

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

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
Параллельное и распределенное программирование

Кафедра дискретной математики и информатики факультета математики и  
компьютерных наук

Образовательная программа  
*02.04.02 - Фундаментальная информатика и информационные технологии*

Направленность (профиль) программы  
*Информационные технологии*

Уровень высшего образования  
*Магистратура*

Форма обучения  
*Очная*

Статус дисциплины: *входит в обязательную часть ОПОП*

Махачкала, 2020

Рабочая программа дисциплины «Параллельное и распределенное программирование» составлена в 2020 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.04.02 - Фундаментальная информатика и информационные технологии (уровень магистратура) от «23» августа 2017 г. №811.

Разработчик(и): кафедра ДМиИ, Гаджиева Т.Ю., к. ф.-м. н., доцент

Рабочая программа дисциплины одобрена:  
на заседании кафедры дискретной математики и информатики  
от «10» марта 2020 г., протокол № 7

Зав. кафедрой Миб Магомедов А.М.  
(подпись)

на заседании Методической комиссии факультета математики и  
компьютерных наук от «15» 03 2020 г., протокол № 4

Председатель Бейбалаев Бейбалаев В.Д.  
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим  
управлением « 26 » 03 2020 г.

Начальник УМУ Гасангаджиева Гасангаджиева А.Г.  
(подпись)

## Аннотация рабочей программы дисциплины

Дисциплина «Параллельное и распределенное программирование» входит в *обязательную часть образовательной программы* по направлению подготовки 02.04.02 - Фундаментальная информатика и информационные технологии.

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

Содержание дисциплины охватывает круг вопросов, связанных с знаниями и навыками построения распределенных офисных приложений с использованием подходов компоновочного объектно-ориентированного проектирования приложений. Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-5; профессиональных – ПК-3, ПК-4.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: *практические занятия, самостоятельная работа.*

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости в форме контрольной работы и промежуточный контроль в форме *зачета.*

Объем дисциплины 2 зачетные единицы, в том числе в академических часах по видам учебных занятий:

Очная форма обучения

Семестр	Учебные занятия							СРС, в том числе экзамен	Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)
	в том числе:								
	всего	Контактная работа обучающихся с преподавателем					консультации		
		всего	Лекции	Лабораторные занятия	Практические занятия	КСР			
3	72	14			14			58	зачет

## 1. Цели освоения дисциплины

Целями освоения дисциплины «Параллельное и распределительное программирование» являются формирование культуры параллельного и распределенного программирования студента, фундаментальная подготовка в области развития методов параллельного программирования, методов программирования на современных компьютерных и суперкомпьютерных системах, овладение аппаратом параллельного и распределенного программирования для дальнейшего использования в приложениях.

## 2. Место дисциплины в структуре ОПОП магистратуры

Дисциплина «Параллельное и распределительное программирование» входит в *обязательную часть образовательной программы* по направлению подготовки 02.04.02 - Фундаментальная информатика и информационные технологии и преподается на 2 курсе в 3 семестре (2 зачетных единицы). Изучение предмета завершается зачетом в конце семестра.

Дисциплина «Параллельное и распределительное программирование» логически и содержательно взаимосвязана с такими дисциплинами, как «Сетевые технологии», «Алгоритмы и анализ сложности», «Дискретная математика».

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

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии с ОПОП)	Планируемые результаты обучения	Процедура освоения
ОПК-5. Способен инсталлировать и сопровождать программное обеспечение информационных систем, осуществлять эффективное управление разработкой программных средств и проектов	ОПК-5.1. Знает методику установки и администрирования информационных систем и баз данных. Знаком с перечнем ПО, входящим в Единый реестр российских программ	Знает: принципы самостоятельного поиска достоверных источников информации. Умеет: обрабатывать, анализировать и синтезировать информацию для выбора метода решения проблемы в стандартных условиях.  Владеет: навыками по общим принципам	Участие в коллективной разработке проектов, в процессе прохождения практики

	ОПК-5.2. Умеет реализовывать техническое сопровождение информационных систем и баз данных	Знает: основы проведения научных исследований в составе группы программистов. Умеет: использовать инструментальные средства. Владеет: навыками коллективной работы с современными вычислительными средствами.	
	ОПК-5.3. Имеет практические навыки установки и инсталляции программных комплексов.	Знает: основы проведения научных исследований в составе группы программистов. Умеет: использовать инструментальные средства. Владеет: навыками работы проектирования локальных сетей, беспроводных локальных сетей.	
ПК-3. Способность проектировать распределенные информационные системы, их компоненты и протоколы их взаимодействия, собирать, обрабатывать и интерпретировать экспериментальные данные, необходимые для проектной и производственно-технологической деятельности; способность к разработке новых алгоритмических, методических и технологических решений в конкретной сфере	ПК-3.1. Знает основы проектирования и элементы архитектурных решений информационных систем.	Знает: основы проведения научных исследований в составе группы программистов. Умеет: использовать инструментальные средства. Владеет: навыками коллективной работы с современными вычислительными средствами.	Участие в коллективной разработке проектов, в процессе прохождения практики
	ПК-3.2. Умеет применять в практической деятельности профессиональные стандарты в области информационных технологий.	Знает: основы проведения научных исследований в составе группы программистов. Умеет: использовать инструментальные средства. Владеет: навыками коллективной работы с современными	

профессиональной деятельности.		вычислительными средствами.	
	ПК-3.3. Имеет практический опыт составления технического задания на разработку информационной системы	Знает: основы проведения научных исследований в составе группы программистов. Умеет: использовать инструментальные средства. Владеет: навыками коллективной работы с современными вычислительными средствами.	
ПК-4. Способность применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии.	ПК-4.1. Знает современные языки программирования и методы параллельной обработки данных.	Знает: основы разработки теоретических моделей. Умеет: разрабатывать модели для задач проектной деятельности. Владеет: навыками разработки простых концептуальных и теоретических моделей.	Участие в коллективной разработке проектов, в процессе прохождения практики
	ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии.	Знает: основы разработки теоретических моделей. Умеет: разрабатывать модели для задач проектной деятельности. Владеет: навыками разработки простых концептуальных и теоретических моделей.	
	ПК-4.3. Имеет практический опыт разработки интеграции информационных систем.	Знает: основы разработки теоретических моделей. Умеет: разрабатывать модели для задач	

		проектной деятельности. Владеет: навыками разработки простых концептуальных и теоретических моделей.	
--	--	---	--

#### 4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 2 зачетные единицы, 72 академических часа.

4.2. Структура дисциплины.

4.2.1. Структура дисциплины в очной форме

№ п/п	Разделы и темы дисциплины по модулям	Семестр	Виды учебной работы, включая самостоятельную работу студентов (в часах)					Самостоятельная работа в	Формы текущего контроля успеваемости и промежуточной аттестации
			Лекции	Практические занятия	Лабораторные занятия	...	Самостоятельная работа в		
Модуль 1. Средства разработки параллельных программ									
1	Цели, задачи и проблемы параллельных вычислений.	3		2			8	Реферат	
2	Принципы построения параллельных вычислительных систем.	3		4			8	Реферат	
3	Архитектура высокопроизводительных ЭВМ	3		2			12	Реферат	
	<i>Итого по модулю 1:</i>	3		8			28		
Модуль 2. Принципы разработки параллельных алгоритмов и программ									
3	Параллельное программирование в потоках	3		2			10	Реферат	
4	Параллельное программирование в процессах	3		2			12	Реферат	
	Параллельное программирование с использованием ускорителей			2			8	Контрольная работа	
	<i>Итого по модулю 2:</i>	3		6			30		
	<b>ИТОГО:</b>	3		14			58		

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

### **4.3.1. Содержание практических занятий по дисциплине.**

#### **Модуль 1. Средства разработки параллельных программ**

Тема 1. Цели, задачи и проблемы параллельных вычислений Цели и задачи параллельной обработки данных. Необходимость и актуальность параллельных вычислений. Различия между многозадачным, параллельным и распределенным режимами выполнения программ.

Тема 2. Обзор параллельных вычислительных систем и их классификация.

Способы организации параллельной обработки данных. Способы построения многопроцессорных вычислительных систем. Краткая история развития высокопроизводительных вычислений. Примеры параллельных вычислительных систем.

Тема 3. Архитектура высокопроизводительных ЭВМ. Введение в вычисления общего назначения с использованием GPU. Основные архитектурные отличия GPU от CPU.

Архитектура современных GPU. Технология CUDA. Модели и шаблоны программирования с использованием технологии CUDA. Модель памяти CUDA. Типы памяти. Оптимизация CUDA-приложений. Модель исполнения CUDA. Компиляция CUDA-приложений.

#### **Модуль 2. Принципы разработки параллельных алгоритмов и программ**

Тема 3. Параллельное программирование в потоках Библиотека OpenMP: модель программирования, синтаксис, понятие параллельной области и режима ее выполнения. Директивы OpenMP. Параллельные циклы for/DO. Параметры директив. Директивы OpenMP. Параллельные секции, директива single.

Параметры директив. Системные переменные и системные подпрограммы времени выполнения библиотеки OpenMP. Синхронизационные конструкции OpenMP: critical, atomic, barrier, master, ordered, flush. Ошибки, возникающие при программировании с общей памятью: состязание и взаимоблокировка.

Тема 4. Параллельное программирование в процессах

Программирование в терминах обмена сообщениями. Основные понятия библиотеки MPI: инициализация и выход, понятие коммутатора, сообщение. Блокирующие коммуникации точка-точка: функции отправки и приема. Неблокирующие коммуникации точка-точка: функции отправки и приема, подпрограммы ожидания/проверки.

Типы данных MPI (базовые и производные). Соответствие типов данных MPI и типов данных языков C и Fortran. Конструирование производных типов: непрерывно размещенный, векторный, структурный. Коллективные операции распределения данных в MPI. Операции глобальной редукции.

Тема 5. Параллельное программирование с использованием ускорителей

Возможности технологии программирования Cuda. Сравнительный анализ технологии Cuda и стандартов OpenCL, OpenACC. Примеры их применения для реализации коммуникационных механизмов.

### **5. Образовательные технологии**

Процесс изложения учебного материала сопровождается презентациями и демонстрацией решения задач в интерактивном режиме с использованием



мультимедийного проектора. Предусмотрено регулярное общение с лектором и представителями российских и зарубежных компаний по электронной почте и по скайпу.

#### **6. Учебно-методическое обеспечение самостоятельной работы студентов.**

Самостоятельная работа студентов складывается из: - проработки лекционного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных на лекциях упражнений); - изучения рекомендованной литературы и материалов соответствующих форумов интернет; - подготовки к отчетам по лабораторным работам; - подготовки к сдаче промежуточных форм контроля.

№	Виды самостоятельной Работы	Вид контроля	Учебно-методич. обеспечения
1	Изучение рекомендованной литературы	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.3, 8, 9 данного документа
2	Подготовка к отчетам по лабораторным работам	Проверка выполнения работ, опрос по теме работы. См. разделы 7.3, 8, 9 данного док	См. разделы 7.3, 8, 9 данного документа
3	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю и прием рефератов.	См. разделы 7.3, 8, 9 данного документа

Пакет заданий для самостоятельной работы выдается в начале семестра, определяются предельные сроки их выполнения и сдачи.

Название раздела и темы	Перечень рассматриваемых вопросов для самостоятельного изучения	Кол-во часов	Литература
<b>Модуль 1. Средства разработки параллельных программ</b>			
Цели, задачи и проблемы параллельных вычислений.	Параллельные вычисления. Общий смысл. Мультипроцессоры и мультикомпьютеры	8	Основная: 2, 3 Дополнительная: 1, 2, 3
Принципы построения параллельных вычислительных систем.	Обзор программной нотации. Программирование с разделяемыми переменными. Процессы и синхронизация.	8	Основная: 2, 3, Дополнительная: 1, 2, 3
Архитектура высокопроизводительных ЭВМ	Программирование с разделяемыми	12	Основная: 2, 3, Дополнительная: 1, 2, 3

	переменными. Семафоры. Мониторы. Реализация. Распределенное программирование. Передача сообщений.		
<b>Модуль 2. Принципы разработки параллельных алгоритмов и программ</b>			
Параллельное программирование потоках	в Синхронное параллельное программирование. Научные вычисления.	10	Основная: 1, 2, 3, Дополнительная: 1, 2, 3
Параллельное программирование процессах	в Синхронное параллельное программирование. Языки, компиляторы, библиотеки и инструментальные средства	12	Основная: 1, 2, 3, Дополнительная: 1, 2, 3
Параллельное программирование использованием ускорителей	с Возможности технологии программирования Cuda. Сравнительный анализ технологии Cuda и стандартов OpenCL, OpenACC.	8	Основная: 1, 2, 3, Дополнительная: 1, 2, 3

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

### **7.1. Типовые контрольные задания**

#### **Контрольная работа 1.**

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 thread from threads!
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`.
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`.

#### **Контрольная работа 2**

1. 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.
- 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: []: calculation of the iteration number . Задайте  $k = 4$ ,  $N = 10$ .
  - Перепишите программу, в которой параллельно вычисляется сумма чисел от 1 до  $N$  (см. задание 1), без использования параметра reduction. Вместо параметра reduction используйте директиву atomic.

### 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.

### Вопросы к зачёту

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 редукции: 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

7.2. 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 - 50% и промежуточного контроля - 50%. Текущий контроль по дисциплине включает: - посещение занятий - 10 баллов, - подготовка реферата – 20 баллов, - выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- прием проекта - 40 баллов,
- письменная контрольная работа - 60 баллов

## **8. Учебно-методическое обеспечение дисциплины.**

а) [http://cathedra.dgu.ru/EducationalProcess\\_Umk.aspx?Value=11&id=6](http://cathedra.dgu.ru/EducationalProcess_Umk.aspx?Value=11&id=6)

б) основная литература:

1. Биллиг В.А. Параллельные вычисления и многопоточное программирование [Электронный ресурс] / В.А. Биллиг. — 2-е изд. — Электрон. текстовые данные. — М.: Интернет Университет Информационных Технологий (ИНТУИТ), 2016. — 310 с. — 2227- 8397. — Режим доступа: <http://www.iprbookshop.ru/73705.html>
2. 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.
3. Антонов А.С. Технологии параллельного программирования MPI и OpenMP: Учебное пособие. М.: Изд-во МГУ, 2012. – 344с.

в) дополнительная литература:

1. Косяков М.С. Введение в распределенные вычисления. – СПб: НИУ ИТМО, 2014. 155 с.
2. Барский А.Б. Параллельные информационные технологии [Электронный ресурс] : учебное пособие / А.Б. Барский. — Электрон. текстовые данные. — Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. — 503 с. — 978-5-4487-0087-3. — Режим доступа: <http://www.iprbookshop.ru/67379.html>
3. Левин М.П. Параллельное программирование с использованием OpenMP [Электронный ресурс] / М.П. Левин. — Электрон. текстовые данные. — М. : Интернет Университет Информационных Технологий (ИНТУИТ), 2016. — 133 с. — 978-5-94774-857-4. — Режим доступа: <http://www.iprbookshop.ru/52216.html>

## **9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.**

*Примеры описания разных видов наименований учебной литературы:*

1. Поток выполнения [Электронный ресурс] // URL = <https://ru.wikipedia.org/wiki/> время доступа: 21.05.2015

2. Операционная платформа. Определения и классификация [Электронный ресурс] // URL = <http://gigabaza.ru/doc/33138.html> / время доступа: 21.05.2015.
3. Процессы и потоки in-depth. Обзор различных потоковых моделей [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.

#### **10. Методические указания для обучающихся по освоению дисциплины.**

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. Самостоятельная работа студентов складывается из: - проработки рекомендованного материала (настоятельно рекомендуется самостоятельное практическое решение всех упражнений); - изучения рекомендованной литературы и материалов соответствующих форумов интернет; - подготовки к отчетам по лабораторным работам; - подготовки к сдаче промежуточных форм контроля (контрольных работ и сдаче реферата). Пакет лабораторных заданий рассчитан на семестр. Рекомендуется выполнять и сдавать задания своевременно с прохождением соответствующего материала.

#### **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: Microsoft Visual Studio Express, Microsoft Windows, Ubuntu Linux, Skype. Также студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

#### **12. 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 необходимым программным обеспечением. Вся основная литература предоставляется студенту в электронном формате.