

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
Математического обеспечения ЭВМ
наименование кафедры, отвечающей за реализацию дисциплины
 Абрамов Г.В.
подпись, расшифровка подписи
25.04.2022 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.В.ДВ.05.02 Параллельное программирование прикладных задач
Код и наименование дисциплины в соответствии с учебным планом

1. Код и наименование направления подготовки/специальности: _____
15.03.06 Мехатроника и робототехника
2. Профиль подготовки/специализация: Интеллектуальные системы управления в мехатронике и робототехнике
3. Квалификация выпускника: бакалавр
4. Форма обучения: Очная
5. Кафедра, отвечающая за реализацию дисциплины: МО ЭВМ
6. Составители программы: Абрамов Г.В., доктор технических наук, профессор
(ФИО, ученая степень, ученое звание)
7. Рекомендована: Научно-методическим советом факультета прикладной математики, информатики и механики, протокол № 8 от 15.04.2022.
(наименование рекомендующей структуры, дата, номер протокола)
8. Учебный год: 2022-2026 Семестр: 8

9. Цели и задачи учебной дисциплины

Целью дисциплины является изучение моделей, методов и технологий параллельного программирования в объеме, достаточном для успешного начала работ в области параллельного программирования. Излагаемый набор знаний и умений составляет теоретическую основу для методов разработки сложных программ и включают такие темы, как принципы построения параллельных вычислительных систем, анализ параллельных вычислений, принципы разработки параллельных алгоритмов и программ, технологии и системы разработки параллельных программ, параллельные численные алгоритмы для решения типовых задач вычислительной математики. Основной задачей изучения дисциплины является формирование у студентов теоретических знаний и практических навыков разработки алгоритмов и программ, и их реализации.

Задачи учебной дисциплины:

- Изучить архитектуры параллельных вычислительных систем.
- Изучить концепции параллельного программирования.
- Изучить технологии многопоточного программирования OpenMP.
- Изучить технологии программирования GPU CUDA..

10. Место учебной дисциплины в структуре ООП:

Дисциплина относится к части, формируемой участниками образовательных отношений (вариативная) блока Б1 (Б1.В.ДВ.05.02). Для освоения дисциплины студент должен владеть компетенциями дисциплин Б1.О.22 Информатика и программирование, Б1.О.23 Структуры данных и алгоритмы. Дисциплина Б1.В.ДВ.05.02 Параллельное программирование прикладных задач является предшествующей для Б2.В.02(П) Производственная практика (проектно-технологическая) и Б3.01(Д) Выполнение, подготовка к процедуре защиты и защита выпускной квалификационной работы.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ПК-3	Способен осуществлять выбор и создание программного обеспечения для системы управления робототехническими системами в машиностроении под руководством специалиста более высокой квалификации	ПК-3.2	Выполняет написание программ управления гибкими производственным и системами, робототехническими системами и программ сопряжения различных программных сред для управления системами под руководством специалиста более высокой квалификации	<p>Знать:</p> <p>Методы параллельных вычислений для задач вычислительной математики (матричные вычисления, решение систем линейных уравнений, сортировка, обработка графов, уравнения в частных производных, многоэкстремальная оптимизация). Основные подходы к разработке параллельных программ.</p> <p>Уметь:</p> <ul style="list-style-type: none">• Применять общие схемы разработки параллельных программ для реализаций собственных алгоритмов.• Оценивать основные параметры получаемых параллельных программ, таких как ускорение, эффективность и масштабируемость. <p>Владеть:</p> <ul style="list-style-type: none">• основами разработки параллельных программ для МВС с применением технологий OpenMP, CUDA.

		ПК-3.3	Осуществляет их отладку и проверку работоспособности	<p>Знать:</p> <ul style="list-style-type: none"> основы концепций, и синтаксической семантической и организации алгоритмических решений в области разработки параллельного программного обеспечения <p>Уметь:</p> <ul style="list-style-type: none"> выполнять разработку и отладку алгоритмических параллельных вычислений в области прикладного программирования
--	--	--------	--	--

12. Объем дисциплины в зачетных единицах/час. — 2 / 72.

Форма промежуточной аттестации зачет

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость	
		Всего	По семестрам
			8
Аудиторные занятия		32	32
в том числе:	лекции	16	16
	лабораторные	16	16
Самостоятельная работа		40	40
Итого:		72	72

13.1. Содержание дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК*
1. Лекции			
1.1	Параллельные вычисления на CPU с использованием OpenMP	<i>Современное состояние суперкомпьютерной вычислительной техники. Описание параллельного выполнения алгоритма, расписание. Каскадные вычисления, их реализация на векторных процессорах. Оценки эффективности параллельных алгоритмов: ускорение и эффективность. Закон Амдала. Влияние времени передачи данных на эффективность алгоритма.</i>	https://edu.vsu.ru/course/view.php?id=3104
1.2		<i>Параллельные и последовательные области, низкоуровневое распараллеливание, параллельные циклы, параллельные секции, синхронизация</i>	https://edu.vsu.ru/course/view.php?id=3104
1.3		<i>Умножение матриц: линейное, блочное разбиение матриц.</i>	https://edu.vsu.ru/course/view.php?id=3104
1.4		<i>Решение систем линейных уравнений. Алгоритмы параллельной сортировки</i>	https://edu.vsu.ru/course/view.php?id=3104
1.5	Параллельные вычисления на GPU с использованием CUDA	<i>Применение графических процессоров в параллельном программировании потоковых вычислений. Расширения языка C.</i>	https://edu.vsu.ru/course/view.php?id=3104

			104
1.6		Работа с памятью.	
1.7		Примеры решения задач	
1.8		Совместное использование технологий OpenMP и CUDA	
3. Лабораторные занятия			
3.1	Параллельные вычисления на CPU с использованием OpenMP	Реализация матричных вычислений с помощью OpenMP	https://edu.vsu.ru/course/view.php?id=3104
3.2		Разработка программ для типовых параллельных алгоритмов с использованием OpenMP	https://edu.vsu.ru/course/view.php?id=3104
3.3	Параллельные вычисления на GPU с использованием CUDA	Реализация матричных вычислений с помощью CUDA	https://edu.vsu.ru/course/view.php?id=3104
3.3		Разработка программ для типовых параллельных алгоритмов с использованием CUDA	https://edu.vsu.ru/course/view.php?id=3104

* заполняется, если отдельные разделы дисциплины изучаются с помощью онлайн-курса. В колонке Примечание необходимо указать название онлайн-курса или ЭУМК. В других случаях в ячейки ставятся прочерки.

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)			
		Лекции	Лабораторные	Самостоятельная работа	Всего
	Параллельные вычисления на CPU с использованием OpenMP	8	8	20	36
	Параллельные вычисления на GPU с использованием CUDA	8	8	20	36
	Итого:	16	16	40	72

14. Методические указания для обучающихся по освоению дисциплины:

Работа с конспектами лекций, выполнение лабораторных заданий, заданий текущей и промежуточной аттестаций. При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей по работе на LMS-платформе, своевременно подключаться к online-занятиям, соблюдать рекомендации по организации самостоятельной работы

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Источник
1	Борзунов, Сергей Викторович. Параллельное программирование : задачи и решения : учебное пособие / С. В. Борзунов, С. Д. Кургалин ; М-во образования и науки РФ, Воронеж. гос. ун-т .— Воронеж : Издательский дом ВГУ, 2018 .— 112
2	

б) дополнительная литература:

№ п/п	Источник
1	Гергель В.П. Высокопроизводительные вычисления для многопроцессорных многоядерных систем: Учебник – М.: Изд-во МГУ, 2010
2	Воеводин, Валентин Васильевич. Параллельные вычисления : Учебное пособие для студ. вузов, обуч. по направлению 510200 "Прикладная математика и информатика" / В.В. Воеводин, Вл.В. Воеводин .— СПб. : БХВ-Петербург, 2002 .— VII, 599 с. : ил., табл. — ISBN 5-94157-160-7 : 217.27.
3	Лупин, Сергей Андреевич. Технологии параллельного программирования : учебное пособие

	для студентов вузов, обуч. по направлению 230100 "Информатика и вычислительная техника" / С. А. Лупин, М. А. Посыпкин. — М. : Форум : Инфра-М, 2008. — 204, [1] с. : ил. — (Высшее образование). — Библиогр.: с. 147. (20+1ч/з, гр)
--	---

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	Ресурс
1	Антонов А.С. Параллельное программирование с использованием технологии OpenMP: Учебн. пособие. – М.: Изд-во МГУ, 2007. [Электронный ресурс]:[сайт]. URL: http://parallel.ru/info/parallel/openmp
2	Гергель В.П. Теория и практика параллельных вычислений. – [Интернет-ресурс] URL: http://intuit.ru/department/calculate/paralltp/
3	www.lib.vsu.ru – ЗНБ ВГУ
4	Портал «Электронный университет ВГУ» https://edu.vsu.ru/course/view.php?id=3104

* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы, онлайн-курсы, ЭУМК

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению практических (контрольных), курсовых работ и др.)

№ п/п	Источник
1	Абрамов Г.В. Параллельное программирование: лабораторный практикум / Абрамов Г.В., Каплиева Н.А. // Учебно-методическое пособие / Воронежский государственный университет. Воронеж : Издательский дом ВГУ, 2021
2	Конспекты лекций, методические указания к выполнению лабораторных работ, задания, форум на сайте университета edu.vsu.ru

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ, электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются модульно-рейтинговая и личностно-ориентированные технологии обучения (ориентированные на индивидуальность студента, компьютерные и коммуникационные технологии). В рамках дисциплины предусмотрены следующие виды лекций: информационная, лекция-визуализация, лекция с применением обратной связи. Дисциплина реализуется с применением электронного обучения и дистанционных образовательных технологий, для организации самостоятельной работы обучающихся используется онлайн-курс, размещенный на платформе Электронного университета ВГУ(LMS Moodle), а также другие Интернет-ресурсы, приведенные в п.15в.

18. Материально-техническое обеспечение дисциплины:

Учебная аудитория для проведения лекций специализированная мебель, компьютер (мультимедийное оборудование (экран, средства звуковоспроизведения).

Учебная аудитория для практических занятий: специализированная мебель, персональные компьютеры в количестве, обеспечивающем возможность индивидуальной работы, компьютер преподавателя, мультимедийное оборудование (экран). ОС Windows 8 (интернет браузер (Google Chrome, Mozilla Firefox), ПО Adobe Reader, пакет стандартных офисных приложений для работы с документами, таблицами (MS Office, МойОфис, LibreOffice).

Учебная аудитория для проведения лабораторных работ: специализированная мебель, персональные компьютеры для индивидуальной работы с возможностью подключения к сети «Интернет», мультимедийное оборудование (экран, средства звуковоспроизведения). ОС Windows 8 (интернет браузер (Google Chrome, Mozilla Firefox), ПО Adobe Reader, пакет стандартных офисных приложений для работы с документами, таблицами (MS Office, МойОфис, LibreOffice), допускается демоверсия или виртуальный аналог ПО

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Параллельные вычисления на CPU с использованием OpenMP	ПК-3	ПК-3.2	Лабораторные работы № 1, 2
2.	Параллельные вычисления на GPU с использованием CUDA	ПК-3	ПК-3.3	Лабораторные работы № 3, 4
Промежуточная аттестация форма контроля – <u>зачет</u>				Практическое задание

20. Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1. Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторные работы

(наименование оценочного средства текущего контроля успеваемости)

1. Реализация матричных вычислений с помощью OpenMP
2. Разработка программ для типовых параллельных алгоритмов с использованием OpenMP
3. Реализация матричных вычислений с помощью CUDA
4. Разработка программ для типовых параллельных алгоритмов с использованием CUDA

Описание технологии проведения

Требования к выполнению заданий (или шкалы и критерии оценивания)

Критерии оценки лабораторной работы:

Зачтено: Студент разработал программу, пояснил ее работу, выбор, провел анализ результатов, представил пояснительную записку, ответил на вопросы.

Незачтено: разработанная программа не выполняет все функции или студент не ответил на вопросы по выполненной работе.

20.2. Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

Собеседование по билетам к зачету

(наименование оценочного средства промежуточной аттестации)

Описание технологии проведения

Примеры заданий:

1. Разработать программу для умножения матриц (алгоритм Кэннона).
2. Разработать программу для умножения матрицы на вектор.

3. Разработать программу для умножения матриц методом Фокса.
4. Разработка программы для нахождения кратчайших маршрутов с помощью алгоритма Дейкстры.
5. Реализация волнового алгоритма нахождения кратчайшего пути.
6. Разработка программы для нахождения кратчайших маршрутов с помощью алгоритма Форда-Беллмана.
7. Разработать программу решения волнового уравнения.
8. Разработать программу решение задачи распространения тепла в стержне.
9. Разработать программу решения краевой задачи для уравнения теплопроводности методом сеток.

Решение задач выполняется с использованием технологий OpenMP и CUDA

Требования к выполнению заданий, шкалы и критерии оценивания

Критерии оценки зачета:

Зачтено: студент выполнил все лабораторные работы и они оценены положительными оценками;

Незачтено: выставляется студенту, если хотя бы одна из лабораторных работ оценена на «неудовлетворительно».

Промежуточная аттестация проводится на основании итогов выполнения студентом лабораторных работ по всем темам и ответов на контрольные вопросы.

В процессе проведения промежуточной аттестации используется шкала со следующими критериями оценок:

Зачтено - выставляется, если студент выполнил все лабораторные работы в соответствии с заданием, в ходе представления обнаружил и исправил ошибки с помощью преподавателя или выполнил все лабораторные работы в соответствии с заданием, представил ответы на два вопроса билета.

Незачтено – выставляется, если студент выполнил не все лабораторные работы или обнаружил существенные пробелы в знаниях основного материала, ответив не более чем на один контрольный вопрос из трех заданных ему контрольных вопросов.

Приведённые ниже примерные задания для оценки остаточных знаний:

Задания открытого типа

1. Выберите правильные утверждения, соответствующие свойству масштабируемости параллельного приложения
 - приложение обеспечивает ускорение, равное предельному, согласно закону
 - приложение может обрабатывать разные объемы данных
 - приложение максимально задействует все имеющиеся вычислительные ресурсы
 - нет правильных вариантов
2. Для каких языков программирования реализован подход OpenMP?
 - BASIC
 - Fortran
 - C#
 - Java

3. С помощью, какой функции можно задать число потоков в параллельной области программы?
- schedule
 - omp_set_num_threads
 - omp_get_num_threads
 - master_thread

4. Для чего нужна процедура синхронизации?

- для передачи данных между потоками
- для согласования начала выполнения параллельных потоков
- для согласования завершения выполнения параллельных потоков
- для передачи данных из главного потока в параллельные

5. Определите класс по умолчанию для переменной numt:

```
int i=0;
int numt = omp_get_max_threads();
#pragma omp parallel for
for(i=0; i< numt; i++)
Work(i);
```

- threadprivate
- private
- shared

6. Найдите ошибку в следующем фрагменте программы:

```
#define N 10
int A[N],B[N];
#pragma omp parallel default(shared)
num_threads(10)
{
int iam=omp_get_thread_num();
int tmp;
tmp=A[iam];
B[iam]=tmp;
}
```

- чтение/изменение переменной tmp выполняется без какой-либо синхронизации
- изменение общей переменной B[iam] выполняется без какой-либо синхронизации
- в данном фрагменте программы ошибки нет

7. Директива master

- определяет блок операторов в программе, который будет выполнен одной нитью группы. Остальные нити группы дожидаются завершения выполнения этого блока

- определяет блок операторов в программе, который будет выполнен master-нитью. Остальные нити группы не дожидаются завершения выполнения этого блока
- определяет блок операторов в программе, который будет выполнен master-нитью. Остальные нити группы дожидаются завершения выполнения этого блока

Задания закрытого типа

8. В приведенном ниже фрагменте кода на месте пропуска поставьте функцию, которая выводит количество потоков OpenMP

```
#pragma omp parallel
int n;
n = _____();
printf("number of threads %d\n",n);
```

9. Фрагмент кода выполняется только одним потоком OpenMP благодаря директивам _____?

10. Константу _____ нужно задать в качестве параметра функции cudaMemcpy для копирования данных из оперативной памяти компьютера в память графического ускорителя

Задания с развернутым ответом

11. Написать фрагмент программы с использованием технологии OpenMP, которая реализует умножение матриц, размерностью 4 x 4.

12. Написать фрагмент программы с использованием технологии CUDA, которая реализует умножение матриц, размерностью 4 x 4.

13. Разработать программу для сортировки пузырьком с использованием технологии OpenMP.

14. Разработать программу для сортировки пузырьком с использованием технологии CUDA.