Минобрнауки России

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой

Борисов Дмитрий Николаевич

Кафедра информационных систем

5.03.2025

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Б1.В.ДВ.04.03 Параллельные алгоритмы обработки данных

1. Код и наименование направления подготовки/специальности:

09.03.02 Информационные системы и технологии

2. Профиль подготовки/специализация:

Инженерия информационных систем и технологий

3. Квалификация (степень) выпускника:

Бакалавриат

4. Форма обучения:

Очная

5. Кафедра, отвечающая за реализацию дисциплины:

Кафедра информационных систем

6. Составители программы:

Фертиков Вадим Валериевич, кандидат физ.-мат. наук, доцент

7. Рекомендована:

Научно-методическим советом ФКН, протокол НМС ФКН № 5 от 05.03.2025

8. Учебный год:

2027-2028

9.Цели и задачи учебной дисциплины:

формирование профессиональных компетенций будущих бакалавров в области информационных систем и технологий через изучение наиболее общих принципов построения параллельных алгоритмов и связанных вопросов классификации их реализующих параллельных вычислительных систем, практических приемов их применения для решения вычислительных задач и при реализации параллельных приложений. В процессе освоения учебных материалов студент получит знание основных методов разработки параллельных алгоритмов, их графического представления, принципов их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение; представление о разновидностях архитектурных решений и основах анализа производительности современных параллельных систем обработки данных, о принципах их классификации, о современных стандартах на системы программирования для

реализации параллельных вычислений и аппаратуру компонентов вычислительных систем; умение применять перечисленные сведения для практической разработки алгоритмов и реализующих их архитектурных элементов; навыки работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью).

10. Место учебной дисциплины в структуре ООП:

дисциплина относится к блоку дисциплин по выбору части, формируемой участниками образовательных
отношений, Блока 1. Дисциплины (модули) (Б1.В.ДВ); входные знания в объеме обязательных курсов, предусмотренных учебным планом для изучения в семестрах 1 – 5: «Введение в программирование», «Теоретические основы информатики», «Методы вычислений», «Языки и системы программирования», «Алгоритмы и структуры данных», «Архитектура ЭВМ». В свою очередь, знание параллельных алгоритмов обработки данных необходимо студентам для изучения общих профессиональных и специальных дисциплин: «Технологии обработки информации», «Моделирование систем».

№ п/п	Наименование дисциплин учебного плана, с которым организована взаимосвязь дисциплины рабочей программы	№ разделов дисциплины рабочей программы, связанных с указанными дисциплинами
1	Алгоритмы и структуры данных	1, 5, 7, 8
2	Архитектура ЭВМ	2–4, 6, 15
3	Язык программирования Си	19–23

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

Код и название компетенции	Код и название индикатора компетенции	Знания, умения, навыки
ПК-3 Выполнение и управление работами по созданию и сопровождению информационных систем	ПК-3.2 Создание программного кода информационной системы в рамках выполнения работ по созданию (модификации) и сопровождению информационной системы	знать: разновидности архитектурных решений и основы анализа производительности параллельных систем обработки данных, принципы их классификации, стандарты на системы программирования для реализации параллельных вычислений и аппаратуру компонентов вычислительных систем, основные методы разработки параллельных алгоритмов, способы их графического представления, принципы их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение

Код и название компетенции	Код и название индикатора компетенции	Знания, умения, навыки
		уметь: применять перечисленные сведения для практической разработки реализующих алгоритмы параллельных архитектурных элементов, применять перечисленные сведения для практической разработки алгоритмов
		владеть (иметь навык(и)): навыками работы с применением интегрированных сред разработки приложений на языке программирования Си (Microsoft Visual Studio), навыками работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью)

12. Объем дисциплины в зачетных единицах/час:

3/108

Форма промежуточной аттестации:

Зачет с оценкой

13. Трудоемкость по видам учебной работы

Вид учебной работы	Семестр 6	Всего
Аудиторные занятия	64	64
Лекционные занятия	32	32
Практические занятия	16	16
Лабораторные занятия	16	16
Самостоятельная работа	44	44
Курсовая работа		0
Промежуточная аттестация	0	0
Часы на контроль		0
Всего	108	108

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
1	Информационный граф, как средство представления параллельных алгоритмов.	Информационный граф общего вида. Отношения предшествования и безразличия между операторами. Виды параллелизма. Ярусно-параллельная форма информационного графа и критерии ее оптимизации.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
2	Производительность и быстродействие систем обработки данных.	Способы оценки, единицы измерения производительности и быстродействия. Классификация тестов для измерения производительности. Список TOP 500 самых высокопроизводительных вычислительных систем.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
3	Теоретический анализ производительности.	Закон Амдала для идеальной параллельной модели вычислений, эффективность параллельного алгоритма.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
4	Принципы классификации параллельных вычислительных систем по элементам архитектуры.	Разновидности систем команд, методы управления процессом вычислений (командное, потоком данных, потоком запросов), механизмы передачи операндов, синхронизация вычислений. Классификация систем обработки данных Флинна.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
5	Общие принципы построения параллельных алгоритмов.	Параллельное программирование и векторизующий компилятор: потери исходной степени параллелизма, присущего методу решения задачи. Ускорение и сложность алгоритма. Классификация задач и соответственно методов построения алгоритмов.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
6	Векторизация и векторные архитектуры.	Основные векторные операции, разновидности векторных архитектур (векторно-параллельные и векторно-конвейерные), особенности векторно-конвейерной обработки данных и теоретическая оценка ее производительности. Основные приемы векторизации.	-
7	Систолические алгоритмы.	Принципы разработки и примеры реализации систолических алгоритмов. Систолические и волновые вычислительные системы.	Дистанционное чтение лекции. Генераторы заданий на самостоятельную работу. https://edu.vsu.ru/course/view.php?id=2097
8	Выявление неявного параллелизма информационного графа.	Практические приемы выявление неявного параллелизма алгоритма. Преобразование информационного графа методами рекурсивного разбиения и циклической редукции.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
9	Основные классы методов декомпозиции.	Декомпозиция данных и функциональная декомпозиция. Показательные примеры реализации алгоритмов.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
10	Численная устойчивость параллельных алгоритмов.	Элементы теории приближенных вычислений. Анализ устойчивости вычислений с плавающей запятой.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
11	Дедлоки и ливлоки при параллельных вычислениях.	Общие черты и особенности дедлоков и ливлоков. Концепция ресурсов вычислительных систем; граф распределения ресурсов, как средство выявления потенциальной возможности дедлока, а также мониторинга дедлоков и ливлоков в реальном времени. Средства защиты от ливлоков.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
12	Средства защиты от дедлоков.	Необходимые условия возникновения дедлока. Обнаружение (редукция графа распределения ресурсов), обход (алгоритм банкира) и предотвращение дедлоков (принципы Хавендера).	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
13	Дедлоки в коммуникационных средах.	Дедлоки в коммуникационных средах и средства их предотвращения (способы структурирования буферного пула).	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
14	Подсистема коммутации параллельных вычислительных систем.	Топологии сетей, используемые для построения коммуникационной среды. Способы описания (граф межмодульных связей, коммуникационная функция, таблица связей) и основные параметры топологий.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
15	Элементы архитектуры параллельных систем из компонентов высокой степени готовности.	Классификация архитектуры параллельных систем из компонентов высокой степени готовности. Преимущества и недостатки подхода. Состав и устройство основных элементов.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
16	Организация памяти систем из компонентов высокой степени готовности: механизмы обеспечения когерентности данных.	Организация памяти: явный и неявный механизмы обеспечения когерентности данных, алгоритм MESI аппаратного обеспечения когерентности в системах с разделяемой памятью, алгоритмы для систем с распределенной памятью.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
17	Коммутаторы вычислительных систем.	Простые коммутаторы с временным и пространственным разделением, составные коммутаторы блокирующие, с множественными путями, перестраиваемые, неблокирующие.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
18	Стандарты на реализации коммуникационной среды.	Основные особенности стандартных реализаций коммуникационной среды: SCI, MYRINET, Raceway, SRC, Memory Channel.	Дистанционное чтение лекции https://edu.vsu.ru/course/view.php?id=2097
19	Организация параллельного вычислительного процесса на локальной сети персональных компьютеров, логически структурированной как асинхронная вычислительная система с распределенной памятью посредством пакета MPICH.	Знакомство с составом пакета MPICH (реализации стандарта MPI). Изучение особенностей сборки параллельного приложения на примере тестового проекта из состава поставки. Реализация тестового приложения. Изучение программы-загрузчика параллельных приложений.	Элемент "Книга": руководство по установке MPICH. Контроль: элемент "Задание". https://edu.vsu.ru/course/view.php?id=2097
20	Методы организации межпроцессорного обмена сообщениями типа "точка-точка" и простейшие функции коллективного обмена в стандарте MPI.	Лабораторная работа №1. Разработка параллельного алгоритма приближенного расчета ряда, бесконечного произведения или интеграла (в зависимости от индивидуального задания), построение соответствующего информационного графа и реализация в виде параллельного приложения с использованием методов организации межпроцессорного обмена сообщениями типа "точка-точка" и простейших функции коллективного обмена.	Элемент "Книга": руководство по установке MPICH. Контроль: элемент "Задание". https://edu.vsu.ru/course/view.php?id=2097

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК
21	Приемы межпроцессорной передачи структурированных данных с преобразованием в стандарте MPI.	Лабораторная работа №2. Реализация приложения путем описания шаблона для определенного преобразования структуры (вектора, матрицы или др., в соответствии с индивидуальным заданием) и применения его при передаче сообщения.	Элемент "Книга": руководство по реализации параллельного алгоритма. Контроль: элемент "Задание". https://edu.vsu.ru/course/view.php?id=2097
22	Стандартные MPI функции коллективного обмена данными процессов.	Лабораторная работа №3. Изучение стандартных MPI функций коллективного обмена данными процессов и применение определенных из них при реализации соответствующего параллельного алгоритма (согласно индивидуальному заданию).	Элемент "Книга": руководство по реализации параллельного алгоритма. Контроль: элемент "Задание". https://edu.vsu.ru/course/view.php?id=2097
23	Использование распределенных операций стандарта MPI.	Лабораторная работа №4. Изучение принципов действия распределенных операций стандарта MPI. Реализация параллельного приложения (в соответствии с индивидуальным заданием), основанного на их использовании.	Элемент "Книга": руководство по реализации параллельного алгоритма. Контроль: элемент "Задание". https://edu.vsu.ru/course/view.php?id=2097

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
1	Информационный граф, как средство представления параллельных алгоритмов.	1			1	2
2	Производительность и быстродействие систем обработки данных.	1			1	2
3	Теоретический анализ производительности.	1			2	3
4	Принципы классификации параллельных вычислительных систем по элементам архитектуры.	1			2	3
5	Общие принципы построения параллельных алгоритмов.	2	1	1	2	6
6	Векторизация и векторные архитектуры.	2	1	1	2	6
7	Систолические алгоритмы.	2	1	1	2	6
8	Выявление неявного параллелизма информационного графа.	2	1	1	2	6
9	Основные классы методов декомпозиции.	2			2	4
10	Численная устойчивость параллельных алгоритмов.	2	1	1	2	6

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
11	Дедлоки и ливлоки при параллельных вычислениях.	2	1	1	2	6
12	Средства защиты от дедлоков.	2			2	4
13	Дедлоки в коммуникационных средах.	2			2	4
14	Подсистема коммутации параллельных вычислительных систем.	2			2	4
15	Элементы архитектуры параллельных систем из компонентов высокой степени готовности.	2			2	4
16	Организация памяти систем из компонентов высокой степени готовности: механизмы обеспечения когерентности данных.	2			2	4
17	Коммутаторы вычислительных систем.	2			2	4
18	Стандарты на реализации коммуникационной среды.	2			2	4

№ п/п	Наименование темы (раздела)	Лекционные занятия	Практические занятия	Лабораторные занятия	Самостоятельная работа	Всего
19	Организация параллельного вычислительного процесса на локальной сети персональных компьютеров, логически структурированной как асинхронная вычислительная система с распределенной памятью посредством пакета MPICH.		2	2	2	6
20	Методы организации межпроцессорного обмена сообщениями типа "точка-точка" и простейшие функции коллективного обмена в стандарте MPI.		2	2	2	6
21	Приемы межпроцессорной передачи структурированных данных с преобразованием в стандарте MPI.		2	2	2	6
22	Стандартные MPI функции коллективного обмена данными процессов.		2	2	2	6
23	Использование распределенных операций стандарта MPI.		2	2	2	6
		32	16	16	44	108

14. Методические указания для обучающихся по освоению дисциплины

При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей, вовремя подключаться к online занятиям, ответственно подходить к заданиям для самостоятельной работы.

Внеаудиторная самостоятельная работа студентов включает проработку материалов лекций, изучение рекомендованной литературы, подготовку к лабораторным работам и их защитам, выполнение двух письменных самостоятельных работ по лекционному материалу, подготовку к устному опросу и итоговой контрольной работе. Учебно-методическое обеспечение самостоятельной работы – рекомендованная программой литература и комплект программ-генераторов индивидуальных заданий на письменные самостоятельные работы.

Самостоятельная работа в аудитории выполняется под непосредственным руководством преподавателя на лабораторных занятиях, призванных обеспечить выборочное использование лекционного материала для более глубокого изучения организации параллельного вычислительного процесса на локальной сети персональных компьютеров, логически структурированной как асинхронная вычислительная система с распределенной памятью посредством пакета MPICH. Для повышения эффективности руководства при проведении лабораторных занятий, демонстрирующих варианты использования отдельных стандартных конструкций MPI и приемов программирования для решения соответствующих практических задач, необходимо ссылаться на материалы лекций, показывая место решаемых задач в общем плане изложения. Для обеспечения каждого студента индивидуальным комплектом из четырех обязательных для выполнения задач, имеется набор заданий соответствующего объема. Проверка результатов выполнения работы включает требования возможной масштабируемости приложения на количество параллельных процессов, достижения максимального параллелизма и равномерного распределения вычислительной нагрузки, а также хорошего стиля программирования, специфического для используемого языка Си. Демонстрация выполненного лабораторного задания включает обязательную проверку функционирования реализованного приложения под управлением нескольких параллельных процессов (возможно, работающих на различных компьютерах лаборатории), что подразумевает использование стандартного загрузчика параллельных приложений.

Учитывая разницу темпов выполнения индивидуальных заданий, преподаватель обеспечивает выполнение студентами дополнительных заданий (общих для студентов, решивших основную задачу) с целью углубленного изучения материала.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

№ п/п	Источник
1	Параллельные вычислительные системы : учебное пособие / Н. Ю. Сиротинина, О. В. Непомнящий, К. В. Коршун, В. С. Васильев. — Красноярск : СФУ, 2019. — 178 с. — ISBN 978-5-7638-4180-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/157580

б) дополнительная литература:

№ п/п	Источник
1	Башашин, М. В. Практическое введение в технологию MPI на кластере HybriLIT : учебное пособие / М. В. Башашин. — Дубна : Государственный университет «Дубна», 2019. — 50 с. — ISBN 978-5-89847-571-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/154468

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
1	http://www.lib.vsu.ru ЗНБ ВГУ
2	https://edu.vsu.ru Образовательный портал "Электронный университет ВГУ"
3	https://e.lanbook.com ЭБС "Лань"

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
1	Рекомендованная программой литература
2	Конспекты лекций.
3	Комплект программ-генераторов индивидуальных заданий
4	Официальный сайт MPICH (https://www.mpich.org)

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

В качестве интегрированной среды разработки при проведении лабораторных работ студентам рекомендуется использование установленного в компьютерных классах факультета компьютерных наук лицензионного программного обеспечения Microsoft Visual Studio и свободно распространяемого пакета MPICH реализации параллельных вычислений. Для текущего контроля успеваемости наряду с прочим используются две самостоятельные письменные работы по автоматически сгенерированным заданиям. Последние предназначены для дополнительного контроля усвоения обучаемыми раздела № 7 дисциплины. Запустив программу-генератор, студент получает индивидуальное задание, например, следующего содержания: «выписать результат пошагового выполнения систолического алгоритма решения системы линейных уравнений» с конкретными числовыми значениями, генерируемыми случайным образом.

Внеаудиторная самостоятельная работа, как правило, предполагает использование студентами среды разработки Microsoft Visual Studio, официальная процедура установки которой на собственные компьютеры студентов обеспечена факультетом компьютерных наук, а также пакета MPICH, варианты которого, размещенные на официальном сайте, охватывают все наиболее популярные платформы и операционные системы.

При реализации дисциплины могут использоваться технологии электронного обучения и

дистанционные образовательные технологии на базе портала edu.vsu.ru, а также другие доступные ресурсы сети Интернет.

18. Материально-техническое обеспечение дисциплины:

компьютерные классы факультета компьютерных наук, лицензионное программное обеспечение Microsoft Visual Studio, свободно распространяемый пакет MPICH реализации параллельных вычислений, программное обеспечение собственной разработки для генерации заданий на самостоятельную работу вне аудитории.

19. Оценочные средства для проведения текущей и промежуточной аттестаций

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

№ п/п	Разделы дисциплины (модули)	Код компетенции	Код индикатора	Оценочные средства для текущей аттестации
1	Разделы 1, 5, 8, 18,19 Информационный граф, как средство представления параллельных алгоритмов. Общие принципы построения параллельных алгоритмов. Выявление неявного параллелизма информационного графа. Организация параллельного вычислительного процесса на локальной сети персональных компьютеров, логически структурированной как асинхронная вычислительная система с распределенной памятью посредством пакета MPICH. Методы организации межпроцессорного обмена сообщениями типа "точка-точка" и простейшие функции коллективного обмена в стандарте MPI.	ПК-3	ПК-3.2	Комплект задач №1
2	Разделы 6, 7, 20 Векторизация и векторные архитектуры. Систолические алгоритмы. Приемы межпроцессорной передачи структурированных данных с преобразованием в стандарте MPI.			Комплект задач №2 Комплект тестов №1
3	Разделы 9, 21, 22 Основные классы методов декомпозиции. Стандартные MPI функции коллективного обмена данными процессов. Использование распределенных операций стандарта MPI.			Комплект задач №3 Комплект тестов №2

Промежуточная аттестация

Форма контроля - Зачет с оценкой

Оценочные средства для промежуточной аттестации

Комплект КИМ

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.

20.1.1. Примерный перечень практических заданий

Индивидуальное задание №1 Параллельный алгоритм расчета значения числового ряда (произведения, интеграла). Основная цель – освоение метода выявления неявного параллелизма информационного графа и реализация алгоритма с использованием механизма обмена сообщениями «точка-точка». Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выдает результат расчета с допустимой точностью, которая задается оператором; программа должна быть масштабируемой на количество процессов и обязательно должна использовать индивидуальный обмен сообщениями между процессами для реализации итеративного вычисления слагаемых ряда. Пример варианта индивидуального задания из комплекта задач №1:

Изобразите информационный граф алгоритма и рассчитайте значение числового ряда. Алгоритм должен по возможности равномерно распределять нагрузку между ветвями, количество которых заранее неизвестно.

$$\sum_{n=1}^{\infty} \frac{1}{(2n-1)2n(2n+1)} = \ln 2 - \frac{1}{2}$$

Предварительно изучите: Comm_size, Comm_rank, Send, Recv, Bcast, Reduce.

Индивидуальное задание №2 Параллельное приложение для структурного преобразования данных. Основная цель – освоение механизмов описания и использования собственных типов MPI. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выполняет необходимые действия по преобразованию одномерного массива или двумерной матрицы с применением механизма описателей типов MPI; особо приветствуется рациональное использование иерархии типов. Пример варианта индивидуального задания из комплекта задач №2:

Создайте описатель типа и используйте его при передаче данных в качестве шаблона для следующего преобразования: при передаче строки дублируются последовательные триады символов.

abcdef→abcabcdefdef

Предварительно изучите: Type_vector, Type_indexed, Type_commit, Type_free.

Индивидуальное задание №3 Параллельная вычислительная процедура с использованием механизмов коллективного взаимодействия процессов. Основная цель – освоение способов распараллеливания алгоритмов для использования инструментов реализации коллективного обмена данными и коллективных операций MPI. Оценка «зачтено» выставляется, если разработанная согласно индивидуальному заданию программа выполняет необходимые вычисления над элементами массива с применением указанных механизмов коллективного взаимодействия процессов MPI; приветствуется реализация программы, масштабируемой на количество процессов. Пример варианта индивидуального задания из комплекта задач №3:

Используйте функции коллективного обмена данными с целью распределения нагрузки между процессами при решении следующей задачи: найти все элементы одномерного массива произвольной длины, имеющие заданное значение.

Предварительно изучите: Gather[v], Scatter[v], Allgather[v], Alltoall[v].

Для решения следующей задачи реализуйте параллельный алгоритм, использующий распределенные операции, возможно, в сочетании с функциями коллективного обмена данными: найти максимальный элемент в каждой строке двумерной матрицы. Результатом должен быть вектор соответствующей длины.

Предварительно изучите функции Reduce, Allreduce, Reduce_scatter, Scan и предопределенные описатели коллективных операций (MAX, MIN, SUM, PROD и т.д.).

20.1.2. Тестовые задания

При помощи специального комплекта программ-генераторов обучаемые получают индивидуальные задания, которые предназначены для дополнительного контроля усвоения раздела № 7 дисциплины. Запустив программу-генератор, студент получает индивидуальное задание, например, следующего содержания: «выписать результат пошагового выполнения систолического алгоритма решения системы линейных уравнений» с конкретными числовыми значениями, генерируемыми случайным образом. Дополнительно выдается конечный результат вычислений, по которому обучаемый сможет осуществить самоконтроль при решении задачи. Примерная структура задания для письменной самостоятельной работы:

Генератор заданий №1 Систолический алгоритм матричного умножения. Оценка «зачтено» выставляется, если письменная работа выполнена согласно сгенерированному индивидуальному заданию, приведены результаты вычислений на каждом шаге алгоритма, а конечный результат совпал с допустимой точностью с вычисленным на этапе генерации задания проверочным результатом;

Генератор заданий №2 Систолический алгоритм решения системы линейных алгебраических уравнений . Оценка «зачтено» выставляется, если письменная работа выполнена согласно сгенерированному индивидуальному заданию, приведены результаты вычислений на каждом шаге алгоритма, а конечный результат совпал с допустимой точностью с вычисленным на этапе генерации задания проверочным результатом.

20.1.3. Описание технологии проведения

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: устного опроса (индивидуальный опрос, фронтальная беседа); защиты лабораторных работ; а также двух самостоятельных письменных работ по автоматически сгенерированным заданиям при помощи специального комплекта программ-генераторов индивидуальных заданий, которые предназначены для дополнительного контроля усвоения обучаемыми раздела № 7 дисциплины. Запустив программу-генератор, студент получает индивидуальное задание, например, следующего содержания: «выписать результат пошагового выполнения систолического алгоритма решения системы линейных уравнений» с конкретными числовыми значениями, генерируемыми случайным образом. При оценивании результатов устного опроса и защиты лабораторных работ используется качественная шкала оценок. Оценивание результатов выполнения письменных самостоятельных работ предполагает использование количественной шкалы.

20.1.4. Требования к выполнению заданий (или шкалы и критерии оценивания)

Конкретные требования к выполнению каждого практического задания изложены в п.20.1.1. Требования к выполнению письменных самостоятельных работ по тестовым заданиям изложены в п.20.1.2. Для оценивания качества выполнения заданий используется 4-балльная шала: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Приведённые ниже задания рекомендуется использовать при проведении диагностических работ для оценки остаточных знаний по дисциплине

1) тестовые задания – 1 балл

1. Какие операции в модели параллельных вычислений могут выполняться параллельно?

a. Смежные в информационном графе
b. Связанные путем в информационном графе
c. Не связанные путем в информационном графе
d. Не смежные в информационном графе

2. Без наличия какой(их) функции(й) не возможна работа любой MPI программы?

a. MPI_Barrier( )
b. MPI_Finalize( )
c. MPI_Comm_rank( )
d. MPI_Init( )
e. MPI_Comm_size( )
f. MPI_Reduce( )

3. Что входит в реализацию MPI (2 варианта)?

a. Язык программирования
b. Набор управляющих комментариев
c. Библиотека, содержащая функции MPI
d. Исполняющая система

4. Что такое ускорение параллельной программы?

a. Отношение времени работы параллельной программы к времени работы последовательной программы
b. Отношение времени работы последовательной программы ко времени работы параллельной программы
c. Отношение времени работы самого медленного процесса к времени работы самого быстрого
d. Отношение времени работы самого быстрого процесса к времени работы самого медленного

5. К какой модели программирования относится модель, в которой все исполнители (потоки/процессоры) запускают одну программу, а разделение логики между ними основано на уникальных идентификаторах исполнителей?

a. MPI
b. Loop Parallelism
c. SPMD
d. Master/Worker

6. Основная особенность баньян-сети:

a. состоит из трех каскадов простых коммутаторов
b. состоит из коммутаторов 2 x 2
c. существует только один путь от каждого входа к каждому выходу
d. существует как минимум два пути от каждого входа к каждому выходу

7. Что такое конвейеризация?

a. Исполнение нескольких команд одновременно
b. Параллельное выполнение различных частей команд
c. Сохранение данных в сверхбыстрой памяти
d. Обработка данных большого размера

8. Какую стратегию использования памяти предполагает MPI?

a. Разделяемую
b. Распределенную
c. Удаленную
d. Локальную

9. Укажите элементарные операции, на которых основан обмен информацией в MPI.

a. Прием и передача сообщений
b. Удаленный вызов процедур
c. Обращение к области общей памятью
d. Чтение и запись семафоров

10. Как называется уникальный номер процесса в MPI?

a. Порядок
b. Ранг
c. Идентификатор
d. Степень

Ответы на вопросы

Номер вопроса	Ответ (буква)
1.	c
2.	b,d
3.	c,d
4.	b
5.	c
6.	c
7.	b
8.	b
9.	a
10.	b

2) задания с коротким ответом – 2 балла

11. В Вашей программе доля последовательных операций равна 0,4. Какое ускорение согласно

закону Амдала Вы получите на компьютере с 4 процессорами? Ответ округлить до сотых.

12. Коммуникационным _____ сети именуется максимальный путь между любыми двумя узлами. Впишите недостающее слово

13. Библиотека функций, обеспечивающая взаимодействие параллельных процессов с помощью механизма передачи сообщений – ____

Ответы на вопросы

Номер вопроса	Ответ (буква)
11.	1,82
12.	диаметром диаметр
13.	MPI Message Passing Interface message passing interface

3) задания с развернутым ответом – 3 балла

14. Опишите особенности систолических алгоритмов и принципы построения систолических вычислительных структур.

Ответы на вопросы

Номер вопроса	Ответ (буква)
14.	Совмещают свойства конвейерной и матричной обработки: все стадии обработки каждого элемента данных выполняются без обращения к памяти – внутри матрицы однотипных процессоров с регулярными связями. Процессоры, обмениваясь элементами и промежуточными результатами обработки, образуют параллельные конвейеры, по которым продвигаются потоки операндов.

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание особенностей систолических алгоритмов, перечисляет основные принципы построения систолических вычислителей.	3 балла

Обучающийся приводит описание особенностей систолических алгоритмов, в котором допущены незначительные неточности, и верно определяет основные преимущества их применения.	2 балла
Представлено словесное описание идеи систолической обработки, правильно отражающее основную часть ее особенностей.	1 балл
В представленном ответе не раскрыты особенности систолической обработки, не описаны принципы построения систолических матриц.	0 баллов

15. Сформулируйте закон Амдала. Проиллюстрируйте численными примерами наличие ограничений роста ускорения параллельного алгоритма с увеличением количества вычислителей.

Ответы на вопросы

Номер вопроса	Ответ (буква)
15.	Ускорение за счет распараллеливания, согласно закону Амдала, зависит от s – доли последовательных операций в общем объеме вычислений: 1 / (s + (1-s)/N), где N – количество вычислителей. Таким образом, неограниченный рост ускорения с увеличением количества процессоров невозможен.

Критерии оценивания	Шкала оценок
Обучающийся приводит вывод формулы закона Амдала, а также иллюстративные численные примеры с ее использованием, согласно заданию.	3 балла
Обучающийся приводит вывод формулы закона Амдала, в котором допущены незначительные неточности, не влияющие на основной результат – ограничение роста производительности вычислений.	2 балла
Представлена верная словесная формулировка закона Амдала, не подтвержденная численными примерами.	1 балл
В представленном ответе не раскрыта суть закона Амдала, либо вывод формулы содержит грубые ошибки.	0 баллов

16. Изложите принципы классификации архитектур Флинна. Перечислите 4 основных класса, а также подклассы, расширившие классификацию с учетом способа организации памяти.

Ответы на вопросы

Номер вопроса	Ответ (буква)
16.	Классификация Флинна основана на понятиях «потока команд» и «потока данных» и производится по признакам наличия параллелизма в этих «потоках». Расширение классификации использует критерий наличия в архитектуре разделяемой или распределенной памяти.

Критерии оценивания	Шкала оценок
Обучающийся приводит полное и безошибочное описание принципов классификации архитектур Флинна, перечисляет основные классы и подклассы архитектур.	3 балла
Обучающийся приводит описание принципов классификации архитектур Флинна, в котором допущены незначительные неточности, не влияющие на результаты классификации параллельных вычислительных систем.	2 балла
Представлена верная словесная формулировка понятий «потока команд» и «потока данных», но перечислены не все классы Флинна.	1 балл
В представленном ответе нет упоминания о «потоке команд» или «потоке данных», либо не раскрыта суть применения этих понятий для построения классификации.	0 баллов

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств.

20.2.1. Комплект заданий для контрольных работ

Каждый контрольно-измерительный материал для проведения промежуточной аттестации включает задачу, требующую решения с применением теоретических сведений одного из разделов курса для контроля знаний в рамках оценки уровня сформированности компетенции. Примеры контрольно-измерительного материала:

1. Изобразите информационный граф алгоритма расчета значения степенного полинома. Напишите «программы», реализующие этот алгоритм на машинах с управлением потоком команд и с управлением потоком данных. Рассчитайте коэффициент ускорения за счет распараллеливания и эффективность разработанного вами алгоритма.

2. Изобразите топологию вычислительной сети с полной тасовкой для 16 элементов и схему ее реализации на агентах с узлами SCI.

3. Изобразите схему составного коммутатора типа «омега» 9x16. Проиллюстрируйте работу алгоритма управления соединениями в таком коммутаторе. Можно ли построить коммутатор данной размерности на основе каких-либо других прямоугольных коммутаторов, кроме использованных вами? Предложите схему расширения топологии путем добавления внешнего

каскада для получения сбалансированной сети с множественными путями без изменения размерности составного коммутатора. Каково значение показателя избыточности разработанной вами схемы?

4. Изобразите информационный граф для расчета полинома 4-той степени по схеме Горнера (с выявлением неявного параллелизма). Рассчитайте ускорение и эффективность полученного алгоритма, а также коэффициент загрузки процессорных элементов.

5. Продемонстрируйте численным примером работу систолического алгоритма поиска вхождения эталона в двоичную последовательность. Рассчитайте сложность и ускорение алгоритма.

6. Изобразите информационный граф алгоритма расчета значения определенного интеграла методом прямоугольников. Продемонстрируйте накопление погрешности вычислений с плавающей запятой (учесть погрешности нормализации и денормализации). Рассчитайте среднюю квадратическую погрешность результата расчета по Вашему алгоритму.

$$\int_a^b y\,dx \approx \frac{b-a}{n}\left(y_0 + y_1 + \cdots + y_{n-1}\right)$$

7. В контексте алгоритма банкира проанализируйте каждое из приведенных ниже состояний. Если состояние надежно, то покажите при помощи графа распределения ресурсов, каким образом могут завершиться все процессы. Если состояние ненадежно, покажите, каким образом может возникнуть тупик.

Состояние А			Состояние Б		
Процесс	Выделено ресурсов	Максимальная потребность	Процесс	Выделено ресурсов	Максимальная потребность
1	2	6	1	4	8
2	4	7	2	3	9
3	5	6	3	5	8
4	0	2	4	0	0
Резерв		1	Резерв		2

Пример оформления контрольно-измерительного материала:

УТВЕРЖДАЮ

Заведующий кафедрой

информационных систем

_____ Д.Н. Борисов

подпись, расшифровка подписи

14.10.2025

Направление подготовки ___Информационные системы и технологии_____

Дисциплина _____Параллельные алгоритмы обработки данных_____

Курс _____3_____

Форма обучения _____очная_____

Вид аттестации _____промежуточная_____

Вид контроля _____дифференцированный зачет_____

Контрольно-измерительный материал № 1

Приведите показательные фрагменты кода MPI–программы для решения следующих задач.

1. Создайте описатель типа и используйте его при передаче данных в качестве шаблона для следующего преобразования: передаваемая последовательность формируется из символов исходной строки с четными порядковыми номерами.

abcdef -> bdf

Прокомментируйте в коде программы использование функций Type_vector, Type_indexed, Type_commit, Type_free.

2. Используйте функции коллективного обмена данными с целью распределения нагрузки между процессами при решении следующей задачи: найти все элементы одномерного массива произвольной длины, имеющие заданное значение.

Прокомментируйте в коде программы использование функций Gather[v], Scatter[v], Allgather[v], Alltoall[v].

3. Для решения следующей задачи реализуйте параллельный алгоритм, использующий распределенные операции, возможно, в сочетании с функциями коллективного обмена данными: найти максимальный элемент в каждой строке двумерной матрицы. Результатом должен быть вектор соответствующей длины.

Прокомментируйте в коде программы использование функций Reduce, Allreduce, Reduce_scatter, Scan и предопределенных описателей коллективных операций (MAX, MIN, SUM, PROD и т.д.).

Преподаватель _____ __В.В. Фертиков__

20.2.2. Описание технологии проведения

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования. Промежуточная аттестация по итогам освоения дисциплины проводится в форме итоговой контрольной работы и включает решение индивидуальной задачи с применением теоретических сведений одного из разделов курса, что позволяет оценить уровень полученных знаний. При этом предполагается, что оценка степени сформированности умений и навыков выводится по результатам текущих аттестаций.

20.2.3. Требования к выполнению заданий, шкалы и критерии оценивания

При оценивании используется следующая шкала:

5 баллов ставится, если обучающийся демонстрирует полное соответствие знаний, умений, навыков приведенным в таблицах показателям, свободно оперирует приобретенными знаниями, умениями, применяет их при решении практических задач;

4 балла ставится, если обучающийся демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателям, но допускает незначительные ошибки, неточности, испытывает затруднения при решении практических задач;

3 балла ставится, если обучающийся демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателям, допускает значительные ошибки при решении практических задач;

2 балла ставится, если обучающийся демонстрирует явное несоответствие знаний, умений, навыков приведенным в таблицах показателям.

При сдаче зачета с оценкой

оценка «отлично» - 5 баллов

оценка «хорошо» - 4 балла

оценка «удовлетворительно» - 3 балла

оценка «неудовлетворительно» - 2 балла.

Для оценивания результатов обучения на зачете с оценкой используются следующие показатели:

1. знание разновидностей архитектурных решений и основ анализа производительности параллельных систем обработки данных, принципов их классификации;
2. умение применять перечисленные сведения для практической разработки реализующих алгоритмы параллельных архитектурных элементов;
3. владение инструментами поиска в различных источниках информации в изучаемой дисциплиной предметной области;
4. знание стандартов на системы программирования для реализации параллельных вычислений и аппаратуру компонентов вычислительных систем;
5. умение применять перечисленные сведения для практической разработки алгоритмов;
6. владение навыками работы с применением интегрированных сред разработки приложений на языке программирования Си (Microsoft Visual Studio);
7. знание основных методов разработки параллельных алгоритмов, способов их графического представления, принципов их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение;
8. умение применять перечисленные сведения для практической разработки алгоритмов;
9. владение навыками работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью).

Соотношение показателей, критериев и шкалы оценивания результатов обучения.

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок

Сформированные знания о разновидностях архитектурных решений и основах анализа производительности параллельных систем обработки данных, принципах их классификации. Сформированное умение применять перечисленные сведения для практической разработки реализующих алгоритмы параллельных архитектурных элементов. Сформированные навыки владения инструментами поиска в различных источниках информации в изучаемой дисциплиной предметной области. Сформированные знания о стандартах на системы программирования для реализации параллельных вычислений и аппаратуру компонентов вычислительных систем. Сформированное умение применять перечисленные сведения для практической разработки алгоритмов. Сформированные навыки работы с применением интегрированных сред разработки приложений на языке программирования Си (Microsoft Visual Studio). Сформированные знания об основных методах разработки параллельных алгоритмов, способах их графического представления, принципах их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение. Сформированное умение применять перечисленные сведения для практической разработки алгоритмов. Сформированные навыки работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью).	*Повышенный уровень*	*Отлично*

Сформированные, но содержащие отдельные пробелы представления о разновидностях архитектурных решений и основах анализа производительности параллельных систем обработки данных, принципах их классификации. Успешное, но содержащее отдельные пробелы умение применять перечисленные сведения для практической разработки реализующих алгоритмы параллельных архитектурных элементов. Успешные, но содержащие отдельные пробелы навыки владения инструментами поиска в различных источниках информации в изучаемой дисциплиной предметной области. Сформированные, но содержащие отдельные пробелы представления о стандартах на системы программирования для реализации параллельных вычислений и аппаратуру компонентов вычислительных систем. Успешное, но содержащее отдельные пробелы умение применять перечисленные сведения для практической разработки алгоритмов. Успешные, но содержащие отдельные пробелы навыки работы с применением интегрированных сред разработки приложений на языке программирования Си (Microsoft Visual Studio). Сформированные, но содержащие отдельные пробелы представления об основных методах разработки параллельных алгоритмов, способах их графического представления, принципах их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение. Успешное, но содержащее отдельные пробелы умение применять перечисленные сведения для практической разработки алгоритмов. Успешные, но содержащие отдельные пробелы навыки работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью).	*Базовый уровень*	*Хорошо*

Неполное представление о разновидностях архитектурных решений и основах анализа производительности параллельных систем обработки данных, принципах их классификации. Успешное, но не системное умение применять перечисленные сведения для практической разработки реализующих алгоритмы параллельных архитектурных элементов. Успешные, но не системные навыки владения инструментами поиска в различных источниках информации в изучаемой дисциплиной предметной области. Неполное представление о стандартах на системы программирования для реализации параллельных вычислений и аппаратуру компонентов вычислительных систем. Успешное, но не системное умение применять перечисленные сведения для практической разработки алгоритмов. Успешные, но не системные навыки работы с применением интегрированных сред разработки приложений на языке программирования Си (Microsoft Visual Studio). Неполное представление об основных методах разработки параллельных алгоритмов, способах их графического представления, принципах их классификации и анализа с использованием таких характеристик, как вид параллелизма, сложность и ускорение. Успешное, но не системное умение применять перечисленные сведения для практической разработки алгоритмов. Успешные, но не системные навыки работы с широко распространенным инструментарием MPICH – реализацией MPI (Message Passing Interface, стандартного интерфейса прикладных параллельных программных средств для вычислительных систем с распределенной памятью).	*Пороговый уровень*	*Удовлетвори-тельно*
Фрагментарные знания или отсутствие знаний. Фрагмен-тарные умения или отсутствие умений. Фрагментарные навыки или отсутствие навыков.	*–*	*Неудовлетвори-тельно*