

Минобрнауки России
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)**

УТВЕРЖДАЮ



Заведующий кафедрой
Кургалин Сергей Дмитриевич
Кафедра цифровых технологий

03.05.2023

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ФТД.01 Технологии параллельных вычислений

1. Код и наименование направления подготовки/специальности:

02.03.01 Математика и компьютерные науки

2. Профиль подготовки/специализация:

Квантовая теория информации, Распределенные системы и искусственный интеллект

3. Квалификация (степень) выпускника:

Бакалавриат

4. Форма обучения:

Очная

5. Кафедра, отвечающая за реализацию дисциплины:

Кафедра цифровых технологий

6. Составители программы:

Борзунов Сергей Викторович, кандидат физико-математических наук, доцент

7. Рекомендована:

протокол НМС ФКН № 7 от 03.05.2023

8. Учебный год:

2026-2027

Семестры:

7

9. Цели и задачи учебной дисциплины:

дать обзор средств параллельного программирования, сформировать представление о технологиях распределённых вычислений и обработки данных, а также дать практические навыки работы с GRID-системами.

10. Место учебной дисциплины в структуре ООП:

Дисциплина относится к факультативным. Для успешного освоения дисциплины требуется предварительное изучение курса «Параллельное программирование».

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-4 Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования. | ПК-4.1 Знает современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования | Знает основные технологии распределённых вычислений и обработки данных, методы и средства параллельной обработки информации. |
| ПК-5 Способен участвовать в разработке технической документации программных продуктов и программных комплексов | ПК-5.1 Знает основные стандарты, нормы и правила разработки технической документации программных продуктов и программных комплексов | Знает стандарты разработки технической документации для программных комплексов, использующих параллельные алгоритмы. |
| ПК-5 Способен участвовать в разработке технической документации программных продуктов и программных комплексов | ПК-5.2 Умеет использовать их при подготовке технической документации программных продуктов | Умеет использовать стандарты разработки технической документации для программных комплексов, использующих параллельные алгоритмы, для подготовки технической документации разрабатываемых программных продуктов. |
| ПК-5 Способен участвовать в разработке технической документации программных продуктов и программных комплексов | ПК-5.3 Имеет практический опыт подготовки технической документации | Владеет навыками разработки технической документации для программных комплексов, использующих параллельные алгоритмы. |
| ПК-4 Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования. | ПК-4.2 Умеет разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования | Умеет применять технологии параллельных вычислений при решении задач профессиональной деятельности, использовать средства программирования параллельных вычислений с учетом особенностей их реализации. |

| Код и название компетенции | Код и название индикатора компетенции | Знания, умения, навыки |
|---|--|--|
| ПК-4 Способен использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования. | ПК-4.3 Имеет практический опыт разработки и реализации алгоритмов на базе языков и пакетов прикладных программ моделирования | Владеет программными средствами для реализации параллельных вычислений, практическими навыками решения вычислительных задач с помощью технологий параллельного 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 |
| | | | 7 семестр |
| Аудиторные занятия | | 32 | 32 |
| в том числе: | лекции | 16 | 16 |
| | практические | | |
| | лабораторные | 16 | 16 |
| Самостоятельная работа | | 40 | 40 |
| в том числе: курсовая работа (проект) | | | |
| Форма промежуточной аттестации (зачет) | | | |
| Итого: | | 72 | 72 |

13.1. Содержание дисциплины

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---|---|--|
| 1 | Введение в высокопроизводительные вычисления. | Введение. Роль и значение высокопроизводительных вычислений в современном мире. Производительность вычислительных систем. Закон Амдала. | |

| п/п | Наименование раздела дисциплины | Содержание раздела дисциплины | Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК |
|-----|---|--|--|
| 2 | Архитектура параллельных вычислительных систем. | Пути повышения производительности процессоров: CISC- и RISC-процессоры. Конвейеризация. Суперскалярная, VLIW, векторная архитектуры. Кэш-память. Многопроцессорные архитектуры с общей и распределенной памятью. Классификация Флинна. Транспьютеры. | |
| 3 | Параллельные алгоритмы. | Информационный граф алгоритма. Показатели эффективности параллельного алгоритма. Умножение матрицы на вектор. Матричное умножение. Сортировка. Обработка графов. | |

13.2. Темы (разделы) дисциплины и виды занятий

| № п/п | Наименование темы (раздела) | Лекционные занятия | Практические занятия | Лабораторные занятия | Самостоятельная работа | Всего |
|-------|---|--------------------|----------------------|----------------------|------------------------|-------|
| 1 | Введение в высокопроизводительные вычисления. | 4 | 0 | 4 | 10 | 18 |
| 2 | Архитектура параллельных вычислительных систем. | 4 | | 4 | 10 | 18 |
| 3 | Параллельные алгоритмы. | 8 | | 8 | 20 | 36 |
| | | 16 | 0 | 16 | 40 | 72 |

14. Методические указания для обучающихся по освоению дисциплины

Освоение дисциплины складывается из аудиторной работы (учебной деятельности, выполняемой под руководством преподавателя) и внеаудиторной работы (учебной деятельности, реализуемой обучающимся самостоятельно).

Аудиторная работа состоит из работы на лекциях и выполнения практических (или лабораторных) заданий в объеме, предусмотренном учебным планом. Лекция представляет собой последовательное и систематическое изложение учебного материала, направленное на знакомство обучающихся с основными понятиями и теоретическими положениями изучаемой дисциплины. Лекционные занятия формируют базу для практических (или лабораторных) занятий, на которых полученные теоретические знания применяются для решения конкретных практических задач. Обучающимся для успешного освоения дисциплины рекомендуется вести

конспект лекций и практических (лабораторных) занятий.

Самостоятельная работа предполагает углублённое изучение отдельных разделов дисциплины с использованием литературы, рекомендованной преподавателем, а также конспектов лекций, презентационным материалом (при наличии) и конспектов практических (лабораторных) занятий. В качестве плана для самостоятельной работы может быть использован раздел 13.1 настоящей рабочей программы, в котором зафиксированы разделы дисциплины и их содержание. В разделе 13.2 рабочей программы определяется количество часов, отводимое на самостоятельную работу по каждому разделу дисциплины. Большее количество часов на самостоятельную работу отводится на наиболее трудные разделы дисциплины. Для самостоятельного изучения отдельных разделов дисциплины используется перечень литературы и других ресурсов, перечисленных в пунктах 15 и 16 настоящей рабочей программы.

Успешность освоения дисциплины определяется систематичностью и глубиной аудиторной и внеаудиторной работы обучающегося.

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

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

| № п/п | Источник |
|-------|---|
| 1 | Карепова, Е.Д. Основы многопоточного и параллельного программирования [Электронный ресурс]: учеб. пособие / Карепова Е.Д. - Красноярск : СФУ, 2016. — Москва : СФУ, 2016. — 356 с. — Основы многопоточного и параллельного программирования [Электронный ресурс]: учеб. пособие / Карепова Е.Д. - Красноярск : СФУ, 2016. — ISBN 5-7638-3385-0. — <URL: http://www.studentlibrary.ru/book/ISBN9785763833850.html >. |
| 2 | Биллиг, В. А. Параллельные вычисления и многопоточное программирование / В.А. Биллиг. — 2-е изд., испр. — Москва : Национальный Открытый Университет «ИНТУИТ», 2016. — 311 с. : ил., схем. — Библиогр. в кн. — http://biblioclub.ru/ .— <URL: http://biblioclub.ru/index.php?page=book&id=428948 >. |
| 3 | Туральчук, К. А. Параллельное программирование с помощью языка C / К.А. Туральчук. — 2-е изд., испр. — Москва : Национальный Открытый Университет «ИНТУИТ», 2016. — 190 с. : ил. — http://biblioclub.ru/ .— <URL: http://biblioclub.ru/index.php?page=book&id=429098 >. |

б) дополнительная литература:

| № п/п | Источник |
|-------|--|
| 1 | Энтони, У. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ [Электронный ресурс] / Энтони У. ; Пер. с англ. Слинкин А.А. — Москва : ДМК Пресс, 2012. — 672 с. — Книга из коллекции ДМК Пресс - Информатика. — ISBN 978-5-94074-448-1. — <URL: http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=4813 >. |

в) информационные электронно-образовательные ресурсы:

| № п/п | Источник |
|-------|---|
| 1 | ЗНБ ВГУ: https://lib.vsu.ru/ |
| 2 | Электронно-библиотечная система "Университетская библиотека online": http://biblioclub.ru/ |
| 3 | Электронно-библиотечная система "Лань": https://e.lanbook.com/ |
| 4 | Электронно-библиотечная система "Консультант студента": http://www.studmedlib.ru |
| 5 | Электронный университет ВГУ: https://edu.vsu.ru/ |

16. Перечень учебно-методического обеспечения для самостоятельной работы

| № п/п | Источник |
|-------|---|
| 1 | Энтони, У. Параллельное программирование на C++ в действии. Практика разработки многопоточных программ [Электронный ресурс] / Энтони У. ; Пер. с англ. Слинкин А.А. — Москва : ДМК Пресс, 2012 .— 672 с. — Книга из коллекции ДМК Пресс - Информатика .— ISBN 978-5-94074-448-1 .— <URL: http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=4813 >. |

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

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:

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 477

Учебная аудитория: специализированная мебель, ноутбук HP Pavilion Dv9000-er, мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 479

Учебная аудитория: специализированная мебель, компьютер преподавателя i5-8400-2,8ГГц, монитор с ЖК 19», мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 505п

Учебная аудитория: специализированная мебель, компьютер преподавателя i5-3220-3.3ГГц,

монитор с ЖК 17", мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 292

Учебная аудитория: специализированная мебель, компьютер преподавателя Pentium-G3420-3,2ГГц, монитор с ЖК 17", мультимедийный проектор, экран. Система для видеоконференций Logitech ConferenceCam

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 297

Учебная аудитория: специализированная мебель, компьютер преподавателя i3-3240-3,4ГГц, монитор с ЖК 17", мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 380

Учебная аудитория: специализированная мебель, компьютер преподавателя i3-3240-3,4ГГц, монитор с ЖК 17", мультимедийный проектор, экран

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 290

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i7-7800x-4ГГц, мониторы ЖК 27» (12 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 291

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-3220-3,3ГГц, мониторы ЖК 19» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 293

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-8100-3,6ГГц, мониторы ЖК 22» (17 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 295

Компьютерный класс: специализированная мебель, персональные компьютеры на базе

i3-9100-3,6ГГц, мониторы ЖК 24» (14 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 382

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i5-9600KF-3,7ГГц, мониторы ЖК 24» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 383

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i7-9700F-3ГГц, мониторы ЖК 27» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 384

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-2120-3,3ГГц, мониторы ЖК 22» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 385

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-2120-3,3ГГц, мониторы ЖК 19» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 301п

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-2120-3,3ГГц, мониторы ЖК 17» (15 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 303п

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-8100-3,9ГГц, мониторы ЖК 24» (13 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 314п

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-7100-3,6ГГц, мониторы ЖК 19» (16 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

394018, г. Воронеж, площадь Университетская, д. 1, ауд. 316п

Компьютерный класс: специализированная мебель, персональные компьютеры на базе i3-9100-3,6ГГц, мониторы ЖК 19» (30 шт.), мультимедийный проектор, экран.

ПО: ОС Windows v.7, 8, 10, Набор утилит (архиваторы, файл-менеджеры), LibreOffice v.5-7, Пакет MPICH реализации параллельных вычислений, Foxit PDF Reader

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 1-3 | ПК-4 | ПК-4.1 | Письменный опрос |
| 2 | Разделы 1-3 | ПК-5 | ПК-5.1 | Письменный опрос |
| 3 | Разделы 1-3 | ПК-5 | ПК-5.2 | Лабораторные работы |
| 4 | Разделы 1-3 | ПК-5 | ПК-5.3 | Лабораторные работы |
| 5 | Разделы 1-3 | ПК-4 | ПК-4.2 | Лабораторные работы |
| 6 | Разделы 1-3 | ПК-4 | ПК-4.3 | Лабораторные работы |

Промежуточная аттестация

Форма контроля - Зачет

Оценочные средства для промежуточной аттестации

Перечень вопросов для письменного опроса

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

Перечень лабораторных работ

1. Табулирование функций.
2. Компиляция и отладка MPI-программ.
3. Базовые параллельные алгоритмы. Задача о сумме.

Типовое задание для лабораторной работы

Лабораторная работа № 1

«Табулирование функций»

Цель работы: изучение и исследование параллельных методов табулирования функций.

Требования к выполнению работы: выполнение лабораторной работы предусматривает написание программы, реализующей изучаемые методы, и проверку её работы на контрольном примере.

Отчёт о работе проводится в виде собеседования и заключается в демонстрации работы программы, объяснении принципов работы алгоритма и ответов на дополнительные вопросы.

Критерии оценки: для получения оценки «зачтено» необходимо показать высокий уровень владения теоретическим материалом, уметь объяснить принцип работы написанной программы, верно ответить на дополнительные вопросы.

Задание: написать программу, реализующую параллельное вычисление таблицы значений функции с использованием OpenMP. Проверить работу программы на контрольном примере. Исследовать сходимость методов и провести сравнение по числу шагов, потребовавшихся для достижения указанной точности.

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью следующих оценочных средств:

- письменный опрос

Перечень вопросов для письменного опроса

Раздел 1. Введение в высокопроизводительные вычисления

1. Современные концепции и средства параллельного программирования.
2. Принципы программирования для ЭВМ с общей и распределенной памятью.
3. Роль и значение высокопроизводительных вычислений в современном мире.
4. Производительность вычислительных систем.
5. Закон Амдала.

Раздел 2. Технологии параллельного программирования

1. Структура MPI-программы.
2. Сообщения, их передача и прием.
3. Синхронное и асинхронное взаимодействие.
4. Коллективный обмен данными.
5. Виды коллективного обмена, барьеры, широковещательная рассылка данных. Коммуникаторы и топологии.
6. Производные типы данных.
7. Компиляция и отладка MPI-программ.
8. Система программирования OpenMP.

Раздел 3. Параллельные алгоритмы

1. Информационный граф алгоритма.
2. Показатели эффективности параллельного алгоритма.
3. Умножение матрицы на вектор.

4. Матричное умножение.
5. Сортировка.

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

тестовые задания - 1 балл

1. Компиляция программы в рамках стандарта MPI может быть выполнена с помощью команды:

- a) `mpirun`
- b) **`mpifort`**
- c) `mpicxx`
- d) `mpicc`

2. Запуск программы в рамках стандарта MPI может быть выполнен с помощью команды:

- a) **`mpirun`**
- b) `mpiexec`
- c) `mpicc`
- d) `mpifort`

3. Получить значение ранга процесса можно с помощью подпрограммы MPI:

- a) `MPI_Comm_size`
- b) **`MPI_Comm_rank`**
- c) `MPI_Init`
- d) `MPI_Finalize`

4. Получить значение размера коммуникатора можно с помощью подпрограммы MPI:

- a) **`MPI_Comm_size`**
- b) `MPI_Comm_rank`
- c) `MPI_Init`
- d) `MPI_Finalize`

5. Стандартный коммуникатор, содержащий все запущенные процессы MPI-программы, называется

- a) **`MPI_COMM_WORLD`**
- b) `MPI_COMM_NULL`
- c) `MPI_ANY_TAG`
- d) `MPI_ANY_SOURCE`

6. Первым по порядку вызовом подпрограммы MPI может быть вызов:

- a) `MPI_Initialized`
- b) **`MPI_Init`**
- c) `MPI_Comm_rank`
- d) `MPI_Comm_size`

7. В коллективной операции обмена данных принимают участие:

- a) не более двух процессов
- b) ровно два процесса
- c) **все процессы, принадлежащие указанному коммуникатору**
- d) не менее двух процессов

8. Для выполнения широковещательной рассылки сообщений применяется:

- a) **`MPI_Bcast`**
- b) `MPI_Bsend`
- c) `MPI_Send`
- d)

9. К классу коллективных операций MPI относятся:

- a) **сбор данных**
- b) **распределение данных**
- c) **операция редукции данных**
- d) обмен по готовности

10. В рамках операции редукции можно выполнить

- a) **суммирование элементов массива**
- b) умножение двух матриц
- c) **определение максимального элемента**
- d) **перемножение элементов массива**

11. Операция широковещательной рассылки

- a) **пересылает одни и те же данные всем процессам**
- b) выполняет частичную редукцию
- c) пересылает всем остальным процессам разные фрагменты данных
- d)

12. Подпрограмма MPI_Gather:

- a) выполняет широковещательную рассылку
- b) **выполняет сбор данных**
- c) **является операцией коллективного обмена данными**
- d)

13. Подпрограмма MPI_Scatter:

- a) выполняет распределение данных между процессами
- b) выполняет широковещательную рассылку
- c) является операцией синхронизации
- d)

14. Какие из следующих операций выполняют сбор данных на одном процессе

- a) **MPI_Gather**
- b) **MPI_Gatherv**
- c) MPI_Allgather
- d)

15. Какие из перечисленных операций относятся к классу векторных

- a) MPI_Alltoall
- b) **MPI_Alltoallv**
- c) MPI_Allgather
- d) **MPI_Allgatherv**

16. Подпрограмма MPI_Alltoall:

- a) выполняет операцию частичной редукции
- b) выполняет операцию полной редукции
- c) выполняет операцию синхронизации
- d) **выполняет передачу данных по схеме «каждый-всем»**

17. Подпрограмма MPI_Scan:

- a) **выполняет операцию частичной редукции**
- b) выполняет операцию полной редукции
- c) выполняет операцию синхронизации
- d) выполняет передачу данных по схеме «точка-точка»

18. В рамках стандарта MPI создание производного типа в MPI выполняется функцией:

- a) MPI_Comm_create
- b) MPI_Type_create
- c) **MPI_Type_create_struct**
- d)

19. В рамках стандарта MPI создание структурного типа в MPI выполняется функцией

- a) **MPI_Type_struct**
- b) MPI_Type_create
- c) MPI_Comm_create
- d)

20. Коммуникатором в MPI называется:

- a) **группа процессов, наделенная общим контекстом обмена данными**
- b) подпрограмма, выполняющая передачу или прием сообщения
- c) подпрограмма для обработки сигналов, поступающих от операционной системы
- d)

21. Для создания коммуникатора предусмотрена команда

- a) MPI_Comm_size
- b) **MPI_Comm_create**
- c) MPI_Comm_rank
- d)

22. Создать группу коммуникаторов можно с помощью команды

- a) **MPI_Group_excl**
- b) **MPI_Group_incl**
- c) **MPI_Group_difference**
- d) MPI_Group_free

23. Подпрограмма MPI_Comm_size:

- a) является операцией двухточечного обмена
- b) **возвращает количество процессов в указанном коммуникаторе**
- c) задает количество процессов в указанном коммуникаторе
- d) проверяет возможность создания коммуникатора с указанным размером

24. Определить количество процессов в коммуникаторе можно с помощью:

- a) **MPI_Comm_size**
- b) MPI_Comm_rank
- c) MPI_Comm_create
- d)

25. В MPI существуют следующие типы двухточечных обменов:

- a) **стандартный обмен**
- b) **обмен по готовности**
- c) **синхронный обмен**
- d) **обмен с буферизацией**

26. Стандартная блокирующая двухточечная передача выполняется

- a) **MPI_Send**
- b) MPI_Ssend
- c) MPI_Bsend
- d) MPI_Rsend

27. Двухточечная передача с буферизацией выполняется функцией

- a) MPI_Send
- b) MPI_Ssend
- c) **MPI_Bsend**
- d) MPI_Rsend

28. Неблокирующий вариант операций передачи сообщений существует для:

- a) **стандартной операции обмена данными**
- b) **обмена с буферизацией**

c) обмена «по готовности»

d)

29. Неблокирующий прием в MPI выполняется :

a) **MPI_Irecv**

b) MPI_Recv

c) MPI_Reduce_Scatter

d)

30. Подпрограмма MPI_Wait предназначена для:

a) **блокирующей проверки выполнения обмена**

b) неблокирующей проверки выполнения обмена

c) приостановки выполнения программы на заданный период времени

d)

31. Подпрограмма MPI_Test предназначена для:

a) блокирующей проверки выполнения обмена

b) **неблокирующей проверки выполнения обмена**

c) приостановки выполнения программы на заданный период времени

d)

32. Подпрограмма MPI_Testall выполняет проверку:

a) завершения любой операции обмена из некоторого множества

b) **завершения всех обменов**

c) завершения одного обмена

d)

33. Подпрограмма MPI_Testany выполняет проверку:

a) **завершения любой операции обмена из некоторого множества**

b) завершения всех обменов

c) завершения одного обмена

d)

34. Неблокирующая передача с буферизацией выполняется :

a) **MPI_IbSEND**

b) MPI_BSEND

c) MPI_Test

d) MPI_Testall

35. Средством разработки многопоточных программ является:

a) **OpenMP**

b) PVM

c) MPI

d)

36. Под "гонкой данных" понимается ситуация, когда:

a) не хватает долговременной памяти для размещения данных

b) **несколько потоков работают с разделяемыми данными, и конечный результат зависит от порядка доступа потоков к данным**

c) несколько потоков работают с данными, локализованными для каждого из потоков, и конечный результат собирается в одну переменную

d) не хватает оперативной памяти для размещения данных

37. Какие существуют подходы к использованию ресурсов параллелизма в разрабатываемой программе

a) **автоматическое распараллеливание последовательной версии программы средствами компилятора**

b) **использование специализированных языков для параллельного программирования**

c) **использование библиотек, предоставляющих возможности параллельного исполнения кода**

d) программирование с использованием особых расширений языка — средств распараллеливания

38. Какие виды зависимости операторов в программе существуют

- a) **зависимость по выходу**
- b) **поточковая зависимость**
- c) **антизависимость**
- d)

39. Какие ключевые слова OpenMP можно использовать после директивы parallel

- a) **firstprivate**
- b) **lastprivate**
- c) **reduction**
- d) runtime

40. Какие ключевые слова OpenMP можно использовать после директивы single

- a) **private**
- b) **firstprivate**
- c) **nowait**
- d) **copyprivate**

41. Произвольный замок в OpenMP может находиться в состоянии:

- a) **неинициализированном**
- b) **разблокированном**
- c) **заблокированном**
- d) **распределенном**

42. Какая из приведенных в лекции топологий (при одинаковом количестве процессоров) обладает наименьшим диаметром:

- a) топология гиперкуб
- b) топология линейка
- c) **топология полный граф**
- d) топология трехмерной решетки

43. Какое сокращение соответствует стандартной одноядерной архитектуре в рамках классификации Флинна?

- a) **SISD**
- b) SIMD
- c) MISD
- d) MIMD

44. Какая архитектура в рамках классификации Флинна наиболее распространена среди суперкомпьютеров?

- a) SISD
- b) SIMD
- c) MISD
- d) **MIMD**

45. Нарушению какого условия Бернштейна соответствует наличие зависимости по выходным данным?

- a) **нарушению первого условия Бернштейна**
- b) **нарушению первого условия Бернштейна**
- c) **нарушению первого условия Бернштейна**
- d) **ни одно из условий Бернштейна не нарушено**

2) задания с коротким ответом — 2 балла

46. Какой аббревиатурой в классификации Флинна обозначается класс вычислительных систем, поддерживающих работу одного потока команд, одного потока данных?
47. Какой аббревиатурой в классификации Флинна обозначается класс вычислительных систем, поддерживающих работу одного потока команд, нескольких потоков данных?
48. Какой аббревиатурой в классификации Флинна обозначается класс вычислительных систем, поддерживающих работу нескольких потоков команд, одного потока данных?
49. Какой аббревиатурой в классификации Флинна обозначается класс вычислительных систем, поддерживающих работу нескольких потоков команд, нескольких потоков данных?
50. В системах с распределенной памятью каждый вычислительный узел имеет доступ только к принадлежащему ему участку памяти — локальной памяти. В этом случае для межпроцессорного обмена данными предусматривается возможность отправки и приема сообщений по коммуникационной сети, объединяющей вычислительную систему. Как называется соответствующая модель программирования?
51. Как называется связный ациклический граф?
52. Какой аббревиатурой принято обозначать модель вычислительной системы — машину с произвольным доступом к памяти?
53. Какой аббревиатурой принято обозначать модель вычислительной системы — параллельную машину с произвольным доступом к памяти?
54. Как называется величина, равная отношению $T1(N)/Tp(N)$, где p — количество вычислительных узлов?
55. Как называется величина, равная отношению $T1(N)/(p \cdot Tp(N))$?
56. Величина, равная произведению времени параллельного решения задачи и числа используемых процессоров называется «... вычислений». Заполните пропуск.
57. Как называется упорядоченный по времени набор векторов, образованных всеми данными в памяти вычислительной системы?
58. Какую аббревиатуру используют для обозначения отдельного устройства вычислительной системы, способное выполнять графический рендеринг, а также высокопроизводительные вычисления наряду с центральным процессором?

Ответы:

46. SISD
47. SIMD
48. MISD
49. SISD
50. модель передачи сообщений
51. дерево
52. RAM
53. PRAM
54. ускорение
55. эффективность
56. стоимость
57. история содержимого памяти
58. GPU

3) задания с развернутым ответом

59. Время работы последовательной версии некоторого алгоритма A равно $T1(N) = 2N \log_2(N) \tau$, где N — размер входных данных, τ — время выполнения одной вычислительной операции. В предположении, что алгоритм допускает максимальное распараллеливание, т.е. время работы на вычислительной системе с p процессорами равно $Tp(N) = T1(N) / p$, вычислите время работы алгоритма A в следующем случае: $N = 32$, $p = 4$.

60. Время работы последовательной версии некоторого алгоритма A равно $T1(N) = 2N \log_2(N) \tau$, где N — размер входных данных, τ — время выполнения одной вычислительной операции. В предположении, что алгоритм допускает максимальное распараллеливание, т.е. время работы на вычислительной системе с p процессорами равно $Tp(N) = T1(N) / p$, вычислите время работы алгоритма A в следующем случае: $N = 32$, $p = 16$.

61. Время работы последовательной версии некоторого алгоритма B равно $T1(N) = 2^N \tau$, где N — размер входных данных, τ — время выполнения одной вычислительной операции. В предположении, что алгоритм допускает максимальное распараллеливание, т.е. время работы на вычислительной системе с p процессорами равно $Tp(N) = T1(N) / p$, вычислите время работы алгоритма B в следующем случае: $N = 32$, $p = 4$.

62. Время работы последовательной версии некоторого алгоритма B равно $T1(N) = 2^N \tau$, где N — размер входных данных, τ — время выполнения одной вычислительной операции. В предположении, что алгоритм допускает максимальное распараллеливание, т.е. время работы на вычислительной системе с p процессорами равно $Tp(N) = T1(N) / p$, вычислите время работы алгоритма B в следующем случае: $N = 32$, $p = 16$.

63. Пусть доля последовательных вычислений f в некоторой программе равна 1/10. Вычислите максимальное ускорение программы $(Sp)_{\max}$ на вычислительной системе с p процессорами с учетом закона Амдала.

64. Пусть доля последовательных вычислений f в некоторой программе равна 1/100. Вычислите максимальное ускорение программы S_{∞} с учетом закона Амдала.

65. Покажите, что условия Бернштейна не являются необходимыми условиями коммутативности двух операторов

66. На экзамене по курсу параллельного программирования студент утверждает, что в рамках коммуникатора любой процесс может отправить с помощью функции `MPI_Send()` сообщение любому процессу, в том числе и самому себе. Прав ли студент?

67. Пусть организуется попарный обмен сообщениями в рамках коммуникатора `MPI_COMM_WORLD`, и количество процессов в параллельной программе равно N. Запишите множество рангов, доступных для использования в качестве рангов процесса-получателя в `MPI_Send()`;

68. Пусть организуется попарный обмен сообщениями в рамках коммуникатора `MPI_COMM_WORLD`, и количество процессов в параллельной программе равно N. Запишите множество рангов, доступных для использования в качестве рангов процесса-отправителя в `MPI_Recv()`.

69. Установите, допускает ли следующий участок кода распараллеливание с помощью директивы среды OpenMP. Если такое распараллеливание возможно, то приведите соответствующую директиву, если невозможно, то объясните, почему.

```
for(int i=0;i<(int)log2(N);i++)
{
    list[i] = 2.0*i*log2(i+1);
    if ((i>=IMIN) && (i<=IMAX)) temp[i] = list[i];
}
```

70. Установите, допускает ли следующий участок кода распараллеливание с помощью директивы среды OpenMP. Если такое распараллеливание возможно, то приведите соответствующую директиву, если невозможно, то объясните, почему.

```
double res = 0.0;
for(int i=0;i<N;i++)
res += list[i]*list[i];
```

71. Установите, допускает ли следующий участок кода распараллеливание с помощью директивы среды OpenMP. Если такое распараллеливание возможно, то приведите соответствующую директиву, если

невозможно, то объясните, почему.

```
for(int i=0;i<N;i++)
{
list[i] = 1.5*exp(i);
if ( list[i] <= LMAX )
break;
}
```

Ответы

59. $Tr(N) = 80\tau$

60. $Tr(N) = 20\tau$

61. $Tr(N) = 2^{(30)} \tau$

62. $Tr(N) = 2^{(28)} \tau$

63. $10p/(p+9)$

64. $S_{\infty} = 100$

65. Рассмотрим следующую пару операторов (o1, o2): o1 : a = a + 1; o2 : a = a - 1;

Ясно, что все три условия Бернштейна в данном случае нарушены. Тем не менее, результат работы программ, содержащих последовательности (o1, o2) и (o2, o1), будет одинаков для всех возможных значений переменной a. Таким образом, условия Бернштейна не являются необходимыми условиями коммутативности двух операторов.

66. Да, в среде MPI процесс коммуникатора может отправить сообщение любому процессу, принадлежащему тому же коммуникатору, в том числе и себе. Тем не менее, указанной возможности в программе желательно избегать, т.к. в данном случае возможно появление ситуации «тупика».

67. $rank \in \{0,1,\dots,N - 1\} \cup \{MPI_PROC_NULL\}$

68. $rank \in \{0,1,\dots,N - 1\} \cup \{MPI_PROC_NULL\} \cup \{MPI_ANY_SOURCE\}$

69. цикл допускает распараллеливание, соответствующая директива:

```
#pragma omp parallel for
```

70. цикл допускает распараллеливание, соответствующая директива:

```
#pragma omp parallel for reduction(+:res)
```

71. наличие оператора break не позволяет воспользоваться директивами распараллеливания;

Для оценивания результатов обучения на зачёте используются оценки: «зачтено» и «не зачтено».

| Критерии оценивания компетенций | Уровень сформированности компетенций | Шкала оценок |
|--|--------------------------------------|-------------------|
| <p>Ответа обучающегося соответствует хотя бы половине из перечисленных критериев. Сформированные знания основных понятий, определений и теорем, изучаемых в курсе, возможно с затруднениями при воспроизведении.</p> | <p>Пороговый уровень</p> | <p>Зачтено</p> |
| <p>Ответ на контрольно-измерительный материал не соответствует более чем половине из перечисленных показателей. Обучающийся демонстрирует отрывочные знания (либо их отсутствие) основных понятий, определений и теорем, используемых в курсе.</p> | <p>-</p> | <p>Не зачтено</p> |