

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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Параллельные вычисления

Кафедра дискретной математики и информатики

Образовательная программа
02.03.02 - Фундаментальная информатика и информационные технологии

Профиль подготовки
Информатика и компьютерные науки

Уровень высшего образования
бакалавриат

Форма обучения
очная

Статус дисциплины: вариативная (по выбору)

Махачкала, 2017

Аннотация рабочей программы дисциплины

Дисциплина “ Параллельные вычисления ” входит в вариативную по выбору часть образовательной программы бакалавриата по направлению 02.03.02 – 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 современными знаниями в области параллельного и распределенного программирования.

Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-4; профессиональных – ПК-3, ПК-8.

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

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

Объем дисциплины 2 зачетных единиц, в том числе в академических часах по видам учебных занятий

Се- местр	Учебные занятия						СРС, в том числе экза- мен	Форма проме- жуточной атте- стации (зачет, дифференциро- ванный зачет, экзамен
	в том числе							
	Контактная работа обучающихся с преподавателем							
	Все го	из них						
Лек- ции		Лабора- торные занятия	Практи- ческие занятия	КСР	Кон- сульта- ции			
8	72	14					58	зачет

1. Цели освоения дисциплины

Цель изучения дисциплины «Параллельные вычисления» - 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 бакалавриата

Дисциплина « Параллельные вычисления » входит в вариативную часть образовательной программы бакалавриата по направлению 02.03.02 – Фундаментальная информатика и информационные технологии и является дисциплиной по выбору.

Дисциплина «Параллельные вычисления» логически и содержательно взаимосвязана с такими дисциплинами, как «Основы программирования» (1-й семестр), «Языки программирования» (2-й семестр), «Введение в анализ ИТ» (3-4 семестры).

В свою очередь, знания, приобретенные в процессе изучения данной дисциплины, послужат базой для выполнения выпускных квалификационных работ.

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-4	способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Знать: языки программирования Уметь: разрабатывать и оценивать алгоритмы Владеть: навыками работы в средах программирования
ПК-3	способность использовать современные информационные и вычислительные средства	Знать: современные программные средства и библиотеки параллельных вычислений – MPI, OpenMP, CUDA Уметь: свободно пользоваться основными системными средствами современных ОС Владеть: средствами параллельного и распределенного программирования
ПК-8	способность применять на практике международные и профессиональные стандарты информационных технологий, современные парадигмы и методологии, инструментальные и вычислительные средства	Знать: архитектуру современных высокопроизводительных вычислительных систем Уметь: обеспечивать передачу информации между приложениями Владеть: навыками работы с современными вы-

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 2 зачетных единиц, 72 академических часов.

4.2. Структура дисциплины.

№ п/п	Разделы и темы дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Самостоятельная работа	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лаборатор-	Контроль самост. раб.		
Модуль 1. Принципы построения параллельных вычислительных систем									
1	Введение в предмет. Основные понятия и определения.	8	1	2					
2	Принципы построения параллельных вычислительных систем.	8	2-3	2				20	Прием самостоятельных работ
3	Классификация параллельных компьютеров и систем.	8	4-5	2				10	Прием самостоятельных работ
<i>Итого по модулю 1:</i>				6				30	Контрольная работа
Модуль 2. Технология OpenMP									
1	Технология OpenMP	8	6-7	2				7	Прием самостоятельных работ
2	Параллельные и последовательные области	8	8-9	2				7	Прием самостоятельных работ
3	Распределение работы	8	10-11	2				7	Прием самостоятельных работ
4	Синхронизация	8	12	2				7	Прием самостоятельных работ
<i>Итого по модулю 2:</i>				8				28	Зачет
ИТОГО:			72	14				58	

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

Темы лекционных занятий

Первый модуль. 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.

Тема 3. 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.

Второй модуль. Технология OpenMP

Тема 1. Технология OpenMP

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

Тема 2. Параллельные и последовательные области

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

Тема 3. Распределение работы

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

Тема 4. Синхронизация

Барьер. Директива ordered. Критические секции. Директива atomic.

5. Образовательные технологии

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

Систематически проводится сравнение реализации потоков и нитей средствами различных языков Delphi, C# и др. Задания, предлагаемые студентам по различным темам, индивидуализированы, рекомендации по их выполнению, а также по исправлению ошибок, даются как непосредственно на лабораторных занятиях, так и по электронной почте.

Часть рекомендаций и указаний к решению заданий, которая относится ко всей учебной группе, размещается на сайте кафедры. Предусмотрено ведение образовательного блога.

Общение студентов по электронной почте с преподавателями кафедры носит регулярный характер, практикуется (в эпизодической форме) и общение с выпускниками кафедры – представителями российских и зарубежных компаний.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

6.1 Виды самостоятельной работы и её контроля

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 сдаче промежуточных форм контроля (контрольных работ).

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Проработка лекционного материала	Контрольный фронтальный опрос	См. разделы 7.3, 8, 9 данного документа
2.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет	Контрольный фронтальный опрос, прием и представление рефератов.	См. разделы 7.3, 8, 9 данного документа
3.	Подготовка к сдаче промежуточных форм контроля	Контрольные работы по каждому модулю.	См. разделы 7.3, 8, 9 данного документа

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-4	Знать: языки программирования Уметь: разрабатывать и оценивать алгоритмы Владеть: навыками работы в средах программирования	Устный опрос, письменный опрос
ПК-3	Знать: современные программные средства и библиотеки параллельных вычислений – MPI, OpenMP, CUDA Уметь: свободно пользоваться основными системными средствами современных ОС Владеть: средствами параллельного и распределенного программирования	Письменный опрос
ПК-8	Знать: архитектуру современных высокопроизводительных вычислительных систем Уметь: обеспечивать передачу информации между приложениями Владеть: навыками работы с современными вычислительными средствами	Устный опрос

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

ОПК-4

Схема оценки уровня формирования компетенции «способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Решение стандартных задач из иллюстративного материала к лекциям	Знать языки программирования.	Уметь разрабатывать и оценивать алгоритмы.	Владеть навыками работы в Microsoft Visual Studio

ПК-3

Схема оценки уровня формирования компетенции «способность использовать современные информационные и вычислительные средства»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Распознавание ситуаций, в которых целесообразно применять распараллеливание вычислений, и грамотное выполнение процедур распараллеливания	Знать современные программные средства и библиотеки параллельных вычислений – OpenMP, CUDA.	Уметь пользоваться основными системными средствами современных ОС	Владеть средствами параллельного и распределенного программирования.

ПК-8

Схема оценки уровня формирования компетенции «способность применять на практике международные и профессиональные стандарты информационных технологий, современные парадигмы и методологии, инструментальные и вычислительные средства»

Уровень	Показатели (что обучающийся должен продемонстрировать)	Оценочная шкала		
		Удовлетворительно	Хорошо	Отлично
Пороговый	Надежная передача информации между приложениями в соответствии с современными стандартами	Знать архитектуру современных высокопроизводительных вычислительных систем	Уметь обеспечивать передачу информации между приложениями	Владеть навыками работы с современными вычислительными средствами

	менными парадигмами	тельных систем	приложениями	средствами
--	---------------------	----------------	--------------	------------

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

7.3. Типовые контрольные задания

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 <Номер нити> thread from <Количество нитей> 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:

```
[<Номер нити>]: calculation of the iteration number <Номер итерации>.
```

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:

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ \cdot & \cdot & \cdot & \dots & \cdot \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} \quad B = \begin{pmatrix} b_{11} & b_{12} & b_{13} & \dots & b_{1n} \\ b_{21} & b_{22} & b_{23} & \dots & b_{2n} \\ \cdot & \cdot & \cdot & \dots & \cdot \\ b_{n1} & b_{n2} & b_{n3} & \dots & b_{nn} \end{pmatrix}$$

$$c_{im} = \sum_{j=1}^n a_{ij} \cdot b_{jm}; i = 1, 2, \dots, n; m = 1, 2, \dots, n$$

$$C = \begin{pmatrix} \sum_{j=1}^n a_{1j} \cdot b_{j1} & \sum_{j=1}^n a_{1j} \cdot b_{j2} & \sum_{j=1}^n a_{1j} \cdot b_{j3} & \dots & \sum_{j=1}^n a_{1j} \cdot b_{jn} \\ \sum_{j=1}^n a_{2j} \cdot b_{j1} & \sum_{j=1}^n a_{2j} \cdot b_{j2} & \sum_{j=1}^n a_{2j} \cdot b_{j3} & \dots & \sum_{j=1}^n a_{2j} \cdot b_{jn} \\ \dots & \dots & \dots & \dots & \dots \\ \sum_{j=1}^n a_{nj} \cdot b_{j1} & \sum_{j=1}^n a_{nj} \cdot b_{j2} & \sum_{j=1}^n a_{nj} \cdot b_{j3} & \dots & \sum_{j=1}^n a_{nj} \cdot b_{jn} \end{pmatrix}$$

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>]: came in section <Номер секции>

Вне секций внутри параллельной области должно выводиться следующее сообщение:

[<Номер нити>]: parallel region

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`.

Темы для рефератов:

1. Сравнение архитектуры CPU и GPU
2. Эволюция GPU
3. Использование нескольких GPU
4. Современные направления развития параллельных вычислительных систем
5. Создание потоков и нитей в Delphi
6. Средства распараллеливания в современных языках программирования
7. Вытесняющая мультизадачность
8. Алгоритмы исключения тупиковых ситуаций
9. Мультизадачность в Windows 10.0
10. Отечественные супер-ЭВМ

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 - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий - 30 баллов,
- выполнение лабораторных заданий – 20 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 30 баллов,
- письменная контрольная работа - 50 баллов,
- тестирование - 20 баллов.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

а) основная литература:

1. Воеводин В. В., Воеводин Вл. В. Параллельные вычисления. – СПб.: БХВ-Петербург, 2002. – 608с.
2. Таненбаум Э., ван Стеен 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.

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.

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 [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.
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: 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.