

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Вычислительные системы и параллельная обработка данных

Кафедра дискретной математики и информатики
Факультета математики и компьютерных наук

Образовательная программа
01.03.02 - Прикладная математика и информатика

Профиль подготовки
Математическое моделирование и вычислительная математика

Уровень высшего образования
бакалавриат

Форма обучения
очная

Статус дисциплины: вариативная по выбору


Махачкала, 2017

Рабочая программа дисциплины «Вычислительные системы и параллельная обработка данных» доработана в 2017 году в соответствии с требованиями ФГОС ВО по направлению подготовки 01.03.02 - Прикладная математика и информатика (уровень бакалавриата) от 12 марта 2015 г. № 228.


Разработчик: кафедра дискретной математики и информатики,
Магомедов А.М., д.ф.-м.н., профессор
Ибрагимова З.И., ст. лаборант

Рабочая программа дисциплины одобрена:

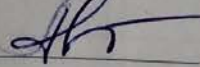
на заседании кафедры дискретной математики и информатики от 6 марта 2017 г.,
протокол № 4.

Зав. кафедрой  Магомедов А.М.
(подпись)

на заседании Методического совета факультета математики и компьютерных наук от 10
марта 2017 г., протокол № 4.

Председатель  Меджидов З.Г.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим

управлением «29» 03 2017 г. 
(подпись)

Аннотация рабочей программы дисциплины

Дисциплина “Вычислительные системы и параллельная обработка данных” входит в вариативную часть образовательной программы бакалавриата по направлению 01.03.02 - Прикладная математика и информатика и является дисциплиной по выбору.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с различными аспектами управления инновационными процессами и инновационными проектами, в том числе управление маркетингом, бизнес- планирование, информационное обеспечение, финансирование инноваций, управление командой проекта, инновационное предпринимательство.

Дисциплина способствует формированию следующих компетенций выпускника: общепрофессиональных: ОПК – 2, профессиональных: ПК – 7, ПК – 13.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, практические занятия, лабораторные занятия.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: в форме 5 контрольных работ конце каждого модуля и итогового экзамена в конце семестра.

Объем дисциплины – 5 зачетных единиц, в том числе в академических часах по видам учебных занятий:

Семес тр	Учебные занятия						СРС	Форма промежуточной аттестации
	в том числе							
	Контактная работа обучающихся с преподавателем							
	Все го	из них						
Лекц ии		Лаборатор ные занятия	Практич еские занятия	КСР	консульт ации			
5	180	18	36	18			108	Экзамен

1. Цели освоения дисциплины

Целями освоения дисциплины «Вычислительные системы и параллельная обработка данных» - приобретение студентами теоретических знаний и устойчивых навыков использования параллельного и распределенного программирования. Основная задача дисциплины: подготовка студентов к работе по созданию и поддержке современных распределенных информационных систем.

2. Место дисциплины в структуре ООП магистратуры

Дисциплина системы управления проектами входит в вариативную часть образовательной программы *бакалавриат* по направлению (специальности) 01.03.02 – Прикладная математика и информатика и является дисциплиной по выбору.

Дисциплина «Вычислительные системы и параллельная обработка данных» логически и содержательно взаимосвязана с такими дисциплинами, как «Основы программирования», «Языки программирования», «Введение в анализ ИТ».

В свою очередь, знания, приобретенные в процессе изучения данной дисциплины, послужат базой для дисциплины «Параллельные вычисления», изучаемой в следующем, 8-м семестре, а также для выполнения выпускных квалификационных работ.

Программа дисциплины предусматривает проведение лекционных занятий, групповую работу студентов на семинарах. Курс предполагает помимо теоретических занятий семинарские занятия. Самостоятельная работа предусматривает освоение предложенной для изучения литературы.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения) .

Компетенции	Формулировка компетенции из ФГОС ВО	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)
ОПК – 2	Способность приобретать новые научные и профессиональные знания, используя современные образовательные и информационные технологии	Знать: стандартные методы синтеза, восстановления, анализа, классификации и распознавания изображений. Уметь: применять на практике изученные подходы и алгоритмы Владеть: навыками постановки научно-исследовательских задач и навыками самостоятельной работы.
ПК – 7	способность к	Знать: основы работы в пакетах

	разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения;	и прикладных программ MathCAD, Mat LAB; Уметь: применять пакеты прикладных программ в области прикладного программирования математических, информационных и имитационных моделей. Владеть: навыками прикладного программирования математических, информационных и имитационных моделей;
ПК – 13	Способность применять существующие и разрабатывать новые методы и средства обучения	Знать: существующие методы моделирования задач. Уметь: составить модели решения задач. Владеть: навыками и средствами в обучении.

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 5 зачетная единица, 180 академических часов.

4.2. Структура дисциплины.

№	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)	
				Лекции	Практические	Лаб. Раб.	Сам. раб	КСР	Общ. тр		
	Модуль 1. Принципы построения параллельных вычислительных систем			4	4	10	18		36	Контрольная работа №1	

1	Введение в предмет. Основные понятия и определения.	5		2	2	4	10		18	Индивидуальный фронтальный опрос.
2	Принципы построения параллельных вычислительных систем.	5		2	2	6	8		18	Устный опрос
Модуль 2. Классификация параллельных компьютеров и систем				4	4	10	18		36	Контрольная работа №2
1	Классификация параллельных компьютеров и систем.	5		2	2	4	6		14	Индивидуальный фронтальный опрос.
2	Технология OpenMP	5		1	1	4	6		12	Устный опрос
3	Параллельные и последовательные области	5		1	1	2	6		10	Устный опрос
Модуль 3. Классификация параллельных компьютеров и систем				4	4	10	18		36	Контрольная работа №3
1	Распределение работы	5		2	2	4	6		14	Индивидуальный фронтальный опрос.
2	Синхронизация	5		1	1	4	6		12	Устный опрос
3	Введение в CUDA C	5		1	1	2	6		10	Устный опрос
Модуль 4. Распараллеливание с разделением памяти				6	6	6	18		36	Лабораторная работа №1

1	Параллельное программирование на CUDA	5		2	2	2	6		12	Индивидуальный фронтальный опрос.	
2	Взаимодействие нитей	5		2	2	2	6		12	Устный опрос	
3	Архитектура высокопроизводительных ЭВМ.	5		2	2	2	6		12	Устный опрос	
Модуль 5											
	Подготовка к экзамену+экс.	5							36	Экзамен	
	Итого								36		
ИТОГО:						18	36	18	108	180	

4.3. Содержание дисциплины, структурированное по темам (разделам).

Темы лекционных занятий

Модуль 1 . Принципы построения параллельных вычислительных систем

Тема 1. Введение в предмет. Основные понятия и определения.

Понятие параллельных вычислений. Необходимость параллельных вычислений. Сдерживающие факторы. Характеристика необходимых знаний и умений.

Тема 2. Принципы построения параллельных вычислительных систем.

Основные виды ЭВМ. Принципы фон-Неймана. Структура традиционных ЭВМ. Способы повышения производительности компьютеров.

Модуль 2. Классификация параллельных компьютеров и систем

Тема 1. Классификация параллельных компьютеров и систем

Классификация Флинна, Хокни, Шнайдера. Взаимосвязь классификаций. Параллельные компьютеры с общей и разделенной памятью. Параллельные компьютеры с сетевой структурой. Характеристика типовых схем коммуникации в многопроцессорных вычислительных системах.

Тема 2. Технология OpenMP

Компиляция программы. Модель параллельной программы. Директивы и функции.
Выполнение программы. Замер времени.

Тема 3. Параллельные и последовательные области

Директива parallel. Переменные среды и вспомогательные функции. Директива single.
Директива master.

Модуль 3. Классификация параллельных компьютеров и систем

Тема 1. Распределение работы

Параллельные циклы. Параллельные секции. Директива workshare. Задачи (tasks).

Тема 2. Синхронизация

Барьер. Директива ordered. Критические секции. Директива atomic.

Тема 3. Введение в CUDA C

Развитие GPU-вычислений. Технология CUDA.

Модуль 4. Распараллеливание с разделением памяти

Тема 1. Параллельное программирование на CUDA C

Организация параллелизма в CUDA. Написание программ на CUDAC.

Тема 2. Взаимодействие нитей

Расщепление параллельных блоков. Разделяемая память и синхронизация.

Темы лабораторных занятий

Модуль 1. Принципы построения параллельных вычислительных систем

Тема 1. Введение в предмет. Основные понятия и определения.

Тема 2. Принципы построения параллельных вычислительных систем.

Модуль 2. Классификация параллельных компьютеров и систем

Тема 1. Классификация параллельных компьютеров и систем

Тема 2. Технология OpenMP

Тема 3. Параллельные и последовательные области

Модуль 3. Вопросы синхронизации

Тема 1. Распределение работы

Тема 2. Синхронизация

Тема 3. Введение в CUDA C

Модуль 4. Распараллеливание с разделением памяти

Тема 1. Параллельное программирование на CUDA C

Тема 2. Взаимодействие нитей

Задания для самостоятельной работы, их содержание и форма контроля приведены в форме таблицы.

Наименование тем	Содержание самостоятельной	Форма контроля
Тема 1. Цели, задачи и проблемы параллельных	Проработка лекционного материала. Изучение	Опрос, оценка выступлений,
Тема 2. Принципы построения	Проработка лекционного материала. Изучение	Опрос, оценка выступлений,
Тема 3. Архитектура высокопроизводительны	Проработка лекционного материала. Изучение	Опрос, оценка выступлений,
Тема 4. Параллельное программирование в	Работа с учебной литературой.	Тестирование, проверка реферата.
Тема 5. Параллельное программирование в	Работа с учебной литературой. Подготовка	Опрос, оценка выступлений,
Тема 6. Параллельное программирование с использованием	Работа с учебной литературой. Подготовка реферата.	Опрос, оценка выступлений, проверка реферата.

Предусмотрено проведение индивидуальной работы (консультаций) со студентами в ходе изучения материала данной дисциплины.

5. Образовательные технологии

Лабораторные работы проводятся в компьютерных классах с использованием меловой доски и мультимедийного проектора. Для проведения лабораторных занятий необходима аудитория, оснащенная компьютерами, мультимедиа-проектором, экраном, доской, ноутбуком (с программным обеспечением для демонстрации слайд-презентаций).

6. Учебно-методическое обеспечение самостоятельной работы студентов.

6.1. Виды и порядок выполнения самостоятельной работы

1. Изучение рекомендованной литературы.
2. Подготовка к отчетам по лабораторным занятиям.
3. Подготовка к коллоквиуму.
4. Подготовка к экзамену.

№	Виды самостоятельной работы	Вид контроля	Учебно-методич. обеспечения
1	Изучение рекомендованной литературы	Устный опрос по разделам дисциплины	См. разделы 6.2, 7.3 данного документа
2	Подготовка к отчетам по лабораторным работам	Проверка выполнения расчетов, оформления работы в лабораторном журнале и проработки вопросов к текущей теме по рекомендованной работе	См. разделы 6.2, 7.3 данного документа
4	Подготовка к коллоквиуму	Промежуточная аттестация в форме контрольной работы	См. разделы 6.2, 7.3 данного документа
5	Подготовка к зачету	Устный опрос, либо компьютерное тестирование	См. разделы 6.2, 7.3 данного документа

Текущий контроль успеваемости осуществляется непрерывно, на протяжении всего курса. Прежде всего, это устный опрос по ходу лабораторных занятий, выполняемый для оперативной активизации внимания студентов и оценки их уровня усвоения тем. Результаты устного опроса учитываются при выборе индивидуальных задач для решения. Каждую неделю осуществляется проверка выполнения заданий, как домашних, так и лабораторных.

Промежуточный контроль проводится в форме коллоквиума, в которых содержатся теоретические вопросы.

Итоговый контроль проводится либо в виде устного зачета, либо в форме тестирования.

Оценка «отлично» ставится за уверенное владение материалом курса.

Оценка «хорошо» ставится при полном выполнении требований к прохождению курса и умении ориентироваться в изученном материале.

Оценка «удовлетворительно» ставится при достаточном выполнении требований к прохождению курса и владении конкретными знаниями по программе курса.

Оценка «неудовлетворительно» ставится, если требования к прохождению курса не выполнены и студент не может показать владение материалом.

6.2. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.

Вопросы для самостоятельного изучения по конкретным разделам (модулям) приведены в п. 7.4 настоящей Программы.

При проведении обучения используются компьютерные программы для редактирования текстов и работы с электронными таблицами, а также доступное через интернет специализированное программное обеспечение и сервисы, не требующие установки. В случаях, когда бесплатного доступа к какому-либо сервису будет недостаточно, преподавателем будет предоставлен доступ с расширенным функционалом. На лекциях и на семинарских занятиях проводится разбор практических задач и кейсов. Возможно проведение деловых, ролевых игр и мастер-классов с участием приглашенных экспертов. Задания для проверочной работы, самостоятельной работы, домашние задания содержатся в пособиях, указанных в списке учебной литературы.

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы.

Перечень компетенций с указанием этапов их формирования приведен в описании образовательной программы.

Компетенция	Знания, умения, навыки	Процедура освоения
ОПК – 2	<p>Знать: стандартные методы синтеза, восстановления, анализа, классификации и распознавания изображений.</p> <p>Уметь: применять на практике изученные подходы и алгоритмы</p> <p>Владеть: навыками постановки научно-исследовательских задач и навыками самостоятельной работы.</p>	Устный опрос, тестирование, Контрольная работа
ПК – 7	<p>Знать: основы работы в пакетах прикладных программ MathCAD, Mat LAB;</p> <p>Уметь: применять пакеты прикладных программ в области прикладного программирования математических, информационных и имитационных моделей.</p> <p>Владеть: навыками прикладного программирования математических,</p>	Устный опрос, тестирование, контрольная работа, лабораторная работа

	информационных и имитационных моделей;	
ПК – 13	Знать: существующие методы моделирования задач. Уметь: составить модели решения задач. Владеть: навыками и средствами в обучении.	Устный опрос, тестирование, контрольная работа, лабораторная работа

7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания.

ОПК – 2

Схема оценки уровня формирования компетенции «способность приобретать новые научные и профессиональные знания, используя современные образовательные и информационные технологии»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Знать: современные теории, методы, системы и средства прикладной математики и информационных технологий для решения научно-исследовательских и прикладных задач;	Имеет неполное представление о методах доказательства математических утверждений	Допускает неточности в анализе информации	Демонстрирует четкое представление об основах получения и обработки информации
Базовый	Уметь: самостоятельно овладевать новыми информационным	Демонстрирует слабое умение осуществлять постановку задач и выполнять	Может осуществлять постановку задач и выполнять	Может эффективно осуществлять постановку задач и

	и технологиями и технологиями программирования в современных средах.	эксперименты по проверке их корректности.	эксперименты по проверке их корректности.	выполнять эксперименты по проверке их корректности.
Продвину- тый	Владеть: навыками поиска необходимой информации и самостоятельного обучения с помощью информационных технологий	Имеет точности при проведении научных исследований, связанных с изучением и обработкой мультимедийных данных.	Владеет навыками проведения научных исследований, связанных с изучением и обработкой мультимедийных данных.	Показывает совершенные знания и навыки проведения научных исследований, связанных с изучением и обработкой мультимедийных данных.

ПК – 7

Схема оценки уровня формирования компетенции «способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Знать: принципы построения, назначение, структуру, функции и основы бизнес-планов научно-прикладных	Демонстрирует слабые знания методов построения математических моделей	Знает построение математической модели той или иной задачи проектной и производственной	Хорошо владеет теоретическим и знаниями по дисциплинам специализации по направлению

	проектов;		технологическо й деятельности	подготовки магистра.
Базовый	Уметь: разрабатывать бизнес -планы научно- прикладных проектов	Слабо применяет на практике компьютерные технологии для решения различных задач.	Умеет применять на практике компьютерные технологии для решения различных задач.	Может эффективно осуществлять постановку задач и выполнять эксперименты по проверке корректности.
Продвину- тый	Владеть: навыками оптимизации бизнес-планов научно- прикладных проектов.	Владеет методами моделирования научно-прикладных проектов.	Владеет способностью разрабатывать модели бизнес- планов.	Отлично владеет навыками оптимизации бизнес-планов.

ПК – 13

Схема оценки уровня формирования компетенции «способность применять существующие и разрабатывать новые методы и средства обучения»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Знать: архитектуру современных высокопроизводи тельных вычислительных	Демонстрирует слабые знания архитектуры современных высокопроизводите льных	Имеет теоретические знания архитектуру современных высокопроизвод	Знает архитектуру современных высокопроизво дительных вычислительны

	систем	вычислительных систем	ительных вычислительных систем	х систем.
Базовый	Уметь: обеспечивать передачу информации между приложениями	Слабо применяет на практике компьютерные технологии для решения различных задач.	Умеет применять на практике компьютерные технологии для решения различных задач.	Может эффективно осуществлять постановку задач и выполнять эксперименты по проверке.
Продвину- тый	Владеть: навыками работы с современными вычислительными средствами	Владеет методами моделирования научно-прикладных проектов.	Владеет способностью работать с вычислительными средствами.	Отлично владеет работы с современными вычислительными средствами

Если хотя бы одна из компетенций не сформирована, то положительная оценки по дисциплине быть не может.

7.3. Типовые контрольные задания

7.3.1. Темы рефератов:

1. Модели параллельного программирования. Мультипроцессорные системы с общей памятью
2. Модели параллельного программирования. Системы с распределенной памятью. Ускорители
3. Обзор современных методов и средств отладки и настройки параллельных программ
4. Архитектуры с распределенной памятью
5. Параллельные методы умножения матрицы на вектор
6. Параллельные методы матричного умножения
7. Введение в Grid - и Cloud - технологии
8. Моделирование и анализ параллельных вычислений
9. Теоретические основы параллельных алгоритмов

10. Оценка коммуникационной трудоемкости параллельных алгоритмов
11. Сравнение архитектуры CPU и GPU
12. Распределенное программирование.
13. Распараллеливание невычислительных задач. Сортировка массивов
14. Автоматическое распараллеливание последовательных программ. Построение ярусно-параллельной формы программы
15. Эволюция GPU
16. Использование нескольких GPU
17. Современные направления развития параллельных вычислительных систем
18. Создание потоков и нитей в одном из языков высокого уровня
19. Оценка эффективности параллельных алгоритмов. Закон Амдала. Задача NC
20. Средства распараллеливания в современных языках программирования
21. Вытесняющая мультизадачность
22. Алгоритмы исключения тупиковых ситуаций
23. Мультизадачность в Windows 10.0
24. Отечественные суперкомпьютеры
25. Алгоритмы маршрутизации. Методы передачи данных. Анализ трудоемкости основных операций передачи данных.
26. Эффективность использования вычислительной системы. Способы оценки показателей.

7.3.2. Перечень заданий к промежуточному (по частям) и итоговому контролю

Задание 1.

Напишите программу, в которой создается k нитей, и каждая нить выводит на экран свой номер и общее количество нитей в параллельной области в формате:

```
I am <Номернити> thread from <Количествонитей> threads!
```

Задание 2.

Изучите конструкции для управления работой с данными shared и private. Напишите программу, в которой создается k нитей, и каждая нить выводит на экран свой номер через переменную rank следующим образом:

```
rank = omp_get_thread_num();  
printf("I am %d thread.\n", rank);
```

Экспериментами определите, общей или частной должна быть переменная rank.

Задание 3.

Напишите программу, в которой две нити параллельно вычисляют сумму чисел от 1 до N. Распределите работу по нитям с помощью оператора if языка C. Для сложения результатов вычисления нитей воспользуйтесь OpenMP-параметром reduction.

Задание 4.

Изучите OpenMP-директиву параллельного выполнения цикла for. Напишите программу, в которой k нитей параллельно вычисляют сумму чисел от 1 до N. Распределите работу по нитям с помощью OpenMP-директивы for.

Задание 5.

Изучите параметр schedule директивы for. Модифицируйте программу «Сумма чисел» из задания 4 таким образом, чтобы дополнительно выводилось на экран сообщение о том, какая нить, какую итерацию цикла выполняет:

[<Номернити>]: calculation of the iteration number <Номеритерации>.

Задайте k = 4, N = 10.

Задание 6.

Напишите OpenMP-программу, которая вычисляет число π с точностью до N знаков после запятой. Используйте следующую формулу:

$$\pi = \left(\frac{4}{1+x_0^2} + \frac{4}{1+x_1^2} + \dots + \frac{4}{1+x_{N-1}^2} \right) \times \frac{1}{N}, \text{ где } x_i = (i+0.5) \times \frac{1}{N}, i = \overline{0, N-1}$$

Распределите работу по нитям с помощью OpenMP-директивы for.

Задание 7.

Напишите OpenMP-программу, которая вычисляет произведение двух квадратных матриц $A \times B = C$ размера $n \times n$. Используйте следующую формулу:

Задание 8.

Изучите OpenMP-директивы создания параллельных секций sections и section. Напишите программу, содержащую 3 параллельные секции, внутри каждой из которых должно выводиться сообщение:

[<Номер нити>]: sameinsection<Номер секции>

Вне секций внутри параллельной области должно выводиться следующее сообщение:

[<Номер нити>]: parallelregion

Запустите приложение на 2-х, 3-х, 4-х нитях. Проследите, как нити распределяются по параллельным секциям.

Задание 9.

Перепишите программу, в которой параллельно вычисляется сумма чисел от 1 до N (см. задание 4), без использования параметра reduction. Вместо параметра reduction используйте директиву atomic.

Задание 10.

Перепишите параллельную программу вычисления числа π (см. задание 6) без использования параметра `reduction`. Вместо параметра `reduction` используйте директиву `critical`.

Перечень вопросов для организации текущего контроля:

1. Параллельные вычисления. Общий смысл.
2. Мультипроцессоры и мультикомпьютеры
3. Обзор программной нотации
4. Программирование с разделяемыми переменными. Процессы и синхронизация.
5. Программирование с разделяемыми переменными. Блокировки и барьеры.
6. Программирование с разделяемыми переменными. Семафоры. Мониторы.
Реализация
7. Распределенное программирование. Передача сообщений.
8. Распределенное программирование. Удаленный вызов процедур и рандеву.
9. Распределенное программирование. Модели взаимодействия процессов.
10. Распределенное программирование. Реализация языковых механизмов
11. Синхронное параллельное программирование. Научные вычисления.
12. Синхронное параллельное программирование. Языки, компиляторы, библиотеки и инструментальные средства

Перечень контрольных вопросов для сдачи зачета:

1. Особенности современных параллельных архитектур. Виды параллельности. Модели параллельного программирования. Характеристики параллельной программы: ускорение, масштабируемость, эффективность.
2. Мультипроцессорные системы с общей памятью. Общая архитектура и подходы к параллельному программированию. Проблемы, возникающие при организации доступа к общим ресурсам.
3. Механизмы управления доступом к критическим ресурсам: активное ожидание, семафоры, мониторы.
4. Библиотека `Pthreads`: управление потоками, управление мютексами, управление условными переменными.
5. Библиотека `OpenMP`: модель программирования, синтаксис, понятие параллельной области и режима ее выполнения.

6. Директивы OpenMP. Параллельные циклы for/do. Параметры директив.
7. Директивы OpenMP. Параллельные секции, директива single. Параметры директив.
8. Системные переменные и системные подпрограммы времени выполнения библиотеки OpenMP.
9. Распараллеливание циклов. Зависимости по данным.
10. Синхронизационные конструкции OpenMP: critical, atomic, barrier, master, ordered, flush.
11. Ошибки, возникающие при программировании с общей памятью: состязание и взаимоблокировка.
12. Кластерные высокопроизводительные вычислительные системы: требования к архитектуре. Коммуникационное оборудование и аппаратно-программные платформы.
13. Модель вычислительной системы с распределенной памятью – LogGP.
14. Программирование в терминах обмена сообщениями. Основные понятия библиотеки MPI: инициализация и выход, понятие коммутатора, сообщение.
15. Блокирующие коммуникации точка-точка: функции отправки и приема.
16. Неблокирующие коммуникации точка-точка: функции отправки и приема, подпрограммы ожидания/проверки.
17. Типы данных MPI (базовые и производные). Соответствие типов данных MPI и типов данных языков C и Fortran. Конструирование производных типов: непрерывно размещенный, векторный, структурный.
18. Коллективные операции распределения данных в MPI: MPI_Bcast, MPI_Scatter, MPI_Gather, MPI_Allgather, MPI_Alltoall. Операции глобальной редукации: minloc и maxloc; определенные пользователем.
19. Односторонние коммуникации.
20. Основные конструкции технологии Cuda.
21. Сравнительный анализ стандартов OpenCL и OpenACC.
22. Средства отладки и настройки параллельных программ.

7.4. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля – 30 % и промежуточного контроля – 70 %.

Текущий контроль по дисциплине включает:

- посещение занятий - 10 баллов,

- участие на лабораторных занятиях - 10 баллов,
- выполнение домашних (аудиторных) контрольных работ - 10 баллов.

Промежуточный контроль по дисциплине включает:

- лабораторная работа - 30 баллов,
- письменная контрольная работа - 40 баллов,

а) основная литература:

1. Воеводин В. В., Воеводин Вл. В. Параллельные вычисления. – СПб.: БХВ-Петербург, 2002. – 608с.
2. Таненбаум Э., ванСтеен М. Распределенные системы. Принципы и парадигмы. – СПб:Питер, 2003. – 877с.
3. Боресков А.В. и др. Параллельные вычисления на GPU. Архитектура и программная модель CUDA. Учеб. пособие. 2-е изд. М.: Издательство Московского университета, 2015. – 336с.
4. Антонов А.С. Технологии параллельного программирования MPI и OpenMP: Учебное пособие. М.: Изд-во МГУ, 2012. – 344с.
5. Лупин С.А., Посыпкин М.А. Технологии параллельного программирования. – М.: ИД «Форум»: ИНФРА-М, 2011. – 208с.
6. Гергель В.П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем: Учебник – М.: Изд-во МГУ, 2010

б) дополнительная литература:

1. Сандерс Дж., Кэндрот Э. Технология CUDA в примерах: введение в программирование графических процессоров. Пер. с англ. Слинкина А.А. М.: ДМК Пресс, 2011. – 232с.
2. Эндрюс Г.Р. Основы многопоточного, параллельного и распределенного программирования. – М.: Вильямс, 2003.
3. Косяков М.С. Введение в распределенные вычисления. – СПб: НИУ ИТМО, 2014. – 155 с.
4. Тель Ж. Введение в распределенные алгоритмы. Пер. с англ. – М.: МЦНМО, 2009. – 616с.
5. Линев А.В., Боголепов Д.К., Бастраков С.И. Технологии параллельного программирования для процессоров новых архитектур: – Учебник. М.: Изд-во МГУ, 2010.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», полезных для освоения дисциплины

1. Поток выполнения [Электронный ресурс] // URL = <https://ru.wikipedia.org/wiki/> время доступа: 21.05.2015
2. [Операционная платформа. Определения и классификация](http://gigabaza.ru/doc/33138.html)[Электронный ресурс] // URL = <http://gigabaza.ru/doc/33138.html> / время доступа: 21.05.2015.
3. [Процессы и потоки in-depth. Обзор различных потоковых моделей](http://habrahabr.ru/post/40227/) [Электронный ресурс] // URL = <http://habrahabr.ru/post/40227/> время доступа: 22.05.2015.
4. Автоматическое распараллеливание программ для распределенных систем. Статическое построение расширенного графа управления [Электронный ресурс] // URL = <http://xreferat.com/33/4673-1/> время доступа: 22.05.2015.
5. <http://www.intuit.ru>[Электронный ресурс]
6. <http://www.parallel.ru> Материалы на сайте Лаборатории параллельных информационных технологий МГУ.

10. Методические указания для обучающихся по освоению дисциплины.

1. При решении лабораторных заданий программистский подход непременно должен присутствовать (без него решение не будет полноценным), однако, он не должен заслонять сугубо математические (доказательство и др.) и алгоритмические (построение, оптимизация, верификация и др.) аспекты.
2. Необходимо обратить внимание на распознавание ситуаций, когда распараллеливание: а) допустимо, б) целесообразно, в) необходимо. Нельзя игнорировать «накладные расходы» ресурсов, выделяемых собственно распараллеливанию вычислений.
3. Важно различать архитектурные и теоретические проблемы распараллеливания.
4. При решении проблемы автоматического распараллеливания особое внимание следует уделить созданию внутреннего представления программы, органично соответствующего проблематике решаемой задачи.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение:

MicrosoftVisualStudioExpress, MicrosoftWindows, UbuntuLinux, Skype. Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

13.Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с современным аппаратным и программным обеспечением. При выполнении лабораторных заданий студенту предоставляется право выбора одного из двух языков программирования из поддерживаемых MS VisualStudio. На сайте кафедры размещаются учебные пособия и презентации к лекции.