметров.

Ожидание события с блокировкой ожидающего потока:

```
void wait_event(event);
```

Перевод события в сигнальное состояние:

```
void set_event(event);
```

Сброс события из сигнального состояния:

```
void reset_event(event);
```

Событие, не находящееся в сигнальном состоянии должно блокировать поток, вызывающий метод `wait_event` над ним.

Когда событие переводится в сигнальное состояние, все потоки, ожидающие событие, разблокируются. Потоки, вызывающие `wait_event` над событием, находящимся в сигнальном состоянии, не блокируются.

Событие остается в сигнальном состоянии до тех пор, пока не будет вызван метод `reset_event` над ним.

Только что созданное событие находится в несигнальном состоянии.

11. Модель параллелизма данных.

12. С помощью переменных условий реализовать (с помощью Posix или C11 или C++11 threads) семафор, под которым понимается механизм синхронизации, который ассоциирован с некоторым целочисленным счетчиком потоков. Завладение семафором означает декремент положительного счетчика или, если счетчик равен нулю, блокировку вызывающего потока выполнения до тех пор, пока счетчик не окажется ненулевым. Инкремент счетчика потоков, агрегируемого семафором, связан с операцией освобождения семафора. Семафор создается с заданным целым числом N – максимальным числом потоков, которые одновременно могут владеть экземпляром семафора.

Требуется реализовать следующий интерфейс.

Создание семафора: на вход процедуры создания семафора подается беззнаковое целое ненулевое N .

Завладение семафором:

```
void acquire_semaphore(semaphore);
```

Освобождение семафора:

```
void release_semaphore(semaphore);
```

13. Модель разделяемой памяти.

14. С помощью CUDA реализовать подсчет в случайном большом векторе V из N 32-х битовых целых элементов, которые делятся на два, три и пять.

Интерфейс:

```
void count_multiplies(const int * V, size_t N, unsigned* m2, unsigned * m3, unsigned * m5);
```

15. Совместное и параллельное выполнение программ (Concurrency vs Parallelism).

16. В языке C и в C-подобных (напр., php, perl) языках существует функция `rand()` без параметров, возвращающая псевдослучайное целое число на основе некоторого глобального состояния, которое переносит в себе энтропию (которой, с помощью функции `srand` генератор был проинициализирован) от генерации к генерации. И во многих случаях доступ к этому глобальному состоянию из множества потоков параллельного выполнения не является потокобезопасным. Сформулируйте возможные подходы к реализации параллельного (псевдо-) случайного генератора последовательностей чисел, записываемых в выходной вектор. Укажите на достоинства и недостатки выбранных подходов.

17. Реализация совместных вычислений средствами операционных систем Unix, Windows. Потоки Posix.

18. С помощью CUDA и центрального процессора реализовать гетерогенные параллельные вычисления для генерации в выходной вектор значений (плавающая точка, одинарная точность) синусоиды для заданных диапазоном (x_0, x_{N-1}) и (большим) шагом N значений абсциссы.

Интерфейс:

```
void generate(float* buffer, size_t N, float x0, float xN_1);
```

19. Языковые инструменты для параллельных и совместных вычислений: OpenMP, потоки C11 и C++11.

20. Составить параллельный алгоритм подсчета частот байтовых символов в векторе байт `input` длины N . Результатом должна являться таблица частот – вектор `weights` из 256-ти целых типа `unsigned`.

Реализовать алгоритм для CUDA.

Интерфейс:

```
void gather_weights(const uint8_t* input, size_t N, unsigned* weights);
```

21. Детерминизм параллельных алгоритмов и программ.

22. Пусть двумя векторами $X = \{x_i\}_{0 \leq i < N}$ и $Y = \{y_i\}_{0 \leq i < N}$ заданы два длинных целых беззнаковых числа. Показать параллельную реализацию умножения «в столбик» потоками центрального процессора. Произведение должно быть записано в предоставленный буфер $Z = \{z_i\}_{0 \leq i < 2N}$.

Интерфейс:

```
void parallel_mul(const unsigned* X, const unsigned* Y, size_t N, unsigned* Z);
```

23. Параллелизм, основанный на задачах.

24. Какова временная вычислительная сложность подсчета суммы всех значений в случайном векторе чисел с помощью редукции, если имеется неограниченное количество параллельных логических процессоров?

25. Инструменты синхронизации при реализации совместных вычислений: мьютексы и мониторы.

26. Пусть задан некоторый двусвязный список $list = \{(L, V_0, R)_i\}_{0 \leq i < N}$, каждый i -ый узел которого хранит адреса соседних узлов, левого – L_i и правого – R_i , а также ассоциированное с ним значение $V_{0,i}$ с плавающей точкой одинарной точности (`float`). Для крайнего левого элемента указатель

L_0 равен NULL. Аналогично, для крайнего правого элемента списка указатель R_{N-1} равен NULL.

Ассоциировать каждый список с задачей и, используя SPMD-параллелизм, а также синхронизацию типа «барьер», реализовать алгоритм, который принимает на вход головной (первый) узел списка, а также значение t и изменяет значения $V_{0,i}$ на $V_{t,i}$ согласно формуле:

$$V_{t,i} = \frac{V_{t-1,i-1} + 2V_{t-1,i} + V_{t-1,i+1}}{4}$$

для $0 < i < N - 1$ и $t > 0$.

Реализовать алгоритм на CUDA с интерфейсом:

```
struct ListNode
{
    struct ListNode* L;
    struct ListNode* R;
    float V;
};
void fin_diff(struct ListNode* list, unsigned t);
```

27. Вычисления на графических процессорах. SIMT-параллелизм.

28. Пусть имеется некоторое дерево, которое задается корнем. Элементы дерева никак не упорядочены. Записать алгоритм параллельного поиска элемента с заданным значением в таком дереве, считая, что все элементы уникальны по ассоциированным с ними значениям. Реализовать алгоритм, используя понятие задачи и динамический параллелизм на центральном процессоре.

29. Ускорители вычислений NVIDIA CUDA. Программно-аппаратная архитектура CUDA.

30. С помощью потоков CUDA и, отдельно, потоков CPU реализовать быструю сортировку вектора $V = \{v_i\}_{0 \leq i < N}$. Учитывать ограничения на доступ к памяти за пределами буфера $v_0 \dots v_{N-1}$.

Интерфейс:

```
void parallel_qsort(const float* V, size_t N, float* V_sorted);
```

Буферы V и V_sorted разные и не пересекаются.

31. Гетерогенные вычисления с использованием центральных и графических процессоров.

32. Пусть на двумерном Евклидовом пространстве задан вектор точек $P = \{p_i\}_{0 \leq i < N}$, $N > 2$. Создать параллельный алгоритм поиска пары ближайших друг к другу точек, используя стратегию «разделяй и властвуй». Реализовать алгоритм на центральном процессоре, используя понятие задачи и динамический параллелизм.

```
struct point_t
{
    float x, y;
};
void closest_points(const point_t* P, size_t N, point_t* p1, point_t* p2);
```